The Dissertation Committee for David Michael Kamensky
certifies that this is the approved version of the following dissertation:

Immersogeometric fluid–structure interaction analysis of bioprosthetic heart valves

Committee:

Michael S. Sacks, Supervisor

Thomas J. R. Hughes, Supervisor

Omar Ghattas

Robert D. Moser

Ming-Chen Hsu

Immersogeometric fluid–structure interaction analysis of bioprosthetic heart valves

by

David Michael Kamensky, B.S.

DISSERTATION
Presented to the Faculty of the Graduate School of
The University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN
August 2016

Acknowledgments

First, I would like to thank my committee members for supervising this research and contributing valuable expertise. My starting point for implementing the numerical methods described in this dissertation was the computational fluid–structure interaction framework developed by Yuri Bazilevs's group at UC San Diego, which I could never have made sense of without the patient supervision of Ming-Chen Hsu throughout the past four years. I would also like to thank John A. Evans, Yue Yu, and Daniele Boffi for helpful discussions on the mathematical analysis contained in this dissertation. Further, the experimental results reported in this dissertation could not have been collected without the diligent work of John G. Lesicko, Jordan L. Graves, Hugo Landaverde, Javier Solis, Mitchell A. Katona, Samuel Petter, and Bruno Rego. Some of the comptuations presented relied on supercompting resources provided by the Texas Advanced Computing Center (TACC). Essential financial support for this work was provided by the CSEM Graduate Fellowship at UT Austin and NIH/NHLBI grants R01 HL108330 and R01 HL070969.

Immersogeometric fluid–structure interaction analysis of bioprosthetic heart valves

Publication No. ____________________

David Michael Kamensky, Ph.D.
The University of Texas at Austin, 2016

Supervisors: Michael S. Sacks
Thomas J. R. Hughes

The purpose of this dissertation is to develop numerical methods for fluid–structure interaction (FSI) analysis that are suitable for modeling and simulating bioprosthetic heart valves (BHVs). BHVs are prosthetic replacements for the valves that regulate blood flow through the heart. BHVs reproduce natural hemodynamic conditions by mimicking the structure of native heart valves: they consist of thin flexible leaflets, passively driven by interaction with surrounding fluid. Current designs frequently require replacement 10–15 years after implantation. Computer simulation may help identify causes of and solutions to durability issues. Despite much previous research into computer simulation of heart valve FSI, inconvenience or inaccuracy of readily available numerical methods have prevented widespread incorporation of FSI into models of heart valve mechanics.

Challenges associated with heart valve FSI simulation include large deformations of the region occupied by fluid, with changes of topology as the valve opens and closes, and low mass of the structure relative to the fluid, which necessitates careful treatment of fluid–structure coupling. The presence of large pressure gradients also requires special attention to the treatment of fluid mass conservation. Further, a useful numerical method for studying and improving designs of BHVs should be able to capture variations of valve geometry without requiring major effort to construct geometry-specific discretizations.

To meet these challenges, I develop a new numerical approach, combining the immersed boundary concept of capturing fluid–structure interfaces on unfitted discretizations with recent developments in isogeometric analysis (IGA), which directly uses geometrical designs of engineered systems as discrete analysis meshes. In this work, I immerse an isogeometric structure discretization into an unfitted analysis mesh of the fluid subproblem. I refer to the immersion of design geometries into unfitted analysis meshes as *immersogeometric analysis*. To reliably couple unfitted discretizations of the fluid and structure subproblems, I introduce a new semi-implicit time integration procedure and analyze its stability and convergence in the context of linear model problems. I verify that this analysis extrapolates to the nonlinear setting through numerical experiments and explore the validity of my modeling assumptions by comparing computer simulations with observations from an *in vitro* experiment.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

This dissertation develops a methodology for computational fluid–structure interaction (FSI) analysis that is suitable for modeling and simulating heart valves. Despite many years of research into computational analysis of heart valve FSI and much interest from the biomedical engineering community, the inconvenience or inaccuracy of readily available numerical methods have so far prevented widespread incorporation of FSI into models of heart valve mechanics.

1.1 Bioprosthetic heart valves

Heart valves are passive structures that open and close in response to hemodynamic forces, ensuring proper unidirectional blood flow through the heart. At least 280,000 diseased heart valves are surgically replaced annually [133, 152]. The most popular replacements are bioprosthetic heart valves

---

I originally prepared Section 1.2 of this chapter for the following paper:

J. S. Soares, K. R. Feaver, W. Zhang, D. Kamensky, A. Aggarwal, M. S. Sacks. Biomechanical behavior of bioprosthetic heart valve heterograft tissues: Characterization, simulation, and performance. *Cardiovascular Engineering and Technology*, Accepted. D. Kamensky prepared the literature review on computational FSI. Other authors prepared the remainder of the document.

(BHVs), which are fabricated from biologically-derived materials. Like native valves, BHVs consist of thin flexible leaflets that are pushed open by blood flow in one direction and closed by flow in the other direction. BHVs have more natural hemodynamics than the older "mechanical" prostheses designs, which consist of rigid moving parts and require life-long anticoagulation therapy [133]. However, the durability of a typical BHV remains limited to about 10–15 years, with failure resulting from structural deterioration, mediated by fatigue and tissue mineralization [133, 152, 158, 182]. While much research has sought to prevent mineralization, methods to extend durability remain largely unexplored. A critical part of such efforts to improve the design of BHVs is understanding the stresses within leaflets over the complete cardiac cycle.

Techniques from computational engineering may be used for stress analysis of heart valves. Some previous computational studies on heart valve mechanics have used (quasi-)static [7, 170] and dynamic [112] structural analysis, with assumed pressure loads on the leaflets. This produces deformation and stress distributions that can be used to understand the mechanical behavior of BHVs. However, the assumed pressure load only crudely approximates the interaction between blood and valvular structures. A purely structural analysis is only applicable to static pressurization of a closed valve, which represents only a portion of the full cardiac cycle. It is therefore important to simulate the dynamics of heart valves interacting with hemodynamics using methods for computational FSI.

1.2 Computational FSI analysis of BHVs

Native and prosthetic valves present a number of unique challenges for FSI analysis. Foremost among these is the fact that the heart valve leaflets contact one another, changing the topology of the fluid subdomain. This means that standard arbitrary Lagrangian–Eulerian (ALE) [46, 47, 93] or deforming-spatial-domain/space–time (DSD/ST) [179, 180] formulations, which continuously deform the fluid subproblem domain from some reference configuration, are no longer directly applicable. To salvage such methods, one must augment them with special techniques to handle extreme deformations like topology changes. One solution is remeshing, i.e. generating a new mesh of finite elements or finite volumes for the fluid subproblem domain whenever its deformation becomes too extreme [99–101, 177]. This allows computations to proceed, but introduces additional computational cost and numerical errors associated with the projection of fluid solutions from old to new meshes. Recent work by Takizawa et al. [172] introduced the space–time with topology change (ST-TC) method, which extended the DSD/ST framework to allow topology changes without remeshing. Reference [173] applied ST-TC to CFD of a heart valve with prescribed leaflet motion, but the application of ST-TC to complex FSI with sliding and/or unpredictable structural self contact remains an open problem. Makhijani et al. [124] reported a boundary-fitted BHV FSI simulation in 1997, but replaced true contact with inverse-square-law repulsive forces between leaflets and a symmetry plane. While the results in [124] appear promising, few methodological details were provided and no further work on BHV FSI analysis using this method has been reported.

In light of the difficulties encountered in boundary-fitted FSI analysis of heart valves, the overwhelming majority of work to-date on native and bioprosthetic heart valve FSI analysis has followed in the tradition of Peskin's famous immersed boundary method [131].[1] While it is not a universal convention, I follow [126, 148, 160] in applying the term "immersed boundary method" quite broadly, using it to describe any numerical method for approximately solving partial differential equations (PDEs) that allows boundaries of the PDE domain to cut arbitrarily through the computational mesh defining the discrete solution space. In my experience, researchers have highly variable interpretations of the term "immersed boundary method", and I would recommend that writers explicitly clarify its meaning within a particular document.

Immersed boundary methods for FSI can greatly simplify the treatment of large structural deformations and structural self-contact, but engender a number of disadvantages relative to ALE and DSD/ST techniques, as reviewed by Tezduyar [178]. In particular, immersed boundary methods struggle to efficiently capture boundary layer solutions near the fluid–structure interface. Takizawa et al. [171] found that the resolution of such layers is essential to obtaining accurate fluid–structure shear stresses in hemodynamic analyses. A comprehensive overview of various immersed boundary methods and their

---

[1] An even more radical departure from boundary-fitted FSI is to discretize the fluid using a mesh-free approach, such as smoothed-particle hydrodynamics (SPH) [65]. SPH is not widely used in the engineering CFD or FSI communities, but has occasionally been applied to heart valve simulation [183].

properties is beyond the scope of this literature review, and we refer the interested reader to the review articles [126, 160]. The present review of immersed boundary methods is limited to their history of application to heart valve FSI analysis.

Peskin introduced the immersed boundary concept in 1972, specifically to meet the demands of heart valve FSI analysis [131]. The numerical method proposed by Peskin has found little if any direct application by bioengineers, though, due to its crude representation of the heart valve as a collection of markers connected by elastic fibers. However, deficient modeling of the structure subproblem is not an inherent feature of immersed boundary methods. In the early 2000s, de Hart et al. [42–44] and van Loon et al. [189–191] used an immersed boundary method introduced by Baaijens [8] to couple finite element discretizations of heart valves and blood flow. This allowed for investigation of various constitutive models, but numerical instabilities prevented analysis at realistic Reynolds numbers and transvalvular pressure levels. Increasing availability of parallel computing resources in the 2010s has led to higher resolution simulations of heart valves in recent years. Griffith [70] adapted Peskin's original immersed boundary approach to modern distributed-memory computer architectures and included adaptive mesh refinement for the fluid subproblem, to compute FSI of a native aortic valve throughout a full cardiac cycle, with physiological flow velocities and pressure differences. Borazjani [29] applied the curvilinear immersed boundary (CURVIB) method [30, 60] to simulate systolic ejection through a bioprosthetic aortic valve, using nearly 10 million grid points in the fluid domain. The valve leaflet models in the studies by Griffith and Borazjani suffered from deficiencies, though, with [70] modeling the leaflets in the style of Peskin, as markers connected by elastic fibers, and [29] omitting bending stiffness. The CURVIB method was extended to include fluid–shell structure interaction in [62, 64], but the efficacy of the approach has not yet been demonstrated for the portion of the cardiac cycle in which the valves are closed and must support a large pressure differential.

The immersed boundary analyses cited above have relied on academic research codes. As early as the late 1990s, immersed-boundary fluid–structure coupling methods in the commercial software LS-DYNA [39] have been used for FSI simulations of bioprosthetic and native aortic valves [35, 36, 169, 199]. Terminology in [35, 36] may lead to some confusion, since the computational method is described as ALE, while the figures clearly show non-matching fluid and structure interfaces. Sturla et al. clarify this point in Appendix B of [169], describing the adopted algorithms in greater detail, confirming that the numerical method is what I would classify as an immersed boundary approach, based on the definition given above. The time-explicit procedures used by LS-DYNA result in severe Courant–Friedrichs–Lewy conditions [40, 41] that limit the maximum stable time step size in hemodynamic computations, because blood is nearly incompressible, rendering the problem effectively parabolic. Complete fluid incompressibility can even render explicit fluid–structure coupling unconditionally unstable when the fluid and structure have similar densities [188]. References [169, 199] circumvented this difficulty by artificially reducing the speed of sound in blood by a significant factor, reporting that the fluid density variations introduced by this deliberate modeling error were negligible. The use of other commercial off-the-shelf analysis software for heart valve FSI analysis may be possible using so-called "black box" coupling algorithms [28] to connect independent finite element analysis and CFD programs without access to their internal details. Specialized methods are required for stable and efficient black box coupling of fluids to thin, light structures such as heart valve leaflets [125, 188]. Astorino et al. [6] applied a novel black box coupling algorithm to FSI analysis of an idealized aortic valve.

1.3 Immersogeometric analysis

Following the majority of the studies cited in Section 1.2, this dissertation pursues an immersed boundary approach to heart valve FSI analysis. The goal of immersed boundary methods has always been to simplify the construction of analysis-suitable computational models from available geometric data (such as design drawings, anatomic models, or medical images) specifying the domain of a PDE system. Traditional immersed boundary analysis eases this process by allowing subproblems to be discretized separately, then coupled through a numerical method.

Another technology for simplifying computational model generation is isogeometric analysis (IGA), introduced by Hughes et al. [90] in 2005. IGA is based on the insight that many geometries in engineering design are specified in spline spaces that can be systematically enriched, then used to approximate solutions of PDEs. As it turns out, these spline spaces also have many desirable mathematical properties that can improve the quality of approximate PDE solutions. These properties include control over smoothness, improved approximation power relative to classical finite element spaces [52], and straightforward constructions of discrete de Rham complexes [33, 34]. The benefits of these properties are evident in both fluid and structural analysis efforts, including studies of incompressible flow [3, 16, 51, 54], thin shells [109–111, 128], extreme mesh distortion [123], and contact problems [45, 127]. All of these topics are relevant to heart valve FSI analysis. IGA encounters several major difficulties, though, when faced with realistic engineering designs. Foremost among these are:

1. Many designs of volumes are specified in terms of bounding spline surfaces. If an analyst wishes to solve a PDE in such a volume, then IGA, as originally conceived, is inapplicable, since the spline space used for the design can only represent functions on the boundary of the PDE domain.

2. Spline surfaces in designs are frequently trimmed along curves that do not conform to the parametric supports of the spline space's basis functions. The analysis space suggested by standard IGA is therefore not fitted to the actual boundaries of the problem domain.

These challenges could be addressed by changing the way in which engineering products are designed: designers could transition to using geometry representations that are analysis-suitable. Changing the habits of designers throughout industry, however, would require an incredible feat of mass persuasion. Undeterred by this challenge, the creators of analysis-suitable design technologies (e.g. T-splines [153–155]) have succeeded at incorporating their work into several major commercial design platforms. It remains, in my opinion, doubtful, though, that analysis-suitable design will become a standard practice in industry any time soon. Further, many engineering designs specified in past or current formats will remain relevant for decades into the future.

One way to make IGA practical without changing the design process is to incorporate numerical methods from immersed boundary analysis. Difficulty 1 can be alleviated by creating a convenient unfitted analysis space covering the volume of interest, then using an immersed boundary method to enforce the desired conditions on the spline surfaces enclosing the PDE domain. Difficulty 2 can be addressed by using the natural isogeometric solution space while treating the trim curves as immersed boundaries. Promising work in both of these directions has already been carried out using an immersed boundary approach known as the finite cell method [137, 140, 141, 150, 151]. In addition to patching weaknesses of first-generation IGA, the direct application of immersed boundary techniques to design geometries can eliminate the meshing and consequent geometrical approximation[2] of subproblems from

---

[2] In practice, immersogeometric methods must frequently approximate integrals over the domain geometry, which may be considered a type of geometrical approximation [168, Sections 4.3 and 4.4], but I would argue that this is conceptually distinct from the direct alteration of domain geometry that occurs in traditional mesh generation.

traditional immersed boundary analysis. In [105], my collaborators and I introduced the term *immersogeometric analysis* to describe this symbiotic union of immersed boundary and isogeometric technologies.[3]

1.4 Verification and validation

Throughout the heart valve FSI analysis literature reviewed in Section 1.2, very little emphasis is placed on ensuring that computations accurately approximate solutions of the desired mathematical models or that these models are accurate representations of physical valves. The remainder of this dissertation shall refer to these two problems as "verification" and "validation". This follows the terminological conventions defined in [9]. Specifically, I draw the reader's attention to two key definitions:

- **Verification:** The process of determining whether the output of a computer simulation accurately approximates solutions to the mathematical problem that is said to model a physical event. This might include
  - *A priori* error analysis of a discretization scheme.
  - Testing convergence of simulation outputs to analytical solutions of the mathematical model.
  - Comparison of numerical solutions to benchmark problems with the results of thoroughly verified codes.

---

[3] The word "immersogeometric" was originally coined in 2014 by T.J.R. Hughes, while traveling in Italy; it is derived from the Italian word *immerso*, meaning "immersed".

- **Validation:** The process of determining whether a mathematical model accurately represents a physical event. This might include
  - Comparison of experimental results with analytical solutions of the mathematical model.
  - Comparison of experimental results with outputs of thoroughly verified computations.

Aphoristically, verification is concerned with solving the equations right while validation is concerned with solving the right equations. Conflation of verification and validation is a surprisingly common conceptual error in computational science and engineering. For instance, the reader can likely think of published works that compare unverified computations with experimental results, then speciously conclude that the numerical method is correctly approximating the solution to a PDE. The bulk of this dissertation is devoted to the development and verification of numerical methods. However, I have included some limited experimental validation, in the form of qualitative comparisons between computational results and *in vitro* experiments.

1.5 Structure and content of the dissertation

This section summarizes the remainder of the dissertation. Chapter 2 states the coupled PDEs that I use to model the fluid–structure system. Chapter 3 describes spatial discretizations for the subproblems corresponding to the fluid and structure. Chapter 4 completes the discretization with a novel semi-implicit time integration scheme to couple the fluid and structure subproblems. In Chapter 4, this time integrator is presented alongside intuitive reasoning. Chapter 5 proves its convergence for a linear parabolic model problem. This analysis provides guidelines for selecting appropriate values of the free parameters of the scheme. The theoretical results are supported by numerical experiments. Chapter 6 finds that the conclusions of this analysis can be extrapolated to the setting of nonlinear Navier–Stokes and large-displacement FSI. I look at both norm convergence and quantities of interest in nonlinear benchmark problems. After verifying that the numerical methods described in Chapters 3 and 4 accurately approximate the mathematical problem put forward in Chapter 2, Chapter 7 applies the new technology to BHV FSI simulation and compares the results to *in vitro* experimental work. Finally, Chapter 8 sketches some future developments that may improve on the technology of this dissertation, connect it to clinical practice, and apply it to other FSI problems.

The contributions summarized above can be grouped into the three concentration areas of the Computational Science, Engineering, and Mathematics (CSEM) graduate program as follows:

- **Area A:** Applicable Mathematics
    - Formulation and *a priori* analysis of linear model problems that are rich enough to provide insight into the stability and accuracy of FSI discretizations, but simple enough to remain analytically-tractable.

  - Interpretation of this analysis, to obtain succinct guidelines for selecting free parameters of numerical methods.

- **Area B:** Numerical analysis and scientific computation
  - Formulation and implementation of an algorithm for fluid–thin structure coupling that permits stable simulations of large-displacement FSI problems using independent fluid and structure discretizations.
  - Application of this algorithm to isogeometric discretizations of fluid and structure subproblems, allowing spline-based structure geometries to be directly analyzed.
  - Numerical experiments to support the Area A contributions.
  - Numerical experiments to support extrapolation of conclusions from the Area A contributions to the context of nonlinear FSI.
  - Enhancements to previous technologies for discretizing fluid and structure subproblems, including
    - penalty-based contact for isogeometric discretizations of shell structures,
    - modifications to stabilized methods for incompressible flow that improve mass conservation near immersed boundaries,
    - and a distributed-memory implementation of divergence-conforming B-spline discretizations of incompressible flow [51] with immersed boundaries.

- **Area C:** Mathematical modeling and applications
  - Formulation of a mathematical model for a BHV interacting with surrounding fluid.
  - Application of the developed fluid–thin structure interaction framework to BHV FSI simulations.
  - Preliminary validation of the mathematical model for BHV FSI, in the form of qualitative comparisons between simulation outputs and results of an original *in vitro* experiment.

Chapter 2

Mathematical model of FSI

To construct a verifiable computer simulation of BHV FSI, one must first assume a correspondence between the physical phenomenon and a mathematical problem that is said to model it. In this dissertation, I follow the thoroughly-validated paradigm of modeling macroscopic fluid and structural dynamics within the framework of continuum mechanics, where the deformations of bulk materials are represented by solutions to partial differential equations. In particular, I model BHV leaflets as thin shells structures, using the Kirchhoff–Love theory, and I model the surrounding fluid, be it human blood or the working fluid of an *in vitro* experiment, as an incompressible Newtonian fluid. These two subproblems are coupled through kinematic and dynamic compatibility conditions along the fluid–solid interface. Due to its thinness, the structure is modeled geometrically as a 2D surface embedded in

---

Some of this chapter's content is derived from the following paper:

D. Kamensky, M.-C. Hsu, Y. Yu, J. A. Evans, M. S. Sacks, T. J. R. Hughes. Immersogeometric cardiovascular fluid–structure interaction analysis using divergence-conforming B-splines. *Computer Methods in Applied Mechanics and Engineering*, In review (preprint: ICES Report 16-14). D. Kamensky implemented the numerical methods, formulated and analyzed model problems, and participated in the experimental work. M.-C. Hsu provided supervision and edited the manuscript extensively. Y. Yu, J. A. Evans, and T. J. R. Hughes supervised the mathematical analysis. M. S. Sacks helped plan and supervise the laboratory experiments.

the 3D fluid domain. A discussion of the validity of these modeling assumptions is deferred to Chapter 7. The present chapter is concerned only with stating the mathematical problem. I state the model in a weak form, which is both more suggestive of finite-dimensional approximations and more conducive to the inclusion of distributional forces associated with immersed boundaries. I am deliberately vague in the statement of the problem; a detailed mathematical study of the necessary and/or sufficient conditions on the regularity of domain geometries, problem data, and function spaces for this nonlinear problem to be well posed is beyond the scope of this dissertation. Some linear model problems are stated and analyzed more rigorously in Chapter 5.

Remark 2.1. The reader may expect the structure subproblem to include some stipulation that the structure cannot deform to intersect itself. Explicit inclusion of such an inequality constraint would, however, be redundant in light of the fluid–structure kinematics, since a single-valued, continuous velocity field is defined throughout the fluid–structure continuum. While it is, in practice, critical to include some specialized treatment of structure-on-structure contact in a numerical method, because the lubrication limit is essentially impossible to fully resolve, I consider the treatment of contact to be a feature of the discretization and not a part of the mathematical model.

2.1 Augmented Lagrangian formulation of FSI

My starting point is the augmented Lagrangian framework for FSI introduced by Bazilevs et al. [19], which I specialize to the case of thin structures as follows. Denote the region occupied by incompressible Newtonian fluid at time t by $\left(\Omega_1\right)_t \subset \mathbb{R}^d$, where d is the number of space dimensions. The structure geometry at time t is modeled by the surface $\Gamma_t \subset (\Omega_1)_t$, with dimension $d-1$. Let $\mathbf{u}_1$ denote the fluid's velocity and p denote its pressure. Let $\mathbf{y}$ denote the structure's displacement from some reference configuration, Γ_0, and $\mathbf{u}_2 \equiv \dot{\mathbf{y}}$ denote the velocity of the structure. The fluid–structure kinematic constraint that $\mathbf{u}_1 = \mathbf{u}_2$ on Γ_t is enforced using the augmented Lagrangian

$$\int_{\Gamma_t} \boldsymbol{\lambda} \cdot (\mathbf{u}_1 - \mathbf{u}_2)\; d\Gamma + \frac{1}{2}\int_{\Gamma_t} \beta |\mathbf{u}_1 - \mathbf{u}_2|^2\; d\Gamma\;, \tag{2.1}$$

where $\boldsymbol{\lambda}$ is a Lagrange multiplier and $\beta \geq 0$ is a penalty parameter. The resulting variational problem is: Find $\mathbf{u}_1 \in \mathcal{S}_u$, $p \in \mathcal{S}_p$, $\mathbf{y} \in \mathcal{S}_d$, and $\boldsymbol{\lambda} \in \mathcal{S}_\ell$ such that, for all test functions $\mathbf{w}_1 \in \mathcal{V}_u$, $q \in \mathcal{V}_p$, $\mathbf{w}_2 \in \mathcal{V}_d$, and $\delta\boldsymbol{\lambda} \in \mathcal{V}_\ell$

$$\begin{aligned}
& B_1(\{\mathbf{u}_1, p\}, \{\mathbf{w}_1, q\}; \hat{\mathbf{u}}) - F_1(\{\mathbf{w}_1, q\}) \\
& \quad + \int_{\Gamma_t} \mathbf{w}_1 \cdot \boldsymbol{\lambda}\; d\Gamma + \int_{\Gamma_t} \mathbf{w}_1 \cdot \beta(\mathbf{u}_1 - \mathbf{u}_2)\; d\Gamma \;=\; 0\;, && (2.2)\\
& B_2(\mathbf{y}, \mathbf{w}_2) - F_2(\mathbf{w}_2) \\
& \quad - \int_{\Gamma_t} \mathbf{w}_2 \cdot \boldsymbol{\lambda}\; d\Gamma - \int_{\Gamma_t} \mathbf{w}_2 \cdot \beta(\mathbf{u}_1 - \mathbf{u}_2)\; d\Gamma \;=\; 0\;, && (2.3)\\
& \int_{\Gamma_t} \delta\boldsymbol{\lambda} \cdot (\mathbf{u}_1 - \mathbf{u}_2)\; d\Gamma \;=\; 0\;, && (2.4)
\end{aligned}$$

where $\mathcal{S}_u$, $\mathcal{S}_p$, $\mathcal{S}_d$, and $\mathcal{S}_\ell$ are the trial solution spaces for the fluid velocity, fluid pressure, structural displacement, and fluid–structure interface Lagrange multiplier solutions, respectively, and $\mathcal{V}_u$, $\mathcal{V}_p$, $\mathcal{V}_d$, and $\mathcal{V}_\ell$ are the corresponding test function spaces. B_1, B_2, F_1, and F_2 are the semi-linear forms and linear functionals corresponding to the (weak) fluid and structural dynamics subproblems.

2.2 Fluid subproblem

As mentioned above, the weak fluid subproblem is defined to be incompressible and Newtonian:

$$\begin{aligned} B_1(\{\mathbf{u},p\},\{\mathbf{w},q\};\hat{\mathbf{u}}) = &\int_{(\Omega_1)_t} \mathbf{w}\cdot\rho_1\left(\left.\frac{\partial\mathbf{u}}{\partial t}\right|_{\hat{\mathbf{x}}} + (\mathbf{u}-\hat{\mathbf{u}})\cdot\boldsymbol{\nabla}\mathbf{u}\right)\,d\Omega \\ &+\int_{(\Omega_1)_t}\boldsymbol{\varepsilon}(\mathbf{w}):\boldsymbol{\sigma}_1(\mathbf{u},p)\,d\Omega + \int_{(\Omega_1)_t} q\boldsymbol{\nabla}\cdot\mathbf{u}\,d\Omega \\ &-\gamma\int_{(\Gamma_{1\mathbf{h}})_t}\mathbf{w}\cdot\rho_1\left\{(\mathbf{u}-\hat{\mathbf{u}})\cdot\mathbf{n}_1\right\}_-\mathbf{u}\,d\Gamma\ , \end{aligned} \tag{2.5}$$

$$F_1(\{\mathbf{w},q\}) = \int_{(\Omega_1)_t}\mathbf{w}\cdot\rho_1\mathbf{f}_1\,d\Omega + \int_{(\Gamma_{1\mathbf{h}})_t}\mathbf{w}\cdot\mathbf{h}_1\,d\Gamma\ , \tag{2.6}$$

where ρ_1 is the fluid mass density, $\boldsymbol{\varepsilon}$ is the symmetric gradient operator, $\boldsymbol{\sigma}_1(\mathbf{u},p) = -p\mathbf{I} + 2\mu\boldsymbol{\varepsilon}(\mathbf{u})$, where μ is the dynamic viscosity, $\mathbf{f}_1$ is a prescribed body force, and $\mathbf{h}_1$ is a prescribed traction on $\Gamma_{1\mathbf{h}} \subset \partial\Omega_1$. I assume that $(\Omega_1)_t$ deforms from some reference configuration, $(\Omega_1)_0$, according to the velocity field $\hat{\mathbf{u}}$, which need not equal $\mathbf{u}_1$. $\partial(\cdot)/\partial t|_{\hat{\mathbf{x}}}$ indicates time differentiation with respect to a fixed point $\hat{\mathbf{x}}$ from $(\Omega_1)_0$. The last term of (2.5) is not usually considered to be part of the weak Navier–Stokes problem, but it enhances the stability of the problem in cases where flow enters through the Neumann boundary $\Gamma_{1\mathbf{h}}$ [50]. The function $\{\cdot\}_-$ isolates the negative part of its argument, i.e.

$$\{x\}_- = \left\{\begin{array}{ll} 0 & x > 0 \\ x & \text{otherwise} \end{array}\right. . \tag{2.7}$$

The coefficient γ controls the strength of this stabilizing term and $\mathbf{n}_1$ is the outward-facing normal to Ω_1. The stabilizing influence of this term can be understood by testing the coercivity of B_1 in the case of $\gamma \geq 1$. This term is often viewed as a detail of the discretization, but γ is typically $\mathcal{O}(1)$, in which case the perturbation introduced by this stabilization is a modification of the mathematical model; it will not converge to zero with refinement of the discretization.

2.3 Thin structure subproblem

Following the Kirchhoff–Love thin shell kinematic hypotheses (see, e.g. [109–111]), B_2 and F_2 from the structure subproblem are defined

$$B_2(\mathbf{y},\mathbf{w}) = \int_{\Gamma_t} \mathbf{w}\cdot\rho_2 h_{\text{th}}\, \frac{\partial^2\mathbf{y}}{\partial t^2}\bigg|_{\mathbf{X}}\, d\Gamma + \int_{\Gamma_0}\int_{-h_{\text{th}}/2}^{h_{\text{th}}/2} D_{\mathbf{w}}\mathbf{E} : \mathbf{S}\; d\xi^3 d\Gamma \tag{2.8}$$

and

$$F_2(\mathbf{w}) = \int_{\Gamma_t} \mathbf{w}\cdot\rho_2 h_{\text{th}}\mathbf{f}_2\; d\Gamma + \int_{\Gamma_t} \mathbf{w}\cdot\mathbf{h}^{\text{net}}\; d\Gamma\ , \tag{2.9}$$

where ρ_2 is the structure mass density, $\mathbf{f}_2$ is a prescribed body force, h_{th} is the thickness of the shell, ξ^3 is a through-thickness coordinate, and I have referred the elasticity term to the reference configuration (cf. [81, (8.60)] or [23, (1.80)]). $\mathbf{E}$ is the Green–Lagrange strain tensor [81, (2.67)] corresponding to the displacement $\mathbf{y}$, $D_{\mathbf{w}}\mathbf{E}$ is its functional derivative in the direction of $\mathbf{w}$, viz.

$$D_{\mathbf{w}}\mathbf{E}(\mathbf{y}) = \frac{d}{d\epsilon}\mathbf{E}(\mathbf{y}+\epsilon\mathbf{w})\bigg|_{\epsilon=0}\ , \tag{2.10}$$

and $\mathbf{S}$ is the second Piola–Kirchhoff stress tensor [81, (3.63)], depending on $\mathbf{E}$. The last term of F_2 sums the prescribed tractions on the two sides of Γ_t: $\mathbf{h}^{\text{net}} = \mathbf{h}(\xi^3 = -h_{\text{th}}/2) + \mathbf{h}(\xi^3 = +h_{\text{th}}/2)$. $\partial(\cdot)/\partial t|_{\mathbf{X}}$ indicates time differentiation with respect to a fixed material point, $\mathbf{X}\in\Gamma_0$.

2.3.1 Definition of E in terms of y

The Green–Lagrange strain $\mathbf{E}$ is simplified to depend entirely on the shell structure's midsurface displacement, $\mathbf{y} : \Gamma_0 \to \mathbb{R}^d$, so as to reduce the dimension of the solid mechanics problem. Stating the precise dependence of $\mathbf{E}$ on $\mathbf{y}$ requires some notation. I shall assume, for now, that $d = 3$ (so that Γ_t is a 2D surface), as this is the highest space dimension in which BHVs are designed to function, and the specialization to $d = 2$ (i.e. a 2D beam theory in which Γ_t is a 1D curve) is straightforward. First, consider a coordinate chart on Γ_0, mapping points $\mathbf{X}$ of the midsurface to coordinate pairs (ξ^1, ξ^2). Then allow a third coordinate ξ^3 to parameterize material points extruded in the normal direction to Γ_0. Letting Greek letter indices have the range $\{1, 2\}$, define covariant basis vectors

$$\mathbf{A}_\alpha = \frac{\partial \mathbf{X}}{\partial \xi^\alpha} \ , \tag{2.11}$$

$$\mathbf{A}_3 = \frac{\mathbf{A}_1 \times \mathbf{A}_2}{|\mathbf{A}_1 \times \mathbf{A}_2|} \tag{2.12}$$

and

$$\mathbf{a}_\alpha = \frac{\partial \mathbf{x}}{\partial \xi^\alpha} \ , \tag{2.13}$$

$$\mathbf{a}_3 = \frac{\mathbf{a}_1 \times \mathbf{a}_2}{|\mathbf{a}_1 \times \mathbf{a}_2|} \tag{2.14}$$

on the reference and deformed configurations, where $\mathbf{x}(\mathbf{X}(\xi^1, \xi^2)) = \mathbf{y}(\mathbf{X}(\xi^1, \xi^2)) + \mathbf{X}(\xi^1, \xi^2)$ is the deformed position of the midsurface material point $\mathbf{X}$ that is mapped to coordinates (ξ^1, ξ^2). The corresponding contravariant basis vectors $\mathbf{A}^i$ and $\mathbf{a}^i$ are such that

$$\mathbf{A}^i \cdot \mathbf{A}_j = \mathbf{a}^i \cdot \mathbf{a}_j = \delta^i{}_j \ . \tag{2.15}$$

These basis vectors can be used to provide formulas for the midsurface metric tensor

$$g_{\alpha\beta} = \mathbf{a}_\alpha \cdot \mathbf{a}_\beta \ , \tag{2.16}$$

$$G_{\alpha\beta} = \mathbf{A}_\alpha \cdot \mathbf{A}_\beta \tag{2.17}$$

and curvature coefficients

$$b_{\alpha\beta} = -\mathbf{a}_\alpha \cdot \frac{\partial \mathbf{a}_3}{\partial \xi^\beta} = \frac{\partial \mathbf{a}_\alpha}{\partial \xi^\beta} \cdot \mathbf{a}_3 \ , \tag{2.18}$$

$$B_{\alpha\beta} = -\mathbf{A}_\alpha \cdot \frac{\partial \mathbf{A}_3}{\partial \xi^\beta} = \frac{\partial \mathbf{A}_\alpha}{\partial \xi^\beta} \cdot \mathbf{A}_3 \ . \tag{2.19}$$

$G_{\alpha\beta}$, $g_{\alpha\beta}$, $B_{\alpha\beta}$, and $b_{\alpha\beta}$ are then used to define the in-plane components of the simplified Green–Lagrange strain at a point $(\mathbf{X}, \xi^3)$ in $\Gamma_0 \times (-h_{\mathrm{th}}/2, h_{\mathrm{th}}/2)$:

$$E_{\alpha\beta} = \varepsilon_{\alpha\beta} + \xi^3 \kappa_{\alpha\beta} \ , \tag{2.20}$$

where

$$\varepsilon_{\alpha\beta} = \frac{1}{2}(g_{\alpha\beta} - G_{\alpha\beta}) \ , \tag{2.21}$$

$$\kappa_{\alpha\beta} = B_{\alpha\beta} - b_{\alpha\beta} \ , \tag{2.22}$$

with respect to the basis $\mathbf{A}^\alpha \otimes \mathbf{A}^\beta$. While not strictly necessary, I find it most convenient to transform these components into a local Cartesian coordinate system. This ensures compatibility with tensor component formulas found in references assuming Cartesian coordinate systems, thereby expediting the implementation of constitutive relations found in the engineering literature. The local Cartesian basis vectors $\{\hat{\mathbf{I}}_j\}_{j=1}^3$ and $\{\hat{\mathbf{i}}_j\}_{j=1}^3$ are obtained by performing Gram–Schmidt orthonormalization of the covariant basis vectors $\{\mathbf{A}_j\}_{j=1}^3$ and $\{\mathbf{a}_j\}_{j=1}^3$, as in [109, (2.43)–(2.45)]. This ensures that the third coordinate remains out-of-plane, i.e. $\hat{\mathbf{I}}_3 = \mathbf{A}_3$ and $\hat{\mathbf{i}}_3 = \mathbf{a}_3$. For the components of the Green–Lagrange strain, this transformation is computed by (cf. [109, (3.41)])

$$E_{\alpha\beta} \leftarrow E_{\gamma\delta} \left(\hat{\mathbf{I}}_\gamma \cdot \mathbf{A}^\alpha\right) \left(\hat{\mathbf{I}}_\delta \cdot \mathbf{A}^\beta\right) \ . \tag{2.23}$$

For the remainder of the discussion on the structure subproblem I will, unless stated otherwise, assume that tensor components are given in such a local Cartesian basis. This effectively hides the distinction between covariant and contravariant objects, so that all indexes can be given as subscripts without ambiguity.

Remark 2.2. The formula (2.23) differs from [109, (3.41)]. The cited formula [109, (3.41)] is stated in terms of contravariant basis vectors that vary through the thickness of the structure. However, [109] tacitly assumes (2.23) in the analytical integration of bending moments through the thickness [109, (3.38) and (3.39)]. Assuming (2.23) can be construed as a thin-shell approximation of the “shifter tensor” defined by [26, (63)]. Alternatively, [111] and [175] formulate Kirchhoff–Love thin shell mechanics without making this assumption. The FSI techniques proposed in this dissertation are independent of how exactly the shell subproblem is formulated; reference [85] uses them with the shell formulation of [111].

The components $E_{\alpha 3}$ and $E_{3\alpha}$ are assumed to be zero as part of the Kirchhoff–Love model. The component E_{33} must remain nonzero to obtain reasonable agreement with standard elasticity for plane stress problems. E_{33} is made unique by introducing the assumption that $S_{33} = 0$. Given $\mathbf{y}$ (and thus all components of $\mathbf{E}$ aside form E_{33}) the scalar equation $S_{33}(\mathbf{E}) = 0$ can be solved for the unknown E_{33}. The exact expression for E_{33} in terms of $\mathbf{y}$ will therefore depend on the constitutive model expressing $\mathbf{S}$ as a function of $\mathbf{E}$.

Due to the assumptions that $E_{\alpha 3} = E_{3\alpha} = 0$ and $S_{33} = 0$ for any displacement $\mathbf{y}$, the product $D_{\mathbf{w}}\mathbf{E} : \mathbf{S}$ involves only indexes ranging over $\{1, 2\}$:

$$D_{\mathbf{w}}\mathbf{E} : \mathbf{S} = D_{\mathbf{w}}E_{\alpha\beta}S_{\alpha\beta} + D_{\mathbf{w}}E_{\alpha 3}S_{\alpha 3} + D_{\mathbf{w}}E_{3\beta}S_{3\beta} + D_{\mathbf{w}}E_{33}S_{33} \tag{2.24}$$

$$= D_{\mathbf{w}}E_{\alpha\beta}S_{\alpha\beta} \; . \tag{2.25}$$

In particular, this conveniently censors $D_{\mathbf{w}}E_{33}$, which would introduce a dependence of the strain variation on the material, due to the fact that the functional form of $E_{33}(\mathbf{y})$ depends on the constitutive model.

For problems in which $d = 2$, the relevant restriction of this shell theory can be obtained by simply extruding the curve Γ_0 out of the plane to form a 2D shell, but constraining all displacments in the out-of-plane direction. This is the model that I use for all 2D fluid–thin structure interaction problems in this dissertation. The resulting “beam” theory for the $d = 2$ case is not Euler–Bernoulli theory. For a shell composed of an isotropic material with Young’s modulus E and Poisson ratio ν, the effective beam stiffness of the constrained shell formulation is $E/(1-\nu^2)$ rather than E.

2.3.2 Constitutive modeling

Any material model that accepts as input a Green–Lagrange strain $\mathbf{E}$ and returns as output a 2nd Piola–Kirchhoff stress $\mathbf{S}$ can be used directly in the structure subproblem defined above. No special limitations on constitutive modeling are introduced by the use of Kirchhoff–Love shell theory. Throughout this dissertation, I model structures as hyperelastic, meaning that

$$\mathbf{S} = \frac{\partial \Psi}{\partial \mathbf{E}} \ , \tag{2.26}$$

where Ψ is some functional mapping strains to scalar energy densities [81, Chapter 6]. I use two particular models in this dissertation.

2.3.2.1 St. Venant–Kirchhoff model

The simplest example of a strain energy functional is motivated by a formal extension of linearized elasticity:

$$\Psi = \frac{1}{2}\mathbf{E} : \mathbb{C} : \mathbf{E} \quad \Rightarrow \quad \mathbf{S} = \mathbb{C} : \mathbf{E} \ , \tag{2.27}$$

where $\mathbb{C}$ is a rank-four elasticity tensor. In the case where $\mathbb{C}$ corresponds to an isotropic material and can be derived from a Young's modulus E and Poisson ratio ν, this model is refered to as a St. Venant–Kirchhoff material. The St. Venant–Kirchhoff model is popular for its simplicity, but suffers from instabilities in compression [81, Section 6.5, Exercise 4]. The St. Venant–Kirchhoff model also allows for the ξ_3 integral in B_2 to be computed analytically [109, (3.38) and (3.39)] to arrive at a simplified formulation [109, (3.47)].

2.3.2.2 Incompressible neo-Hookean model

A model that is more physically accurate at large strains is the incompressible neo-Hookean model:

$$\Psi = \frac{\mu_s}{2}\left(\mathrm{tr}\mathbf{C} - 3\right) + p\left(\left(\det\mathbf{C}\right)^{1/2} - 1\right) \ , \tag{2.28}$$

where $\mu_s > 0$ is the single material parameter, known as the shear modulus, $\mathbf{C} = 2\mathbf{E} + \mathbf{I}$ is the right Cauchy–Green deformation tensor [81, (2.63)], and p is a Lagrange multiplier to enforce the constraint of incompressibility. The incompressible neo-Hookean model is a special case of the structural constitutive model for soft tissues introduced by Fan and Sacks [57]. Fan and Sacks use an incompressible neo-Hookean term in their strain energy functional to model the effects of extra-cellular matrix, alongside other terms to model the effects of collagen fibers. They show that, for realistic (experimentally calibrated) choices of parameters, the neo-Hookean matrix term dominantes the bending behavior of thin sheets of tissue [57, Figure 4a]. The opening and closing kinematics of BHV leaflets are largely dominated by bending. I have, accordingly, found little sensitivity of leaflet kinematics to additional terms in Ψ modeling collagen fiber behavior at large tensile strains. For simplicity of presentation, I have therefore omitted these additional terms.

Chapter 3

Discretization of subproblems

Distinct fluid and structure subproblems may be isolated from the coupled problem stated in Chapter 2 by setting the test function corresponding to the other subproblem and the test function corresponding to the kinematic constraint to zero. Each of these subproblems may be discretized by adapting existing techniques for computational fluid and structural dynamics. This chapter focuses on semi-discretizations in space. The treatment of time discretization will be tied to the coupling of the two subproblems, which is the

---

Some of this chapter's content is derived from the following publications:

D. Kamensky, M.-C. Hsu, D. Schillinger, J. A. Evans, A. Aggarwal, Y. Bazilevs, M. S. Sacks, T. J. R. Hughes. An immersogeometric variational framework for fluid–structure interaction: Application to bioprosthetic heart valves. *Computer Methods in Applied Mechanics and Engineering*, 284:1005–1053, 2015. D. Kamensky developed the techniques used for fluid–thin structure interaction and structure-on-structure contact. M.-C. Hsu provided supervision and implemented the finite cell method for flow around bulky objects. D. Schillinger helped formulate the finite cell approach used. J. A. Evans provided mathematical advice. A. Aggarwal developed the geometrical model of the valve. Y. Bazilevs, M. S. Sacks, and T. J. R. Hughes supervised the work.

D. Kamensky, M.-C. Hsu, Y. Yu, J. A. Evans, M. S. Sacks, T. J. R. Hughes. Immersogeometric cardiovascular fluid–structure interaction analysis using divergence-conforming B-splines. *Computer Methods in Applied Mechanics and Engineering*, In review (preprint: ICES Report 16-14). D. Kamensky implemented the numerical methods, formulated and analyzed model problems, and participated in the experimental work. M.-C. Hsu provided supervision and edited the manuscript extensively. Y. Yu, J. A. Evans, and T. J. R. Hughes supervised the mathematical analysis. M. S. Sacks helped plan and supervise the laboratory experiments.

subject of Chapter 4.

3.1 Fluid subproblem

The fluid subproblem may be isolated by setting $\mathbf{w}_2 = \delta\boldsymbol{\lambda} = \mathbf{0}$, which yields (2.2), in which the structure velocity $\mathbf{u}_2$ and the Lagrange multiplier $\boldsymbol{\lambda}$ should be viewed as prescribed data. I describe two ways of discretizing this subproblem: the variational multiscale (VMS) approach[1] (Section 3.1.1) and the divergence-conforming B-spline approach (Section 3.1.2). Section 3.1.3 compares the advantages and disadvantages of these two approaches.

3.1.1 Variational multiscale formulation

Galerkin's method is unstable when applied directly to the fluid subproblem (2.2). The usable combinations of discrete velocity and pressure approximation spaces are restricted by inf-sup stability considerations [74] and it is well known that, when advective phenomena dominate diffusion, the Galerkin discrete solution can become highly oscillatory [32, 58]. Further, there would be no turbulence model, and high Reynolds number flows would need to be resolved at viscous length scales. The issues of stability and turbulence modeling are simultaneously addressed by the variational multiscale (VMS) method [91] of Bazilevs et al. [16]. In short, it substitutes an ansatz for

---

[1] My use of the term "VMS" in this dissertation refers to the specific VMS formulation explained in Section 3.1.1, applied to equal-order pressure–velocity discretizations. My choice of terminology should not be taken to mean that the concept of VMS analysis is incompatible with div-conforming B-splines, which is demonstrably [193] not true.

subgrid velocities and pressures into the weak fluid subproblem. This ansatz is consistent with the strong form of the Navier–Stokes equations, so that the resulting formulation smoothly transitions to a high-order-accurate direct numerical simulation as the approximation spaces are refined.

The mesh-dependent VMS formulation is posed on a collection of disjoint fluid elements $\{\Omega^e\}$ such that $\overline{\Omega_1} = \cup_e \overline{\Omega^e}$. $\{\Omega^e\}$, Ω_1, and Γ remain time-dependent, but, when there is no risk of confusion, I drop the subscript t to simplify notation. I introduce a superscript h to indicate association with discrete spaces defined over these elements. The mesh $\{\Omega^e\}$ deforms with velocity $\hat{\mathbf{u}}^h$. Let $\mathcal{V}_u^h$ and $\mathcal{V}_p^h$ be discrete velocity and pressure spaces defined over $\{\Omega^e\}$. The semi-discrete VMS fluid subproblem is: Find $\mathbf{u}_1^h \in \mathcal{V}_u^h$ and $p^h \in \mathcal{V}_p^h$ such that, for all $\mathbf{w}_1^h \in \mathcal{V}_u^h$ and $q^h \in \mathcal{V}_p^h$,

$$\begin{aligned} & B_1^{\mathrm{VMS}}(\{\mathbf{u}_1^h, p^h\}, \{\mathbf{w}_1^h, q^h\}; \hat{\mathbf{u}}^h) - F_1^{\mathrm{VMS}}(\{\mathbf{w}_1^h, q^h\}) \\ & + \int_\Gamma \mathbf{w}_1^h \cdot (\lambda \mathbf{n}_2)\, d\Gamma + \int_\Gamma \mathbf{w}_1^h \cdot \beta(\mathbf{u}_1^h - \mathbf{u}_2)\, d\Gamma \ = \ 0\ , \end{aligned} \tag{3.1}$$

where

$$\begin{aligned} B_1^{\mathrm{VMS}}(\{\mathbf{u},p\},\{\mathbf{w},q\};\hat{\mathbf{u}}) = & \int_{\Omega_1} \mathbf{w}\cdot\rho_1\left(\left.\frac{\partial\mathbf{u}}{\partial t}\right|_{\hat{\mathbf{x}}} + (\mathbf{u}-\hat{\mathbf{u}})\cdot\nabla\mathbf{u}\right)\ d\Omega \\ & + \int_{\Omega_1} \boldsymbol{\varepsilon}(\mathbf{w}) : \boldsymbol{\sigma}_1\ d\Omega + \int_{\Omega_1} q\nabla\cdot\mathbf{u}\ d\Omega \\ & - \gamma\int_{\Gamma_{\mathbf{1h}}} \mathbf{w}\cdot\rho_1\left\{(\mathbf{u}-\hat{\mathbf{u}})\cdot\mathbf{n}_1\right\}_{-}\mathbf{u}\ d\Gamma \\ & - \sum_e \int_{\Omega^e} \left((\mathbf{u}-\hat{\mathbf{u}})\cdot\nabla\mathbf{w} + \frac{1}{\rho_1}\nabla q\right)\cdot\mathbf{u}'\ d\Omega \\ & - \sum_e \int_{\Omega^e} p'\nabla\cdot\mathbf{w}\ d\Omega \\ & + \sum_e \int_{\Omega^e} \mathbf{w}\cdot(\mathbf{u}'\cdot\nabla\mathbf{u})\ d\Omega \\ & - \sum_e \int_{\Omega^e} \frac{1}{\rho_1}\nabla\mathbf{w} : (\mathbf{u}'\otimes\mathbf{u}')\ d\Omega \\ & + \sum_e \int_{\Omega^e} (\mathbf{u}'\cdot\nabla\mathbf{w})\,\overline{\tau}\cdot(\mathbf{u}'\cdot\nabla\mathbf{u})\ d\Omega\ , \end{aligned} \tag{3.2}$$

and

$$F_1^{\mathrm{VMS}}(\{\mathbf{w},q\}) = F_1(\{\mathbf{w},q\})\ . \tag{3.3}$$

The forms B_1^{VMS} and F_1^{VMS} are the VMS semi-discrete counterparts of B_1 and F_1. $\mathbf{u}'$ is the fine scale velocity ansatz,

$$\mathbf{u}' = -\tau_{\mathrm{M}}\left(\rho_1\left(\left.\frac{\partial\mathbf{u}}{\partial t}\right|_{\hat{\mathbf{x}}} + (\mathbf{u}-\hat{\mathbf{u}})\cdot\nabla\mathbf{u} - \mathbf{f}\right) - \nabla\cdot\boldsymbol{\sigma}_1\right)\ , \tag{3.4}$$

and p' is the fine scale pressure,

$$p' = -\rho_1\tau_{\mathrm{C}}\nabla\cdot\mathbf{u}\ . \tag{3.5}$$

These clearly correspond to residuals of the strong momentum and continuity equations forming the incompressible Navier–Stokes system. The stabilization parameters τ_{M}, τ_{C}, and $\overline{\tau}$ are defined as

$$\begin{aligned}
\tau_{\mathrm{M}} &= \left(s \left(\frac{4}{\Delta t^2} + (\mathbf{u} - \hat{\mathbf{u}}) \cdot \mathbf{G}(\mathbf{u} - \hat{\mathbf{u}}) + C_I \left(\frac{\mu}{\rho_1} \right)^2 \mathbf{G} : \mathbf{G} \right) \right)^{-1/2} \ , && (3.6)\\
\tau_{\mathrm{C}} &= \left(\tau_{\mathrm{M}} \mathrm{tr} \mathbf{G} \right)^{-1} \ , && (3.7)\\
\overline{\tau} &= \left(\mathbf{u}' \cdot \mathbf{G} \mathbf{u}' \right)^{-1/2} \ , && (3.8)
\end{aligned}$$

where Δt is a timescale associated with the as-yet-unspecified temporal discretization, C_I is a dimensionless positive constant derived from element-wise inverse estimates [31, 53], and $\mathbf{G}$ generalizes the element diameter "h" to physical elements mapped through $\mathbf{x}(\boldsymbol{\xi})$ from a parametric parent element:

$$G_{ij} = \sum_{k=1}^{d} \frac{\partial \xi_k}{\partial x_i} \frac{\partial \xi_k}{\partial x_j} \ . \tag{3.9}$$

For a quasi-uniform shape-regular mesh with diam Ω^e asymptotically bounded above and below by h, we have $\mathbf{n} \cdot \mathbf{G} \mathbf{n} \sim h^{-2}$ for a unit vector $\mathbf{n}$. The factor s in the definition of τ_{M} is a dimensionless quantity that is allowed to vary in space. In most of Ω_1, $s = 1$, but, in an $\mathcal{O}(h)$ neighborhood of Γ, $s = s^{\text{shell}} \geq 1$. I introduced this factor in [105] to improve mass conservation near immersed boundaries. A theoretical motivation for this scaling is given in Section 5.4, and a numerical investigation of its effect is given in Section 6.4. In all computations presented in this dissertation, $s(\mathbf{x})$ is constructed to be in the pressure discrete space. Suppose this space is spanned by n basis functions

$\{N_i(\mathbf{x})\}_{i=1}^n$, forming a partition of unity for every $\mathbf{x} \in \Omega_1$ and having $\mathcal{O}(h)$-diameter supports. (This property is satisfied by, e.g., the standard B-spline or NURBS basis functions.) $s(\mathbf{x})$ is expressed as

$$s(\mathbf{x}) = \sum_{i=1}^{n} s_i N_i(\mathbf{x}) \ . \tag{3.10}$$

For $i \in \{1, \ldots, n\}$, $s_i = s^{\text{shell}}$ if the support of N_i intersects Γ and $s_i = 1$ otherwise. This construction implies that $s(\mathbf{x}) = s^{\text{shell}}$ over the entirety of each fluid element intersecting Γ and $s(\mathbf{x}) = 1$ everywhere outside of an $\mathcal{O}(h)$ region containing Γ.

Remark 3.1. The $\overline{\tau}$ term of B_1^{VMS} is not derived from VMS analysis; it is an extra term, introduced by Taylor et al. [174], to provide stabilizing dissipation near steep solution gradients.

3.1.2 Divergence conforming B-splines

As mentioned in the previous section, I found it necessary to scale the stabilization parameters of the VMS formulation in an unusual way to ensure mass conservation in immersed boundary computations. A way to totally eliminate mass loss and obtain pointwise divergence-free velocity solutions is to discretize the fluid in a so-called “structure-preserving”, “divergence-conforming”, or “div-conforming” manner, such that the divergence of every vector-valued function in the discrete velocity space is a member of the scalar-valued discrete pressure space. If this property is satisfied, then, assuming the Galerkin discrete problem is well-posed and a discrete velocity solution $\mathbf{u}_1^h$

exists, the weak continuity equation must be satisfied for $q^h = \nabla \cdot \mathbf{u}_1^h$:

$$\forall q^h \in \mathcal{V}_p^h \left((q^h, \nabla \cdot \mathbf{u}_1^h)_{L^2(\Omega_1)} = 0 \right) \quad \text{and} \quad \nabla \cdot \mathbf{u}_1^h \in \mathcal{V}_p^h \tag{3.11}$$

$$\Rightarrow (\nabla \cdot \mathbf{u}_1^h, \nabla \cdot \mathbf{u}_1^h)_{L^2(\Omega_1)} = 0 \tag{3.12}$$

$$\Rightarrow \|\nabla \cdot \mathbf{u}_1^h\|^2_{L^2(\Omega_1)} = 0 \tag{3.13}$$

$$\Rightarrow \nabla \cdot \mathbf{u}_1^h(\mathbf{x}) = 0 \qquad \text{for a.e. } \mathbf{x} \in \Omega_1 \ . \tag{3.14}$$

A discretization of this type was developed for Stokes and Navier–Stokes flows by Evans and Hughes [51, 54, 55]. Evans and Hughes used B-splines to construct velocity and pressure spaces satisfying the necessary properties, then directly posed the weak problem $B_1(\{\mathbf{u}_1^h, p^h\}, \{\mathbf{w}_1^h, q^h\}; \mathbf{0}) = F_1(\{\mathbf{w}_1^h, q^h\})$[2] over these discrete spaces (augmenting it with Nitsche's method to enforce no-slip boundary conditions).[3] A caveat to the above reasoning is that, to truly obtain velocities that conform to the incompressibility constraint, one would need to solve the discrete algebraic problem *exactly*, which is almost always impractical for real problems. I demonstrate in the 3D numerical examples of Sections 7.2 and 7.3, however, that the benefits of divergence-conforming discretizations are robust enough to persist through commonly-used approximations in the assembly and solution of the discrete problem.

As mentioned above, Evans and Hughes used Nitsche's method to en-

---

[2] In the discussion of div-conforming discretizations, I assume that Ω_1 is static, i.e. $\hat{\mathbf{u}} = \mathbf{0}$. There is probably no serious practical issue with inserting some reasonable mesh velocity, as it does not enter into the continuity equation, but I have not explored this possibility and some nice theoretical properties depend on having a solenoidal advection velocity.

[3] The presentation of Evans and Hughes formulates the advection term of B_1 differently, but this is easily shown to be equivalent to the present formulation, because $\nabla \cdot \mathbf{u}_1^h \equiv 0$.

force no-slip boundary conditions in a strongly consistent manner that allows for optimal convergence to sufficiently-regular solutions. For the immersogeometric computations of this paper, the regularity of the fluid velocity solution is always low (at most $H^{3/2-\epsilon}(\Omega_1)$) and I use, for simplicity,[4] a "naive" velocity penalization, i.e. I alter the problem to be

$$B_1(\{\mathbf{u}_1, p\}, \{\mathbf{w}_1, q\}; \mathbf{0}) + C_{\text{pen}} \int_{\Gamma_{\text{pen}}} (\mathbf{u}_1 - \mathbf{g}) \cdot \mathbf{w}_1 \, d\Gamma = F_1 (\{\mathbf{w}_1, q\}) \ , \quad (3.15)$$

where $C_{\text{pen}} > 0$ is a penalty parameter and $\mathbf{g}$ is the desired velocity on $\Gamma_{\text{pen}} \subset \partial\Omega_1$. If the normal component of the Dirichlet boundary condition is strongly enforced (i.e. built directly into the spaces in which $\mathbf{u}_1$ and $\mathbf{w}_1$ live), the formulation (3.15) can be used unaltered to penalize just the tangential portion of the boundary condition. If the penalty constant C_{pen} scales like μ/h, then this remains weakly consistent with the Navier–Stokes problem.

Div-conforming B-splines for incompressible flow are a specific application of a narrow subset of discrete de Rahm complexes. This is an important topic not only for approximation of incompressible flows but for computational electromagnetics and magnetohydrodynamics as well. In this dissertation, I will focus exclusively on the application to incompressible flow, with an eye toward implementation. Readers interested in generalizations and theoretical aspects should refer to [4] for a discussion of discrete exterior calculus and [33, 34] for its development within IGA.

---

[4] Actually, the consistency of Nitsche's method relies on having an exact velocity solution in $H^{3/2+\epsilon}(\Omega_1)$ (cf. [51, Section 7.2]), which is marginally more regular than we would expect from an immersed boundary approach.

3.1.2.1 Construction for rectangular domains

Suppose, for now, that Ω_1 is an axis-aligned d-dimensional rectangle. Then physical space can serve directly as a d-variate B-spline parameter space.[5] Define a d-variate scalar B-spline space for the pressure on Ω_1. Then, for $1 \leq i \leq d$, we can k-refine the pressure space once in the i^{th} parametric direction to obtain a scalar space for the i^{th} Cartesian velocity component. Due to well-known properties of B-splines under differentiation [134], the i^{th} partial derivative of the i^{th} velocity component will then be in the pressure space. The scalar basis functions of the d velocity component spaces can be multiplied by their respective unit vectors to obtain a vector-valued basis for the discrete velocity space. The divergence of a vector-valued velocity function will therefore be a sum of d scalar functions in the pressure space.

A statement of all possible structure-preserving B-spline spaces can (after defining the appropriate notation) be written in a compact formula, as in [51, Section 5.2]. In the notation of the cited reference, the velocity space is $\widehat{\mathcal{RT}}_h$ and the pressure space is $\widehat{\mathcal{W}}_h$. Following the terminology of [51], if the pressure space has polynomial degree k' in all directions, the entire pressure and velocity discretization is said to be of degree k', despite the presence of $(k'+1)$-degree splines in the velocity component spaces.

To clarify the construction, I spell out an example of degree $k' = 1$.

---

[5] For readers unfamiliar with the construction and basic properties of B-splines, a comprehensive explanation can be found in [134]. The aspects essential to understanding IGA are reviewed concisely in [90].

Suppose $d = 2$ and the pressure space has degree one ($= k'$) in both the x_1 and x_2 directions. Its (open) knot vectors in the x_1 and x_2 directions are both

$$(1, 1, 2, 3, 4, 5, 6, 7, 7)\ . \tag{3.16}$$

Then the scalar B-spline space $\mathcal{S}_{u_1}$ for the x_1 component of $\mathbf{u}_1$ would have knot vector

$$(1, 1, 1, 2, 3, 4, 5, 6, 7, 7, 7) \tag{3.17}$$

and degree two ($= k' + 1$) in the x_1 direction and knot vector

$$(1, 1, 2, 3, 4, 5, 6, 7, 7) \tag{3.18}$$

and degree one ($= k'$) in the x_2 direction. The partial derivative $\frac{\partial}{\partial x_1}$ maps functions from this space to the pressure space. Vector-valued basis functions for the velocity are obtained by multiplying the scalar basis functions of $\mathcal{S}_{u_1}$ by the unit vector $\mathbf{e}_1$. Similarly, the space $\mathcal{S}_{u_2}$ for the x_2 component of $\mathbf{u}_1$ would have knot vector

$$(1, 1, 2, 3, 4, 5, 6, 7, 7) \tag{3.19}$$

and degree one in the x_1 direction and knot vector

$$(1, 1, 1, 2, 3, 4, 5, 6, 7, 7, 7) \tag{3.20}$$

and degree two in the x_2 direction. The corresponding vector-valued velocity basis functions are obtained by multiplying scalar basis functions of S_{u_2} by $\mathbf{e}_2$. The extensions to higher polynomial orders, more space dimensions, different knot multiplicities, periodic domains, and so on should be straightforward.

3.1.2.2 Generalization to non-rectangular domains

Div-conforming B-splines are not limited to rectangular domains. A point $\mathbf{X}$ in a rectangular parametric domain $\widehat{\Omega}$ may be mapped to a point $\mathbf{x}$ in a non-rectangular physical domain Ω, using a motion $\mathbf{x} = \boldsymbol{\phi}(\mathbf{X})$. (The regularity requirements of this mapping are given in [51, Section 4.3], with some differences in notation.) To obtain divergence-conforming velocity and pressure spaces on the physical domain Ω, vector-valued velocity basis functions defined on $\widehat{\Omega}$ can be pushed forward using the Piola transform. For an arbitrary parametric-space velocity function $\widehat{\mathbf{u}}$, its physical-space counterpart $\mathbf{u}$ is

$$\mathbf{u}(\mathbf{x}) = \frac{1}{J(\mathbf{X})}\mathbf{F}(\mathbf{X})\widehat{\mathbf{u}}(\mathbf{X}) \ , \tag{3.21}$$

where, using Cartesian index notation [81, Section 1.1] and symbols analogous to those frequently seen in nonlinear elasticity, $\mathbf{F}$ is the deformation gradient of $\boldsymbol{\phi}$, viz.

$$\mathbf{F} = \frac{\partial \boldsymbol{\phi}}{\partial \mathbf{X}} \qquad \Longleftrightarrow \qquad F_{iJ} = \frac{\partial \phi_i}{\partial X_J} = \phi_{i,J} \ , \tag{3.22}$$

and J is the determinant of $\mathbf{F}$. Using Nanson's formula [81, (2.54)] and integration by parts, this implies the well-known Piola identity

$$\mathrm{div}\ \mathbf{u} = \frac{1}{J}\mathrm{DIV}\ \widehat{\mathbf{u}} \ , \tag{3.23}$$

where

$$\mathrm{div}\ \mathbf{u} = \frac{\partial u_j}{\partial x_j} = u_{j,j} \quad \text{and} \quad \mathrm{DIV}\ \widehat{\mathbf{u}} = \frac{\partial \widehat{u}_B}{\partial X_B} = \widehat{u}_{B,B} \ . \tag{3.24}$$

(For readers unfamiliar with this identity, it also follows easily from (3.27), derived below.) To ensure pointwise divergence-free velocity solutions, we would like the divergence of every pushed-forward velocity function to exist in the pushed-forward pressure space. Recall that, for every $\widehat{\mathbf{u}}$ in the parametric velocity space, there exists $\widehat{q}$ in the parametric pressure space such that $\widehat{q} =$ DIV $\widehat{\mathbf{u}}$. Then, in view of (3.23), the parametric pressure space function should be pushed forward by

$$q(\mathbf{x}) = \frac{1}{J(\mathbf{X})}\widehat{q}(\mathbf{X}) \ , \tag{3.25}$$

so that $q =$ div $\mathbf{u}$ and the argument (3.11)–(3.14) remains valid.

The weak Navier–Stokes equations involve the spatial gradient of the pushed-forward velocity $\mathbf{u}$, viz. $u_{i,j}$. Given the B-spline control point values for components of $\widehat{\mathbf{u}}$, it is only immediately straightforward to evaluate $\widehat{u}_{A,B}$. I derive here a formula for the physical velocity gradient in terms of the $\mathbf{X}$-derivatives of $\widehat{\mathbf{u}}(\mathbf{X})$ and $\boldsymbol{\phi}(\mathbf{X})$. I use the identity

$$\frac{\partial J}{\partial F_{iA}} = J F^{-1}_{Ai} \ , \tag{3.26}$$

which is equivalent to [81, (1.241)]. Underlines and colors are purely for visual clarity.

$$\begin{aligned}
u_{i,j} &= \left(\frac{1}{J} F_{iA} \hat{u}_A\right)_{,j} \\
&= \left\{ \left(\frac{1}{J}\right)_{,j} F_{iA} + \frac{1}{J} F_{iA,j} \right\} \hat{u}_A + \frac{1}{J} F_{iA} \hat{u}_{A,j} \\
&= \left\{ \left(\frac{-1}{J^2} \frac{\partial J}{\partial F_{\ell B}} \frac{\partial F_{\ell B}}{\partial X_C} \frac{\partial X_C}{\partial x_j} \right) F_{iA} + \frac{1}{J} \frac{\partial F_{iA}}{\partial X_C} \frac{\partial X_C}{x_j} \right\} \hat{u}_A + \frac{1}{J} F_{iA} \frac{\partial \hat{u}_A}{\partial X_C} \frac{\partial X_C}{\partial x_j} \\
&= \left\{ \frac{-1}{J^2} \frac{\partial J}{\partial F_{\ell B}} F_{\ell B,C} F^{-1}_{Cj} F_{iA} + \frac{1}{J} F_{iA,C} F^{-1}_{Cj} \right\} \hat{u}_A + \frac{1}{J} F_{iA} \hat{u}_{A,C} F^{-1}_{Cj} \\
&= \left\{ \frac{-1}{\underline{\underline{J}}} F^{-1}_{B\ell} F_{\ell B,C} \underline{F^{-1}_{Cj}} F_{iA} + \frac{1}{\underline{\underline{J}}} F_{iA,C} \underline{F^{-1}_{Cj}} \right\} \hat{u}_A + \frac{1}{\underline{\underline{J}}} F_{iA} \hat{u}_{A,C} \underline{F^{-1}_{Cj}} \\
&= \frac{1}{\underline{\underline{J}}} \left(\left\{ -F^{-1}_{B\ell} F_{\ell B,C} F_{iA} + F_{iA,C} \right\} \hat{u}_A + F_{iA} \hat{u}_{A,C} \right) \underline{F^{-1}_{Cj}} \ . \qquad (3.27)
\end{aligned}$$

By substituting j for i and invoking the symmetry of $\boldsymbol{\phi}$'s Hessian, one may easily derive (3.23) from the last line of (3.27).

Div-confomring B-splines may be used on wider classes of geometries by joining deformed rectangular meshes together with a discontinuous Galerkin approach, as described in [51, Section 6.5], but this possibility is not exploited in the present work.

3.1.2.3 Stabilization of advection

As discussed while introducing the VMS discretization in Section 3.1.1, Galerkin's method is not necessarily stable for practically-coarse discretizations of high-Reynolds-number flows. The Galerkin discretization used by Evans and Hughes can be straightforwardly augmented to include SUPG stabilization [32] (but *without* its frequent accomplice pressure stabilizing Petrov–

Galerkin (PSPG) [92], which would clearly disrupt the pointwise mass conservation). However, the appearance of the pressure gradient in the momentum equation residual removes the property of the Galerkin approximation that the error in the velocity solution is independent of pressure interpolation error [54, (6.32)]. This property is valuable in the presence of immersed boundaries that induce large discontinuities in the exact pressure solution. In the computations of this dissertation, I stabilize div-conforming discretizations in a weakly-consistent manner, by introducing an $\mathcal{O}(h)$ streamline diffusion. Specifically, I add[6]

$$+\sum_e (\tau\rho_1 \mathbf{u}_1^h \cdot \nabla \mathbf{u}_1^h, \mathbf{u}_1^h \cdot \nabla \mathbf{w}_1^h)_{L^2(\Omega^e)} \tag{3.28}$$

to $B_1(\{\mathbf{w}_1^h, q^h\}, \{\mathbf{u}_1^h, p^h\})$, where $\{\Omega^e\}_{e=1}^{N_\text{el}}$ are the N_el Bézier elements of the B-spline mesh and

$$\tau = \begin{cases} \left(\mathbf{u}_1^h \cdot \mathbf{G}\mathbf{u}_1^h\right)^{-1/2} & \mathbf{u}_1^h \cdot \mathbf{G}\mathbf{u}_1^h > 0 \\ 0 & \text{otherwise} \end{cases} . \tag{3.29}$$

The components of $\mathbf{G}$ are again defined by (3.9), with $\boldsymbol{\xi} \in (-1,1)^d$ a normalized parametric coordinate in Ω^e. While this form of stabilization is only weakly consistent, we do not expect high order convergence rates from immersed boundary discretizations (of the type developed in this dissertation), due to low-order interpolation errors. The stabilization term (3.28) should therefore not harm asymptotic convergence rates. Further, the artificial diffusion acts only in the flow direction, and is minimally disruptive to laminar solutions.

---

[6] I assume here that $\hat{\mathbf{u}} = \mathbf{0}$, as in my statement of the problem to be solved using div-conforming B-splines.

3.1.3 Comparison of VMS and div-conforming B-splines

In this section, I list the advantages and disadvantages of the VMS and div-conforming approaches. The advantages of VMS are as follows:

- The discrete formulation is stable for arbitrary combinations of pressure and velocity spaces. This includes the "equal-order" discretization family, in which the pressure and each Cartesian component of the velocity are in the same scalar discrete space. This choice allows for greater optimization of CFD codes than would be possible for inf-sup-stable pressure–velocity pairs. In principle, one could stably apply VMS to the div-conforming B-spline spaces, although the ∇q^h–∇p^h interaction would spoil the mass conservation.
- The formulation has been demonstrated to be an effective turbulence model over a wide range of Reynolds numbers, while retaining high-order accuracy in the limit of direct numerical simulation.
- The effective application of this formulation to a wide variety of challenging FSI problems is thoroughly documented in the computational mechanics literature.

The disadvantages are:

- The absence of strong mass conservation leads to severe mass loss near immersed boundaries, unless the stabilization constants are scaled by an *ad hoc* factor. Improving the mass conservation in this way requires some degree of empirical tuning and harms the conditioning of the discrete problem. Further, it may reduce accuracy in some situations. It should be noted that this is not a unique disadvantage of the VMS formulation, as other non-div-conforming discretizations suffer in the presence of large pressure gradients. See, e.g., [59], in which the authors resorted to drastic scaling of grad-div stabilization to prevent catastrophic mass loss.

The advantages of the div-confomring B-spline approach are:

- No special tuning is required to obtain satisfactory mass conservation.
- Errors in laminar solutions are independent of Reynolds number [51, Section 7.6.2].
- The error in the velocity field is independent of the pressure interpolation error [54, (6.32)]. In particular, when the exact pressure solution contains a large discontinuity cutting through element interiors, such as the pressure jump across a thin immersed structure, this does not affect the accuracy of the velocity solution.

The disadvantages are:

- The best way to stabilize advection is not currently clear; the naive use of SUPG will not interfere with the pointwise mass conservation, but it eliminates the advantage of having velocity error estimates that are independent of pressure interpolation. The stabilization (3.28) that I have used in this dissertation is practically effective, but only weakly consistent and cannot achieve high-order accuracy.

- The technology is relatively new, so there is very little documented experience applying it to challenging problems such turbulent flow and FSI.

- The saddle-point structure of the discrete problem following from implicit time discretization makes it more difficult to solve than the discrete problem emanating from VMS.

- The requisite tensor product structure of the B-spline mesh places a major constraint on mesh generation.

In light of the above, neither discretization of the fluid subproblem stands out as clearly superior and I feel that the best approach is to study both. At the time of writing, I find my implementation using VMS to be a more practical solution, but most of the difficulties with div-conforming B-splines are tied to their novelty, which, by definition, will wear off over time.

3.2 Structure subproblem

Setting $\mathbf{w}_1 = \delta\boldsymbol{\lambda} = \mathbf{0}$ isolates the structure subproblem (2.3), in which $\mathbf{u}_2$ and $\boldsymbol{\lambda}$ are considered prescribed data.

3.2.1 Isogeometric spline discretization

This subproblem can be stably discretized using a straightforward conforming Galerkin method. However, this has rarely been done prior to the introduction of IGA, because, for $B_2(\mathbf{y}, \mathbf{w})$ to remain bounded, $\mathbf{y}$ and $\mathbf{w}$ need to be in $H^2(\Gamma)$; the formulation involves $L^2(\Gamma)$ inner products of second derivatives of test and trial functions. This essentially corresponds to the constraint that discrete spaces be subsets of $C^1(\Gamma)$.[7] Traditional finite element spaces do not satisfy this constraint. However, the spline spaces used in IGA can be made as smooth as the geometry allows. The geometry of typical BHV leaflets can be faithfully modeled by smooth, single-patch B-spline surfaces, so, for the purposes of this dissertation, the semidiscrete structure subproblem amounts to choosing $\mathcal{V}_y^h$ to be a smooth B-spline space and adding superscript hs to the test and trial functions of (2.3). The implementation of such discretizations for arbitrary hyperelastic constitutive laws is documented exhaustively by Kiendl et al. [111].

3.2.2 Linearization by automatic differentiation

To apply a Newton (or Newton-based) iteration to the system of nonlinear algebraic equations emanating from the discretized structure subproblem, one must compute partial derivatives of the discrete structure subproblem residual with respect to the coefficient of each basis function of the discrete

---

[7] This is not strictly mandated by the Sobolev embedding theorem unless the dimension of Γ is one (cf. [2, Theorem 5.4, Case C], with $j = 1$, $p = 2$, and $m = 1$), but any practical discrete subspace of $H^2(\Gamma)$ composed of piecewise polynomial functions will be smooth.

trial solution space. These derivatives will depend on the material model used, i.e. the functional dependence of **S** on **E** in the variational forms given in Section 2.3. Complex material models may, in general, have complicated algorithms (with conditionals, loops, and so on) relating **S** to **E**, rather than simple formulas. Computing the necessary derivatives of such models for all possible cases may involve significant labor, slowing down the pace of development and introducing possibilities for programming errors. Fortunately, however, the computation of derivatives can be automated. For materials other than the St. Venant–Kirchhoff material (for which a verified implementation was already available to me), I employ *automatic differentiation* to compute material stiffness contributions to the structure subproblem residual Jacobian matrix. Automatic differentiation is not a novel concept, and has previously been applied in the context of computational mechanics [115, 116]. However, the technology is not widely understood, so I provide an overview here.

3.2.2.1 The concept of automatic differentiation

Automatic differentiation is most easily understood through examples. I will therefore walk through a minimal Fortran implementation, in lieu of stating a formal definition. (More comprehensive explanations can be found in [115, 116].) Consider the following code snippets to be part of a Fortan module called `autodiff`. First, within the module `autodiff`, define a derived type, the "augmented real number", abbreviated "`areal`":

```
type areal
   real(8) :: x
```

```
    real(8) :: xprime
  end type areal
```

The goal is to render this type interchangeable with the standard double-precision floating point type `real(8)`, by overloading all of the relevant operations. The attribute `x` is the value of the augmented real number. The additional attribute `xprime` stores a partial derivative with respect to some other augmented real number (evaluated at a particular point). To clarify what I mean by this, I will demonstrate how to overload the addition operator, `+`. The `+` operator can tied to a procedure `arplus` adding augmented reals

```
  interface operator(+)
     module procedure arplus
  end interface
```

which is implemented

```
    elemental function arplus(a,b) result(c)
      implicit none
      type(areal), intent(in) :: a,b
      type(areal) :: c
      c%x = a%x + b%x
      c%xprime = a%xprime + b%xprime
    end function arplus
```

Suppose, then, that `x`, `y`, and `z` are `areal`s corresponding to real-valued variables x, y, and z. Suppose that, intially, `x%xprime=1.0d0` and `y%xprime=0.0d0`. Then, if we assign `z = x + y`, `z%x` will store the sum of x and y, while `z%xprime` will be equal to $\partial z/\partial x$, evaluated at the point ($x =$`x%x`$, y =$`y%x`). To obtain $\partial z/\partial y$, one would instead initialize the `xprime` attributes of `x` and `y` like `x%xprime=0.0d0` and `y%xprime=1.0d0` before executing `z = x + y`. It is straightforward to see, then, how this concept could be extended to subtraction, multiplication, negation, transcendental functions, vectorized operations, matrix intrinsics, operations with mixed arguement types (e.g. `integer`s added to `areal`s, `real(8)`s multiplied by `areal`s, etc.), assignment to and from other numeric types, and so on. As a reinforcing example, consider the interface for overloading `exp`,

```
interface exp
   module procedure arexp
end interface
```

and its corresponding implementation

```
elemental function arexp(a) result(b)
  type(areal), intent(in) :: a
  type(areal) :: b
  b%x = exp(a%x)
  b%xprime = a%xprime * exp(a%x)
end function arexp
```

Thus, for any routine with numerical input and output parameters of type `areal`, the partial derivatives of the ouptut variables with respect to input variable `x` can be readily obtained by setting `x%xprime=1.0d0`, setting all of the other input variables' `xprime` members to zero, running the algorithm, and querying the `xprime` attributes of the output variables. If a sufficiently exhaustive `autodiff` module is implemented, code from subroutines written around `real(8)`s can be re-used verbatim. The operations contributing the the output can be in loops, conditionals, etc. Implementation would be similar in any language which allows operator overloading, such as C++ or Python.

3.2.2.2 Application to computing residual Jacobians

From the above, it is clear that using automatic differentiation to compute all entries of a dense Jacobian of an N-dimensional N-variate function would cost N function evaluations. Assembling all N entries of a finite element residual once for each of N degrees of freedom would be intractable. However, the sparse structure of the problem can be exploited to make automatic differentiation a practical technology for computing tangent matrices.

A finite element residual is a sum of contributions from elements. Each such element contains only $m \ll N$ nonzero entries that depend on m input variables. These contributions can be viewed as m-dimensional m-variate functions (or "element right-hand side vectors", in common parlance). For a given element type, the number of degrees of freedom per element, m, is independent of the residual dimension, N. Thus the cost of assembling the m-dimensional residual m times to automatically compute the corresponding nonzero Jacobian contributions (or the "element left-hand side matrix") is independent of the problem size.

This method of computing the residual Jacobian is less efficient than hard-coding judicious approximations of element Jacobians. In my experience, the use of automatic differentiation increases the cost of assembly by a factor of about 5 to 10 in typical situations.[8] In the context of immersed fluid–thin

---

[8] This is based on an operator-overloading implementation, as suggested by the explanation in Section 3.2.2.1. Automatic differentiation can also be implemented as a source code transformation, which usually yields improved performance [115, 116].

structure interaction, though, the additional cost of using this method in the structure subproblem is barely noticeable, since the lower-dimensional structure subproblem demands only a small fraction of the total computation and the straightforward parallelization of element-by-element assembly procedures allows this cost to be easily spread over the (typically many) processors used for the fluid subproblem. I therefore prefer automatic differentiation in this context, due to its versatility. While I employ only relatively simple constitutive models for BHV leaflets in this dissertation, a variety of more baroque models can be found in the biomedical literature and a practical framework for BHV FSI analysis should anticipate their use.

3.2.3 Contact

As mentioned in Chapter 2, fluid–structure kinematics prevent structural self-intersection in the exact solution to the mathematical model of FSI. As two disjoint portions of the fluid–structure interface approach one another, the incompressible fluid in between will provide a lubrication effect and prevent complete contact. This phenomenon has been exploited in boundary-element discretizations of inflatable shell structures interacting with Stokes flow [192]. The asymptotic behavior of squeeze flow between smooth surfaces can be estimated from the analytical results of Rukmani and Usha [142]. However, simulations performed at finite resolution with weakly-enforced fluid–structure kinematic conditions may permit structural self-intersection in the approximate solution. I therefore incorporate a penalty-based contact method into the semi-discrete structure subproblem to prevent pre-asymptotic self-intersections from becoming excessive.

3.2.3.1 Formulation of contact penalization

The leaflets are modeled geometrically as surfaces of co-dimension one to $\mathbb{R}^d$, so I must first clarify what is meant by penetration. A lower-dimensional surface has no interior in which to detect penetrating geometry. However, a BHV leaflet, operating under normal conditions, will contact other leaflets on only one side, motivating the following definition of penetration.

Consider leaflets S_1 and S_2 to be smooth parametric surfaces in $\mathbb{R}^3$. For $\mathbf{x}_1 \in S_1$, with surface normal $\mathbf{n}_1$ determining the side on which contact will occur, $\mathbf{x}_1$ is said to contact leaflet S_2 if the following conditions are met:

1. There exists a point $\mathbf{x}_2 \in S_2$ with normal $\mathbf{n}_2$ such that $(\mathbf{x}_1 - \mathbf{x}_2)$ is perpendicular to S_2. I refer to $\mathbf{x}_2$ as the closest point on S_2 to $\mathbf{x}_1$, but, without additional assumptions on S_2, the defining conditions guarantee neither that $\mathbf{x}_2$ is unique nor that it minimizes the Euclidean distance $\|\mathbf{x}_1 - \mathbf{x}_2\|_{\ell^2}$. In practice, $\mathbf{x}_2$ is determined by iteratively solving the nonlinear problem of finding $\boldsymbol{\xi} = (\xi_1, \xi_2)$ in the parameter space for S_2 such that

$$\left\{ \begin{array}{rcl} (\mathbf{x}_1 - \mathbf{x}_2(\boldsymbol{\xi})) \cdot \dfrac{\partial \mathbf{x}_2(\boldsymbol{\xi})}{\partial \xi_1} & = & 0 \\ \\ (\mathbf{x}_1 - \mathbf{x}_2(\boldsymbol{\xi})) \cdot \dfrac{\partial \mathbf{x}_2(\boldsymbol{\xi})}{\partial \xi_2} & = & 0 \end{array} \right. . \tag{3.30}$$

2. $\|\mathbf{x}_1 - \mathbf{x}_2\|_{\ell^2} < c$, where $c > 0$ is a parameter chosen to avoid false positive contact of distant geometry. I assume that penalties will be strong enough to prevent penetrations larger than c.

For a contacting point $\mathbf{x}_1$, its signed penetration is defined as $d = (\mathbf{x}_2 - \mathbf{x}_1) \cdot \mathbf{n}_2$. $\mathbf{x}_1$ is said to penetrate S_2 if $d > -h_\mathrm{c}$, where $c > h_\mathrm{c} \geq 0$ indicates a minimum desired distance between the contacting sides of S_1 and S_2. When $d > 0$, I add the condition that $|\mathbf{n}_1 \cdot \mathbf{n}_2| > \alpha$, for some $0 \leq \alpha < 1$. Choosing $\alpha > 0$ allows a hinge-like boundary between S_1 and S_2 that can open through angles larger than 270° without immediately incurring a contact penalty. This notation is illustrated for a pair of contacting points in Figure 3.1.

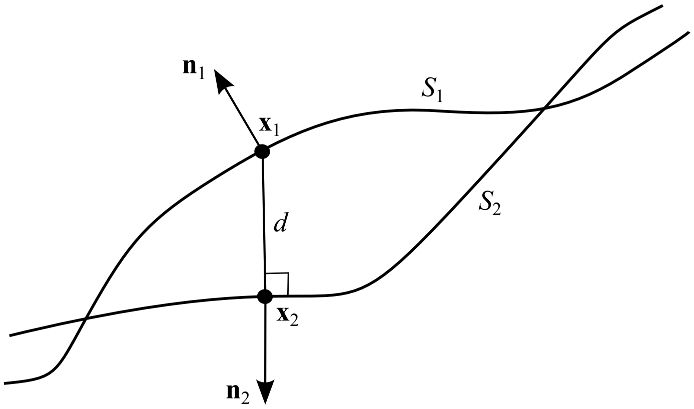


Figure 3.1: Illustration of contact notation.

Non-penetration is enforced weakly, by penalizing $d > -h_\mathrm{c}$. To motivate the spatially-discrete contact algorithm, consider adding the following term

$$+\int_{S_1} \left(\left((\mathbf{w}_2)^2 - (\mathbf{w}_2)^1\right) \cdot \mathbf{n}_2\right) \left(k d^+\right) \, d\Gamma \tag{3.31}$$

to the left-hand side of (2.3). This term tests a penetration residual against a difference of weighting functions, $(\mathbf{w}_2)^1$ and $(\mathbf{w}_2)^2$, where $(\mathbf{w}_2)^i$ is the structure weighting function restricted to surface S_i. The addition of the term (3.31) is not a rigorous formulation because the change-of-variables to integrate $(\mathbf{w}_2)^2$ over S_1 is not precisely defined and the definition of d is ambiguous. With some regularity assumptions on S_1 and S_2, and c sufficiently small, we could treat the leaflets as smooth manifolds and use the tubular neighborhood theorem of differential geometry to assert the existence of a well-behaved mapping between contacting regions, but I do not have a constructive estimate for the bound on c, and will instead disambiguate the formulation in an *ad hoc* manner, by simply detailing my discrete implementation below.

To assemble the contact algorithm's contribution to the structure subproblem's nonlinear residual, test for penetration and apply penalty forces at a discrete set of contact points, $\{\mathbf{x}_1^1, \dots, \mathbf{x}_1^n\} \subset S_1$, in accordance with the definitions given above. For the subset $\{\mathbf{x}_1^{j_k}\}$ of these points contacting $\{\mathbf{x}_2^{j_k}\} \subset S_2$, opposing forces on S_2 must be added to conserve linear momentum. To conserve angular momentum, the contact forces between $\mathbf{x}_1$ and $\mathbf{x}_2$ are along their separation $\mathbf{x}_1 - \mathbf{x}_2$, which is, by construction, parallel to $\mathbf{n}_2$. The force on $\mathbf{x}_1$ is $\mathbf{f}_1 = -w(P_k(d))\mathbf{n}_2$ and the force on $\mathbf{x}_2$ is $\mathbf{f}_2 = -\mathbf{f}_1$, where w is a weight associated to $\mathbf{x}_1$ and $P_k(d)$ penalizes penetration. I define the penalty function as follows:

$$P_k(d) = \begin{cases} \frac{k}{2h_\mathrm{c}}(d + h_\mathrm{c})^2 & , \quad d \in (-h_\mathrm{c}, 0) \\ kh_\mathrm{c}/2 + kd & , \quad d \geq 0 \\ 0 & , \quad \text{otherwise} \end{cases} , \qquad (3.32)$$

where k decides the strength of the position penalty. The behavior of P_k on the interval $-h_{\mathrm{c}} < d < 0$, illustrated in Figure 3.2, ensures that the penalty activates smoothly as contact begins. This smoothing greatly improves the nonlinear convergence of the structure subproblem residual. Motivated by (3.31), I choose $\{\mathbf{x}_1^j\}$ to be Gaussian integration points on elements of S_1 and weight forces using the corresponding integration rule. Based on dimensional analysis and asymptotic considerations, one can estimate reasonable contact parameters

$$k = c_1 E/h \tag{3.33}$$

$$h_{\mathrm{c}} = c_2 h \tag{3.34}$$

where h is a measure of the structural element size, E is an estimate of the material stiffness, and c_1 and c_2 are modest dimensionless constants. Throughout this work, however, I simply use uniform values of k and h_{c} that I determined to be effective through numerical experiments.

The above method does not preserve geometrical symmetries. To see this, consider contacting planes at an angle; the directions of contact forces depend on the choice of S_1 and S_2, as shown in Figure 3.3. To ensure that results are independent of this arbitrary distinction, one must compute forces with both choices and sum the results.

Remark 3.2. In the terminology of Sauer and De Lorenzis [147], this method of symmetrizing the contact forces is a classical two-pass contact algorithm. One could alternatively consider omitting the forces on $\{\mathbf{x}_2^{j_k}\}$ during each

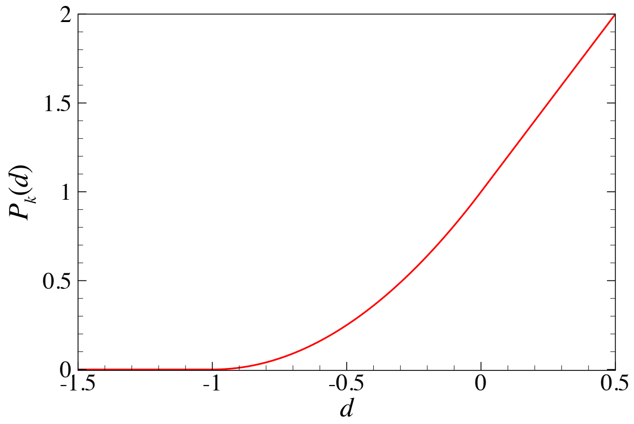


Figure 3.2: Illustration of the function $P_k(d)$ for $k = 2$ and $h_c = 1$.

application of the contact algorithm, which would correspond to the double half-pass technique proposed by Sauer and De Lorenzis. This does not, in general, enforce momentum balance, but Sauer and De Lorenzis found the double half-pass algorithm to be more stable and computationally efficient, while recovering momentum balance to high accuracy at reasonable levels of refinement.

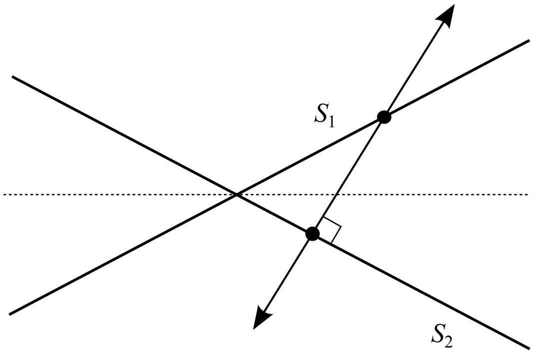


Figure 3.3: Symmetrical geometry results in asymmetrical contact forces.

3.2.3.2 Approximate linearization

The nonlinear residual assembled by the above procedure is not generally differentiable with respect to the structure's displacement field, so the applicability of typical Newton or quasi-Newton iteration to (or even well-posedness of) the nonlinear algebraic problems arising from implicit time discretization of the resulting semi-discrete structure subproblem is questionable. However, for sufficiently- (and in the context of BHV dynamics, not excessively-) small time steps, an approximate linearization of the contact forces is sufficient to stably execute predictor–multi-corrector methods based on Newton iteration. In particular, when approximating the linearizations of contact forces of the form

$$\mathbf{f}_1 = -\mathbf{f}_2 = -wP_k((\mathbf{x}_2(\boldsymbol{\xi}) - \mathbf{x}_1) \cdot \mathbf{n}_2)\mathbf{n}_2 \ , \tag{3.35}$$

I ignore the dependencies of w, $\boldsymbol{\xi}$, and $\mathbf{n}_2$ on the shell structure midsurface displacement. This results in symmetric contributions to the approximate Jacobian of the structure subproblem residual, which allows for the use of optimized iterative linear solvers for the tangent system of the structure subproblem.

3.2.3.3 Searching for contacting points

The application of Newton iteration to solve problem (3.30) can run into a number of difficulties:

- Depending on the starting point, the iteration could converge to a solution with $\|\mathbf{x}_1 - \mathbf{x}_2\|_{\ell^2} \geq c$, despite the existence of a solution with $\|\mathbf{x}_1 - \mathbf{x}_2\|_{\ell^2} < c$. This would cause a contacting point to be spuriously ignored.

- S_2 may not be mapped to its parameter space by a single smooth coordinate chart as in the simple case of a B-spline or NURBS patch. For instance, S_2 might be given as a collection of many Bézier elements, as would be the typical representation of a more complicated spline structure, such as a T-spline, in an analysis code. Attempting to solve (3.30) on each element would be prohibitively expensive.

- Even if S_2 has a single coordinate chart and a unique closest point to $\mathbf{x}_1$, the iterative solution of (3.30) would benefit from an informed initial guess.

It is therefore desirable to be able to rapidly obtain an accurate initial guess of $\boldsymbol{\xi}$, to expedite the process of solving (3.30). I developed a method to do this based on spatial hashing (also sometimes referred to as bucket sorting in the computational mechanics literature).

The problem of finding a starting iterate on S_2 can be given abstractly as follows: Given $\mathbf{x}_1 \in S_1$ and $\{\mathbf{x}_2^i\}_{i=1}^{N_p} \subset S_2$ with known parametric coordinates, distributed roughly evenly over S_2, find the closest $\mathbf{x}_2^i$ to $\mathbf{x}_1$ such that $\|\mathbf{x}_1 - \mathbf{x}_2^i\|_{\ell^2} < C$, for some distance $C \geq c$, if such a point exists. A natural choice of $\{\mathbf{x}_2^i\}$ is the set of quadrature points used to integrate the structure subproblem over S_2. Setting $C = c$ (i.e. the cutoff used in the definition of contact) can potentially fail to find a starting iterate even though $\mathbf{x}_1$ is in contact with S_2 (in the sense defined above). The absence of any point from the finite set $\{\mathbf{x}_2^i\} \subset S_2$ that is closer than c to $\mathbf{x}_1$ does not imply that there does not exist $\mathbf{x}_2 \in S_2$ such that $\mathbf{x}_1$ is in contact with $\mathbf{x}_2$. In practice, C should therefore be somewhat larger than c. I have not endeavored to derive precise sufficient conditions for guaranteeing that the solution of this problem will identify a starting iterate whenever $\mathbf{x}_1$ is in contact with S_2, but assessing $\mathbf{x}_1$'s potential for contacting S_2 based on its proximity to elements of $\{\mathbf{x}_2^i\}$ seems to be practically effective whenever $C \geq c$.

To solve the problem stated in the previous paragraph, it is clearly unnecessary to compare any points that are further apart from one another than C. I therefore sort the elements of $\{\mathbf{x}_2^i\}$ into cells of a uniform d-dimensional grid of N_c cells $\{\omega_i\}_{i=1}^{N_c}$ covering S_2, then compare only the $\mathbf{x}_2^i$s in the same cell as $\mathbf{x}_1$, or adjacent cells. If the cells of the grid have edge lengths greater than C, this procedure will make all of the necessary comparisons. While conceptually simple, this engenders an implementational challenge: the fact that S_2 is of a lower dimension than the grid covering it means that the vast majority of the N_c grid cells are unoccupied. N_c will typically be quite large, so operations of cost $\Theta(N_c)$ should be avoided. It is therefore grossly inefficient to iterate over all elements of $\{\omega_i\}$. Aside from allocating memory for the grid cells while initializing the data structure, operations to re-sort the points $\{\mathbf{x}_2^i\}$ into new cells as S_2 deforms should only iterate over occupied cells, of which there can be at most N_p. If quadrature points are used for $\{\mathbf{x}_2^i\}$, then operations of cost $\Theta(N_p)$ are necessarily acceptable. (Otherwise, integrating the shell structure subproblem would be considered unacceptably costly, even without any contact.)

To describe the algorithm that I used to accomplish this task, I must first introduce the necessary data structures:

- L is an array of N_p integers. This will be used to store linked lists of indexes of points on S_2. Because points fall within unique grid cells, these lists can all fit within a single array of size N_p. The storage and traversal of these lists will be clarified below.
- S is a stack of integers. This will be used to keep track of the indexes of occupied grid cells. S is initialized to be empty.
- O is an array of N_c boolean variables. $O(i)$[9] indicates whether or not grid cell i is occupied. O is initialized to false once, when it is first allocated, costing $\Theta(N_c)$ operations.
- H is an array of N_c integers. If $O(i)$ is true, then $H(i)$ indicates the head of a linked list of indexes of points from the set $\{\mathbf{x}_2^j\}$ contained in grid cell ω_i. The array H is initialized to zero once, when it is first allocated, costing $\Theta(N_c)$ operations.

The procedure for sorting elements of $\{\mathbf{x}_2^i\}$ into grid cells $\{\omega_i\}$ after S_2 changes its configuration can be divided into two phases: purge the data structures of

---

[9] N.b. that this is distinguished from "big $\mathcal{O}$" asymptotic notation by font.

information about the previous configuration, then populate the data structures with information about the new configuration. The first phase, purging, is accomplished through the following steps:

1. Set the list L to zero. (Cost: $\Theta(N_p)$, because L is of size N_p.)
2. While S is not empty, pop index i from S, then set $O(i)$ to false and $H(i)$ to zero. (Cost: $\mathcal{O}(N_p)$, because at most N_p cells can be occupied by N_p points, assuming each is in a unique cell.)

The second phase is accomplished by doing the following for each element $\mathbf{x}_2^k$ of the set $\{\mathbf{x}_2^i\}$:

1. Identify the index i such that $\mathbf{x}_2^k \in \omega_i$. This can be accomplished in $\mathcal{O}(1)$ time, because the $\{\omega_j\}$ are elements of a uniform grid.
2. If $O(i)$ is false, set it to true and push i onto the stack S. (If $O(i)$ is already true, do not redundantly push i.) This step also has cost $\mathcal{O}(1)$.
3. Update the linked list of points within this cell:
    (a) Store the old head of the linked list: $h_0 \leftarrow H(i)$.
    (b) Set $\mathbf{x}_2^k$'s index as the new head: $H(i) \leftarrow k$.
    (c) Connect the new head to the linked list: $L(k) \leftarrow h_0$.

    These steps cost $\mathcal{O}(1)$ time.

The cost of updating the data structures to reflect a change in the configuration of S_2 is therefore $\Theta(N_p)$, which is asymptotically equivalent to the cost of integrating the structure formulation. Once the data structures have been populated by the above algorithm, the list of points from $\{\mathbf{x}_2^j\}$ contained in an occupied cell ω_i can be traversed by looking at the point $\mathbf{x}_2^{i_1}$, where $i_1 = H(i)$, then looking at $\mathbf{x}_2^{i_2}$, where $i_2 = L(i_1)$, and so on, until $L(i_n) = 0$, indicating the end of the list.[10] If we assume that the points $\{\mathbf{x}_2^i\}$ are distributed such that the number falling within a single cell is $\mathcal{O}(1)$, then the cost of finding the closest point of distance at most C to $\mathbf{x}_1$ is $\mathcal{O}(1)$.

3.3 Discretization of surface integrals

The evaluation of integrals over Γ_t deserves some remark, since Γ_t does not conform to boundaries of the fluid elements $\{\Omega^e\}$.

3.3.1 Definition of surface quadrature rule

I employ a variant of the approach used by Düster et al. [49] to integrate immersed boundary traction in finite cell solutions of solid mechanics problems. I define a Gaussian quadrature rule with respect to a parameterization of the reference configuration Γ_0 of the immersed boundary and weight it by the Jacobian determinant of the mapping from Γ_0 to Γ_t. For a quasi-uniform fluid mesh with elements of diameter $\Theta(h)$, this practice suggests that surface quadrature elements should be of diameter $\mathcal{O}(h)$, but I do not enforce

---

[10] This structure is inspired by the cluster maps used in FAT file systems.

this condition strictly in computations. The fact that the surface quadrature elements are not aligned with the mesh cells is certainly a “variational crime”, although studies from the finite cell literature (e.g. [87, 149, 164]) suggest that the influence of quadrature errors on co-dimension-one boundaries is small relative to the effects of errors in volume quadrature.

3.3.2 Locating quadrature points in the parameterization of the fluid domain

The relevant integrals involve traces of functions defined on the fluid domain. To evaluate these traces, I must be able to locate the quadrature points of the surface in the parameter space of the background mesh. The physical location, $\mathbf{x}_g \in \mathbb{R}^d$, of an integration point can be obtained straightforwardly by evaluating the surface parameterization. Finding the corresponding element index e and parametric point $\boldsymbol{\xi}$ in the parameter space of Ω^e, given a physical point $\mathbf{x}_g$, is typically a more difficult problem. If the fluid is represented on a rectangular grid, the solution is trivial. For more general fluid discretizations, I have explored a number of different approaches.

3.3.2.1 Locating quadrature points in elements

Due to the flexibility of the VMS formulation described in Section 3.1.1, it can be used with a wide variety of unstructured meshes of finite elements $\{\Omega^e\}$, each of which is mapped to physical space from a parametric parent element Ξ^e by $\mathbf{x}^e : \Xi^e \to \Omega^e$. The problem is then to find an element index e such that there exists $\boldsymbol{\xi} \in \Xi^e$ with $\mathbf{x}^e(\boldsymbol{\xi}) = \mathbf{x}_g$. The brute force approach to solving this problem is to iterate over all element indexes, executing a Newton iteration to find $\boldsymbol{\xi}$ in each element, with an appropriate divergence criterion to handle the vast majority of elements for which there is no solution. The number of Newton iterations that need to be attempted can be greatly reduced, though, by using some simple techniques:

- **Bounding boxes:** If an axis aligned bounding box is computed for fluid element Ω^e, this provides a much cheaper test than divergent Newton iteration to conclusively determine that $\mathbf{x}_g$ is *not* in Ω^e. For some types of elements, e.g. linear tetrahedra, the computation of bounding boxes is obvious. For higher-order elements, it may not be. For the computations in this dissertation, I compute bounding boxes for curved elements, such as Bézier elements of spline discretizations, by simply evaluating the physical locations of the parametric corners, then inflating the bounding box of this discrete point set by a factor of two. Such a procedure is not guaranteed to be correct for extremely distorted elements, and it thus introduces an implicit shape regularity restriction on the fluid elements. A more general approach to computing tight bounding boxes for isogeometric elements is given in [117, Section 4], but I have not explored applying it.

- **Continuous motion:** The surface quadrature points are transported with the deforming Γ_t in a Lagrangian fashion. Given an appropriate choice of time step, it is reasonable to expect that a quadrature point will remain in the same Eulerian (or ALE) fluid element, or move to a neighboring element from one time step to the next. These elements should be tested first before resorting to more general search methods. In my computations, I split up the fluid subproblem into subdomains of contiguous elements, to be processed in parallel, building on the technology described in [82]. After applying the cheap continuous motion heuristic in parallel on all subdomains, a collective communication can synchronize the parallel tasks' knowledge of which (small minority of) quadrature points still need to be located by less efficient means.

- **Octree space partitioning:** The indexes of elements whose bounding boxes intersect the leaves of a spatial octree covering Ω_1 can be stored in dynamically-growing lists of integers associated with the leaves of the octree, as detailed in [200, Section 3.3]. This avoids testing for intersection of $\mathbf{x}_g$ with the overwhelming majority of fluid elements. My collaborators and I have, so far, only applied this technique with tetrahedral fluid meshes [87, 200], but it is not fundamentally limited to tetrahedral elements.

3.3.2.2 Locating quadrature points in smoothly-deformed domains

The deformation $\boldsymbol{\phi}$ (in the notation of Section 3.1.2.2) mapping the parametric domain of a div-conforming B-spline discretization into physical space does not need to be related at all to the B-spline spaces used to define the solution variables. In the div-conforming B-spline computations of this dissertation, I employ analytical mappings.

I implement these in an object-oriented fashion, as subclasses implementing the virtual methods of a general geometrical mapping interface, which is most succinctly described by its own C++ source code:

```
class GeometricalMapping{
public:
  virtual ~GeometricalMapping(){}
  virtual void phi(double (*xi)[3],
    double (*x)[3]){}
  virtual void firstDerivs(double (*xi)[3],
    double (*F)[3][3],
    double (*Finv)[3][3],
    double *J){}
  virtual void hessian(double (*xi)[3],
    double (*hessian)[3][3][3]){}
  virtual void phi_inverse(double (*x)[3],
    double (*xi)[3]){}
};
```

The methods `phi`, `firstDerivs`, and `hessian` implement the mapping and derivatives needed to evaluate pushed-forward velocities and pressures and the necessary spatial gradients. The meanings of the argument names should be obvious upon review of the notation from Section 3.1.2.2. The method `phi_inverse` is the first non-trivial step needed to evaluate traces of fluid solution variables at quadrature points on the immersed boundary Γ. For some mappings, an analytical form of $\boldsymbol{\phi}^{-1}$ is available, but, more typically, I use `firstDerivs` to implement a Newton iteration on the entire parametric space. Given an implementation of `phi_inverse`, however, one still cannot (efficiently) evaluate the fluid solution fields at $\mathbf{X}_g = \boldsymbol{\phi}^{-1}(\mathbf{x}_g)$. Efficient evaluation of spline functions requires knowledge of which basis functions have the parametric point $\mathbf{X}_g$ in their supports. This reduces to the problem of determining which Bézier element $\mathbf{X}_g$ lies in. The tensor product structure of the B-spline space allows for an efficient solution of this problem by performing a binary search in each parametric direction.

Chapter 4

Discretization of fluid–structure coupling

The augmented Lagrangian coupling the fluid and structure subproblems is discretized using a semi-implicit time integration scheme, in which the penalty term is treated implicitly and the Lagrange multiplier is updated explicitly. This circumvents difficulties encountered in fully-implicit coupling, while forbidding leakage of fluid through the structure in steady-state solutions

---

Some of this chapter's content is derived from the following papers:

D. Kamensky, M.-C. Hsu, D. Schillinger, J. A. Evans, A. Aggarwal, Y. Bazilevs, M. S. Sacks, T. J. R. Hughes. An immersogeometric variational framework for fluid–structure interaction: Application to bioprosthetic heart valves. *Computer Methods in Applied Mechanics and Engineering*, 284:1005–1053, 2015. D. Kamensky developed the techniques used for fluid–thin structure interaction and structure-on-structure contact. M.-C. Hsu provided supervision and implemented the finite cell method for flow around bulky objects. D. Schillinger helped formulate the finite cell approach used. J. A. Evans provided mathematical advice. A. Aggarwal developed the geometrical model of the valve. Y. Bazilevs, M. S. Sacks, and T. J. R. Hughes supervised the work.

D. Kamensky, J. A. Evans, M.-C. Hsu. Stability and conservation properties of collocated constraints in immersogeometric fluid-thin structure interaction analysis. *Communications in Computational Physics*. 18(4):1147–1180, 2015. D. Kamensky formulated and analyzed the improvements to the semi-implicit time integration and performed the numerical experiments. J. A. Evans and M.-C. Hsu supervised the work.

D. Kamensky, M.-C. Hsu, Y. Yu, J. A. Evans, M. S. Sacks, T. J. R. Hughes. Immersogeometric cardiovascular fluid–structure interaction analysis using divergence-conforming B-splines. *Computer Methods in Applied Mechanics and Engineering*, In review (preprint: ICES Report 16-14). D. Kamensky implemented the numerical methods, formulated and analyzed model problems, and participated in the experimental work. M.-C. Hsu provided supervision and edited the manuscript extensively. Y. Yu, J. A. Evans, and T. J. R. Hughes supervised the mathematical analysis. M. S. Sacks helped plan and supervise the laboratory experiments.

and retaining the stability that eludes fully-explicit approaches. This chapter is mainly concerned with clearly describing the solution algorithm applied to the nonlinear FSI problem stated in Chapter 2 and discussing its qualitative properties. A more precise analysis of the algorithm is carried out in the context of a linearized model problem in Chapter 5.

4.1 Separation of normal and tangential fluid–structure coupling

The constraint that $\mathbf{u}_1 = \mathbf{u}_2$ on Γ can be formally separated into two constraints: the no-penetration constraint on the normal velocities

$$\mathbf{u}_1 \cdot \mathbf{n}_2 = \mathbf{u}_2 \cdot \mathbf{n}_2 \tag{4.1}$$

and the no-slip constraint on the tangential velocities

$$\mathbf{u}_1 - (\mathbf{u}_1 \cdot \mathbf{n}_2)\,\mathbf{n}_2 = \mathbf{u}_2 - (\mathbf{u}_2 \cdot \mathbf{n}_2)\,\mathbf{n}_2 \ , \tag{4.2}$$

where $\mathbf{n}_2$ is the normal vector to Γ (i.e. $\mathbf{a}_3$ defined in (2.14)). These constraints are enforced by the normal and tangential components of the multiplier field $\boldsymbol{\lambda}$ and the penalty force $\beta\,(\mathbf{u}_1 - \mathbf{u}_2)$.

The no-penetration constraint is critical to the qualitative structure of solutions. In the application to BHV analysis, for instance, the valve leaflets must be able to stop flow when the valve is closed. The no-slip constraint is less essential and its strong enforcement may even be detrimental to the qualitative character of discrete solutions on coarse meshes [15, 20–22, 83].[1] I

---

[1] The cited works attribute improved solution quality to the tangential slippage allowed

therefore discretize these two constraints differently. For the no-penetration constraint, I discretize a scalar Lagrange multiplier field on Γ, which I denote $\lambda = \boldsymbol{\lambda} \cdot \mathbf{n}_2$. For the no-slip constraint, I approximate the tangential component of $\boldsymbol{\lambda}$ by a weakly-consistent penalty force. The weakly-consistent penalty approximation of the tangential constraint may be seen as a degenerate case of Nitsche's famous strongly-consistent penalty method [129], as I explained in [105, Section 4.1]. Because the structure midsurface Γ_t can cut through the fluid domain in arbitrary ways, I do not attempt to construct inf-sup stable combinations of velocity and multiplier spaces. Instead, I circumvent the inf-sup condition by regularizing the no-penetration constraint residual in the following way:

$$(\mathbf{u}_1 - \mathbf{u}_2) \cdot \mathbf{n}_2 \quad \to \quad (\mathbf{u}_1 - \mathbf{u}_2) \cdot \mathbf{n}_2 - \frac{r}{\beta}\lambda \ , \tag{4.3}$$

where $r \geq 0$ is a dimensionless constant. This is essentially the perturbed Lagrangian approach that has previously been used to stabilize contact problems [159]. Much as the slip penalization can be derived as a degernate case of Nitsche's method [105, Section 4.1], the regularization of the no-penetration constraint can be viewed as a degenerate case of strongly-consistent Barbosa–Hughes stabilization [14].

Thus the problem that I proceed to discretize in time may be written:

---

by weak boundary conditions on coarse meshes. Obviously it is impossible for flow fields on either side of an immersed boundary to both slip in independent directions if they are represented on a single mesh, but I have observed that excessive enforcement of tangential boundary conditions leads to Gibbs-like phenomena in the tangential velocity profile, which can generate spurious eddies.

Find $\mathbf{u}_1 \in \mathcal{S}_u$, $p \in \mathcal{S}_p$, $\mathbf{y} \in \mathcal{S}_d$, and $\lambda \in \mathcal{S}_\ell$ such that, for all test functions $\mathbf{w}_1 \in \mathcal{V}_u$, $q \in \mathcal{V}_p$, $\mathbf{w}_2 \in \mathcal{V}_d$, and $\delta\lambda \in \mathcal{V}_\ell$

$$
\begin{aligned}
&B_1(\{\mathbf{w}_1, q\}, \{\mathbf{u}_1, p\}; \hat{\mathbf{u}}) - F_1(\{\mathbf{w}_1, q\}) \\
&\quad + B_2(\mathbf{w}_2, \mathbf{y}) - F_2(\mathbf{w}_2) \\
&\quad + \int_{\Gamma_t} (\mathbf{w}_1 - \mathbf{w}_2) \cdot \lambda \mathbf{n}_2 \; d\Gamma && (4.4) \\
&\quad + \int_{\Gamma_t} (\mathbf{w}_1 - \mathbf{w}_2) \cdot \tau^B_{\text{NOR}} \left((\mathbf{u}_1 - \mathbf{u}_2) \cdot \mathbf{n}_2 \right) \mathbf{n}_2 \; d\Gamma && (4.5) \\
&\quad + \int_{\Gamma_t} (\mathbf{w}_1 - \mathbf{w}_2) \cdot \tau^B_{\text{TAN}} \left((\mathbf{u}_1 - \mathbf{u}_2) - \left((\mathbf{u}_1 - \mathbf{u}_2) \cdot \mathbf{n}_2 \right) \mathbf{n}_2 \right) \; d\Gamma && (4.6) \\
&\quad + \int_{\Gamma_t} \delta\lambda \cdot \left((\mathbf{u}_1 - \mathbf{u}_2) \cdot \mathbf{n}_2 - \frac{r\lambda}{\tau^B_{\text{NOR}}} \right) \; d\Gamma \;=\; 0 \;, && (4.7)
\end{aligned}
$$

where I have split the penalty term from the original variational problem into normal and tangential components. Inspired by applications of Nitsche's method to viscous incompressible flows, I propose to scale the tangential penalty like

$$
\tau^B_{\text{TAN}} = C_{\text{TAN}} \frac{\mu}{h} \;, \tag{4.8}
$$

where C_{TAN} is a dimensionless $\mathcal{O}(1)$ constant and h is a measure of the fluid element diameter, with units of length. This causes the no-slip portion of the boundary condition on Γ to disappear in the inviscid limit of $\mu \to 0$. To ensure that the normal penalty does not suffer the same fate in this limit, I propose that it scale like

$$
\tau^B_{\text{NOR}} = \max \left\{ C^{\text{inert}}_{\text{NOR}} \frac{\rho_1 h}{\Delta t}, C^{\text{visc}}_{\text{NOR}} \frac{\mu}{h} \right\} \;, \tag{4.9}
$$

where $C^{\text{inert}}_{\text{NOR}}$ and $C^{\text{visc}}_{\text{NOR}}$ are dimensionless constants and Δt is a time scale associated with the temporal discretization.[2]

4.2 Time integration algorithm

This section states the time-marching procedure for the fluid–structure system. The algorithm computes approximate solutions at a set of time levels, indexed by n and separated by steps of size Δt. Suppose that, at time level n, the discrete fluid velocity is defined by a vector of coefficients $\mathbf{U}^n$, the fluid acceleration by $\dot{\mathbf{U}}^n$, the fluid pressure by $\mathbf{P}^n$, and the structure displacement, velocity, and acceleration by $\mathbf{Y}^n$, $\dot{\mathbf{Y}}^n$, and $\ddot{\mathbf{Y}}^n$, respectively. I refer to the multiplier at time level n as λ^n, considering it a function defined over Γ_t, with the understanding that it is represented discretely as a set of samples at quadrature points of the (Lagrangian) integration rule on Γ_t. (Recall Section 3.3.) Considering the solution variables at time level n known, the first step of the algorithm is to construct a system of equations for all $(n + 1)$-level

---

[2] An alternative formula for τ^B_{NOR} might be $C\rho_1 h/\tau_{\mathrm{M}}$, where the time scale τ_{M} is a typical SUPG stabilization constant. Standard definitions of τ_{M} would capture both branches of (4.9).

unknowns, excluding λ^{n+1}, which is initially set equal to λ^n:

$$\text{Res}\left(\mathbf{U}^{n+\alpha_f}, \dot{\mathbf{U}}^{n+\alpha_m}, \mathbf{Y}^{n+\alpha_f}, \dot{\mathbf{Y}}^{n+\alpha_f}, \ddot{\mathbf{Y}}^{n+\alpha_m}, \mathbf{P}^{n+1}, \lambda^{n+1}(=\lambda^n)\right) = \mathbf{0}\ , \tag{4.10}$$

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \Delta t\left((1-\gamma)\dot{\mathbf{U}}^n + \gamma\dot{\mathbf{U}}^{n+1}\right)\ , \tag{4.11}$$

$$\dot{\mathbf{U}}^{n+\alpha_m} = \dot{\mathbf{U}}^n + \alpha_m\left(\dot{\mathbf{U}}^{n+1} - \dot{\mathbf{U}}^n\right)\ , \tag{4.12}$$

$$\mathbf{U}^{n+\alpha_f} = \mathbf{U}^n + \alpha_f\left(\mathbf{U}^{n+1} - \mathbf{U}^n\right)\ , \tag{4.13}$$

$$\mathbf{Y}^{n+1} = \mathbf{Y}^n + \Delta t\dot{\mathbf{Y}}^n + \frac{\Delta t^2}{2}\left((1-2\beta)\ddot{\mathbf{Y}}^n + 2\beta\ddot{\mathbf{Y}}^{n+1}\right)\ , \tag{4.14}$$

$$\dot{\mathbf{Y}}^{n+1} = \dot{\mathbf{Y}}^n + \Delta t\left((1-\gamma)\ddot{\mathbf{Y}}^n + \gamma\ddot{\mathbf{Y}}^{n+1}\right)\ , \tag{4.15}$$

$$\ddot{\mathbf{Y}}^{n+\alpha_m} = \ddot{\mathbf{Y}}^n + \alpha_m\left(\ddot{\mathbf{Y}}^{n+1} - \ddot{\mathbf{Y}}^n\right)\ , \tag{4.16}$$

$$\dot{\mathbf{Y}}^{n+\alpha_f} = \dot{\mathbf{Y}}^n + \alpha_f\left(\dot{\mathbf{Y}}^{n+1} - \dot{\mathbf{Y}}^n\right)\ , \tag{4.17}$$

$$\mathbf{Y}^{n+\alpha_f} = \mathbf{Y}^n + \alpha_f\left(\mathbf{Y}^{n+1} - \mathbf{Y}^n\right)\ , \tag{4.18}$$

where α_m, α_f, β, and γ are parameters of the time integration scheme. The function Res(...) is the nonlinear residual corresponding to the discretization of (4.7) with $\delta\lambda = 0$. The multiplier test function is set to zero to exclude the FSI kinematic constraint equation, which, because λ^{n+1} is held fixed, would lead to an ill-posed system with more equations than unknowns. While the multiplier is considered fixed in this problem, the penalty terms are still treated implicitly. This penalty-coupled problem is resolved by a block iterative procedure, which alternates between solving for fluid and structure increments. Block iteration is described further in Section 4.3. The formulas (4.10)–(4.18) are based on the generalized-α method of time integration [38]. Following Bazilevs et al. [18, Section 4.4], I work within a subset of generalized-α methods, parameterized by as single scalar, $\rho_\infty \in [0, 1]$, which controls numerical damping and defines the four free parameters as

$$\alpha_m = \frac{1}{2}\left(\frac{3-\rho_\infty}{1+\rho_\infty}\right) \ , \tag{4.19}$$

$$\alpha_f = \frac{1}{1+\rho_\infty} \ , \tag{4.20}$$

$$\gamma = \frac{1}{2} + \alpha_m - \alpha_f \ , \tag{4.21}$$

$$\beta = \frac{1}{4}\left(1 + \alpha_m + \alpha_f\right)^2 \ . \tag{4.22}$$

For a discussion of the effects of this parameter on stabilized finite element computations of unsteady Navier–Stokes, see [97]. Alternatively, one may select the generalized-α parameters as follows to produce the backward Euler method:

$$\alpha_m = \alpha_f = \gamma = \beta = 1 \ . \tag{4.23}$$

Remark 4.1. A more canonical implementation of the generalized-α scheme might introduce

$$\lambda^{n+\alpha_f} = \lambda^n + \alpha_f\left(\lambda^{n+1} - \lambda^n\right) \tag{4.24}$$

and

$$\mathbf{P}^{n+\alpha_f} = \mathbf{P}^n + \alpha_f\left(P^{n+1} - P^n\right) \tag{4.25}$$

for use in Eq. (4.10), but the formulation has no time derivatives of the corresponding fields and their α-level coefficients would be uniquely determined by the fully-discrete formulation, leaving Eqs. (4.24) and (4.25) as post-processing steps for the $(n+1)$-level unknowns. I follow [16] in simply renaming these

α-level unknowns to $(n+1)$-level unknowns. This has no effect on the values of the fluid velocity and structure displacement coefficients and essentially translates the pressure and multiplier solutions by a fraction of Δt in time.

Because (4.10)–(4.18) do not include the fluid–structure interface kinematic constraint, the (regularized) α-level constraint residual

$$R^{n+\alpha} = \left(\left(\mathbf{u}_1^h\right)^{n+\alpha_f} - \left(\mathbf{u}_2^h\right)^{n+\alpha_f} \right) \cdot \mathbf{n}_2^{n+\alpha_f} - \frac{r\lambda^{n+1}}{\tau_{\mathrm{NOR}}^B} \tag{4.26}$$

is not necessarily zero on $\Gamma_{t+\alpha_f}$. In (4.26), $\left(\mathbf{u}_1^h\right)^{n+\alpha_f}$ is the fluid velocity defined by coefficients $\mathbf{U}^{n+\alpha_f}$, $\left(\mathbf{u}_2^h\right)^{n+\alpha_f}$ is the structure velocity defined by coefficients $\dot{\mathbf{Y}}^{n+\alpha_f}$, and $\mathbf{n}_2^{n+\alpha_f}$ is the normal to $\Gamma_{t+\alpha_f}$, as determined by the displacement coefficients $\mathbf{Y}^{n+\alpha_f}$.

To motivate the development of the multiplier update step, consider the case of $r = 0$. If $R^{n+\alpha} = 0$ and $r = 0$, then the normal component of the α-level penalty force, $\tau_{\mathrm{NOR}}^B R^{n+\alpha}$, will be zero and the normal α-level fluid–structure force will be due only to the Lagrange multiplier, λ^{n+1}. This suggests the explicit update

$$\lambda^{n+1} \leftarrow \lambda^{n+1} + \tau_{\mathrm{NOR}}^B R^{n+\alpha} \ , \tag{4.27}$$

in which λ^{n+1} is set equal to the α-level fluid–structure forcing. (4.10)–(4.18) are of course no longer satisfied with the updated λ^{n+1}, but one may attempt to iterate the steps

1. Solve (4.10)–(4.18) with λ^{n+1} fixed.

2. Upate λ^{n+1} by (4.27):

$$\lambda^{n+1} = \lambda^n + \tau_{\text{NOR}}^B R^{n+\alpha} \ . \tag{4.28}$$

until $\|R^{n+\alpha}\|_{L^2(\Gamma_t)}$ is converged to some tolerance. Note that, in the case of $r > 0$, (4.28) is an implicit formula, because $R^{n+\alpha}$ depends on λ^{n+1}. It can be trivially recast in explicit form, though:

$$\lambda^{n+1} = \frac{\lambda^n + \tau_{\text{NOR}}^B \left(\left(\mathbf{u}_1^h\right)^{n+\alpha_f} - \left(\mathbf{u}_2^h\right)^{n+\alpha_f} \right) \cdot \mathbf{n}_2^{n+\alpha_f}}{1+r} \ . \tag{4.29}$$

As explained in [105, Section 4.2.1], the $r = 0$ case of this iteration corresponds to the classic augmented Lagrangian algorithm of Hestenes [78] and Powell [135], which is an implicit variant of the well-known Uzawa iteration [10, 187] for solving saddle point problems. For $r = 0$, though, the convergence criterion of $\|R^{n+\alpha}\|_{L^2(\Gamma_t)} < \epsilon$ is too strict to arrive at a non-locking solution; it effectively demands pointwise constraint satisfaction between the non-matching discrete velocity spaces of the fluid and structure. I found, accordingly, in [105], that the iteration does not typically converge, but I circumvented this difficulty by truncating to a single pass, leading to the semi-implicit time marching scheme of first solving (4.10)–(4.18) with $\lambda^{n+1} = \lambda^n$, then updating λ^{n+1} by (4.28) and continuing directly to the next time step. This time splitting approach proved effective for transient problems, but may be expected to run into difficulties in problems that approach steady solutions. Choosing $r > 0$ can improve robustness.

Although the stabilization provided by choosing $r > 0$ affords the possibility of fully-implicit time integration, which is typically recommended for complex FSI problems [23], semi-implicit integration procedures can greatly reduce computational cost. The present semi-implicit algorithm is in fact stable, in an energetic sense, even when $r = 0$. This is discussed physically in [104, Section 3.2] and analyzed mathematically in the context of linear model problems in Chapter 5. This stability is in contrast to "staggered" or "loosely coupled" FSI methods which are notoriously unstable, especially when the fluid is incompressible [188], prompting widespread preference for implicit methods. The use of $r > 0$ allows for robustness even when energy is continuously added to the system, as through an inhomogeneous boundary condition. Some caution is warranted, however, in perturbing the kinematic constraint. Section 4.4.4 provides an illustrative example of the effects of this consistency error.

4.3 Block iterative solution of the implicit problem

The implicit step of the semi-implicit time integration algorithm of Section 4.2 amounts to a penalty regularization of fluid–structure coupling, with a prescribed loading $\lambda^n \mathbf{n}^{n+\alpha_f}$ along $\Gamma^{n+\alpha_f}$. Because the penalty is not solely responsible for fluid–structure coupling, its value can be moderate, rendering the regularized problem much easier to solve than fully-implicit fluid–structure coupling. A simple block-iterative procedure that alternates between fluid and structure solutions turns out to be practical, even for "difficult" applications, such as BHV simulation, in which a light structure interacts with a heavy, incompressible fluid.

Schematically, consider $R_\text{f}(u_\text{f}, u_\text{s})$ to be the nonlinear residual for the fully-discrete fluid subproblem at a particular time step, which depends on the discrete fluid and structure solutions, u_{f} and u_{s}. Likewise, $R_{\mathrm{s}}(u_{\mathrm{f}}, u_{\mathrm{s}})$ is the residual for the discrete structure subproblem. Then the block-iterative procedure to find a root of $(R_{\mathrm{f}}, R_{\mathrm{s}})$ is to start with guesses for u_{f} and u_{s}, then repeat the steps

1. Assemble $R_{\mathrm{f}}(u_{\mathrm{f}}, u_{\mathrm{s}})$ and a (typically approximate) tangent matrix, $A_{\mathrm{f}} \approx \partial R_{\mathrm{f}}/\partial u_{\mathrm{f}}$.
2. Solve the linear system $A_{\mathrm{f}} \Delta u_{\mathrm{f}} = -R_{\mathrm{f}}$ for the fluid solution increment.
3. Update the fluid solution: $u_{\mathrm{f}} \leftarrow u_{\mathrm{f}} + \Delta u_{\mathrm{f}}$.
4. Assemble $R_{\mathrm{s}}(u_{\mathrm{f}}, u_{\mathrm{s}})$ and $A_{\mathrm{s}} \approx \partial R_{\mathrm{s}}/\partial u_{\mathrm{s}}$.
5. Solve $A_{\mathrm{s}} \Delta u_{\mathrm{s}} = -R_{\mathrm{s}}$ for the structure solution increment.
6. Update the structure solution: $u_{\mathrm{s}} \leftarrow u_{\mathrm{s}} + \Delta u_{\mathrm{s}}$.

until R_{f} and R_{s} are sufficiently converged. Note that this resembles Newton iteration with an inexact tangent, wherein off-diagonal blocks of the tangent matrix for the combined system,

$$\begin{pmatrix} A_{\mathrm{f}} & (\partial R_{\mathrm{f}}/\partial u_{\mathrm{s}}) \\ (\partial R_{\mathrm{s}}/\partial u_{\mathrm{f}}) & A_{\mathrm{s}} \end{pmatrix} \begin{pmatrix} \Delta u_{\mathrm{f}} \\ \Delta u_{\mathrm{s}} \end{pmatrix} = - \begin{pmatrix} R_{\mathrm{f}} \\ R_{\mathrm{s}} \end{pmatrix} \ , \tag{4.30}$$

are neglected. However, the update of the fluid solution in step 3 distinguishes block iteration from an inexact tangent method. To ensure predictable running times and avoid stagnation in pathological configurations, I typically select the resolution of the nonlinear algebraic solution by choosing a fixed number of iterations rather than a percentage by which the residual must be reduced. This may be interpreted as a predictor–multi-corrector scheme based on Newton's method [18]. While it is possible that error from isolated, poorly-solved time steps can pollute the future of an unsteady solution, I find that, within reasonable limits, quantities of engineering interest are typically more sensitive to spatial and temporal discretizations than nonlinear solution tolerance.

The fact that this procedure is stable when applied to the problem of BHV FSI, in which the fluid is much more massive than the structure, is perhaps surprising to researchers familiar with Dirichlet-to-Neumann fluid–structure coupling, where the structure velocity is applied as a Dirichlet boundary condition on the fluid and the fluid traction is applied as a Neumann boundary condition on the structure. With that style of coupling, block iteration may be unstable for any time step size, no matter how small, if the fluid is incompressible and heavy (relative to the structure) [188]. Section 5.3 analyzes the stability and convergence of the block iteration algorithm in the context of penalty coupling between two linear elliptic problems, confirming that it is unconditionally stable, but perhaps slow to converge for excessively-large penalty parameters.

In all computations shown in this dissertation, the structure tangent matrix A_{s} is approximated to be symmetric and solved iteratively using a serial implementation of the conjugate gradient method [79]. For computations using the VMS formulation for the fluid subproblem, the matrix A_{f} is inverted approximately using a parallel implementation (by Y. Bazilevs and collaborators) of the generalized minimum residual (GMRES) method [144]. For computations using div-conforming B-spline discretizations of the fluid subproblem, A_{f} is inverted approximately or directly using linear algebra routines from the Portable Extensible Toolkit for Scientific Computation (PETSc) [11–13]. In some computations, A_{f} is assembled only on the first block iteration of each time step, then reused in subsequent iterations. Some of the numerical examples reverse the order of the fluid and structure solutions, but, in agreement with the analysis of Section 5.3, this does not appear to confer and advantage (or disadvantage) relative to the algorithm stated above. Details of the solution procedures for specific numerical examples are provided when relevant in the sequel.

4.4 Discussion

Some alternate interpretations and qualitative analysis of the algorithm stated in Section 4.2 help to build intuitive understanding and lay the groundwork for the more precise analysis given in Chapter 5.

4.4.1 Modified equation interpretation of semi-implicit integration

When $r = 0$, the multiplier becomes an accumulation of penalty tractions from previous time steps. This is equivalent to replacing the multiplier and normal penalty terms

$$\begin{aligned}\int_{\Gamma_t} & (\mathbf{w}_1 - \mathbf{w}_2) \cdot (\lambda \mathbf{n}_2)\ d\Gamma \\ & + \int_{\Gamma_t} ((\mathbf{w}_1 - \mathbf{w}_2) \cdot \mathbf{n}_2)\, \tau^B_{\mathrm{NOR}} \left((\mathbf{u}_1 - \mathbf{u}_2) \cdot \mathbf{n}_2\right)\ d\Gamma \end{aligned} \tag{4.31}$$

by a penalization of (a backward Euler evaluation of) the time integral of pointwise normal velocity differences on the immersed surface Γ_t

$$\begin{aligned}\int_{\Gamma_t} \Bigg\{ & \frac{\tau^B_{\mathrm{NOR}}}{\Delta t} \left(\mathbf{w}_1(\mathbf{x},t) - \mathbf{w}_2(\mathbf{x},t)\right) \cdot \mathbf{n}_2(\mathbf{x},t) \\ & \int_0^t \big(\mathbf{u}_1(\boldsymbol{\varphi}_\tau(\boldsymbol{\varphi}_t^{-1}(\mathbf{x})),\tau) - \mathbf{u}_2(\boldsymbol{\varphi}_\tau(\boldsymbol{\varphi}_t^{-1}(\mathbf{x})),\tau)\big) \\ & \quad \cdot \mathbf{n}_2(\boldsymbol{\varphi}_\tau(\boldsymbol{\varphi}_t^{-1}(\mathbf{x})),\tau)\ d\tau \Big\}\ d\Gamma\ , \end{aligned} \tag{4.32}$$

where $\boldsymbol{\varphi}_\tau(\mathbf{X})$ gives the spatial position at time τ of material point $\mathbf{X} \in \Gamma_0$ and the measure $d\Gamma$ corresponds to the integration variable $\mathbf{x} \in \Gamma_t$. That the time integral in (4.32) is evaluated using the backward Euler method is demonstrated in the following exposition. First define (at fixed $\mathbf{X}$)

$$\lambda^{\mathrm{reg}}(t) = \frac{\tau^B_{\mathrm{NOR}}}{\Delta t} \int_0^t (\mathbf{u}_1(\tau) - \mathbf{u}_2(\tau)) \cdot \mathbf{n}_2(\tau)\ d\tau\ . \tag{4.33}$$

The time rate-of-change of the integral λ^{reg} will be its integrand

$$(\dot{\lambda^{\mathrm{reg}}}) = \left.\frac{\partial \lambda^{\mathrm{reg}}}{\partial t}\right|_{\mathbf{X}} = \frac{\tau^B_{\mathrm{NOR}}}{\Delta t} (\mathbf{u}_1 - \mathbf{u}_2) \cdot \mathbf{n}_2\ . \tag{4.34}$$

The normal forcing on Γ in the implicit step of the semi-implicit time integrator is designated

$$(\lambda^{\mathrm{reg}})^{n+1} = (\lambda^{\mathrm{reg}})^n + \Delta t (\dot{\lambda^{\mathrm{reg}}})^{n+1} \tag{4.35}$$

where $(\lambda^{\mathrm{reg}})^n$ is a sum of all previous approximations of λ and $\Delta t(\dot{\lambda^{\mathrm{reg}}})^{n+1}$ is the current time step's penalty forcing, i.e. $\tau^B_{\mathrm{NOR}}\left(\mathbf{u}_1^{n+\alpha_f} - \mathbf{u}_1^{n+\alpha_f}\right)\cdot\mathbf{n}_2^{n+\alpha_f}$. For the time step indexing of λ^{reg}, I have followed the same convention employed for λ; see Remark 4.1 in Section 4.2 if clarification is needed. (4.35) is precisely the backward Euler algorithm for computing λ^{reg}. Thus the forcing (4.32) is accounted for in a fully implicit manner within the discrete solution process, using a manifestly stable time integrator. This is only first-order accurate in time, but, in the application to BHVs, other considerations have driven the time step down to small enough values that time integration is not a dominant source of discretization error.

In the case of $r > 0$, we can draw a similar analogy. If the α-level normal penalty force and λ^{n+1} are again lumped together and denoted $(\lambda^{\mathrm{reg}})^{n+1}$, it is straightforward to see that λ^{reg} advances through time by backward Euler evaluation of the differential equation

$$\frac{1}{(1+r)}\left.\frac{\partial\lambda^{\mathrm{reg}}}{\partial t}\right|_{\mathbf{X}} = \frac{\tau^B_{\mathrm{NOR}}}{\Delta t}\left(\mathbf{u}_1 - \mathbf{u}_2\right)\cdot\mathbf{n}_2 - \frac{r}{\Delta t(1+r)}\lambda^{\mathrm{reg}}\ . \tag{4.36}$$

Intuitively, it is clear that the additional term causes an exponential decay of λ^{reg} in the absence of constraint violation, which highlights its stabilizing effect on the multiplier field. One can quickly check that this reduces to (4.34) in the case of $r = 0$.

4.4.2 Analogy to artificial compressibility

Integrating a constraint residual in time is not a new concept for approximation of a Lagrange multiplier. The differential equation given in

(4.34) resembles the method of artificial compressibility, devised by Chorin [37] in 1967 and widely used since to simulate incompressible flows (see, e.g., [32, 63, 118, 138]). In the artificial compressibility scheme, the approximated Lagrange multiplier p representing the pressure evolves through time in an analogous way to λ^{reg} (in the case $r = 0$):

$$\partial_t p = -\frac{1}{\delta}\nabla \cdot \mathbf{u}_1 \ , \tag{4.37}$$

where the constraint is $\nabla \cdot \mathbf{u}_1 = 0$ (instead of $(\mathbf{u}_1 - \mathbf{u}_2) \cdot \mathbf{n}_2 = 0$), $1/\delta$ is the penalty parameter, and the difference in sign is due to the arbitrary choice of sign with which $\boldsymbol{\lambda}$ enters the augmented Lagrangian formulation (2.1). A physical interpretation of this, similar to Chorin's original formulation of (4.37) in terms of a fictitious density variable, is that the $r = 0$ case of the algorithm penalizes a displacement penetration of the fluid through the structure, using the penalty $\tau^B_{\text{NOR}}/\Delta t$. This interpretation makes clear how penalizing the time integral of velocity prevents the steady creep of flow through a barrier. The displacement penalty interpretation becomes less clear in the case of $r > 0$, though.

4.4.3 Relation to feedback boundary conditions

The degeneration of Nitsche's method to a velocity penalty and the time-continuous interpretation of the semi-implicit algorithm with $r = 0$ may both be interpreted as special cases of an existing framework for enforcing Dirichlet boundary conditions on the unsteady Navier–Stokes equation. Goldstein et al. [66] proposed to apply concentrated surface forcing of the form

[66, (3)]

$$\mathbf{f}(\mathbf{x}_s, t) = \alpha \int_0^t \mathbf{u}_1(\mathbf{x}_s, \tau)\ d\tau + \beta \mathbf{u}_1(\mathbf{x}_s, t)\ , \tag{4.38}$$

for all $\mathbf{x}_s$ on a stationary solid boundary (i.e. $\mathbf{u}_2 = \mathbf{0}$) with (dimensional) parameters $\alpha \leq 0$ and $\beta \leq 0$. Goldstein et al. interpreted this method, which I refer to here as the feedback method, in the context of control theory, arguing heuristically that it provides negative feedback in the case of constraint violation. This method is frequently passed over as a historical curiosity in literature reviews of immersed boundary CFD and FSI, and dismissed with criticisms of its arbitrary penalty parameters and numerical stiffness, but the feedback boundary condition and related methods remain in use today by numerous research groups, for both direct numerical simulation (DNS) of flow physics phenomena and engineering analysis of difficult FSI problems.

The initial implementation of [66] used a spectral discretization of the fluid (based on the DNS method famously applied in [113]) and applied $\mathcal{O}(h)$ smoothing to filter the concentrated forces, so as to reduce pollution effects due to the global nature of the spectral basis functions (cf. [194, Chapter I, Section 2]). Goldstein and collaborators continue to use this methodology for DNS of turbulent flows with nontrivial boundary geometries [48, 67–69, 167, 184]. Saiki and Biringen [145, 146] extended the concept of feedback forcing to finite difference fluid discretizations, using bilinear interpolation within grid cells to evaluate velocity at quadrature points of the immersed boundary and also to distribute concentrated feedback forces to grid points. [145] was the first application of the approach to moving boundaries, in which (4.38) becomes (cf. [145, (1)] and (4.32))

$$\begin{aligned}\mathbf{f}(\boldsymbol{\varphi}_t(\mathbf{X}_s),t) =&\alpha\int_0^t\left(\mathbf{u}_1(\boldsymbol{\varphi}_\tau(\mathbf{X}_s),\tau)-\mathbf{U}_2(\mathbf{X}_s,\tau)\right)\,d\tau\\&+\beta\left(\mathbf{u}_1(\boldsymbol{\varphi}_\tau(\mathbf{X}_s),\tau)-\mathbf{U}_2(\mathbf{X}_s,\tau)\right)\ ,\end{aligned}\tag{4.39}$$

where $\boldsymbol{\varphi}_t(\mathbf{X}_s)$ represents the position at time t of a material point $\mathbf{X}_s$ on the moving boundary, which moves with velocity $\mathbf{U}_2(\mathbf{X}_s,t)$. This extension naturally suggests application to FSI, and a recent series of papers by Huang, Sung, and collaborators has demonstrated that feedback forcing is a robust and accurate approach for the simulation of light flexible structures immersed in incompressible flows [88, 89, 143, 157, 185]. A similar immersed boundary approach has been used in the commercial code LS-DYNA [39] for decades, to study automobile airbag inflation and other challenging FSI problems [76, 161–163], including heart valve simulation [35, 36, 169, 199]. LS-DYNA documentation refers to this capability as the "constrained Lagrange in solid" formulation. I have seen no document explicitly relating this to Goldstein et al.'s feedback approach, and assume that it was arrived at independently. The repeated rediscovery of this formulation by engineers studying difficult CFD and FSI problems suggests an inherent robustness to the approach.

The above studies all relied on explicit or semi-implicit time integration schemes, which placed stability restrictions on α and β relative to Δt. Much attention has therefore been paid to the temporal stability of explicitly-integrated feedback forces. The most comprehensive study of the temporal stability of feedback forcing is due to Lee [121]. To my knowledge, though, no previous attempt has been made by numerical analysts to prove the convergence of the feedback boundary condition method. In Chapter 5, I study the convergence of feedback boundary conditions in the context of a linear parabolic model problem, as a stepping-stone to the analysis of my semi-implicit time integrator for the augmented Lagrangian system.

4.4.4 Qualitative effects of multiplier stabilization

The case of $r > 0$ is perhaps less physically intuitive than the $r = 0$ case, which admits direct analogies to artificial compressibility (Section 4.4.2) and negative feedback (Section 4.4.3). To provide some intuition for the influence of r, I look, in detail, at the simple model of plug flow through a blocked tube: a rigid barrier cuts across a channel filled with a fluid that I assume, *a priori*, to have a single velocity, $u\mathbf{e}_1$, that is constant across space, but may vary with time. To allow nonzero velocity solutions with this kinematic assumption, I apply slip boundary conditions on the channel walls. This is illustrated in Figure 4.1.

4.4.4.1 Leakage

Suppose that the ends of the channel are subject to pressures P_1 and P_2, which define the pressure drop, $\Delta P = P_1 - P_2$. Suppose also, for now, that the Lagrange multiplier field takes on a single constant value across the barrier. Then the steady state solution of the semi-implicit time integration procedure described in Section 4.2 will reduce to the conditions

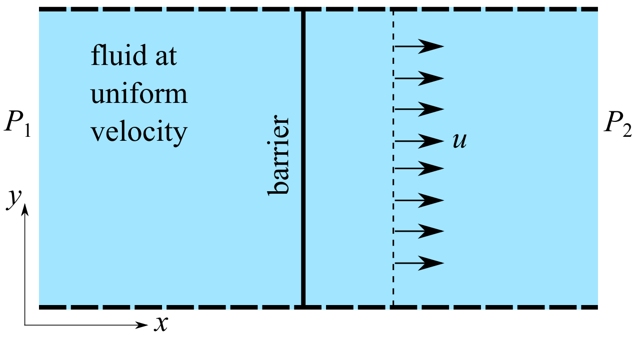


Figure 4.1: Plug flow through a tube, blocked by a barrier. In the exact solution, u must be zero, but weak enforcement techniques can allow leakage.

1. Steadiness: $\lambda^{n+1} = \lambda^n$. Define this constant value to be λ^∞. It must satisfy $\lambda^\infty = \left(\lambda^\infty + \tau^B_{\text{NOR}} u\right)/(1+r)$.
2. Equilibrium: $\lambda^\infty + \tau^B_{\text{NOR}} u = \Delta P$; the multiplier and penalty must balance the pressure drop to preclude acceleration of the fluid plug.

It follows that the leakage, u, is given by

$$u = \frac{r \Delta P}{\tau^B_{\text{NOR}}(1+r)} \ , \tag{4.40}$$

which asymptotes to inverse scaling with the penalty parameter as $r \to \infty$ and to zero as $r \to 0$. For a fixed nonzero value of r, steady leakage converges to zero with refinement at the same rate as it would for a pure penalty method, but, if r is an adjustable parameter, one may scale the leakage down to arbitrarily small levels without impacting the solvability of the discrete problem at each time step (because r appears only in the explicit multiplier update).

4.4.4.2 Spurious modes of λ

If λ is permitted to vary across the cross-section of the pipe, there are many choices which could satisfy the equilibrium condition. Any λ with $\int_\Gamma \lambda \, d\Gamma = 0$ could be added to an existing solution and corresponds to a "spurious mode" of the Lagrange multiplier field. Such modes will exist for richer fluid approximation spaces as well. Any λ that is $L^2(\Gamma)$-orthogonal to the discrete space of fluid–structure velocity differences will constitute such a spurious mode. The solution algorithm of Section 4.2 explicitly constructs the multiplier approximation as a linear combination of discrete fluid–structure velocity differences, and is therefore, in principle, immune to such spurious modes. In practice, though, the multiplier field can develop oscillations that are *nearly* orthogonal to the space of velocity differences. Without stabilization, they may become quite large, as demonstrated by the numerical experiments of Section 6.3. The analysis of Chapter 5 indicates that these oscillations may be viewed as storing energy, which can later be unleashed into the fluid and structure sub-problems. In principle, if oscillations in λ grow to extreme magnitudes, the subsequent release of energy could be catastrophic for the fluid velocity and structure displacement, but I have not been able to elicit such behavior, even with deliberate effort.

4.4.4.3 Conflicting boundary conditions

If a Dirichlet condition is applied to the plug flow, constraining u to have some nonzero value, then, when $r = 0$, λ will clearly diverge as $t \to \infty$. This corresponds, of course, to an ill-posed problem with contradictory boundary conditions, but such situations may be approached in practice, if, for instance, the immersed structure is forced into a fluid element whose nodes are subject to strongly-enforced Dirichlet boundary conditions. When $r > 0$, the semi-implicit time integration remains robust in this extreme limit: assuming again that λ takes on a single value across the barrier, the steadiness condition, $\lambda^\infty = \left(\lambda^\infty + \tau^B_{\text{NOR}} u\right) / (1 + r)$ implies that λ stabilizes at a value of $\tau^B_{\text{NOR}} u / r$ as $t \to \infty$.

Chapter 5

Analysis of linear model problems

An *a priori* convergence analysis of my discretization of the nonlinear problem stated in Chapter 2 is beyond the scope of this dissertation. Analysis of several simpler problems, however, can provide valuable insights into the behavior of the numerical method.

5.1 Convergence of the semi-implicit time integration

A critical question to address is whether the proposed semi-implicit time integration of the augmented Lagrangian is a fundamentally sound approach for enforcing Dirichlet boundary conditions in parabolic problems. Despite the closely-related feedback boundary condition's decades-long history of successful application to CFD and FSI (as reviewed in Section 4.4.3), I could not find any *a priori* analysis of its convergence. To investigate the conver-

---

Some of this chapter's content is derived from the following paper:

D. Kamensky, M.-C. Hsu, Y. Yu, J. A. Evans, M. S. Sacks, T. J. R. Hughes. Immersogeometric cardiovascular fluid–structure interaction analysis using divergence-conforming B-splines. *Computer Methods in Applied Mechanics and Engineering*, In review (preprint: ICES Report 16-14). D. Kamensky implemented the numerical methods, formulated and analyzed model problems, and participated in the experimental work. M.-C. Hsu provided supervision and edited the manuscript extensively. Y. Yu, J. A. Evans, and T. J. R. Hughes supervised the mathematical analysis. M. S. Sacks helped plan and supervise the laboratory experiments.

gence of the approach, I first introduce a linear, scalar model problem and prove convergence in that setting. I then discuss some simple extensions to problems that are more representative of FSI. I include numerical experiments indicating that the *a priori* analysis is not sharp; convergence is, in practice, faster than predicted. Throughout this chapter, I follow the common practice of considering the symbol "C" to stand for a generic constant that is independent of refinement parameters, but may represent different numerical values in different places.

5.1.1 Scalar parabolic model problem

I first analyze the behavior of the semi-implicit time integrator applied to a scalar parabolic problem with immersed boundaries. To summarize, the main steps of the analysis are:

- Relate the semi-implicit time-integration of the scalar parabolic problem to implicit time integration of a regularized feedback boundary condition problem with h- and Δt-dependent coefficients (Section 5.1.1.2).
- Show that solutions of the regularized problem converge to solutions of the original parabolic problem (Section 5.1.1.3).
- Analyze the spatial discretization error of the semi-discrete regularized problem (Section 5.1.1.5).
- Quantify the truncation error in time of implicit time integration of the semi-discrete regularized problem (Section 5.1.1.7).

5.1.1.1 Scalar parabolic problem statement

As in Chapter 2, I begin from a problem stated in weak form to more naturally accommodate the singular distributional forcing associated with immersed boundaries. A scalar field u, which I refer to as temperature, evolves through time according to a second-order parabolic PDE resembling the heat equation on a domain $\Omega \subset \mathbb{R}^d$, while satisfying homogeneous Dirichlet boundary conditions on $\partial\Omega$ and being constrained to have its trace on the immersed surface Γ equal to the function g, defined on Γ. An example of such a configuration is shown in Figure 5.1. In weak form, using a Lagrange multiplier to enforce the constraint on Γ, the problem is: Find $u \in L^2(0,T;H^1_0(\Omega))$ with $\partial_t u \in L^2(0,T;H^{-1}(\Omega))$ and $\lambda \in L^2(0,T;H^{-1/2}(\Gamma))$ such that for every $v \in H^1_0(\Omega)$ and $\delta\lambda \in H^{-1/2}(\Gamma)$ at a.e. $t \in [0,T]$,

$$\begin{aligned}
&\rho_{H^{-1}(\Omega)}\langle \partial_t u(t), v\rangle_{H^1(\Omega)} + a(u(t),v) \\
&\qquad + {}_{H^{-1/2}(\Gamma)}\langle \lambda(t), \gamma v\rangle_{H^{1/2}(\Gamma)} - {}_{H^{-1/2}(\Gamma)}\langle \delta\lambda, \gamma u(t)\rangle_{H^{1/2}(\Gamma)} \\
&\quad = {}_{H^{-1/2}(\Gamma)}\langle \delta\lambda, -g(t)\rangle_{H^{1/2}(\Gamma)} + (f(t),v)_{L^2(\Omega)}
\end{aligned} \tag{5.1}$$

and

$$u(0) = u_0 \in L^2(\Omega)\ , \tag{5.2}$$

where $\rho > 0$ is a scalar coefficient, γ is the trace operator mapping from $H^1(\Omega)$ to $H^{1/2}(\Gamma)$, ${}_{A^*}\langle\cdot,\cdot\rangle_A$ is a duality pairing between a space A and its dual, a is a bilinear form that is coercive and bounded over $H^1_0(\Omega)$, u_0 is an initial condition for u at time $t = 0$, $g(t)$ is the Dirichlet boundary data on Γ at time t, and $f(t)$ is a prescribed source term driving the temperature. The trace operator may be omitted in the sequel, when there is no risk of confusion. or background on the time-dependent spaces used in defining this problem, and an appropriate weak definition of the time derivative ∂_t, see [56, Section 5.9.2]. In particular, the meaningfulness of assigning an $L^2(\Omega)$ initial condition to u at time $t = 0$ (despite satisfying the equation at only a.e. time) is assured by [56, Section 5.9.2, Theorem 3(i)]. If Γ divides Ω into two subdomains, as in Figure 5.1, the existence, uniqueness, and regularity theory for second-order parabolic problems with Dirichlet boundary conditions can be applied in each subdomain. I make the important assumption that the Lagrange multiplier λ is in $L^2(\Gamma) \subset H^{-1/2}(\Gamma)$. This greatly simplifies comparisons between (5.1) and the regularized problem (5.6) introduced below, in which the surface force field corresponding to a regularized multiplier is in $L^2(\Gamma)$. I introduce further regularity assumptions on u and λ, as needed to complete the arguments below. Implicit in these assumptions are regularity constraints on Ω, Γ, f, and g, but, for brevity, I simply state their effects directly. A discussion of the effects of problem data on regularity of solutions to second-order parabolic problems

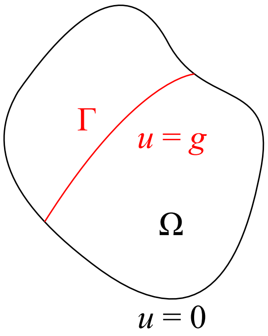


Figure 5.1: The domain Ω and the immersed boundary Γ.

may be found in [56, Section 7.1.3].

5.1.1.2 Semi-implicit time integration

I now define a semi-implicit algorithm for this scalar model problem that is analogous to the scheme proposed for nonlinear FSI in Chapter 4. At each time step,

1. Given u^n and λ^n, find u^{n+1} such that for all $v \in H_0^1(\Omega)$,

   $$\begin{aligned}\rho(u_t^{n+1}, v)_{L^2(\Omega)} + a(u^{n+1}, v) + (\lambda^n, \gamma v)_{L^2(\Gamma)} + \beta(\gamma u^{n+1} - g(t^{n+1}), \gamma v)_{L^2(\Gamma)} \\ = (f(t^{n+1}), v)_{L^2(\Omega)} \ , \end{aligned} \tag{5.3}$$

   where

   $$u_t^{n+1} = \frac{u^{n+1} - u^n}{\Delta t} \ . \tag{5.4}$$

2. Update the multiplier such that, for all $\delta\lambda \in L^2(\Gamma)$,

   $$(\lambda^{n+1}, \delta\lambda)_{L^2(\Gamma)} = \frac{1}{1+r} \left(\lambda^n + \beta(\gamma u^{n+1} - g(t^{n+1})), \delta\lambda\right)_{L^2(\Gamma)} \ . \tag{5.5}$$

As explained in Section 4.4.1, in the context of FSI, this is a backward Euler time integration of a regularized problem: Find $u^{\mathrm{reg}} \in L^2(0,T;H_0^1(\Omega))$ with $\partial_t u^{\mathrm{reg}} \in L^2(0,T;H^{-1}(\Omega))$ and $\lambda^{\mathrm{reg}} \in L^2(0,T;L^2(\Gamma))$ with $\partial_t \lambda^{\mathrm{reg}} \in L^2(0,T;L^2(\Gamma))$ such that for all $v \in H_0^1(\Omega)$ and $\delta\lambda \in L^2(\Gamma)$ at a.e. $t \in [0,T]$,

$$\begin{aligned}&\rho_{H^{-1}(\Omega)}\langle \partial_t u^{\mathrm{reg}}, v\rangle_{H^1(\Omega)} + a(u^{\mathrm{reg}}, v) + (\lambda^{\mathrm{reg}}, \gamma v)_{L^2(\Gamma)} \\ &\quad + \left(\frac{\Delta t}{(1+r)\beta}\partial_t \lambda^{\mathrm{reg}} - (\gamma u^{\mathrm{reg}} - g) + \frac{r}{(1+r)\beta}\lambda^{\mathrm{reg}}, \delta\lambda\right)_{L^2(\Gamma)} \\ &\quad = (f, v)_{L^2(\Omega)} \ , \end{aligned} \tag{5.6}$$

$$u^{\text{reg}}(0) = u_0 \in L^2(\Omega) \ , \tag{5.7}$$

and

$$\lambda^{\text{reg}}(0) = \lambda_0 \in L^2(\Gamma) \ . \tag{5.8}$$

The meaningfulness of assigning an $L^2(\Gamma)$ initial condition to λ^{reg} at the point $t = 0$ is assured by [56, Section 5.9.2, Theorem 2(i)], with $X = L^2(\Gamma)$.

The existence and uniqueness of solutions to the problem (5.6) for constant values of β and Δt follows from the usual Faedo–Galerkin argument for parabolic problems, which is executed in detail for the heat equation in [56, Section 7.1.2]. Briefly, existence theory for ordinary differential equations is applied to a sequence of Galerkin approximations and uniform energy bounds on the sequence of solutions are used to obtain a unique weak limit satisfying the weak PDE. In adapting the proof from [56, Section 7.1.2], a suitable basis for approximating λ^{reg} would be the eigenfunctions of the Laplace–Beltrami operator on Γ. This does *not* imply any sort of uniformity of energy estimates with respect to the limits $\beta \to \infty$ and/or $\Delta t \to 0$. Robustness in those limits is derived separately in the sequel.

5.1.1.3 Convergence of the regularized problem

My first step toward showing that the output of the semi-implicit algorithm converges to a solution of the parabolic problem (5.1) is to show that the solution of the regularized problem (5.6) converges to the solution of (5.1). This portion of the analysis may be of interest beyond the narrow context of studying the semi-implicit algorithm proposed in this dissertation, as the convergence of (5.6) to (5.1) is applicable to other discretizations of feedback boundary conditions. (Nearly all of the examples cited in Section 4.4.3 used explicit time integration.) Denote the error between the solutions to (5.1) and (5.6) by

$$(e_u, e_\lambda) = (u^{\text{reg}} - u, \lambda^{\text{reg}} - \lambda) \ . \tag{5.9}$$

One can derive a bound on the $L^2(\Omega)$ error in temperature at time T by bounding the energy norm

$$|||e_u(T), e_\lambda(T)|||^2 = \frac{1}{2}\rho \left\| e_u(T) \right\|^2_{L^2(\Omega)} + \frac{\Delta t}{2(1+r)\beta} \left\| e_\lambda(T) \right\|^2_{L^2(\Gamma)} \ . \tag{5.10}$$

Taking the difference between (5.6) and (5.1) (restricting $\delta\lambda$ in (5.1) to $L^2(\Gamma) \subset H^{-1/2}(\Gamma)$ and assuming that $\lambda(t)$ and $\partial_t\lambda(t)$ are in $L^2(\Gamma)$ for a.e. $t \in [0, T]$),

$$\begin{aligned} &\rho_{H^{-1}(\Omega)} \left\langle \partial_t e_u, v \right\rangle_{H^1(\Omega)} + a(e_u, v) + (e_\lambda, \gamma v)_{L^2(\Gamma)} \\ &\quad + \left(\frac{\Delta t}{(1+r)\beta} \partial_t \lambda^{\text{reg}} - \gamma e_u + \frac{r}{(1+r)\beta} \lambda^{\text{reg}}, \delta\lambda \right)_{L^2(\Gamma)} = 0 \ . \end{aligned} \tag{5.11}$$

Add and subtract

$$\left(\frac{\Delta t}{(1+r)\beta} \partial_t \lambda + \frac{r}{(1+r)\beta} \lambda, \delta\lambda \right)_{L^2(\Gamma)} \tag{5.12}$$

from the left-hand side of (5.11) to obtain

$$\begin{aligned} &\rho_{H^{-1}(\Omega)} \left\langle \partial_t e_u, v \right\rangle_{H^1(\Omega)} + a(e_u, v) + (e_\lambda, v)_{L^2(\Gamma)} \\ &\quad + \left(\frac{\Delta t}{(1+r)\beta} \partial_t e_\lambda - e_u + \frac{r}{(1+r)\beta} e_\lambda, \delta\lambda \right)_{L^2(\Gamma)} \\ &\quad + \left(\frac{\Delta t}{(1+r)\beta} \partial_t \lambda + \frac{r}{(1+r)\beta} \lambda, \delta\lambda \right)_{L^2(\Gamma)} = 0 \ . \end{aligned} \tag{5.13}$$

Setting $v = e_u$ and $\delta\lambda = e_\lambda$, then applying [56, Section 5.9.2, Theorem 3(ii)] to simplify the $H^1(\Omega)$ duality pairing,

$$\begin{aligned}\partial_t |||e_u, e_\lambda|||^2 = & -a(e_u, e_u) - \frac{r}{(1+r)\beta}\|e_\lambda\|^2_{L^2(\Gamma)} \\ & - \frac{\Delta t}{(1+r)\beta}(\partial_t \lambda, e_\lambda)_{L^2(\Gamma)} - \frac{r}{(1+r)\beta}(\lambda, e_\lambda)_{L^2(\Gamma)} \ . \end{aligned} \tag{5.14}$$

Applying Young's inequality to the last two terms above,

$$\begin{aligned}\partial_t |||e_u, e_\lambda|||^2 \leq & - a(e_u, e_u) - \frac{r}{2(1+r)\beta}\|e_\lambda\|^2_{L^2(\Gamma)} \\ & + \frac{\Delta t}{2(1+r)\beta}\left(T\|\partial_t \lambda\|^2_{L^2(\Gamma)} + \frac{1}{T}\|e_\lambda\|^2_{L^2(\Gamma)}\right) \\ & + \frac{r}{2(1+r)\beta}\|\lambda\|^2_{L^2(\Gamma)} \ . \end{aligned} \tag{5.15}$$

Adding the non-negative term

$$+\frac{\rho}{2}\|e_u\|^2_{L^2(\Omega)} + a(e_u, e_u) + \frac{r}{2(1+r)\beta}\|e_\lambda\|^2_{L^2(\Gamma)} \tag{5.16}$$

to the right-hand side,

$$\begin{aligned}\partial_t |||e_u, e_\lambda|||^2 \leq & \frac{1}{T}|||e_u, e_\lambda|||^2 + \frac{T\Delta t}{2(1+r)\beta}\|\partial_t \lambda\|^2_{L^2(\Gamma)} \\ & + \frac{r}{2(1+r)\beta}\|\lambda\|^2_{L^2(\Gamma)} \ . \end{aligned} \tag{5.17}$$

Then, assuming $e_u(0) = 0$ and $e_\lambda(0) = 0$, Grönwall's lemma bounds the error at time T:

$$\begin{aligned}|||e_u(T), e_\lambda(T)|||^2 \leq & C\int_0^T \left(\frac{T\Delta t}{2(1+r)\beta}\|\partial_t \lambda\|^2_{L^2(\Gamma)} \right. \\ & \left. + \frac{r}{2(1+r)\beta}\|\lambda\|^2_{L^2(\Gamma)}\right) \, dt \ . \end{aligned} \tag{5.18}$$

Thus

$$\frac{\rho}{2}\|e_u\|^2_{L^2(\Omega)} \leq |||e_u, e_\lambda|||^2 \to 0 \tag{5.19}$$

as $\beta \to \infty$. In the case of $r = 0$, we have this convergence when $\Delta t \to 0$ at fixed β. A formally similar argument produces an analogous estimate for $(\rho/2)\|\partial_t^N e_u\|^2_{L^2(\Omega)}$ with $N \geq 1$, so long as the solution to (5.1) is sufficiently regular:

$$\begin{aligned}\left|\left|\left|\partial_t^N e_u(T), \partial_t^N e_\lambda(T)\right|\right|\right|^2 \leq & C\int_0^T \left(\frac{T\Delta t}{2(1+r)\beta}\|\partial_t^{N+1}\lambda\|^2_{L^2(\Gamma)}\right. \\ & \left. + \frac{r}{2(1+r)\beta}\|\partial_t^N \lambda\|^2_{L^2(\Gamma)}\right)\, dt\ .\end{aligned} \tag{5.20}$$

Ensuring that $\partial_t^N e_u(0) = \partial_t^N e_\lambda(0) = 0$ (as assumed for the $N = 0$ case above, to arrive at (5.18)) requires the assumption that the multiplier solution $\lambda(t)$ of (5.1) evolves sufficiently smoothly from an initial value of zero. This is a somewhat restrictive assumption, but still admits many nontrivial solutions. For example, it can easily be satisfied by starting up the forcing functions $f(t)$ and $g(t)$ smoothly, to disturb a homogeneous initial temperature field. I ignore this condition altogether in numerical experiments, with no apparent effect on the convergence of the method.

Remark 5.1. Similar assumptions on problem data derivatives at $t = 0$ are made in [71, Section 4.3], while analyzing the convergence of an artificial compressibility scheme. (Recall the analogy of Section 4.4.2.) The cited work acknowledges that these restrictions are not especially realistic and [71, Remark 4.2] suggests that they might be weakened by using time-weighted norms, referring to [156, Lemma 3.2] as an example. The numerical examples of [71, Section 6] ignore the assumptions about data at $t = 0$ while still exhibiting the desired convergence rates.

Remark 5.2. It is clear from the numerical experiments in the sequel that the rates of convergence with respect to β and Δt are not sharp. Y. Yu has sketched proofs of some sharper estimates of the error between solutions to (5.1) and (5.6), based on adapting the duality arguments in [72, Section 2.3].

Remark 5.3. One might also try to optimize the choices of r and β, to minimize e_u and e_λ, but such optimizations can come into conflict with competing demands on these parameters, from the spatial discretization. For instance, faster convergence of e_u with respect to Δt can be obtained by selecting $r = C\Delta t/T$ above, but this hurts convergence of the bound (5.46) on spatial discretization error below.

5.1.1.4 Uniform bound in $H^{3/2-\epsilon}(\Omega)$

Useful interpolation error bounds in finite element spaces require a bound on $u^{\mathrm{reg}}(T)$ in a norm stronger than $\|\cdot\|_{H^1(\Omega)}$. Such bounds need to be uniform in the refinement limits of $\beta \to \infty$ and $\Delta t \to 0$. A uniform bound on u^{reg} in the $H^{3/2-\epsilon}(\Omega)$ norm may be found using elliptic regularity given a uniform $L^2(\Gamma)$ bound on λ^{reg}. In the case of $r = \mathcal{O}(\Delta t)$, such a uniform bound follows immediately from (5.18). If r goes to zero more slowly than $C\Delta t$, the uniform bound on $\|\lambda^{\mathrm{reg}}\|_{L^2(\Gamma)}$ must be derived separately. This is the case of interest, since the suggested scaling of r from Section 4.1 is $r = C$.

Assume that $\beta > 0$, $\Delta t = C/\beta$, and $r = C$, and consider norms of (time derivatives of) the solution to the original problem (5.1) to be constant (since this solution is independent of β and Δt). Then, for β sufficiently large, (5.20) can be re-written as

$$|||\partial_t e_u(T), \partial_t e_\lambda(T)|||^2 \leq \frac{C}{\beta}\ . \tag{5.21}$$

This implies

$$\frac{\rho}{2}\|\partial_t e_u(T)\|^2_{L^2(\Omega)} \leq \frac{C}{\beta} \tag{5.22}$$

$$\Rightarrow \|\partial_t e_u(T)\|_{L^2(\Omega)} \leq \frac{C}{\sqrt{\beta}} \tag{5.23}$$

and

$$\frac{\Delta t}{2(1+r)\beta}\,\|\partial_t e_\lambda(T)\|^2_{L^2(\Gamma)} \leq \frac{C}{\beta} \tag{5.24}$$

$$\Rightarrow \|\partial_t e_\lambda(T)\|^2_{L^2(\Gamma)} \leq C\beta \tag{5.25}$$

$$\Rightarrow \left\|\frac{\Delta t}{(1+r)\beta}\partial_t e_\lambda(T)\right\|_{L^2(\Gamma)} \leq \frac{C}{\beta^{3/2}}\ . \tag{5.26}$$

The above-bounded terms may be cast as part of the prescribed forcing in an elliptic problem for the error at time T. Consider re-arranging (5.13) at time T into the problem: Find $e_u(T)$ and $e_\lambda(T)$ such that for all v and $\delta\lambda$

$$\begin{aligned}
&a(e_u(T), v) + (e_\lambda(T), v)_{L^2(\Gamma)} - (e_u(T), \delta\lambda)_{L^2(\Gamma)} \\
&\quad + \frac{r}{(1+r)\beta}\,(e_\lambda(T), \delta\lambda)_{L^2(\Gamma)} \\
&= -\rho(\partial_t e_u(T), v)_{L^2(\Omega)} - \left(\frac{\Delta t}{(1+r)\beta}\partial_t e_\lambda(T), \delta\lambda\right)_{L^2(\Gamma)} \\
&\quad - \left(\frac{\Delta t}{(1+r)\beta}\partial_t\lambda(T) + \frac{r}{(1+r)\beta}\lambda(T), \delta\lambda\right)_{L^2(\Gamma)}\ ,
\end{aligned} \tag{5.27}$$

where $\partial_t e_u(T)$ and $\partial_t e_\lambda(T)$ on the right-hand side of the equation are *not* considered to be unknown. In (5.27), these functions are fixed data, subject to the bounds (5.23) and (5.26). The left-hand side bilinear form of (5.27),

$$\begin{aligned} B_{\text{steady}}((u,\lambda),(v,\delta\lambda)) =& a(u,v) + (\lambda, v)_{L^2(\Gamma)} - (u, \delta\lambda)_{L^2(\Gamma)} \\ &+ \frac{r}{(1+r)\beta} (\lambda, \delta\lambda)_{L^2(\Gamma)} \ , \end{aligned} \tag{5.28}$$

is coercive and bounded in the norm

$$\|u,\lambda\|^2_{\text{steady}} = \beta \|u\|^2_{H^1(\Omega)} + \|\lambda\|^2_{L^2(\Gamma)} \ . \tag{5.29}$$

The coercivity constant is clearly seen to be C/β:

$$\begin{aligned} B_{\text{steady}}((u,\lambda),(u,\lambda)) &= a(u,u) + (\lambda, v)_{L^2(\Gamma)} - (\lambda, v)_{L^2(\Gamma)} \\ &\quad + \frac{r}{(1+r)\beta} \|\lambda\|^2_{L^2(\Gamma)} && (5.30) \\ &\geq C \left(\|u\|^2_{H^1(\Omega)} + \frac{1}{\beta} \|\lambda\|^2_{L^2(\Gamma)} \right) && (5.31) \\ &\geq \frac{C}{\beta} \|u,\lambda\|^2_{\text{steady}} \ . && (5.32) \end{aligned}$$

The dual norm (induced by (5.29)) of the right-hand side functional,

$$\begin{aligned} F_{\text{steady}}((v,\delta\lambda)) =& -\rho(\partial_t e_u(T), v)_{L^2(\Omega)} - \left(\frac{\Delta t}{(1+r)\beta} \partial_t e_\lambda(T), \delta\lambda \right)_{L^2(\Gamma)} \\ & - \left(\frac{\Delta t}{(1+r)\beta} \partial_t \lambda(T) + \frac{r}{(1+r)\beta} \lambda(T), \delta\lambda \right)_{L^2(\Gamma)} \ , \end{aligned} \tag{5.33}$$

is also C/β, for β sufficiently large. Using the Lax–Milgram theorem to bound the solution of (5.27) in terms of the coercivity constant and the right-hand side functional norm,

$$\|e_u(T), e_\lambda(T)\|_{\text{steady}} \leq \frac{1}{C/\beta} \|F_{\text{steady}}(\cdot)\|_{\text{steady}} \leq C \ . \tag{5.34}$$

Based on the definition of $\|\cdot\|_{\text{steady}}$, (5.34) provides a β-independent bound on $\|e_\lambda(T)\|_{L^2(\Gamma)}$. Using the uniform bound on $\|\lambda^{\text{reg}}\|_{L^2(\Gamma)}$ which immediately follows,

$$(\lambda^{\text{reg}}, \gamma(\cdot))_{L^2(\Gamma)} \tag{5.35}$$

is bounded independently of β over the set of functions on Ω with traces in $L^2(\Gamma)$. This set of functions is $H^{1/2+\epsilon}(\Omega)$. Thus the functional (5.35) is in the dual space $H^{-(1/2+\epsilon)}(\Omega)$, and its induced norm is bounded uniformly with respect to β. Using elliptic regularity theory and the uniform bound on the $L^2(\Omega)$ norm of $\partial_t e_u(T)$, $u^{\text{reg}}(T)$ is bounded in $H^{3/2-\epsilon}(\Omega)$, independently of β.

5.1.1.5 Spatial discretization of the modified equation

Consider, now, a semi-discrete counterpart of problem (5.6), posed over finite element spaces for $u^{\text{reg}}(t)$. I formally consider $\lambda^{\text{reg}}(t)$ and $\delta\lambda$ to be in the infinite-dimensional space $L^2(\Gamma)$, yet, due to the structure of the problem, λ^{reg} will clearly stay within the finite-dimensional trace space of the discrete temperature so long as its initial condition and the data g are also in this space. This can be verified in the semi-discrete setting by deriving a solution of $\lambda^{\text{reg}}(t)$ in terms of $u^{\text{reg}}(t)$ at a fixed point on Γ (cf. [104, (3.35)–(3.38)]). In the fully-discrete setting, the finite dimensionality of λ^n is clear from its explicit update formula and the closure of the discrete trace space under linear combination. The semi-discrete numerical method is to find $u^h \in V_u^h \subset H_0^1(\Omega)$

and $\lambda^h \in L^2(\Gamma)$ such that, for every $v^h \in V_u^h$ and $\delta\lambda \in L^2(\Gamma)$,

$$\begin{aligned}&\rho(\partial_t u^h, v^h)_{L^2(\Omega)} + a(u^h, v^h) + (\lambda^h, v^h)_{L^2(\Gamma)} \\ &\quad + \left(\frac{\Delta t}{(1+r)\beta}\partial_t \lambda^h - \big(u^h - g\big) + \frac{r}{(1+r)\beta}\lambda^h, \delta\lambda\right)_{L^2(\Gamma)} = \big(f, v^h\big)_{L^2(\Omega)} \ .\end{aligned} \tag{5.36}$$

The $L^2(\Omega)$ inner product used above to represent $\partial_t u^h \in H^{-1}(\Omega)$ is appropriate in the finite dimensional setting. The semi-discrete errors in the velocity and multiplier fields are

$$e_u^{\text{reg}} = u^h - u^{\text{reg}} = (u^h - \bar{u}^h) + (\bar{u}^h - u^{\text{reg}}) = \xi_u^h + \eta_u \tag{5.37}$$

and

$$e_\lambda^{\text{reg}} = \lambda^h - \lambda^{\text{reg}} \ , \tag{5.38}$$

The function $\bar{u}^h \in V_u^h$ is an arbitrary interpolant, used to split the velocity error into discrete and interpolation components. Because, as discussed above, the multiplier test space is considered to be all of $L^2(\Gamma)$, there is no reason to perform a splitting of the multiplier error, since the "interpolation" component will be zero. The velocity interpolation error is defined as an elliptic projection of u^{reg} into the discrete space V_u^h. Specifically,

$$a(\bar{u}^h, v^h) = a(u^{\text{reg}}, v^h) \qquad \forall v^h \in V_u^h \ . \tag{5.39}$$

As argued earlier, u^{reg} is uniformly bounded in $H^{3/2-\epsilon}(\Omega)$, so it is reasonable to assume

$$\|\eta_u\|^2_{H^1(\Omega)} \le Ch^{1-2\epsilon} \ . \tag{5.40}$$

An Aubin–Nitsche-type duality argument produces

$$\|\eta_u\|^2_{L^2(\Omega)} \leq Ch^{3-2\epsilon} \ , \tag{5.41}$$

from which one can obtain

$$\|\eta_u\|^2_{L^2(\Gamma)} \leq Ch^{2-2\epsilon} \tag{5.42}$$

by assuming a trace estimate on Γ. Using Galerkin orthogonality (i.e. consistency of (5.36) with (5.6)) and setting $v^h = \xi^h_u$ and $\delta\lambda = e^{\mathrm{reg}}_\lambda$,

$$\begin{aligned} &\rho(\partial_t e^{\mathrm{reg}}_u, \xi^h_u)_{L^2(\Omega)} + a(e^{\mathrm{reg}}_u, \xi^h_u) + (e^{\mathrm{reg}}_\lambda, \xi^h_u)_{L^2(\Gamma)} \\ &\quad + \left(\frac{\Delta t}{(1+r)\beta}\partial_t e^{\mathrm{reg}}_\lambda - e^{\mathrm{reg}}_u + \frac{r}{(1+r)\beta} e^{\mathrm{reg}}_\lambda, e^{\mathrm{reg}}_\lambda\right)_{L^2(\Gamma)} = 0 \ . \end{aligned} \tag{5.43}$$

Invoking the velocity error splitting (5.37), this becomes

$$\begin{aligned} &\rho(\partial_t \eta_u, \xi^h_u)_{L^2(\Omega)} + \rho(\partial_t \xi^h_u, \xi^h_u)_{L^2(\Omega)} + a(\xi^h_u, \xi^h_u) \\ &\quad + \left(\frac{\Delta t}{(1+r)\beta}\partial_t e^{\mathrm{reg}}_\lambda - \eta_u + \frac{r}{(1+r)\beta} e^{\mathrm{reg}}_\lambda, e^{\mathrm{reg}}_\lambda\right)_{L^2(\Gamma)} = 0 \ . \end{aligned} \tag{5.44}$$

Re-arranging terms,

$$\begin{aligned} &\partial_t \left(\frac{\rho}{2}\|\xi^h_u\|^2_{L^2(\Omega)} + \frac{\Delta t}{2(1+r)\beta}\|e^{\mathrm{reg}}_\lambda\|^2_{L^2(\Gamma)}\right) \\ &\quad = -\rho(\partial_t \eta_u, \xi^h_u)_{L^2(\Omega)} - a(\xi^h_u, \xi^h_u) + (\eta_u, e^{\mathrm{reg}}_\lambda)_{L^2(\Gamma)} \\ &\qquad - \frac{r}{(1+r)\beta}\|e^{\mathrm{reg}}_\lambda\|^2_{L^2(\Gamma)} \ . \end{aligned} \tag{5.45}$$

Applying Young's inequality on the right and adding strictly positive terms to the upper bound,

$$\partial_t \left|\left|\left| \xi^h_u, e^{\mathrm{reg}}_\lambda \right|\right|\right|^2 \leq \frac{1}{T}\left|\left|\left| \xi^h_u, e^{\mathrm{reg}}_\lambda \right|\right|\right|^2 + \frac{T\rho}{2}\|\partial_t \eta_u\|^2_{L^2(\Omega)} + \frac{2(1+r)\beta}{r}\|\eta_u\|^2_{L^2(\Gamma)} \ . \tag{5.46}$$

Grönwall's lemma then gives a bound on $\left|\left|\left|\xi_u^h(T), e_\lambda^{\text{reg}}(T)\right|\right|\right|$ (and, by triangulation, $|||e_u^{\text{reg}}(T), e_\lambda^{\text{reg}}(T)|||$) in terms of the interpolation error η_u. The only hazard is that $\|\eta_u\|^2_{L^2(\Gamma)}$ must converge faster than β/r diverges. If $\beta = C/h$, $r = C$, and the interpolation error is bounded like (5.42), the temperature should converge at a rate $\geq 1/2$ in $L^2(\Omega)$. A similar argument can bound time derivatives of the semi-discrete error, if the problem data is sufficiently nice at $t = 0$ to ensure that $\partial_t^N e_u^{\text{reg}}(0) = \partial_t^N e_\lambda^{\text{reg}}(0) = 0$.

5.1.1.6 Semi-discrete convergence for $r = 0$

The error bound that follows from (5.46) clearly fails in the limit of $r \to 0$. Computations with $r = 0$, on the other hand, seem to proceed without major issues and enjoy better conservation properties in the steady limit. Let us first attempt to backtrack in the argument of the previous section. Let us try an alternate approach, starting over from (5.45). In the case of $r = 0$, (5.45) becomes

$$\begin{aligned}\partial_t &\left(\frac{\rho}{2}\|\xi_u^h\|^2_{L^2(\Omega)} + \frac{1}{2k}\|e_\lambda^{\text{reg}}\|^2_{L^2(\Gamma)}\right) \\ &= -\rho(\partial_t\eta_u, \xi_u^h)_{L^2(\Omega)} - a(\xi_u^h, \xi_u^h) + (\eta_u, e_\lambda^{\text{reg}})_{L^2(\Gamma)} \ , \end{aligned} \tag{5.47}$$

where I have defined $k = \beta/\Delta t$, to simplify notation. Applying Young's inequality to terms in the right-hand side of (5.47) and adding strictly non-negative terms to the upper bound,

$$\partial_t\left|\left|\left|\xi_u^h, e_\lambda^{\text{reg}}\right|\right|\right|^2 \leq \frac{1}{T}\left|\left|\left|\xi_u^h, e_\lambda^{\text{reg}}\right|\right|\right|^2 + \frac{T\rho}{2}\left\|\partial_t\eta_u\right\|^2_{L^2(\Omega)} + \frac{Tk}{2}\left\|\eta_u\right\|^2_{L^2(\Gamma)} \ . \tag{5.48}$$

Using the interpolation estimate (5.42) and Grönwall's lemma,

$$\left|\left|\left|\xi_u^h(T), e_\lambda^{\mathrm{reg}}(T)\right|\right|\right|^2 \leq Ckh^{2-\epsilon} \tag{5.49}$$

(for k sufficiently large). Applying $\partial_t^N(\cdot)$ to the trial functions and forcing throughout the entire semi-discrete problem and carrying out a similar analysis,

$$\left|\left|\left|\partial_t^N \xi_u^h(T), \partial_t^N e_\lambda^{\mathrm{reg}}(T)\right|\right|\right|^2 \leq Ckh^{2-\epsilon} \ , \tag{5.50}$$

given sufficient time-regularity of the original problem (5.1). As with (5.20), the use of Grönwall's lemma with zero initial time derivatives of the error relies on the assumption that source terms in (5.1) evolve smoothly from zero, although, as evidenced by the numerical examples, this assumption is likely not strictly necessary for convergence of the numerical method. Based on the scalings $\beta \sim 1/h$ and $\Delta t \sim h$ assumed above ($\Rightarrow k \sim 1/h^2$), these error estimates are not very appealing. The semi-discrete solution is essentially bounded, but not convergent. One can try to improve on this situation using a duality argument.

To execute the duality argument, I first need to establish $H^1(\Omega)$ stability of the semi-discrete solution e_u^{reg}, i.e. $\|e_u^{\mathrm{reg}}(s)\|_{H^1(\Omega)} \leq C$ for all $s \in (0, T)$. In keeping with the goals of the present section, consider only the case $r = 0$. Recalling the error splitting (5.37) and the definition of the interpolant $\overline{u}^h$, it is clear that the semi-discrete error satisfies

$$\begin{aligned}\rho\left(\partial_t e_u^{\mathrm{reg}}, v^h\right)_{L^2(\Omega)} + a\left(\xi_u^h, v^h\right) + \left(e_\lambda^{\mathrm{reg}}, v^h\right)_{L^2(\Gamma)}& \\ + \left(\frac{1}{k}\partial_t e_\lambda^{\mathrm{reg}} - e_u^{\mathrm{reg}}, \delta\lambda\right)_{L^2(\Gamma)} &= 0 \qquad \forall\left(v^h, \delta\lambda\right)\ . \end{aligned} \tag{5.51}$$

Setting $\left\{v^h, \delta\lambda\right\} = \left\{\partial_t \xi_u^h, 0\right\}$ and using the symmetry of a,

$$\rho\left(\partial_t e_u^{\mathrm{reg}}, \partial_t \xi_u^h\right)_{L^2(\Omega)} + \frac{1}{2}\partial_t a\left(\xi_u^h, \xi_u^h\right) + \left(e_\lambda^{\mathrm{reg}}, \partial_t \xi_u^h\right)_{L^2(\Gamma)} = 0\ . \tag{5.52}$$

For all $\delta\lambda \in L^2(\Gamma)$, $\left(\xi_u^h + \eta_u, \delta\lambda\right)_{L^2(\Gamma)} = \left(\frac{1}{k}\partial_t e_\lambda^{\mathrm{reg}}, \delta\lambda\right)_{L^2(\Gamma)}$, so (5.52) implies that

$$\rho\left(\partial_t e_u^{\mathrm{reg}}, \partial_t \xi_u^h\right)_{L^2(\Omega)} + \frac{1}{2}\partial_t a\left(\xi_u^h, \xi_u^h\right) + \left(e_\lambda^{\mathrm{reg}}, \frac{1}{k}\partial_t^2 e_\lambda^{\mathrm{reg}} - \partial_t \eta_u\right)_{L^2(\Gamma)} = 0\ . \tag{5.53}$$

Integrating (5.53) in time and applying the Cauchy–Schwarz and Young inequalities,

$$\begin{aligned} & a\left(\xi_u^h(s), \xi_u^h(s)\right) \\ &\quad \le 2\int_0^s \Bigg(\rho\left\|\partial_t e_u^{\mathrm{reg}}(t)\right\|_{L^2(\Omega)}\left\|\partial_t \xi_u^h(t)\right\|_{L^2(\Omega)} + \frac{1}{2k}\Bigg(\left(\frac{1}{T^2} + \frac{1}{T}\right)\left\|e_\lambda^{\mathrm{reg}}(t)\right\|_{L^2(\Gamma)}^2 \\ &\qquad\qquad + T^2\left\|\partial_t^2 e_\lambda^{\mathrm{reg}}(t)\right\|_{L^2(\Gamma)}^2 + Tk^2\left\|\partial_t \eta_u\right\|_{L^2(\Gamma)}^2\Bigg)\Bigg)\, dt \\ &\qquad + a\left(\xi_u^h(0), \xi_u^h(0)\right) \qquad\qquad (5.54) \\ &\quad \le 2\int_0^T \Bigg(\rho\left\|\partial_t e_u^{\mathrm{reg}}(t)\right\|_{L^2(\Omega)}\left\|\partial_t \xi_u^h(t)\right\|_{L^2(\Omega)} + \frac{1}{2k}\Bigg(\left(\frac{1}{T^2} + \frac{1}{T}\right)\left\|e_\lambda^{\mathrm{reg}}(t)\right\|_{L^2(\Gamma)}^2 \\ &\qquad\qquad + T^2\left\|\partial_t^2 e_\lambda^{\mathrm{reg}}(t)\right\|_{L^2(\Gamma)}^2 + Tk^2\left\|\partial_t \eta_u\right\|_{L^2(\Gamma)}^2\Bigg)\Bigg)\, dt \\ &\qquad + a\left(\xi_u^h(0), \xi_u^h(0)\right)\ . \qquad\qquad (5.55) \end{aligned}$$

Recalling (5.50) and interpolation estimates bounding η_u, then using the $H^1(\Omega)$ coercivity of a, we have (to within a factor of $h^{-\epsilon}$) the desired $H^1(\Omega)$

stability of e_u^{reg}:

$$\|e_u^{\mathrm{reg}}(s)\|_{H^1(\Omega)} \leq Ch^{-\epsilon} \ . \tag{5.56}$$

I now proceed with the duality argument, to sharpen (5.49) into a convergent error estimate. Consider the dual problem: Find $w \in L^2(0,T;H_0^1(\Omega))$ and $\omega \in L^2(0,T;H^{-1/2}(\Gamma))$ such that for all $\delta w \in H_0^1(\Omega)$ and $\delta\omega \in H^{-1/2}(\Gamma)$

$$\begin{aligned}\rho_{H^{-1}(\Omega)} \left\langle \partial_t w(t), \delta w \right\rangle_{H^1(\Omega)} - a\left(w(t), \delta w\right) +_{H^{-1/2}(\Gamma)} \left\langle \omega(t), \gamma \delta w \right\rangle_{H^{1/2}(\Gamma)} \\ -_{H^{-1/2}(\Gamma)} \left\langle \delta\omega, \gamma w(t) \right\rangle_{H^{1/2}(\Gamma)} = \left(e_u^{\mathrm{reg}}(t), \delta w\right)_{L^2(\Omega)} \ ,\end{aligned} \tag{5.57}$$

subject to the *final* condition that $w(T) = 0$. This problem should be viewed as evolving backwards through time; by examining the signs of the first two terms, it is clear that (5.57) is unstable in the forward time direction. Note that the trace of w is constrained to be zero on Γ, by the Lagrange multiplier ω. In view of the regularity of the source term, consider $\partial_t w(t) \in L^2(\Omega)$, so that the duality pairing can be re-written as an $L^2(\Omega)$ inner product. Inserting the test functions $\delta w = e_u^{\mathrm{reg}}(t)$ and $\delta\omega = e_\lambda^{\mathrm{reg}}(t)$,

$$\begin{aligned}\|e_u^{\mathrm{reg}}\|_{L^2(\Omega)}^2 = \rho\left(\partial_t w, e_u^{\mathrm{reg}}\right)_{L^2(\Omega)} - a\left(w, e_u^{\mathrm{reg}}\right) +_{H^{-1/2}(\Gamma)} \left\langle \omega, \gamma e_u^{\mathrm{reg}} \right\rangle_{H^{1/2}(\Gamma)} \\ -_{H^{-1/2}(\Gamma)} \left\langle e_\lambda^{\mathrm{reg}}, \gamma w \right\rangle_{H^{1/2}(\Gamma)} \ .\end{aligned} \tag{5.58}$$

Adding and subtracting $\rho\left(w, \partial_t e_u^{\mathrm{reg}}\right)_{L^2(\Omega)}$ on the right and using the symmetry of a and the fact that $\gamma w = 0$ (in an appropriate weak sense),

$$\begin{aligned}\|e_u^{\mathrm{reg}}\|_{L^2(\Omega)}^2 = \ \rho\partial_t\left(w, e_u^{\mathrm{reg}}\right)_{L^2(\Omega)} - \Big(\rho\left(\partial_t e_u^{\mathrm{reg}}, w\right)_{L^2(\Omega)} + a\left(e_u^{\mathrm{reg}}, w\right)\Big) \\ +_{H^{-1/2}(\Gamma)} \left\langle \omega, \gamma e_u^{\mathrm{reg}} \right\rangle_{H^{1/2}(\Gamma)} \ .\end{aligned} \tag{5.59}$$

Let w^h be a function in the discrete temperature space that satisfies the condition $\gamma w^h = 0$. A non-trivial finite element function in the discrete space satisfying this condition could be constructed, for instance, by assigning all nodes of elements intersecting Γ to zero, while allowing effective interpolation away from Γ. Due to the consistency of the semi-discrete problem with the regularized problem (5.6),

$$\begin{aligned}\|e_u^{\mathrm{reg}}\|^2_{L^2(\Omega)} = \rho\partial_t\left(w, e_u^{\mathrm{reg}}\right)_{L^2(\Omega)} - \Big(\rho\left(\partial_t e_u^{\mathrm{reg}}, w - w^h\right)_{L^2(\Omega)} + a\left(e_u^{\mathrm{reg}}, w - w^h\right)\Big) \\ + {}_{H^{-1/2}(\Gamma)}\langle\omega, \gamma e_u^{\mathrm{reg}}\rangle_{H^{1/2}(\Gamma)} \ . \end{aligned} \tag{5.60}$$

Suppose that Γ divides Ω into two portions, Ω_1 and Ω_2, and is sufficiently smooth that $w|_{\Omega_i} \in H^2(\Omega_i), i = 1, 2$. Then

$$\|w|_{\Omega_i}\|_{L^2(0,T;H^2(\Omega_i))} \leq C\|e_u^{\mathrm{reg}}\|_{L^2(0,T;L^2(\Omega))} \tag{5.61}$$

and

$$\|\omega\|_{L^2(0,T;L^2(\Gamma))} \leq C\|e_u^{\mathrm{reg}}\|_{L^2(0,T;L^2(\Omega))} \ , \tag{5.62}$$

where I have abused notation slightly, in using the same symbol "ω" to denote both $\omega \in H^{-1/2}(\Gamma)$ and $\omega \in L^2(\Gamma)$ such that $(\omega, v)_{L^2(\Gamma)} = {}_{H^{-1/2}(\Gamma)}\langle\omega, v\rangle_{H^{1/2}(\Gamma)}$ for all $v \in H^{1/2}(\Gamma)$. The bounds (5.61) and (5.62) follow from regularity theory for the equivalent parabolic Dirichlet boundary value problem [56, Section 7.1.3, Theorem 5] and boundedness of the normal derivative of $w|_{\Omega_i}$ [2, Theorem 7.53] (where the multiplier ω is the jump in normal derivative of w across Γ). Now assume that there exists an interpolant w^h of w, with $\gamma w^h = 0$, such that

$$\|w - w^h\|_{L^2(\Omega_i)} \leq Ch\|w\|_{H^2(\Omega_i)} \tag{5.63}$$

and

$$\|w - w^h\|_{H^1(\Omega_i)} \leq Ch^{1/2}\|w\|_{H^2(\Omega_i)} \ . \tag{5.64}$$

This can be shown for Q_1 finite elements by applying the results of [136] on either side of Γ. It basically forces w^h to be zero on the $\mathcal{O}(h)$-thickness band of elements containing Γ, which introduces a first-order stair-step approximation of the boundary to each of Ω_1 and Ω_2. I speculate that the interpolation estimates (5.63) and (5.64) hold for many other finite element spaces as well. Because $\delta\lambda$ is quantified over all of $L^2(\Gamma)$,

$${}_{H^{-1/2}(\Gamma)}\left\langle \omega, \gamma e_u^{\text{reg}} \right\rangle_{H^{1/2}(\Gamma)} = \left(\omega, \gamma e_u^{\text{reg}}\right)_{L^2(\Gamma)} = \left(\omega, \frac{1}{k}\partial_t e_\lambda^{\text{reg}}\right)_{L^2(\Gamma)} \ . \tag{5.65}$$

Applying Cauchy–Schwarz to (5.65),

$$\begin{aligned} \left|\left(\omega, \gamma e_u^{\text{reg}}\right)_{L^2(\Gamma)}\right| &\leq \frac{\|\omega\|_{L^2(\Gamma)}}{\sqrt{k}}\sqrt{\frac{1}{k}\left\|\partial_t e_\lambda^{\text{reg}}\right\|^2_{L^2(\Gamma)}} \leq \frac{C\sqrt{kh^{2-\epsilon}}}{\sqrt{k}}\|\omega\|_{L^2(\Gamma)} \\ &\leq Ch^{1-\epsilon}\|\omega\|_{L^2(\Gamma)} \ , \end{aligned} \tag{5.66}$$

where the value of ϵ has been allowed to absorb a positive constant. Cauchy–Schwarz, (5.50), and (5.63) give us

$$\rho\left|\left(\partial_t e_u^{\text{reg}}, w - w^h\right)_{L^2(\Omega)}\right| \leq Ch\sqrt{kh^{2-\epsilon}}\|w\|_{H^2(\Omega\setminus\Gamma)} \leq Ch^{1-\epsilon}\|w\|_{H^2(\Omega\setminus\Gamma)} \ , \tag{5.67}$$

where the value of ϵ has again been allowed to shift by a constant factor and $H^2(\Omega \setminus \Gamma)$ indicates that the space is broken across Γ. $H^1(\Omega)$ boundedness of a, $H^1(\Omega)$ stability (5.56) of e_u^{reg}, and the interpolation estimate (5.64) provide

$$\left|a\left(e_u^{\text{reg}}, w - w^h\right)\right| \leq Ch^{1/2-\epsilon}\|w\|_{H^2(\Omega\setminus\Gamma)} \ . \tag{5.68}$$

Thus, using the preceding bounds in (5.60),

$$\begin{aligned}\|e_u^{\mathrm{reg}}\|^2_{L^2(\Omega)} \leq & \rho\partial_t \left(w, e_u^{\mathrm{reg}}\right)_{L^2(\Omega)} + \left(C_1 h^{1-\epsilon} + C_2 h^{1/2-\epsilon}\right)\|w\|_{H^2(\Omega\setminus\Gamma)} \\ & + C_3 h^{1-\epsilon}\|\omega\|_{L^2(\Gamma)} \ ,\end{aligned} \tag{5.69}$$

where C_1, C_2, and C_3 are independent of h and k, but may carry different units in a physical problem. The preasymptotic convergence regimes associated with these constants are therefore connected to the physics of the system being modeled. The practical implications of these preasymptotic regimes for specific problem classes are, however, beyond the scope of the present analysis. Integrating in time, assuming $e_u^{\mathrm{reg}}(0) = 0$, using the final condition on $w(T)$, and recalling the bounds on the solution to the dual problem,

$$\|e_u^{\mathrm{reg}}\|^2_{L^2(0,T;L^2(\Omega))} \leq C\left(C_1 h^{1-\epsilon} + C_2 h^{1/2-\epsilon} + C_3 h^{1-\epsilon}\right)\|e_u^{\mathrm{reg}}\|_{L^2(0,T;L^2(\Omega))} \ . \tag{5.70}$$

The bound (5.70) implies, for h sufficiently small, that

$$\|e_u^{\mathrm{reg}}\|_{L^2(0,T;L^2(\Omega))} \leq C h^{1/2-\epsilon} \ . \tag{5.71}$$

5.1.1.7 Discretization in time

Consider the following problem template: Find x such that, for all y in an appropriate test space,

$$(\dot{x}(t), y) = -B(x(t), y) + F(y) \ , \tag{5.72}$$

where $(\cdot,\cdot)$ is an inner product, B is a bilinear form and F is a bounded linear functional. In the case of the semi-discrete regularized problem (5.36), $x(t)$

and y are in the Cartesian product space of temperatures and multipliers:

$$x = \left\{u^h, \lambda^h\right\} \tag{5.73}$$

and

$$y = \left\{v^h, \delta\lambda\right\} \ , \tag{5.74}$$

and the inner product, bilinear, and linear forms are defined by

$$(\dot{x}, y) = \frac{\rho}{2}\left(\partial_t u^h, v^h\right)_{L^2(\Omega)} + \frac{\Delta t}{2(1+r)\beta}\left(\partial_t \lambda^h, \delta\lambda\right)_{L^2(\Gamma)} \ , \tag{5.75}$$

$$\begin{aligned} 2B(x, y) =& a\left(u^h, v^h\right) + \left(\lambda^h, v^h\right)_{L^2(\Gamma)} - \left(\delta\lambda, u^h\right)_{L^2(\Gamma)} \\ &+ \frac{r}{(1+r)\beta}\left(\lambda^h, \delta\lambda\right)_{L^2(\Gamma)} \ , \end{aligned} \tag{5.76}$$

and

$$2F(y) = (f(t), v^h)_{L^2(\Omega)} - (g(t), \delta\lambda)_{L^2(\Gamma)} \ . \tag{5.77}$$

Note that $B(y, y) \geq 0$ for all y in the test/trial space. Note also that the norm induced by the inner product $(\cdot, \cdot)$ is exactly $|||\cdot|||$ (defined earlier by (5.10)). Test the local truncation error (LTE) that results from inserting the semi-discrete solution into the backward Euler time discretization:

$$((\mathrm{LTE})_{n+1}, y) = \left(\frac{x(t^{n+1}) - x(t^n)}{\Delta t}, y\right) + B(x(t^{n+1}), y) - F(y) \tag{5.78}$$

$$\begin{aligned} &= \left(\frac{x(t^{n+1}) - (x(t^{n+1}) - \Delta t \dot{x}(t^{n+1}) - R_1(t^n))}{\Delta t}, y\right) \\ &\quad + B(x(t^{n+1}), y) - F(y) \end{aligned} \tag{5.79}$$

$$= \left(\frac{R_1(t^n)}{\Delta t}, y\right) \ , \tag{5.80}$$

where $R_1(t^n)$ in the above is the remainder of a Taylor series expansion of x about t^{n+1}, evaluated at t^n. Thus the LTE is bounded like

$$\begin{aligned} |||(\mathrm{LTE})_{n+1}||| &= \frac{|||R_1(t^n)|||}{\Delta t} && (5.81)\\ &\le \sup_{\xi\in(t^n,t^{n+1})} \frac{\frac{1}{2}|||\ddot{x}(\xi)|||\Delta t^2}{\Delta t} && (5.82)\\ &\le \sup_{\xi\in(0,T)} \frac{\frac{1}{2}|||\ddot{x}(\xi)|||\Delta t^2}{\Delta t} && (5.83)\\ &\le C\Delta t\ , && (5.84) \end{aligned}$$

where C is independent of n, Δt and β. The last inequality follows from the uniform stability of (time derivatives of) the regularized problem in the $|||\cdot|||$ norm. Testing the error e^{n+1} between the backward Euler solution x^{n+1} and the semi-discrete solution $x(t^{n+1})$,

$$\begin{aligned} \left(e^{n+1},y\right) &= \left(x^{n+1}-x(t^{n+1}),y\right) && (5.85)\\ &= (e^n,y) - \Delta t\left(B(e^{n+1},y) + ((\mathrm{LTE})_{n+1},y)\right)\ . && (5.86) \end{aligned}$$

Inserting test function $y = e^{n+1}$, using the Cauchy–Schwarz inequality (with the norm induced by the inner product $(\cdot,\cdot)$), and using the coercivity of B,

$$\begin{aligned} \left|\left|\left|e^{n+1}\right|\right|\right|^2 &= \left(e^n,e^{n+1}\right) - \Delta t\left(B(e^{n+1},e^{n+1}) + \left((\mathrm{LTE})_{n+1},e^{n+1}\right)\right) && (5.87)\\ &\le \left|\left|\left|e^{n+1}\right|\right|\right| \cdot |||e^n||| + \Delta t|||(\mathrm{LTE})_{n+1}||| \cdot \left|\left|\left|e^{n+1}\right|\right|\right| && (5.88)\\ \Rightarrow\ \left|\left|\left|e^{n+1}\right|\right|\right| &\le |||e^n||| + \Delta t|||(\mathrm{LTE})_{n+1}|||\ . && (5.89) \end{aligned}$$

Using the bound (5.84) on LTE, (5.89) becomes

$$\left|\left|\left|e^{n+1}\right|\right|\right| \le |||e^n||| + C\Delta t^2\ , \qquad (5.90)$$

where C is independent of Δt and β. Assuming for simplicity that $e^0 = 0$,

$$\left|\left|\left|e^1\right|\right|\right| \leq C'\Delta t^2 \tag{5.91}$$

$$\left|\left|\left|e^2\right|\right|\right| \leq \left(\left|\left|\left|e^1\right|\right|\right| + C'\Delta t^2\right) \leq 2C'\Delta t^2 \tag{5.92}$$

$$\left|\left|\left|e^3\right|\right|\right| \leq \left(\left|\left|\left|e^2\right|\right|\right| + C'\Delta t^2\right) \leq 3C'\Delta t^2 \tag{5.93}$$

$$\dots$$

$$\left|\left|\left|e^N\right|\right|\right| \leq \left(\left|\left|\left|e^{N-1}\right|\right|\right| + C'\Delta t^2\right) \leq NC'\Delta t^2 \ , \tag{5.94}$$

where $N = T/\Delta t$ and the prime on C' is merely to prevent it from absorbing numerical constants as is typically allowed with the symbol "C". Taking T to be constant, then

$$\left|\left|\left|e^N\right|\right|\right| \leq \frac{T}{\Delta t} C'\Delta t^2 \leq C\Delta t \ , \tag{5.95}$$

where C does not depend on Δt or β. This implies first-order temporal convergence in the norm $|||\cdot|||$, which controls the temperature in $L^2(\Omega)$.

5.1.1.8 Numerical experiment: scalar parabolic problem

In this section, I construct an instance of the parabolic model problem and test the convergence of the discretization. In this problem instance, the space dimension, d, is two, $a(u,v) = (\nabla u, \nabla v)_{L^2(\Omega)}$, and $\rho = 1$. The prescribed functions f and g are zero. Ω is the square $(-W/2, W/2)^2 \subset \mathbb{R}^2$, with $W = 2.5$ and Γ is the unit circle $\{\mathbf{x} \in \mathbb{R}^2 \ : \ \|\mathbf{x}\|_{\ell^2} = 1\}$. The time interval terminates at time $t = T = 0.1$ and the initial temperature at $t = 0$ is

$$u_0(r,\theta) = \left\{ \begin{array}{ll} J_0(Rr) & r < 1 \\ 0 & \text{otherwise} \end{array} \right. \ , \tag{5.96}$$

where r and θ are standard 2D polar coordinates and R is the first root of the Bessel function J_0. This implies that the exact solution is

$$u(r, \theta, t) = J_0(Rr)e^{-R^2 t} \; . \tag{5.97}$$

For discretization, I use a linear uniform B-spline space with $2^N \times 2^N$ elements, for $N \in \{3, \ldots, 10\}$, to represent trial and test functions. I define the penalty by $\beta = C_{\text{pen}}/h$, where $h = W/2^N$ is element width of the uniform B-spline space and $C_{\text{pen}} = 1$. The discrete initial condition is set by nodal interpolaton of u_0. The time step is proportional to h, viz. $\Delta t = T/2^N$. An illustrative snapshot of a numerical solution is shown in Figure 5.2.

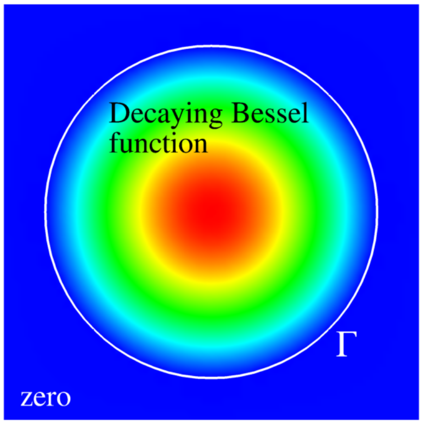


Figure 5.2: Annotated snapshot of a solution to the scalar parbolic test problem.

To test the robustness of the formulation for small perturbation parameters $r \ll 1$, I first compute with $r = 0.1$ and then compare with results for $r = 0$. Figure 5.3 shows the convergence of $L^2(\Omega)$ and $H^1(\Omega)$ norms of the error $u - u^h$ at time T for $r = 0.1$ and $r = 0$, suggesting convergence rates of $1/2$ in $H^1(\Omega)$ and 1 in $L^2(\Omega)$ for both cases.

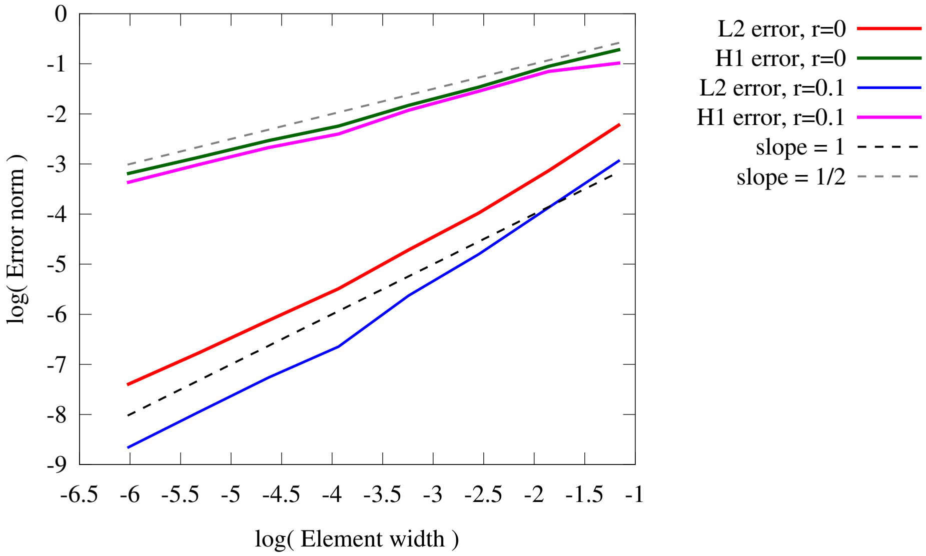


Figure 5.3: Convergence of L^2 and H^1 errors for different values of r.

Integrals over Γ are evaluated using 32×2^N quadrature points, spaced evenly along the arc length of Γ, where 2^N is the number of elements across the width of the domain. A scalar sample of λ is stored and updated at each of these points. This high density of points rules out the possibility of accidental inf-sup stability following from reduced quadrature of the boundary constraint. Consistent with the absence of any proven convergence for λ^h, the computed multiplier field is highly oscillatory and bears no resemblance whatsoever to the spatially-uniform exact solution. Figure 5.4 shows a representative plot of the multiplier as a function of polar angle around Γ. In light of such inaccurate results for the multiplier field, I would recommend to consider the Lagrange multiplier a by-product of constraint enforcement rather than a meaningful component of the solution. This is consistent with the results obtained by Kallemov et al. [103, Figure 8] who solved implicitly for Lagrange multipliers at boundary points immersed in Stokes flow. The cited study found that, for high spatial densities of markers, the multiplier "traction" is highly oscillatory. (While rarely reported or visualized, such boundary force oscillations are presumably present in many immersed boundary computations, since, as pointed out by [103], high densities of markers are frequently recommended to prevent leakage.) Also in agreement with [103], I observe that low-order moments of the Lagrange multiplier field (e.g. net drag or torque on immersed objects) remain accurate in spite of egregious spatial oscillations.

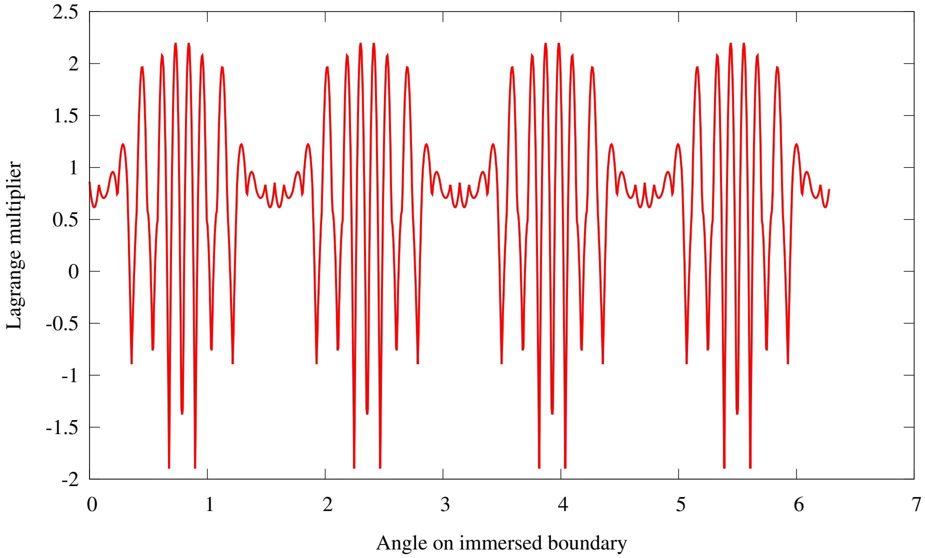


Figure 5.4: The value of λ^h as a function of the angle around Γ. (Linear interpolation is used between surface quadrature point samples of λ^h.)

Remark 5.4. The error norms used to assess convergence are integrated over Ω with the same 2×2 quadrature rule used to assemble the equation systems in the analysis. Due to the low regularity of the exact solution, this incurs a significant integration error, but this extra error is not asymptotically worse than what one would expect from approximation and temporal truncation considerations.

5.1.2 Extension to related linear problems

The analysis of the scalar second-order parabolic equation suggests some extensions to other linear model problems that more closely resemble FSI. This section will address some of those extensions, albeit with a lesser degree of rigor and completeness than the analysis of the scalar parabolic problem.

5.1.2.1 Unsteady Stokes flow

The analysis of the heat equation can be formally extrapolated to divergence-conforming discretizations of unsteady Stokes flow, by posing the problem over the solenoidal subspace of $V_0(\Omega) \subset \left(H^1(\Omega)\right)^d$ and seeking discrete solutions in a finite dimensional subspace (cf. [27, Section 10]). This is in contrast to non-div-conforming discretizations, in which the subspace of discrete velocities that weakly satisfy incompressibility with respect to a finite dimensional pressure test space are not pointwise divergence-free and do not form a proper subset of solenoidal H^1 functions. To simplify discussion, I eliminate the tangential portion of the multiplier on Γ from the outset, with the understanding that this approaches consistentcy with the true no-slip-on-Γ Stokes problem as the penalty coefficient $\tau_{\text{TAN}} \to \infty$.

More precisely, consider the problem: Find velocity $\mathbf{u} \in L^2(0,T;V_0(\Omega))$ with $\partial_t \mathbf{u} \in L^2(0,T;V_0^*(\Omega))$ and normal traction jump $\lambda \in L^2(0,T;H^{-1/2}(\Gamma))$ such that for every $\mathbf{v} \in V_0(\Omega)$ and $\delta\lambda \in H^{-1/2}(\Gamma)$ at a.e. $t \in [0,T]$,

$$\begin{aligned}
&\rho\, {}_{V_0^*(\Omega)}\langle \partial_t \mathbf{u}(t), \mathbf{v} \rangle_{V_0(\Omega)} + a(\mathbf{u}(t), \mathbf{v}) \\
&\qquad + {}_{H^{-1/2}(\Gamma)}\langle \lambda(t), \mathbf{v}\cdot\mathbf{n} \rangle_{H^{1/2}(\Gamma)} - {}_{H^{-1/2}(\Gamma)}\langle \delta\lambda, \mathbf{u}(t)\cdot\mathbf{n} \rangle_{H^{1/2}(\Gamma)} \\
&\qquad + \tau_{\text{TAN}} \left(\mathbf{u}(t) - (\mathbf{u}(t)\cdot\mathbf{n})\mathbf{n}, \mathbf{v} \right)_{L^2(\Gamma)} \\
&\quad = {}_{H^{-1/2}(\Gamma)}\langle \delta\lambda, \mathbf{g}(t)\cdot\mathbf{n} \rangle_{H^{1/2}(\Gamma)} + \tau_{\text{TAN}} \left(\mathbf{g}(t) - (\mathbf{g}(t)\cdot\mathbf{n})\mathbf{n}, \mathbf{v} \right)_{L^2(\Gamma)} \\
&\qquad + {}_{V_0^*(\Omega)}\langle \mathbf{f}(t), \mathbf{v} \rangle_{V_0(\Omega)} \qquad (5.98)
\end{aligned}$$

and

$$\mathbf{u}(0) = \mathbf{u}_0 \in L^2(\Omega) \;, \qquad (5.99)$$

where $\mathbf{n}$ is a normal vector to the surface Γ, $\mathbf{g}(t)$ is the prescribed velocity on Γ at time t, $\mathbf{f}(t)$ is some functional in the dual of $V_0(\Omega)$, ρ takes on the physical interpretation of mass density, and the bilinear form a is now defined

$$a(\mathbf{u}, \mathbf{v}) = 2\mu \left(\boldsymbol{\varepsilon}(\mathbf{u}), \boldsymbol{\varepsilon}(\mathbf{v}) \right)_{L^2(\Omega)} \;, \qquad (5.100)$$

where μ is the dynamic viscosity and $\boldsymbol{\varepsilon}$ is the symmetric gradient operator. This problem statement is subtly incomplete in that, depending on the geometry of Γ, there may be some compatibility condition on the data $\mathbf{g}$ to ensure consistency with mass conservation (e.g., if Γ encloses a region of Ω, the integral of the normal component of $\mathbf{g}$ must be zero to have a solenoidal $\mathbf{u} \in V_0$ satisfying the immersed boundary condition). The steps of the semi-implicit time integration scheme become

1. Given $\mathbf{u}^n$ and λ^n, find $\mathbf{u}^{n+1}$ such that for all $\mathbf{v}$,

$$\begin{aligned}
\rho \left(\mathbf{u}_t^{n+1}, \mathbf{v}\right)_{L^2(\Omega)} &+ a(\mathbf{u}^{n+1}, \mathbf{v}) + (\lambda^n, \mathbf{v} \cdot \mathbf{n})_{L^2(\Gamma)} \\
&+ \tau_{\mathrm{NOR}} \left(\mathbf{u}^{n+1} \cdot \mathbf{n}, \mathbf{v} \cdot \mathbf{n}\right)_{L^2(\Gamma)} \\
&+ \tau_{\mathrm{TAN}} \left(\mathbf{u}^{n+1} - (\mathbf{u}^{n+1} \cdot \mathbf{n})\mathbf{n}, \mathbf{v}\right)_{L^2(\Gamma)} \\
= \tau_{\mathrm{NOR}} &\left(\mathbf{g}(t^{n+1}) \cdot \mathbf{n}, \mathbf{v} \cdot \mathbf{n}\right)_{L^2(\Gamma)} \\
&+ \tau_{\mathrm{TAN}} \left(\mathbf{g}(t^{n+1}) - (\mathbf{g}(t^{n+1}) \cdot \mathbf{n})\mathbf{n}, \mathbf{v}\right)_{L^2(\Gamma)} \\
&+ {}_{V_0^*(\Omega)}\langle \mathbf{f}(t^{n+1}), \mathbf{v} \rangle_{V_0(\Omega)} \ ,
\end{aligned} \tag{5.101}$$

   where

$$\mathbf{u}_t^{n+1} = \frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} \ . \tag{5.102}$$

2. Update the multiplier such that, for all $\delta\lambda \in L^2(\Gamma)$,

$$(\lambda^{n+1}, \delta\lambda)_{L^2(\Gamma)} = \frac{1}{1+r} \left(\lambda^n + \tau_{\mathrm{NOR}}(\mathbf{u}^{n+1} - \mathbf{g}(t^{n+1})) \cdot \mathbf{n}, \delta\lambda\right)_{L^2(\Gamma)} \ . \tag{5.103}$$

The corresponding implicitly-integrated regularized problem is: Find the regularized velocity $\mathbf{u}^{\mathrm{reg}} \in L^2(0, T; V_0(\Omega))$ with $\partial_t \mathbf{u}^{\mathrm{reg}} \in L^2(0, T; V_0^*(\Omega))$ and normal traction jump $\lambda^{\mathrm{reg}} \in L^2(0, T; L^2(\Gamma))$ with $\partial_t \lambda^{\mathrm{reg}} \in L^2(0, T; L^2(\Gamma))$ such that for every $\mathbf{v} \in V_0(\Omega)$ and $\delta\lambda \in L^2(\Gamma)$ at a.e. $t \in [0, T]$,

$$\begin{aligned}
&\rho_{V_0^*(\Omega)}\langle \partial_t \mathbf{u}^{\text{reg}}(t), \mathbf{v}\rangle_{V_0(\Omega)} + a(\mathbf{u}^{\text{reg}}(t), \mathbf{v}) + (\lambda^{\text{reg}}(t), \mathbf{v}\cdot\mathbf{n})_{L^2(\Gamma)} \\
&\quad + \left(\frac{\Delta t}{(1+r)\tau_{\text{NOR}}}\partial_t \lambda^{\text{reg}}(t) - \mathbf{u}^{\text{reg}}(t)\cdot\mathbf{n} + \frac{r}{(1+r)\tau_{\text{NOR}}}\lambda^{\text{reg}}(t), \delta\lambda\right)_{L^2(\Gamma)} \\
&\quad + \tau_{\text{TAN}}\left(\mathbf{u}^{\text{reg}}(t) - (\mathbf{u}^{\text{reg}}(t)\cdot\mathbf{n})\mathbf{n}, \mathbf{v}\right)_{L^2(\Gamma)} \\
&= (\mathbf{g}(t)\cdot\mathbf{n}, \delta\lambda)_{L^2(\Gamma)} + \tau_{\text{TAN}}\left(\mathbf{g}(t) - (\mathbf{g}(t)\cdot\mathbf{n})\mathbf{n}, \mathbf{v}\right)_{L^2(\Gamma)} \\
&\quad + {}_{V_0^*(\Omega)}\langle \mathbf{f}(t), \mathbf{v}\rangle_{V_0(\Omega)}
\end{aligned} \tag{5.104}$$

and

$$\mathbf{u}^{\text{reg}}(0) = \mathbf{u}_0 \in L^2(\Omega)\ , \quad \lambda^{\text{reg}}(0) = \lambda_0 \in L^2(\Gamma)\ . \tag{5.105}$$

Notice that the compatibility condition on $\mathbf{g}$ is no longer strictly required in the regularized problem, although I would anticipate bad results if it is violated. Recognizing the coercivity of the tangential penalty term and the formal similarity of this problem to the scalar parabolic problem of Section 5.1.1, I would not expect to encounter major difficulties adapting the program of Section 5.1.1 to this setting, to bound errors at time T in the norm

$$|||\mathbf{u}, \lambda|||^2 = \frac{1}{2}\rho\,\|\mathbf{u}\|^2_{L^2(\Omega)} + \frac{\Delta t}{2(1+r)\tau_{\text{NOR}}}\,\|\lambda\|^2_{L^2(\Gamma)}\ . \tag{5.106}$$

The first term now carries the physical interpretation of the kinetic energy of the fluid. In the case of $r = 0$, the multiplier represents a normal displacement of fluid through Γ, and the second term of the energy norm becomes a spring-like potential energy (foreshadowing the inclusion of a structural potential energy in linearized FSI problems).

5.1.2.2 Numerical experiment: linearized Taylor–Green vortex

The following velocity field is a solution to the 2D Navier–Stokes equations posed on the domain $\Omega = [-\pi, \pi]^2$ with periodic boundary conditions and no external forcing:

$$\mathbf{u}_{\mathrm{TG}}(\mathbf{x}, t) = (\sin(x_1)\cos(x_2)\mathbf{e}_1 - \cos(x_1)\sin(x_2)\mathbf{e}_2)\, e^{-2\mu t/\rho} \ . \tag{5.107}$$

This is known as the Taylor–Green vortex. The velocity field is also an exact solution to the Stokes equations, with the body force field $\mathbf{f}_{\mathrm{TG}} = -\rho\mathbf{u}_{\mathrm{TG}}\cdot\nabla\mathbf{u}_{\mathrm{TG}}$. One may construct an interesting test problem by prescribing $\mathbf{u} = \mathbf{u}_{\mathrm{TG}}$ as an initial condition at $t = 0$ and also as a time-dependent Dirichlet boundary condition on a closed immersed boundary Γ, then adding a spatially-uniform body force $\mathbf{f}_x = \mathbf{e}_1$ in the x_1-direction, so that the total body force is $\mathbf{f} = \mathbf{f}_{\mathrm{TG}} + \mathbf{f}_x$. This body force induces a pressure gradient in the region enclosed by Γ without perturbing the velocity solution in that region. The velocity outside of the region enclosed by Γ is no longer equal to $\mathbf{u}_{\mathrm{TG}}$ for $t > 0$. There are jumps in the pressure and velocity derivatives along Γ. The regularity of the velocity solution is therefore representative of typical usage of an immersed boundary method.

I have not attempted to derive an exact solution on the entire domain, but one can easily measure the error in the subset Ω^{err}, enclosed by Γ. Let Γ be a circle centered at $(x, y) = (0, 0)$, with radius 2 and let $\Omega^{\mathrm{err}} = \{\mathbf{x} \in \mathbb{R}^2 \mid |\mathbf{x}|_{\ell^2} < 2\}$. I integrate errors on Ω^{err} approximately, using points from a 3×3 Gaussian quadrature rule on each element that fall inside of Ω^{err}.

This is only a first order approximation, but I do not expect higher than first order convergence of errors in any norm. The setup is illustrated, along with a representative numerical solution, in Figure 5.5.

I discretize this problem using $2^N \times 2^N$ div-conforming B-spline elements of degree $k' = 1$, for $N \in \{4, \dots, 9\}$. Due to the low regularity of the exact solution, I would not expect to obtain improved convergence rates with higher k'. The problem is posed over the interval $(0, T)$ with $T = 0.2$, using time steps of size $\Delta t = T/(2^{N-2})$. The initial condition is set using H^1 projection. Penalty values are $\tau_{\text{NOR}} = \tau_{\text{TAN}} = 100\mu/h$, where $h = 2\pi/(2^N)$ is the mesh element size. The convergence of $L^2(\Omega^{\text{err}})$ and $H^1(\Omega^{\text{err}})$ errors at time T is shown in Figure 5.6. As with the heat equation, the convergence for $r > 0$ is first order in $L^2(\Omega^{\text{err}})$ and one-half order in $H^1(\Omega^{\text{err}})$, with robustness in the limit of $r \to 0$.

5.1.2.3 Coupled second-order problems

To look at coupling between d-dimensional and $(d-1)$-dimensional subproblems without immediately facing the complexities of fluid–structure interaction, I outline a model problem in which two second-order parabolic subproblems are coupled: Find $u \in L^2(0, T; H_0^1(\Omega))$ with $\partial_t u \in L^2(0, T; H^{-1}(\Omega))$, $y \in L^2(0, T; H_0^1(\Gamma))$ with $\partial_t y \in L^2(0, T; H^{-1}(\Gamma))$ and $\lambda \in L^2(0, T; H^{-1/2}(\Gamma))$

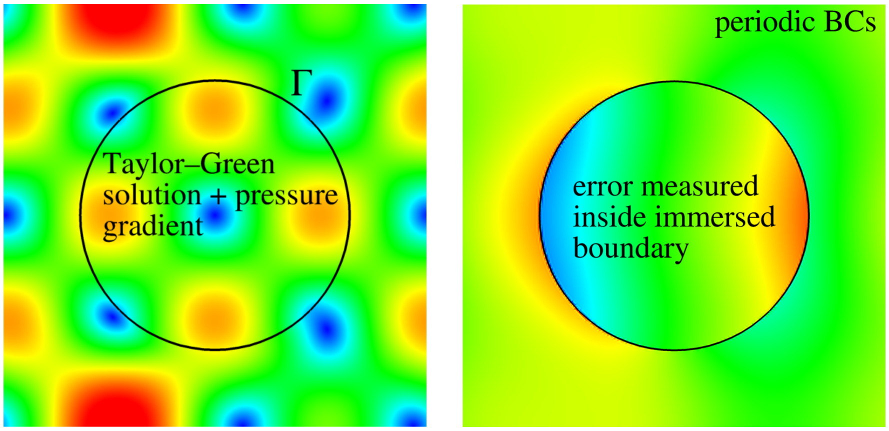


Figure 5.5: Simultaneous velocity magnitude (left) and pressure (right) snapshots of the Stokes Taylor–Green problem, with annotations describing the problem setup.

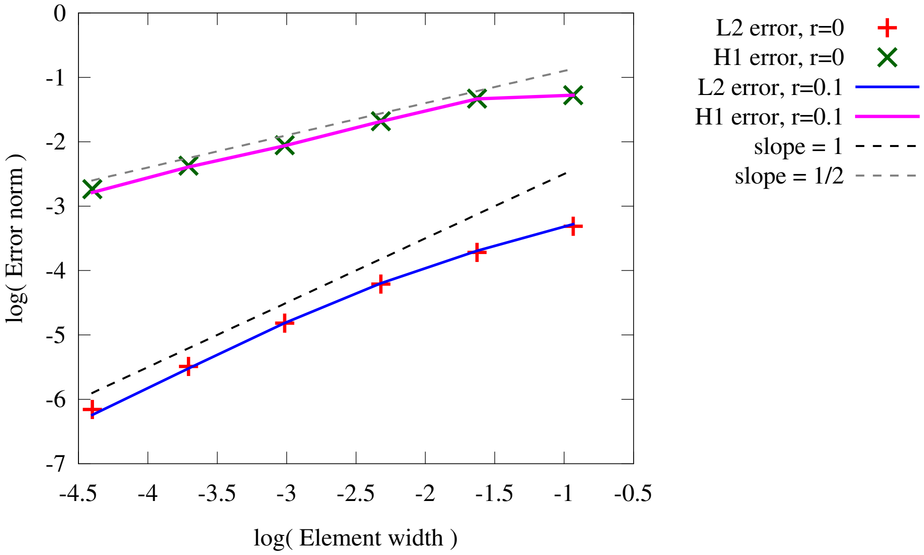


Figure 5.6: Convergence of $L^2(\Omega^{\text{err}})$ and $H^1(\Omega^{\text{err}})$ errors at time T for different values of r.

such that for every $v \in H^1_0(\Omega)$, $z \in H^1_0(\Gamma)$ and $\delta\lambda \in H^{-1/2}(\Gamma)$ at a.e. $t \in [0,T]$,

$$\begin{aligned}\rho_1\ {}_{H^{-1}(\Omega)}\langle \partial_t u(t), v\rangle_{H^1(\Omega)} &+ a_1(u(t),v)\\ &+ \rho_2\ {}_{H^{-1}(\Gamma)}\langle \partial_t y(t), z\rangle_{H^1(\Gamma)} + a_2(y(t),z)\\ &+\ {}_{H^{-1/2}(\Gamma)}\langle \lambda(t), \gamma w - z\rangle_{H^{1/2}(\Gamma)}\\ &-\ {}_{H^{-1/2}(\Gamma)}\langle \delta\lambda, \gamma u(t) - y(t)\rangle_{H^{1/2}(\Gamma)} = F(v,z) \end{aligned} \tag{5.108}$$

and

$$u(0) = u_0 \in L^2(\Omega) \ , \quad y(0) = y_0 \in L^2(\Gamma) \ . \tag{5.109}$$

a_1 is coercive over $H_0^1(\Omega)$ and a_2 is coercive over $H_0^1(\Gamma)$. Following the pattern set by the scalar parabolic problem of Section 5.1.1 and the unsteady Stokes problem of Section 5.1.2.1, it should be clear what the semi-implicit algorithm and equivalent implicitly-integrated problem are for the coupled problem. Much of the program of Section 5.1.1 can then be repeated nearly unchanged to obtain analogous error estimates in the norm

$$|||u, y, \lambda|||^2 = \frac{1}{2}\rho_1 \|u\|^2_{L^2(\Omega)} + \frac{1}{2}\rho_2 \|y\|^2_{L^2(\Gamma)} + \frac{\Delta t}{2(1+r)\beta} \|\lambda\|^2_{L^2(\Gamma)} \ . \tag{5.110}$$

I use this model coupled problem to demonstrate the efficacy of block iteration in Section 5.3.

5.2 Localization of error

Numerical experiments indicate that the rate of convergence of H^1 error with h is one in subdomains Ω^{err} with $\mathcal{O}(1)$ separation from Γ. This is higher than would be possible on all of Ω, based on the regularity of the exact solution and approximation considerations. For peicewise polynomial spaces of degree one, this is optimal convergence in $H^1(\Omega^{\text{err}})$, although the $L^2(\Omega^{\text{err}})$ convergence remains first-order after restricting to Ω^{err} and higher-degree spaces do not yeild higher rates of convergence on Ω^{err}. (The details of these numerical experiments have been omitted for brevity.) This may, at first, seem puzzling, in light of the well-developed field of local error analysis. In the current section, I will dispel this conundrum, by clarifying the distinction between applying a given concentrated source term to a system and using a concetrated source term to enforce a constraint.

In the immersed boundary method developed in this dissertation, the influence of the boundary on the fluid subproblem is felt through a concentrated surface force in $H^{-1}(\Omega)$. For an elliptic problem subject to such a forcing, one can quite easily derive optimal local error estimates in $H^1(\Omega^{\text{err}})$, using the theory developed in [130]. This theory was extended to Stokes flow in [5]. The results are sufficiently robust that they extend almost entirely to forces less regular than $H^{-1}(\Omega)$, including point forces, as shown by [24, 114] for the Poisson problem and by [119] for Stokes flow.

As a model problem to illustrate the pollution effects of jumps in derivatives due to the bilinear form of a weak elliptic problem rather than the linear functional driving it, consider the following modified Poisson problem: Find $u \in H_0^1(\Omega)$ such that for all $v \in H_0^1(\Omega)$,

$$B(u, v) = F(v) \ , \tag{5.111}$$

where

$$B(u, v) = \int_\Omega \nabla u \cdot \nabla v \ d\Omega + \int_\Gamma \beta uv \ d\Gamma \tag{5.112}$$

and F is some bounded linear functional defined on $H_0^1(\Omega)$. Γ is some interface of co-dimension one to Ω. β is a positive penalty coefficient. Assuming a reasonable trace inequality, the modified B is still bounded and coercive on $H_0^1(\Omega)$. The penalty force can be construed as a concentrated source $-\beta u$ on the immersed boundary Γ in an ordinary Poisson problem, but this analogy is strained by the presence of u in the "source" term, and I demonstrate below that the local error analysis for concentrated sources in the Poisson equation does not extend to the problem (5.111).

In what follows, assume that $\Omega_1 \subset\subset \Omega$ has some constant separation from the surface Γ along which the exact solution u is non-smooth. The integral over Γ in B does not affect the interior equations satisfied by the exact and discrete solutions corresponding to discrete test functions whose supports are fully contained in Ω_1. The interior duality and error estimates due to Nitsche and Schatz [130, Sections 4 and 5] are therefore unchanged. In particular, the result [130, Theorem 5.1(i)] still holds: Let $\Omega_0 \subset\subset \Omega_1 \subset\subset \mathbb{R}^d$, $u \in H^\ell(\Omega_1)$, $u_h \in S^h_{k,r}(\Omega_1)$, where $1 \leq k < r$ and p is a non-negative integer, arbitrary but fixed. (Following the notation defined in [130, Section 2], $S^h_{k,r}(A)$ is a finite element subspace of $H^k(A)$ (for some domain $A \subset \mathbb{R}^d$) with shape functions of polynomial order $r - 1$. $\mathring{S}^h_{k,r}(A)$ is the subspace spanned by basis functions with supports contained in A.) Suppose that standard approximation, stability, and superapproximation[1] assumptions hold for the discrete space $S^h_{k,r}$. (Refer to [130, Section 2] for the precise conditions.) Then there exists a $0 < h_1 \leq 1$ such that, if $e = u - u_h$ satisfies

$$B(e, \phi_h) = 0 \qquad \forall \phi_h \in \mathring{S}^h_{k,r}(\Omega_1) \ , \tag{5.113}$$

---

[1] This term does not appear in [130]. It shows up frequently in later discussions of local error estimates, though, and refers to [130, Assumption A.2] and/or other related conditions.

then, for all $h \in (0, h_1]$ and $1 \le \ell \le r$,

$$\|e\|_{H^1(\Omega_0)} \le C \left(h^{\ell-1} \|u\|_{H^\ell(\Omega_1)} + \|e\|_{H^{-p}(\Omega_1)} \right) \ , \tag{5.114}$$

where $C = C(p, \Omega_0, \Omega_1)$.

The difference between the modified and ordinary Poisson problems becomes apparent, though, when attempting to estimate $\|e\|_{H^{-p}(\Omega_1)}$. The usual duality argument no longer delivers optimal local convergence rates for the problem (5.111). The reason for this is that the lack of regularity originates in the bilinear form, and is therefore present as well in the dual problem. To clarify what I mean by this, I retrace the steps of the duality argument.

Consider u_h computed by applying Galerkin's method to (5.111). The error $e = u - u_h$ will clearly satisfy the interior equation (5.113). Now look at the convergence of the term $\|e\|_{H^{-p}(\Omega_1)} \le \|e\|_{H^{-p}(\Omega)}$ from the error estimate (5.114). First consider the definition of the negative norm:

$$\|e\|_{H^{-p}(\Omega)} = \sup_{\|\phi\|_{H^p(\Omega)}=1} (e, \phi) \ . \tag{5.115}$$

For fixed $\phi \in H^p(\Omega)$ with $\|\phi\|_{H^p(\Omega)} = 1$, there exists a unique $v_\phi \in H_0^1(\Omega)$ such that $(\eta, \phi) = B(\eta, v_\phi)$ for all $\eta \in H_0^1(\Omega)$. Then

$$(e, \phi) = B(e, v_\phi) \ . \tag{5.116}$$

From Galerkin orthogonality, the error e satisfies the equation

$$B(e, w_h) = 0 \qquad \forall w_h \in \mathring{S}^h_{k,r}(\Omega) \ . \tag{5.117}$$

One can therefore subtract an arbitrary w_h from the test function v_ϕ in (5.116):

$$(e, \phi) = B(e, v_\phi - w_h) \ . \tag{5.118}$$

Setting w_h equal to the adjoint elliptic projection $P^* v_\phi$ of v_ϕ into $\mathring{S}^h_{k,r}(\Omega)$,

$$(e, \phi) = B(e, v_\phi - P^* v_\phi) \ . \tag{5.119}$$

Since, by definition, $v_\phi - P^* v_\phi$ satisfies

$$B(\phi_h, v_\phi - P^* v_\phi) = 0 \qquad \forall \phi_h \in \mathring{S}^h_{k,r}(\Omega) \ , \tag{5.120}$$

one can add an arbitrary member of $\mathring{S}^h_{k,r}(\Omega)$ to e on the right side of (5.119):

$$(e, \phi) = B(u - \chi_h, v_\phi - P^* v_\phi) \ , \tag{5.121}$$

where $\chi_h \in \mathring{S}^h_{k,r}(\Omega)$ has absorbed the Galerkin discrete solution $u_h \in \mathring{S}^h_{k,r}(\Omega)$ from the error $e = u - u_h$. Due to H^1 boundedness of B,

$$\begin{aligned} |(e, \phi)| &= |B(u - \chi_h, v_\phi - P^* v_\phi)| && (5.122) \\ &\leq C \|u - \chi_h\|_{H^1(\Omega)} \|v_\phi - P^* v_\phi\|_{H^1(\Omega)} \ . && (5.123) \end{aligned}$$

Taking the supremum over eligible ϕs and recalling the definition of the $H^{-p}(\Omega)$ norm,

$$\|e\|_{H^{-p}(\Omega)} \leq C \|u - \chi_h\|_{H^1(\Omega)} \sup_\phi \left(\|v_\phi - P^* v_\phi\|_{H^1(\Omega)} \right) \ . \tag{5.124}$$

Despite the arbitrarily-smooth forcing $\phi \in H^p(\Omega)$ in the dual problem, v_ϕ is not necessarily any more regular than $H^{3/2}(\Omega)$, because of the singular coefficients in B. Let us assume that $v_\phi \in H^{3/2-\epsilon}(\Omega)$ with $\|v_\phi\|_{H^{3/2-\epsilon}(\Omega)} \leq C$, for ϵ arbitrarily small. Then $\|v_\phi - P^* v_\phi\|_{H^1(\Omega)} \leq Ch^{1/2-\epsilon}$, where C is independent of ϕ, since $\|\phi\|_{H^p(\Omega)} = 1 \; \forall \phi$. If we further assume that $u \in H^{3/2-\epsilon}(\Omega)$, then, by appropriate selection of an interpolant χ_h, $\|u - \chi_h\|_{H^1(\Omega)} \leq Ch^{1/2-\epsilon}$. This provides the following estimate of pollution effects:

$$\|e\|_{H^{-p}(\Omega_1)} \leq Ch^{1-2\epsilon} \; . \tag{5.125}$$

Wahlbin's analysis of a similar problem [194, Chapter III, Section 17] suggests that this estimate is sharp. One cannot do any better if the source of local non-smoothness is a singular coefficient instead of a singular source term. Further, Wahlbin's analysis suggests that one will not get a higher rate in the local L^2 norm, since the sharpness of this result will hold for *any* norm of the local error. On a brighter note, though, this analysis doubles the rate of convergence that would have been obtained by a global analysis in the energy norm and therefore provides some improvement over naive estimates.

Remark 5.5. A clever way to circumvent this limitation for elliptic problems is the so-called "fat boundary method" [25], but I have not yet attempted to generalize this to the case of unsteady Navier–Stokes flow.

5.3 Block iterative convergence

This section uses a linearized model problem to study the convergence of the block iterative procedure introduced in Section 4.3 to resolve a penalty-coupled FSI system.

5.3.1 A generic model problem

The Lagrange multipliers are held fixed in the block iteration, so only the penalty coupling is of concern when investigating the stability of block iteration. I therefore introduce the following linear model problem, in which two linear elliptic subproblems, indexed 1 and 2, are coupled along an interface by penalty forces: Find $u_1 \in \mathcal{V}_1$ and $u_2 \in \mathcal{V}_2$ such that, for all test functions $w_1 \in \mathcal{V}_1$ and $w_2 \in \mathcal{V}_2$,

$$B_1(u_1, w_1) + k(u_1 - u_2, w_1)_\Gamma = F_1(w_1) \tag{5.126}$$

$$B_2(u_2, w_2) + k(u_2 - u_1, w_2)_\Gamma = F_2(w_2) \ . \tag{5.127}$$

In this problem, B_1 and B_2 are coercive and bounded bilinear forms, F_1 and F_2 are bounded linear functionals, $k > 0$ is the penalty constant coupling the two subproblems, and $(\cdot, \cdot)_\Gamma$ is an inner product of bounded traces of functions from $\mathcal{V}_1$ and $\mathcal{V}_2$. The block iterative algorithm for this problem is to start with $i = 0$ and an initial guess for u_2^0, then repeat

1. Holding u_2^i constant, find $u_1^{i+1} \in \mathcal{V}_1$ such that, for all $w_1 \in \mathcal{V}_1$,

$$B_1(u_1^{i+1}, w_1) + k(u_1^{i+1}, w_1)_\Gamma = k(u_2^i, w_1)_\Gamma + F_1(w_1) \ . \tag{5.128}$$

2. Holding u_1^{i+1} constant (at the value computed in the previous step) find $u_2^{i+1} \in \mathcal{V}_2$ such that, for all $w_2 \in \mathcal{V}_2$,

$$B_2(u_2^{i+1}, w_2) + k(u_2^{i+1}, w_2)_\Gamma = k(u_1^{i+1}, w_2) + F_2(w_2) \ . \tag{5.129}$$

3. $i \leftarrow i + 1$.

Remark 5.6. Note that, when the subproblems are linear, no initial guess is required for u_1^0. u_1^1 is completely determined by u_2^0.

The goal of this analysis is to determine whether this procedure will converge to a fixed point. Define the norms

$$|||u|||_i^2 = \|u\|_i^2 + k\|u\|_\Gamma^2 \ , \tag{5.130}$$

for $i \in \{1, 2\}$, where $\| \cdot \|_i$ is some norm in which B_i is coercive with unit constant (such as energy for B_i symmetric) and $\| \cdot \|_\Gamma$ is the norm induced by $(\cdot, \cdot)_\Gamma$. Then the bilinear form

$$\mathcal{B}_i(u, v) = B_i(u, v) + k(u, v)_\Gamma \tag{5.131}$$

will be coercive in the norm $|||\cdot|||_i$ with unit constant. We have Green's operators $\{\mathcal{G}_i\}_{i=1}^2$ such that if u_i satisfies

$$\mathcal{B}(u_i, w_i) = F(w_i) \qquad \forall w_i \in \mathcal{V}_i \ , \tag{5.132}$$

then

$$u_i = \mathcal{G}_i(F) \ , \tag{5.133}$$

i.e., $\mathcal{G}_i$ is a map from right-hand-side functionals to solutions, for subproblem i. Because the subproblems are linear, so are their solution operators. Using the Lax–Milgram theorem, with unit coercivity constant,

$$|||\mathcal{G}_i(F)|||_i \leq |||F|||_i \ , \tag{5.134}$$

where the norm on the right-hand side is understood as the induced norm on the dual space $\mathcal{V}_i^*$. Expressing the solution u_1^{i+1} of Step 1 of the block iteration algorithm in terms of $\mathcal{G}_1$, re-write Step 2 as

$$\mathcal{B}_2(u_2^{i+1}, w_2) = k\left(\mathcal{G}_1\left(k(u_2^i, \cdot)_\Gamma + F_1\right), w_2\right)_\Gamma + F_2(w_2)\ . \tag{5.135}$$

Expressing the solution of Step 2 in terms of its Green's operator, there is then a mapping from u_2^i to u_2^{i+1}:

$$u_2^{i+1} = \mathcal{G}_2\left(k\left(\mathcal{G}_1\left(k(u_2^i, \cdot)_\Gamma + F_1\right), \cdot\right)_\Gamma + F_2\right) = \mathcal{H}(u_2^i)\ . \tag{5.136}$$

For the block iteration to be stable, it is sufficient that $\mathcal{H}$ be a contraction mapping. Continuity of $(\cdot, w_1)_\Gamma$ and coercivity of $\mathcal{B}_1$ are sufficient to show that the convergence of subproblem 2 implies the convergence of subproblem 1. Using the linearity of $\mathcal{G}_1$ and $\mathcal{G}_2$ and bilinearity of the inner product $(\cdot, \cdot)_\Gamma$, it is easy to see that

$$\mathcal{H}(u) - \mathcal{H}(v) = \mathcal{G}_2\left(k\left(\mathcal{G}_1\left(k(u - v, \cdot)_\Gamma\right), \cdot\right)_\Gamma\right)\ . \tag{5.137}$$

Recalling (5.134),

$$|||\mathcal{H}(u) - \mathcal{H}(v)|||_2 \leq C_1 C_2 |||u - v|||_2\ , \tag{5.138}$$

where C_1 and C_2 are defined such that

$$|||k(u, \cdot)_\Gamma|||_2 \leq C_1 |||u|||_1 \qquad \forall u \in \mathcal{V}_1 \tag{5.139}$$

and

$$|||k(u, \cdot)_\Gamma|||_1 \leq C_2 |||u|||_2 \qquad \forall u \in \mathcal{V}_2\ . \tag{5.140}$$

To show stability of block iteration, it is therefore sufficient to show that $C_1 \leq 1$ and $C_2 \leq 1$. Let us first proceed in a general way. For $(i, j) \in \{(1, 2), (2, 1)\}$,

$$\begin{aligned} \left|\left|\left| k(u, \cdot)_\Gamma \right|\right|\right|_i &= \sup_{v \neq 0} \frac{k(u, v)_\Gamma}{\sqrt{\|v\|_i^2 + k\|v\|_\Gamma^2}} && (5.141) \\ &\leq \sup_{v \neq 0} \frac{\sqrt{k}(u, v)_\Gamma}{\|v\|_\Gamma} && (5.142) \\ &\leq \sqrt{k}\|u\|_\Gamma && (5.143) \\ &\leq \left|\left|\left| u \right|\right|\right|_j \ . && (5.144) \end{aligned}$$

This demonstrates that $C_1 \leq 1$ and $C_2 \leq 1$, and therefore that block iteration is at worst non-divergent.

5.3.2 Application to FSI

To gain greater insight into the rate of convergence of the iteration, and its dependence on the nature of the subproblems and the penalty parameter k, I now use further assumptions on the structures of B_1 and B_2 and trace and trace-inverse inequalities to sharpen the estimate in the step from (5.143) to (5.144). Consider the problem of dynamic linearized FSI, with subproblem 1 an incompressible Stokesian fluid occupying $\Omega_1 \subset \mathbb{R}^3$ and subproblem 2 a thin immersed structure modeled geometrically as the surface Γ, of codimension one to Ω_1. The inner product $(\cdot, \cdot)_\Gamma$ is the $L^2(\Gamma)$ inner product, with appropriate traces taken of its arguments when necessary. Discretize the problem implicitly in time, with the backward Euler method, using time step Δt. The time-discrete fluid physics are then given by

$$B_1(u, v) = \frac{\rho_1}{\Delta t}(u, v)_{L^2(\Omega_1)} + a_1(u, v) \ , \tag{5.145}$$

where ρ_1 is the fluid mass density and a_1 is an $H^1(\Omega_1)$-coercive bilinear form. Rather than introducing a pressure to enforce incompressibility, simply consider $\mathcal{V}_1$ to be the space of discretely divergence-free velocities, to remain in the simpler setting of coercive problems. (Alternatively, in the case of pressure-stabilizing methods, the method is coercive over the whole pressure–velocity product space.) The structure physics are given by

$$B_2(u,v) = \frac{\rho_2 \ell_{\text{th}}}{\Delta t}(u,v)_\Gamma + \Delta t a_2(u,v)\ , \tag{5.146}$$

where ρ_2 is the structural mass density, ℓ_{th} is the structure's thickness, and a_2 is an $H^2(\Gamma)$-coercive bilinear form. For Stokes flow, a_1 is symmetric, so we can define the norms $\{\|\cdot\|_i\}_{i=1}^2$ by

$$\|u\|_i^2 = B_i(u,u)\ . \tag{5.147}$$

Suppose the coercivity constants for a_1 and a_2 are given by

$$|a_1(u,u)| \geq A_1 \|u\|^2_{H^1(\Omega_1)} \tag{5.148}$$

and

$$|a_2(u,u)| \geq A_2 \|u\|^2_{H^2(\Gamma)}\ . \tag{5.149}$$

Suppose further that the trace inequality

$$\|u\|^2_{H^1(\Omega_1)} \geq T \|u\|^2_\Gamma \tag{5.150}$$

and the trace-inverse inequality

$$\|u\|^2_{L^2(\Omega_1)} \geq Ih \|u\|^2_\Gamma \tag{5.151}$$

hold, where h is a mesh parameter. Then it is clear that

$$\left(\frac{\rho_1 I h}{\Delta t} + A_1 T + k\right) \|u\|_\Gamma^2 \le |||u|||_1^2 \tag{5.152}$$

and

$$\left(\frac{\rho_2 \ell_{\text{th}}}{\Delta t} + \Delta t A_2 + k\right) \|u\|_\Gamma^2 \le |||u|||_2^2 \ . \tag{5.153}$$

Using these estimates in lieu of the duller bound (5.143)–(5.144),

$$C_1^2 = \frac{k}{\frac{\rho_1 I h}{\Delta t} + A_1 T + k} \tag{5.154}$$

and

$$C_2^2 = \frac{k}{\frac{\rho_2 \ell_{\text{th}}}{\Delta t} + \Delta t A_2 + k} \ . \tag{5.155}$$

Obviously $C_i < 1$, but if $|C_1 C_2 - 1| \ll 1$, then convergence of block iteration will be quite slow. Suppose the penalty parameter is given by

$$k = K/h \tag{5.156}$$

and the time step is given by

$$\Delta t = \tau h \ , \tag{5.157}$$

where K and τ are independent of the mesh parameter h. Then

$$C_1^2 \to 1 \quad \text{as} \quad h \to 0 \tag{5.158}$$

and

$$C_2^2 \to \frac{K}{\frac{\rho_2 \ell_{\text{th}}}{\tau} + K} \quad \text{as} \quad h \to 0 \ . \tag{5.159}$$

The convergence therefore approaches a fixed rate under refinement. That rate can be improved by shrinking the time step–mesh size proportionality constant τ and/or decreasing the mesh-independent penalty parameter K. This is consistent with the trends noted in [105, Remark 14] and demonstrated in numerical examples of the sequel.

5.3.3 Relation to Newton iteration

In [105, Section 4.6] and Section 4.3, I introduced block iteration as a modification of Newton iteration with an approximate tangent. This section examines precisely how block iteration and inexact Newton iteration are related and indicates why block iteration is preferable. In the abstract model problem of Section 5.3.1, Newton iteration would have the tangent system

$$\begin{aligned}&\begin{pmatrix} B_1(\cdot,w_1)+k(\cdot,w_1)_\Gamma & -k(\cdot,w_1)_\Gamma \\ -k(\cdot,w_2)_\Gamma & B_2(\cdot,w_2)+k(\cdot,w_2)_\Gamma \end{pmatrix}\begin{pmatrix}\Delta u_1 \\ \Delta u_2\end{pmatrix} \\ &\qquad = -\begin{pmatrix} B_1(u_1^i,w_1)+k(u_1^i-u_2^i,w_1)_\Gamma-F_1(w_1) \\ B_2(u_2^i,w_2)+k(u_1^i-u_2^i,w_2)_\Gamma-F_2(w_2)\end{pmatrix}\ , \end{aligned} \tag{5.160}$$

which would, for a linear problem, arrive at the exact solution in a single iteration, from any initial guess. Eliminating off-diagonal blocks produces two independent equations to update u_1 and u_2:

$$B_1(\Delta u_1,w_1)+k(\Delta u_1,w_1)_\Gamma = -B_1(u_1^i,w_1)-k(u_1^i-u_2^i,w_1)_\Gamma+F_1(w_1) \tag{5.161}$$

and

$$B_2(\Delta u_2,w_2)+k(\Delta u_2,w_2)_\Gamma = -B_2(u_2^i,w_2)-k(u_2^i-u_1^i,w_2)_\Gamma+F_2(w_2)\ . \tag{5.162}$$

Using $u_j^{i+1}=u_j^i+\Delta u_j$ for $j\in\{1,2\}$, and the linearity of $B_j(\cdot,w_j)$ and $(\cdot,w_j)_\Gamma$, these two update equations are clearly equivalent to

$$B_1(u_1^{i+1},w_1)+k(u_1^{i+1},w_1)=k(u_2^i,w_1)_\Gamma+F_1(w_1) \tag{5.163}$$

and

$$B_2(u_2^{i+1}, w_2) + k(u_2^{i+1}, w_2)_\Gamma = k(u_1^i, w_2) + F_2(w_2) \ . \tag{5.164}$$

Unlike the problem of Section 5.3.1, inexact Newton iteration will require an initial guess for u_1^0, to compute u_2^1. For u_2^i, $i > 1$, one can follow analogous steps to those spelled out in Section 5.3.1 to derive a counterpart to (5.136):

$$u_2^{i+1} = \mathcal{H}(u_2^{i-1}) \ , \tag{5.165}$$

where $\mathcal{H}$ is the same as that defined in (5.136). Block iteration is therefore an acceleration of the inexact Newton approach that converges twice as quickly when the subproblems are linear.

Remark 5.7. Notice that the inexact Newton iteration, when applied to the problem of Section 5.3.2, would be equivalent to the following: putting the coupling force in explicitly and adding extra mass along the interface Γ in the tangent matrix for each subproblem. This is suggested heuristically in [181, Section 5.1], as a way to improve the robustness of classical Dirichlet-to-Neumann block iteration. In block iteration for penalty-coupled problems, there is a precise way to determine the amount of extra mass needed to guarantee stability. This interpretation suggests, however, that under-converging the block iteration may cause the structure to behave as if it has extra mass.

5.3.4 Numerical test

I now test the convergence of block iteration for the model coupled problem suggested in Section 5.1.2.3. In particular, I choose $\Omega = (-W/2, W/2)^2 \subset$

$\mathbb{R}^2$ with $W = 2.5$ and select Γ to be the intersection of the line $x_2 = 3x_1 + W/4$ with Ω. I set

$$a_1(u, v) = (\nabla u, \nabla v)_{L^2(\Omega)} \ , \tag{5.166}$$

$$a_2(y, z) = (\nabla y, \nabla z)_{L^2(\Gamma)} \ , \tag{5.167}$$

$\rho_1 = \rho_2 = 1$, and select the functional F so as to strongly enforce $u = 1$ on the left edge of the domain. (I have both abused notation and stretched the problem definition in this example, by first conflating $u \in H_0^1(\Omega)$ with $u + g$, where g satisfies the inhomogeneous boundary condition, then selecting $g \notin H^1(\Omega)$. In computations, the discontinuous boundary data is implemented analogously to the "leaky lid" discretization of the lid-driven cavity benchmark, as depicted in [92, Figure 1].) A representative solution snapshot is shown in Figure 5.7. Notice that the Lagrange multiplier values, plotted as coloration along the physical image of Γ, are highly oscillatory, while the temperatures u and y remain qualitatively smooth.

To investigate the effect of time step on block iterative convergence, I set $T = 10$ and take $\Delta t = T/N$ for $N \in \{1, 10, 100\}$, holding $\beta > 0$ fixed and $r = 0$. The (log of the) ℓ^2 norm of the discrete residual for the Γ subproblem during the first time step is shown as a function of the number of block iterations in Figure 5.8. This differs from the choice of norm used in the convergence analysis, but, for a fixed number of degrees of freedom in the spatial discretization, all norms of the finite-dimensional solution space are equivalent. The linear convergence rate and improvement with temporal refinement

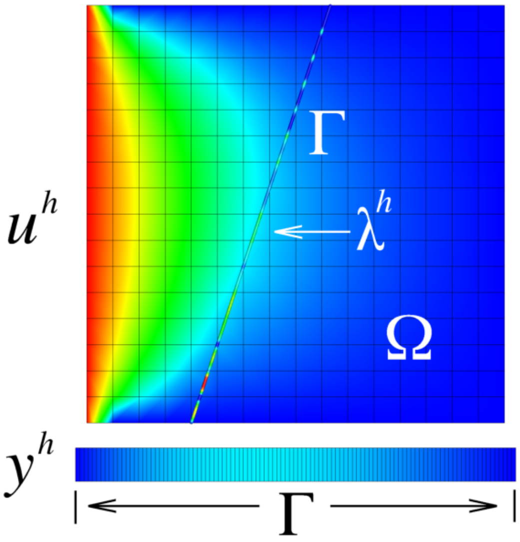


Figure 5.7: Annotated snapshot of a solution to the coupled model problem. The solution to the Γ subproblem, y^h, is plotted below Ω and the colors on Γ cutting through Ω represent point values of λ^h.

predicted in the preceding analysis are confirmed.

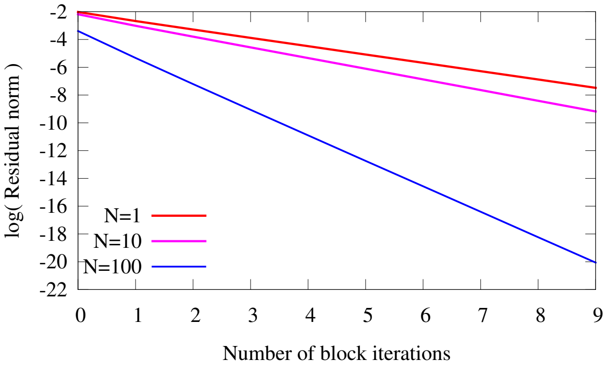


Figure 5.8: The norm of the discrete residual for the Γ subproblem converges linearly with a rate that improves as the time step $\Delta t = T/N$ decreases.

To investigate the effect of penalty parameter, I use $\Delta t = T/100$ and choose $\beta = C/h$, for $C \in \{1, 10, 100\}$ and h the mesh element size shown in Figure 5.7. The block iterative convergence is shown in Figure 5.9. This illustrates that block iterative convergence slows down with increasing penalty value. This highlights the value of including a semi-implicitly-integrated Lagrange multiplier rather than simply using a naive penalty method, in spite of these approaches having the same asymptotic convergence rates. The inclusion of the Lagrange multiplier allows for satisfactory constraint enforcement with lower (and therefore more computationally tractable) penalty values.

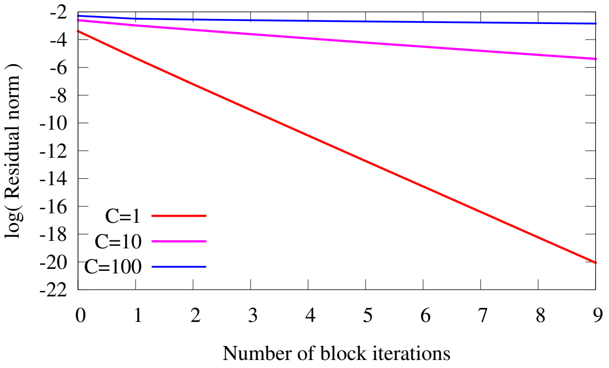


Figure 5.9: The norm of the discrete residual for the Γ subproblem converges linearly with a rate that slows down as the penalty C/h increases.

5.4 Improving mass conservation in PSPG

If an immersed boundary induces a large pressure jump in the fluid, it is acting as a concentrated irrotational force. I isolate the effects of such a force by looking at a linear model problem: the Stokes "no-flow" problem. This problem is introduced precisely by Galvin et al. [59]. Breifly, it is a Stokes flow with homogeneous Dirichlet boundary conditions and an irrotational body force. The exact solution is that the pressure is equal to the potential generating this irrotational vector field and the fluid velocity is zero. (This is also an exact solution to the nonlinear Navier–Stokes equation.) In a numerical method, if the pressure interpolation error enters the bound on the velocity error, then the discrete velocity may be nonzero. If the pressure gradient in the exact solution is very large relative to other data in the problem, then the discrete solution can be very far from hydrostatic. This problem occurs quite dramatically in BHV FSI analysis with VMS discretizations of the fluid subproblem. The effect of the pressure interpolation error on velocity manifests, in the valve problem, as poor mass conservation near the valve, which leads to a *de facto* leakage through it, even when the kinematic constraints are well-enforced. This effect is what motivates the introduction of the scaling factor $s(\mathbf{x})$ in the VMS stabilization parameters.

Galvin et al. [59] investigated the phenomenon of poor mass conservation in incompressible flows with irrotational forcing using inf-sup stable velocity/pressure pairs, and found that, in the presence of large irrotational forces, it was beneficial to use unusually-high grad-div stabilization constants. (Recall that grad-div stabilization is exactly the same thing as the τ_{C} term of the VMS formulation stated in Section 3.1.1.) Galvin et al. scaled τ_{C} globally by factors of up to 10^4, but found that excessive scaling could lead to bad results. When doing immersed boundary analysis with the VMS fluid formulation, there are two key departures from the program of Galvin et al.:

1. We know *a priori* where the large irrotational forces (and thus large pressure interpolation errors) will be located. Specifically, we know that the forces will be in elements containing the immersed structure.
2. The pressure interpolation error contributes to the velocity error not only through the Galerkin term of the weak continuity equation, but also through the appearance of the pressure gradient in the momentum residual $\mathbf{r}_{\mathrm{M}}$, in the stabilization terms. These contributions to the velocity error are controlled by τ_{M}. We might therefore expect to benefit from modifying τ_{M} in addition to τ_{C}.

Interestingly, the issue of poor mass conservation in immersed boundary computations goes all the way back to Peskin's work, as discussed in [132], but Peskin and Printz were not working in a variational framework and therefore did not have available the machinery of functional analysis to clearly explain the problem.

Let us now do some simple error analysis of the VMS formulation for the generic no-flow problem. The VMS analysis used to obtain B_1^{VMS} for unsteady Navier–Stokes reduces, in the case of steady Stokes flow, to the pressure-stabilizing/Petrov–Galerkin (PSPG) formulation [92] augmented with least-squares stabilization of the incompressibility constraint (LSIC, also known as grad-div) [73]. The PSPG/LSIC discrete problem is: Find $(\mathbf{u}^h, p^h) \in \mathcal{S}_u^h \times \mathcal{S}_p^h$ such that $\forall(\mathbf{w}^h, q^h) \in \mathcal{V}_u^h \times \mathcal{V}_p^h$,

$$B^{\mathrm{PSPG}}(\{\mathbf{u}^h, p^h\}, \{\mathbf{w}^h, q^h\}) = F^{\mathrm{PSPG}}(\{\mathbf{w}^h, q^h\}) \ , \tag{5.168}$$

with

$$\begin{aligned} B^{\text{PSPG}}(\{\mathbf{u}^h, p^h\}, \{\mathbf{w}^h, q^h\}) = \mu \int_\Omega \nabla \mathbf{u}^h : \nabla \mathbf{w}^h \; d\Omega \\ + \int_\Omega q^h \nabla \cdot \mathbf{u}^h \; d\Omega - \int_\Omega p^h \nabla \cdot \mathbf{w}^h \; d\Omega \\ + \sum_e \int_{\Omega^e} \tau_{\text{M}}^e \nabla q^h \cdot (-\mu \Delta \mathbf{u}^h + \nabla p^h) \; d\Omega \\ + \sum_e \int_{\Omega^e} \tau_{\text{C}}^e \nabla \cdot \mathbf{u}^h \nabla \cdot \mathbf{w}^h \; d\Omega \; , \end{aligned} \tag{5.169}$$

where the PSPG stabilization constant $\tau_{\text{M}}^e = \mathcal{O}(h^2/\mu)$ and the LSIC stabilization constant $\tau_{\text{C}}^e = \mathcal{O}(h^2/\tau_{\text{M}}^e)$ are steady Stokes flow counterparts to the synonymous stabilization constants that appear in the unsteady Navier–Stokes VMS formulation. τ_{M}^e must obey an upper bound (derived from inverse estimates that bound higher derivatives of discrete polynomial test and trial functions in terms of lower derivatives) for B^{PSPG} to be coercive in the so-called "stability norm",

$$|||\{u, p\}|||^2 = \frac{1}{2}\mu \|\nabla \mathbf{u}\|^2_{L^2(\Omega)} + \frac{1}{2}\sum_e \tau_{\text{M}}^e \|\nabla p\|^2_{L^2(\Omega^e)} + \frac{1}{2}\sum_e \tau_{\text{C}}^e \|\nabla \cdot \mathbf{u}\|^2_{L^2(\Omega^e)} \; , \tag{5.170}$$

introduced by Hughes et al. [92] and extended here to include an LSIC contribution. Following the analysis of [92], I begin by decomposing the error into discrete and interpolation errors. Let the exact solution be $\{\mathbf{u}, p\}$. The error is then defined to be

$$\{\mathbf{e}_u, e_p\} = \{\mathbf{u}^h - \mathbf{u}, p^h - p\} \; . \tag{5.171}$$

Adding and subtracting arbitrary discrete functions $\bar{\mathbf{u}}^h \in \mathcal{V}_u^h$ and $\bar{p}^h \in \mathcal{V}_p^h$,

$$\mathbf{e}_u = \left(\mathbf{u}^h - \bar{\mathbf{u}}^h\right) + \left(\bar{\mathbf{u}}^h - \mathbf{u}\right) = \mathbf{e}_u^h + \boldsymbol{\eta}_u \ , \tag{5.172}$$

$$e_p = \left(p^h - \bar{p}^h\right) + \left(\bar{p}^h - p\right) = e_p^h + \eta_p \ , \tag{5.173}$$

where $\boldsymbol{\eta}_u = \bar{\mathbf{u}}^h - \mathbf{u}$ is the velocity interpolation error and $\eta_p = \bar{p}^h - p$ is the pressure interpolation error. These interpolation errors are unrelated to the numerical method, and depend only on the exact solution and choice of discrete spaces.

In the particular case of the no-flow problem, one can exactly interpolate the velocity $\mathbf{u} = \mathbf{0}$ by choosing $\bar{\mathbf{u}}^h = \mathbf{0}$, so $\boldsymbol{\eta}_u = \mathbf{0}$. Thus $\mathbf{e}_u = \mathbf{e}_u^h = \mathbf{u}^h$. We can therefore bound the H^1 seminorm of the spurious flow in the discrete solution in terms of the stability norm of the discrete error:

$$\frac{1}{2}\mu|\mathbf{u}^h|^2_{H^1(\Omega)} \leq \left|\left|\left|\{\mathbf{e}_u^h, e_p^h\}\right|\right|\right|^2 \ . \tag{5.174}$$

Because the PSPG/LSIC bilinear form is, by design, coercive with unit constant in the stability norm,

$$\left|\left|\left|\{\mathbf{e}_u^h, e_p^h\}\right|\right|\right|^2 \leq \left|B^{\text{PSPG}}\left(\{\mathbf{e}_u^h, e_p^h\}, \{\mathbf{e}_u^h, e_p^h\}\right)\right| \ . \tag{5.175}$$

PSPG/LSIC is in residual form and therefore consistent with the exact solution, so

$$B^{\text{PSPG}}\left(\{\mathbf{e}_u, e_p\}, \{\mathbf{w}^h, q^h\}\right) = 0 \tag{5.176}$$

for all discrete test functions $\{\mathbf{w}^h, q^h\}$. Thus

$$\left|\left|\left|\{\mathbf{e}_u^h, e_p^h\}\right|\right|\right|^2 \leq \left|B^{\text{PSPG}}\left(\{\boldsymbol{\eta}_u, \eta_p\}, \{\mathbf{e}_u^h, e_p^h\}\right)\right| \ . \tag{5.177}$$

Recalling that $\boldsymbol{\eta}_u = \mathbf{0}$ for the no-flow problem and using the definition of B^{PSPG},

$$B^{\mathrm{PSPG}}\left(\{\boldsymbol{\eta}_u, \eta_p\}, \{\mathbf{e}_u^h, e_p^h\}\right) = -\int_{\Omega} \eta_p \nabla \cdot \mathbf{e}_u^h \; d\Omega + \sum_e \int_{\Omega^e} \tau_{\mathrm{M}} \nabla e_p^h \cdot \nabla \eta_p \; d\Omega \; . \tag{5.178}$$

Splitting the first integral into a sum over elements and using Young's inequality in each term,

$$\begin{aligned} \left|B^{\mathrm{PSPG}}\left(\{\boldsymbol{\eta}_u, \eta_p\}, \{\mathbf{e}_u^h, e_p^h\}\right)\right| \leq & \sum_e \left(\frac{\|\eta_p\|_{L^2(\Omega^e)}^2}{2\epsilon_1^e} + \frac{\epsilon_1^e \|\nabla \cdot \mathbf{e}_u^h\|_{L^2(\Omega^e)}^2}{2} \right) \\ & + \sum_e \tau_{\mathrm{M}}^e \left(\frac{\|\nabla e_p^h\|_{L^2(\Omega^e)}^2}{2\epsilon_2} + \frac{\epsilon_2 \|\nabla \eta_p\|_{L^2(\Omega^e)}^2}{2} \right) \; , \end{aligned} \tag{5.179}$$

for arbitrary $\epsilon_1^e > 0$ and $\epsilon_2 > 0$. Choose $\epsilon_1^e = \tau_{\mathrm{C}}^e/2$ and $\epsilon_2 = 2$. Then terms involving the discrete errors $\mathbf{e}_u^h$ and e_p^h may be hidden behind the corresponding terms in the stability norm, so it follows that

$$\frac{1}{2} \left|\left|\left| \{\mathbf{e}_u^h, e_p^h\} \right|\right|\right|^2 \leq \sum_e \left(\frac{\|\eta_p\|_{L^2(\Omega^e)}^2}{\tau_{\mathrm{C}}^e} + \tau_{\mathrm{M}}^e \|\nabla \eta_p\|_{L^2(\Omega^e)}^2 \right) \; . \tag{5.180}$$

Recalling the asymptotics of the stabilization constants, let

$$\tau_{\mathrm{M}}^e \sim \alpha^e \frac{h^2}{\mu} \; , \tag{5.181}$$

$$\tau_{\mathrm{C}}^e \sim \frac{h^2}{\tau_{\mathrm{M}}^e} \sim \frac{\mu}{\alpha^e} \; , \tag{5.182}$$

where $\alpha^e > 0$ is a dimensionless scalar on each element. Recalling that the stability norm bounds H^1 velocity error,

$$|\mathbf{u}^h|_{H^1(\Omega)}^2 \leq \sum_e \frac{C\alpha^e}{\mu^2} \left(\|\eta_p\|_{L^2(\Omega^e)}^2 + h^2 \|\nabla \eta_p\|_{L^2(\Omega^e)}^2 \right) \; , \tag{5.183}$$

where C is independent of α^e and h. Thus, the spurious leakage in the discrete solution can be scaled down to arbitrarily small levels by reducing α^e in elements with pressure interpolation errors. This has the obvious consequence, however, of destabilizing the pressure field, and, in problems with nonzero $\mathbf{u}$, this could, based on standard PSPG error analysis [92], magnify the effects of velocity interpolation errors. It is straightforward to see that shrinking α^e in elements with pressure interpolation errors is akin to increasing the value of s^{shell} in the modified VMS formulation of Section 3.1.1.

Chapter 6

Nonlinear numerical experiments

This chapter tests, through numerical experiments, how well the convergence results derived within the linearized theory of Chapter 5 extrapolate to the discretization described in Chapters 3 and 4 of the nonlinear mathematical model specified in Chapter 2.

Some of the chapter's content is derived from the following publications:

D. Kamensky, M.-C. Hsu, D. Schillinger, J. A. Evans, A. Aggarwal, Y. Bazilevs, M. S. Sacks, T. J. R. Hughes. An immersogeometric variational framework for fluid–structure interaction: Application to bioprosthetic heart valves. *Computer Methods in Applied Mechanics and Engineering*, 284:1005–1053, 2015. D. Kamensky developed the techniques used for fluid–thin structure interaction and structure-on-structure contact. M.-C. Hsu provided supervision and implemented the finite cell method for flow around bulky objects. D. Schillinger helped formulate the finite cell approach used. J. A. Evans provided mathematical advice. A. Aggarwal developed the geometrical model of the valve. Y. Bazilevs, M. S. Sacks, and T. J. R. Hughes supervised the work.

D. Kamensky, J. A. Evans, M.-C. Hsu. Stability and conservation properties of collocated constraints in immersogeometric fluid-thin structure interaction analysis. *Communications in Computational Physics*. 18(4):1147–1180, 2015. D. Kamensky formulated and analyzed the improvements to the semi-implicit time integration and performed the numerical experiments. J. A. Evans and M.-C. Hsu supervised the work.

D. Kamensky, M.-C. Hsu, Y. Yu, J. A. Evans, M. S. Sacks, T. J. R. Hughes. Immersogeometric cardiovascular fluid–structure interaction analysis using divergence-conforming B-splines. *Computer Methods in Applied Mechanics and Engineering*, In review (preprint: ICES Report 16-14). D. Kamensky implemented the numerical methods, formulated and analyzed model problems, and participated in the experimental work. M.-C. Hsu provided supervision and edited the manuscript extensively. Y. Yu, J. A. Evans, and T. J. R. Hughes supervised the mathematical analysis. M. S. Sacks helped plan and supervise the laboratory experiments.

6.1 Extrapolation to Navier–Stokes flow

Before looking at the full FSI problem, consider the rigid-structure limit of Navier–Stokes flow with Dirichlet conditions enforced on immersed boundaries. While this seems like a simpler problem, it is, in fact, especially challenging for a wide class of immersed boundary methods that rely on the structure's constitutive model to compute forces on the fluid. The rigid-structure limit is important in the application of BHV simulation, as many BHV designs include effectively rigid stents.

6.1.1 Taylor–Green vortex

This section considers a variant of the numerical experiment from Section 5.1.2.2, but with the full Navier–Stokes equations. Recall that Section 5.1.2.2 solved the Stokes equations, with the advection term of the Taylor–Green vortex solution (5.107) prescribed as a body force. I now treat the advection term nonlinearly. As in Section 5.1.2.2, div-conforming B-splines of degree $k' = 1$ are used to discretize the velocity and pressure spaces and backward Euler integration is applied in time. In this section, I consider the case of low Reynolds number flow, and choose $\mu = 0.01$. (A high Reynolds-number stress test is carried out in Section 6.1.3.)

An interesting phenomenon that I have noticed in nonlinear computations is that prescribing an immersed boundary velocity that differs from the actual movement of Γ_t leads to severely degraded performance and, at high Reynolds numbers and/or over long time intervals, an apparent lack of convergence. For this reason, I use the boundary of the square $[-\pi, \pi]$ for Γ_t. No fluid flows across this boundary in (5.107). To avoid any special behavior associated mesh-aligned immersed boundaries, I distort the background mesh in a periodic manner, shown in Figure 6.1. In the notation of Section 3.1.2.2, this corresponds to a deformation of

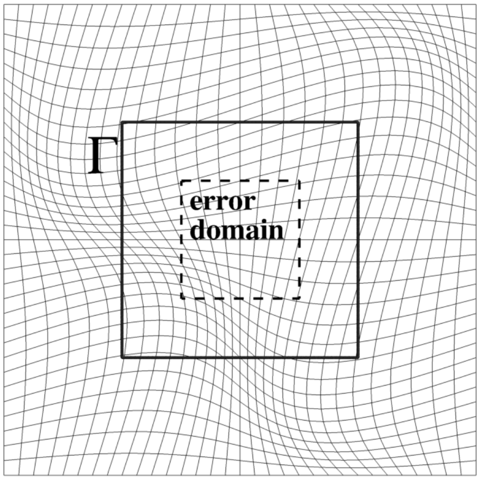


Figure 6.1: The non-rectilinear mesh of Ω avoids grid alignment with Γ.

$$\boldsymbol{\phi}(\mathbf{X}) = \left(X_1 + A\sin\left(\frac{\pi X_1}{W/2}\right)\sin\left(\frac{\pi X_2}{W/2}\right), X_2 + A\sin\left(\frac{\pi X_1}{W/2}\right)\sin\left(\frac{\pi X_2}{W/2}\right)\right) \tag{6.1}$$

applied to the parametric domain $\widehat{\Omega} = (-W/2, W/2)^2 \subset \mathbb{R}^2$, with $A = 1$ and $W = 4\pi$. To test convergence, I divide the parametric domain $\widehat{\Omega}$ evenly into $2^N \times 2^N$ elements, for $N \in \{4, 5, 6, 7, 8\}$. The time interval $(0, T = 0.7)$ is divided into steps of size $\Delta t = T/2^{N-3}$. Penalty parameters are defined by (4.8) and (4.9), with h defined to be $W/2^N$ (regardless of mesh distortion), $C_{\text{NOR}}^{\text{inert}} = C_{\text{NOR}}^{\text{inert}} = 1000$, and $C_{\text{TAN}} = 100$. The initial condition is set by $H^1(\Omega)$

projection onto the divergence-free discrete subspace. As in Section 5.1.2.2, I measure error on a subset Ω^{err} of the domain. For Navier–Stokes flow, it is important to consider the advection-dominated limit, in which the H^1 norm of the exact velocity solution diverges near the boundary (due to a discontinuous velocity field, which is $\notin H^1(\Omega)$), so I define the error domain to have an $\mathcal{O}(1)$ separation from Γ_t, as shown in Figure 6.1. I choose $\Omega^{\text{err}} = [-W/8, W/8]$. Due to the mesh distortion, Ω^{err} is not a union of elements. I integrate errors on Ω^{err} inexactly, by using whatever analysis quadrature points happen to fall in Ω^{err}. While this is a crude, first-order quadrature scheme, it shouldn't influence asymptotic convergence rates, based on the *a priori* analysis of and numerical experience with the simpler linear problems, for which at most first-order convergence is found, regardless of norm.

An annotated snapshot of a solution illustrates the problem setup in Figure 6.2. The convergence of errors on Ω^{err} is shown in Figure 6.3. The nearly first-order convergence rates obtained suggest that the analysis of linear parabolic problems extrapolates reasonably well to Navier–Stokes flow.

6.1.2 Translating Taylor–Green vortex

The addition of a uniform velocity to an initial condition in a periodic domain yields a Galilean transformation of the original solution. In this section, I superpose velocity $\mathbf{v} = -0.87\mathbf{e}_1 - 0.5\mathbf{e}_2$ on top of the initial condition of the problem from Section 6.1.1 and translate the boundary Γ_t at the same velocity. A snapshot of the solution at time T is shown in Figure 6.4. Fig-

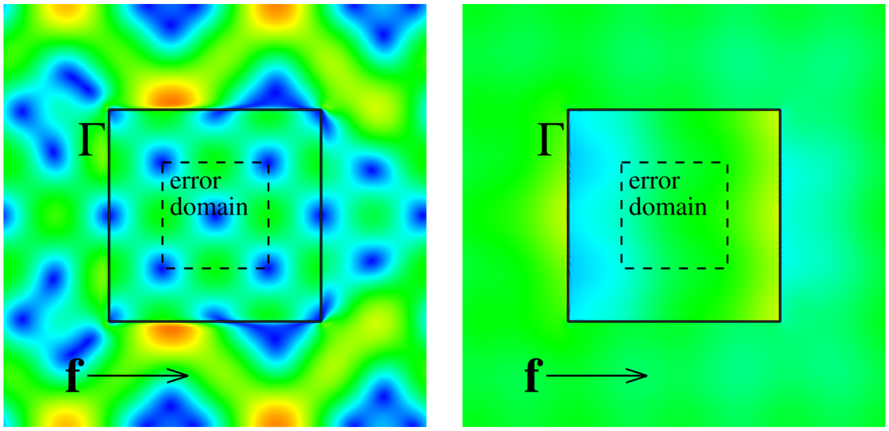


Figure 6.2: Simultaneous velocity magnitude (left) and pressure (right) snapshots of the Navier–Stokes Taylor–Green problem, with annotations describing the problem setup.

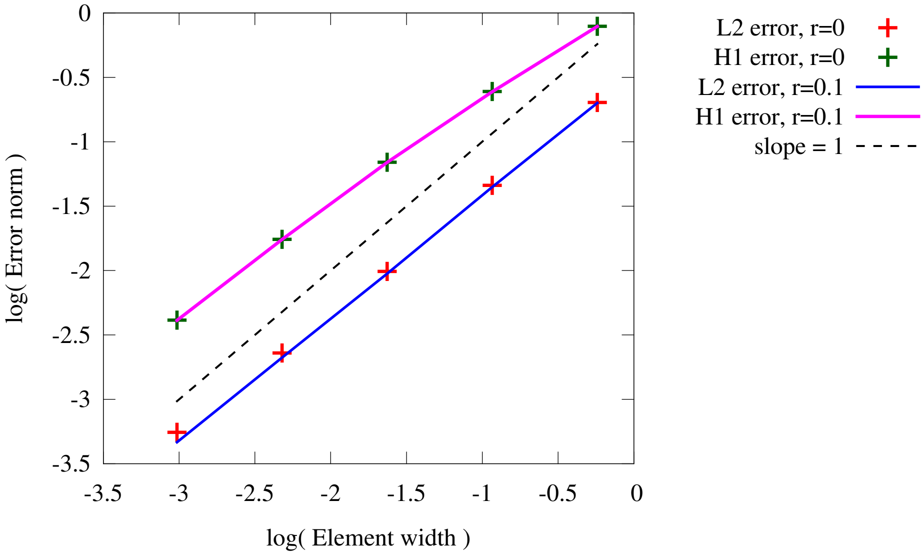


Figure 6.3: Convergence of the $L^2(\Omega^{\mathrm{err}})$ and $H^1(\Omega^{\mathrm{err}})$ errors for $r = 0$ and $r = 0.1$ for Navier–Stokes flow with a stationary boundary and positive viscosity.

ure 6.5 illustrates how the near-first-order convergence on Ω^{err} remains intact.

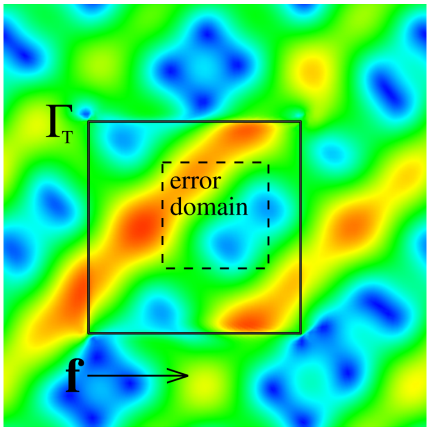


Figure 6.4: Annotated snapshot of velocity magnitude at time T for Navier–Stokes flow with moving boundaries and positive viscosity. (Note the translation of Γ relative to its initial position, shown in Figure 6.2.)

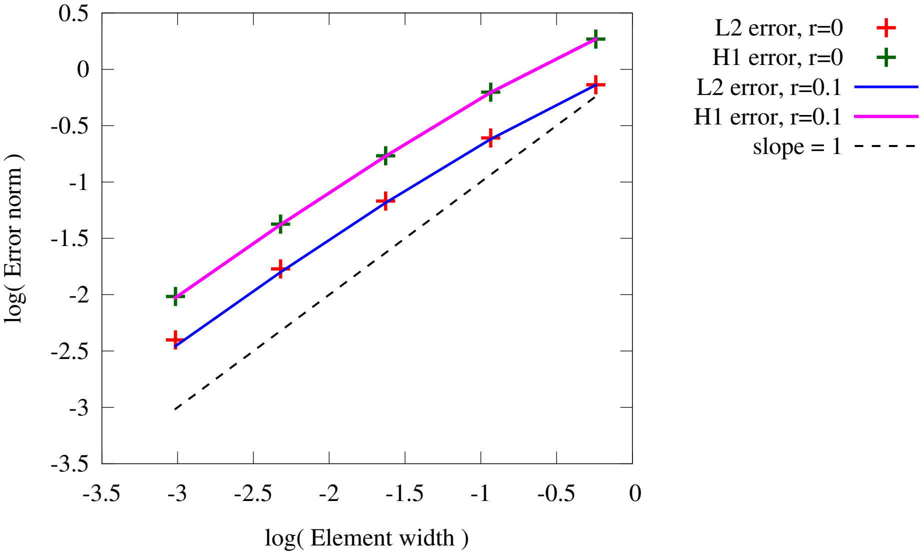


Figure 6.5: Convergence of the $L^2(\Omega^{\text{err}})$ and $H^1(\Omega^{\text{err}})$ errors for $r = 0$ and $r = 0.1$ for Navier–Stokes flow with moving boundaries and positive viscosity.

6.1.3 Infinite Reynolds number

To demonstrate the robustness of the proposed methodology at realistic Reynolds numbers, I repeat the test of Section 6.1.2 with $\mu = 0$. The exact solution becomes tangentially discontinuous at Γ_t. This behavior is captured reasonably well, as shown in Figure 6.6, in spite of the application of (4.8) and (4.9), which provides no enforcement of tangential boundary conditions when $\mu = 0$. The nearly-linear convergence rates in local $L^2(\Omega^{\text{err}})$ and $H^1(\Omega^{\text{err}})$ seminorms are maintained as well, as shown in Figure 6.7, despite the fact that the global $H^1(\Omega)$ norm of the discontinuous exact solution is not well-defined. For $\mu = 0$ and the Δt–h relationship used here, the normal penalty (4.9) is $\mathcal{O}(1)$ as $h \to 0$. Using $r > 0$ therefore introduces an $\mathcal{O}(1)$ perturbation into the no-penetration constraint on Γ_t, so I would not expect asymptotic convergence. In this example, I therefore use $r = 0$. (Even for significant values of $r > 0$, though, I cannot practically refine the mesh enough for the resulting perturbation to dominate errors in this problem.)

Remark 6.1. Recall that the discrete formulation includes h-dependent artificial streamline diffusion (of the form (3.28)) to stabilize advection. In the absence of this diffusion, the solution becomes highly oscillatory.

6.2 2D non-coapting valve

This section looks at a 2D valve-inspired benchmark problem investigated previously by [61, 77, 102, 197]. Because the structure does not contact itself and the low-Reynolds number fluid dynamics are stable, it is straightforward to compute converged solutions using thoroughly verified body-fitted methods. This problem is therefore a valuable verification test for new immersed approaches. I originally performed the body-fitted reference and im-

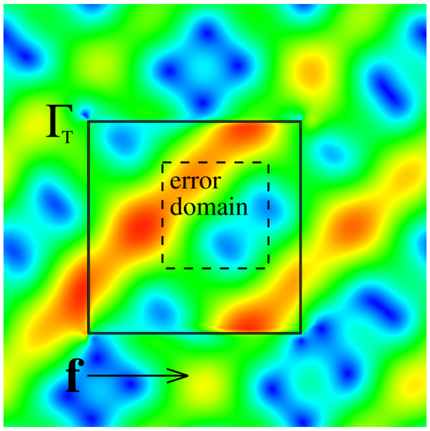


Figure 6.6: Annotated snapshot of velocity magnitude at time T for Navier–Stokes flow with moving boundaries and zero viscosity.

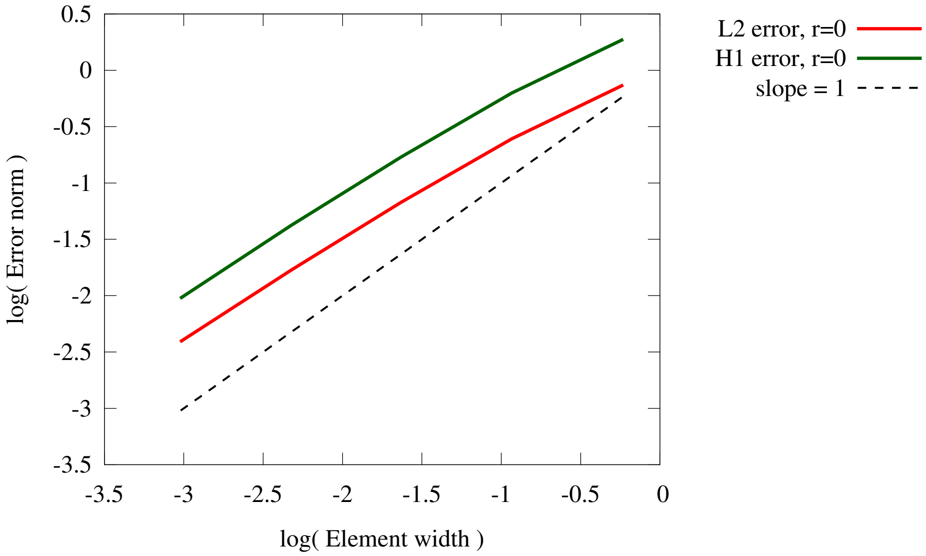


Figure 6.7: Convergence of the $L^2(\Omega^{\mathrm{err}})$ and $H^1(\Omega^{\mathrm{err}})$ errors for $r = 0$ for Navier–Stokes flow with moving boundaries and zero viscosity.

mersogeometric VMS computations for [105, Section 4.7], and have adapted text from that paper in this section.

6.2.1 Description of the problem

The problem consists of two cantilevered elastic beams immersed in a 2D channel filled with incompressible Newtonian fluid, as shown in Figure 6.14. The fluid and structure have equal densities of $\rho_1 = \rho_2 = 100$. The viscosity of the fluid is $\mu = 10$. Gil et al. studied a variety of material models for the beams while Hesch et al. used a nearly incompressible neo-Hookean material, with Young's modulus $E = 5.6 \times 10^7$ and Poisson ratio $\nu = 0.4$. I use the St. Venant–Kirchhoff model described in Section 2.3.2.1 with $E = 5.6 \times 10^7$ and Poisson ratio $\nu = 0.4$. The top and bottom sides of the channel have no-slip boundary conditions, the left end has a prescribed, time-dependent velocity profile, and the right end is a traction-free outflow. The velocity Dirichlet condition on the left end of the channel is given by the formula

$$\mathbf{u}_1\left(y\mathbf{e}_2, t\right) = \begin{cases} 5(\sin(2\pi t) + 1.1)y(1.61 - y)\mathbf{e}_1 & , \quad t > 0 \\ \mathbf{0} & , \quad \text{otherwise} \end{cases} , \tag{6.2}$$

where the origin of the spatial coordinate system is at the bottom left corner of the domain. At times $t < 0$, the fluid and structure are at rest. Taking the channel width of 1.61 as a characteristic length scale and the peak inflow speed of 6.8 as a characteristic flow speed, the Reynolds number is approximately 110. At such low Reynolds numbers, there is little risk of backflow divergence, so the parameter γ in (2.5) is set to zero.

6.2.2 Body-fitted reference computation

The mesh for the body-fitted reference computation is shown in Figure 6.9. The fluid domain consists of 7626 quadratic B-spline elements. Each

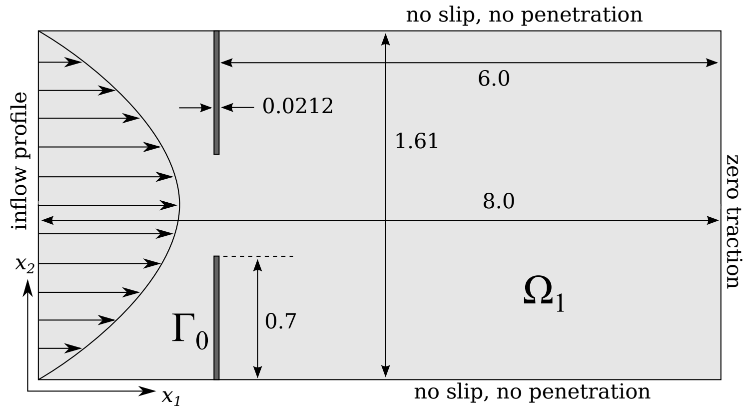


Figure 6.8: Geometry and boundary conditions of the 2D heart valve benchmark. Not to scale. The inflow profile is given by (6.2).

beam consists of 31 quadratic B-spline elements and is coincident with a line of C^0 continuity in the fluid B-spline space, permitting strong enforcement of fluid–structure kinematic constraints. I use generalized-α time integration with $\rho_\infty = 0.5$ and a time step of $\Delta t = 0.005$ for the body-fitted computation. The selected spatial and temporal resolutions ensure that the displacement history of the upper beam tip changes negligibly (~ 0.001 length units) with further refinement in both space and time.

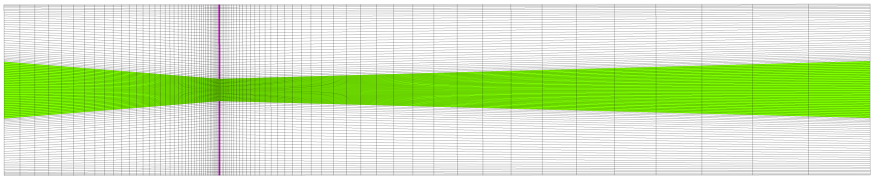


Figure 6.9: The reference configuration of the body-fitted mesh for the 2D valve problem, with leaflets highlighted in magenta and areas of softened mesh highlighted in green.

The fluid mesh deforms from one time step to the next according to the solution of a fictitious isotropic linear elastic problem that takes the location of the beam as a displacement boundary condition. The velocity of this deformation enters into the fluid formulation (3.2) as $\hat{\mathbf{u}}^h$. This velocity is derived from displacements of the mesh in consecutive time steps. Mesh quality is preserved throughout this deformation by stiffening the fictitious material in response to compression: the material tensor is modified such that the mesh Young's modulus, E_{mesh}, scales inversely with the square of the Jacobian determinant, J_ξ, of the mesh's parametric mapping in the previous time step. More detailed discussions of Jacobian-based mesh stiffening can be found in [18, 23, 98, 165, 166, 176]. In the present problem, I also find it necessary to soften the fictitious material governing the deformation of elements between the leaflets. This is accomplished by making its Young's modulus (prior to Jacobian-based stiffening) 1000 times smaller than that of the material adjacent to the leaflets. The regions of softened mesh are highlighted in green in Figure 6.9. A snapshot of the resulting deformed mesh at time $t = 0.5$ is in Figure 6.10. The non-smooth deformation visibly demonstrates the effect of the jump in fictitious material parameter.

The parabolic inflow profile given by (6.2) is represented exactly, using the trace space of the B-spline basis functions. Under the assumption that the geometrical mapping from the B-spline parameter ξ_2 to the physical y-coordinate is time-independent, linear, and invertible at the inflow face of the domain, the velocity profile may be applied by first pre-computing x-direction velocity coefficients for the left-most row of control points such that

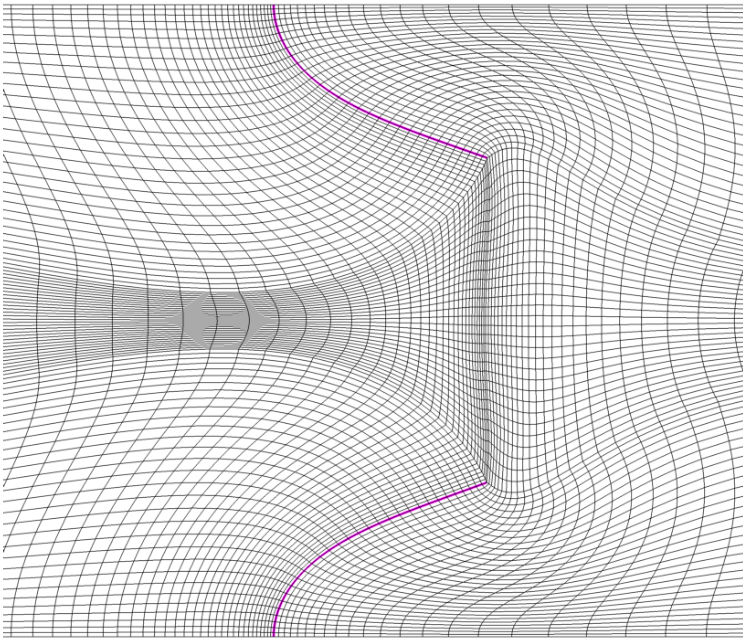


Figure 6.10: The deformation of the body-fitted fluid mesh at $t = 0.5$.

the resulting B-spline curve interpolates the function $y(1.61-y)$ at its Greville abscissae. These coefficients may be scaled by $5(\sin(2\pi t) + 1.1)$ during the computation, to obtain the desired velocity profile at time t.

6.2.3 Immersogeometric computations

I test three immersogeometric discretizations of the problem. The first, which I refer to here as M1, evenly divides the fluid domain into 128×32 quadratic B-spline elements and each beam into 64 quadratic B-spline elements. The other two discretizations are uniform refinements of M1: M2 contains 256×64 fluid elements and 128 shell elements in each beam, while M3 contains 512×128 fluid elements and 256 shell elements in each beam. As in the reference computation, the inflow velocity profile is captured exactly on these meshes. I refine in time alongside spatial refinement, using $\Delta t = 0.01$ with M1, $\Delta t = 0.005$ with M2, and $\Delta t = 0.0025$ with M3.

The time integration of the fluid–structure coupling is done using unstabilized semi-implicit algorithm (i.e. $r = 0$, in the notation of Chapter 4) with the generalized-α parameters determined by $\rho_\infty = 0.5$. Following (4.8) and the low-Reynolds number branch of (4.9), I scale the penalty parameters $\tau^B_{(\cdot)}$ inversely with mesh size, choosing $\tau^B_{(\cdot)} = 10^4$ on M1, $\tau^B_{(\cdot)} = 2 \times 10^4$ on M2, and $\tau^B_{(\cdot)} = 4 \times 10^4$ on M3. The VMS stabilization parameters are scaled near the structure using $s^{\text{shell}} = 10^6$.

6.2.4 Comparison of results

Figure 6.11 shows the x- and y-direction displacements of the upper beam tip for the body-fitted and immersed computations. The displacement histories extracted from immersogeometric discretizations M1, M2, and M3 converge toward the body-fitted result. Comparisons of the pressure contours at time $t = 0.5$ are given in Figure 6.12, showing agreement between the immersogeometric and body-fitted flow fields in regions outside of an $\mathcal{O}(h)$ neighborhood of the immersed beams. Velocity streamlines at $t = 0.5$ for the background mesh M1 are shown in Figure 6.13, demonstrating that the velocity field remains smooth on this coarse mesh, in spite of the pressure error evident from Figure 6.12. This is in contrast to the findings of Baaijens [8], who observed excessive pollution effects in the velocity field when discretizing the pressure about an immersed beam with a continuous approximation space. Compare the velocity streamlines of Figure 6.13 with [8, Figure 2]. Baaijens concluded that the use of a discontinuous pressure space “appears to be mandatory” [8, p. 749], but, in the present computations, the use of $s^{\text{shell}} > 1$ diminishes the pollution effects of the localized pressure interpolation error, as demonstrated also in Section 6.4.2, allowing acceptable results with continuous and equal-order pressure/velocity pairs.

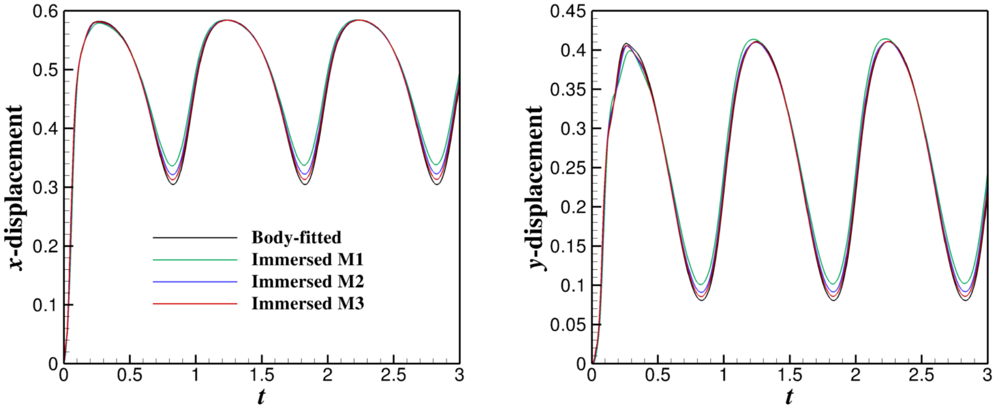
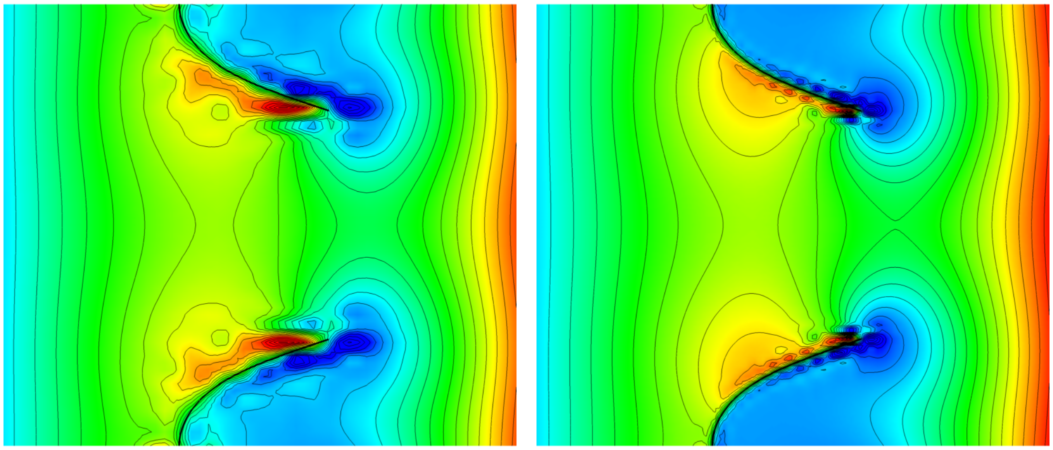
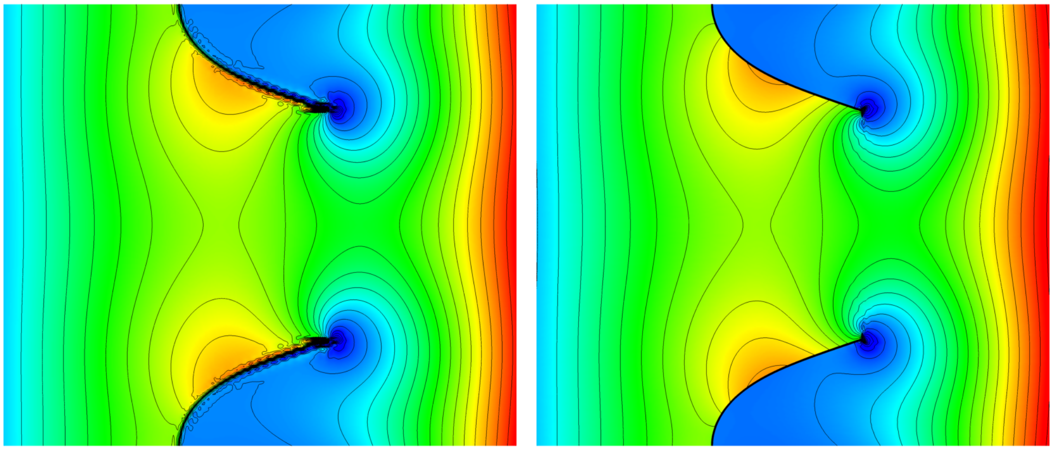


Figure 6.11: The x- and y-displacements of the upper leaflet tip, computed on the immersed and body-fitted meshes.

In regions where the beams are skew to the grid of knot lines in the B-spline background mesh, the pressure contours of Figure 6.12 reveal oscillations at a spatial frequency corresponding to the spacing of knots (i.e. the grid size, h). This is due to the fact that the modified VMS stabilization scaling factor s defined by (3.10) takes on the constant value of s^{shell} on entire elements. The transition from $s = 1$ to $s = s^{\text{shell}}$ therefore occurs over a staircase-shaped

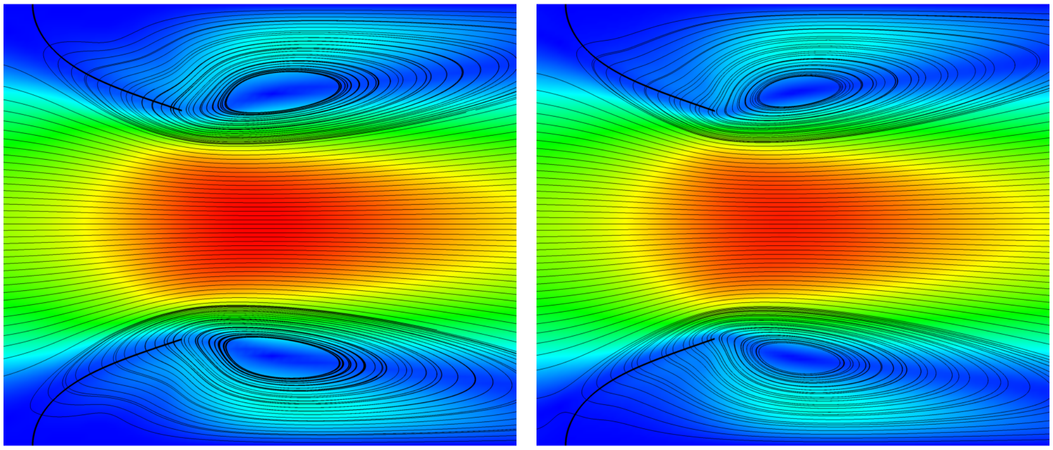


(a) Immersed M1 (b) Immersed M2

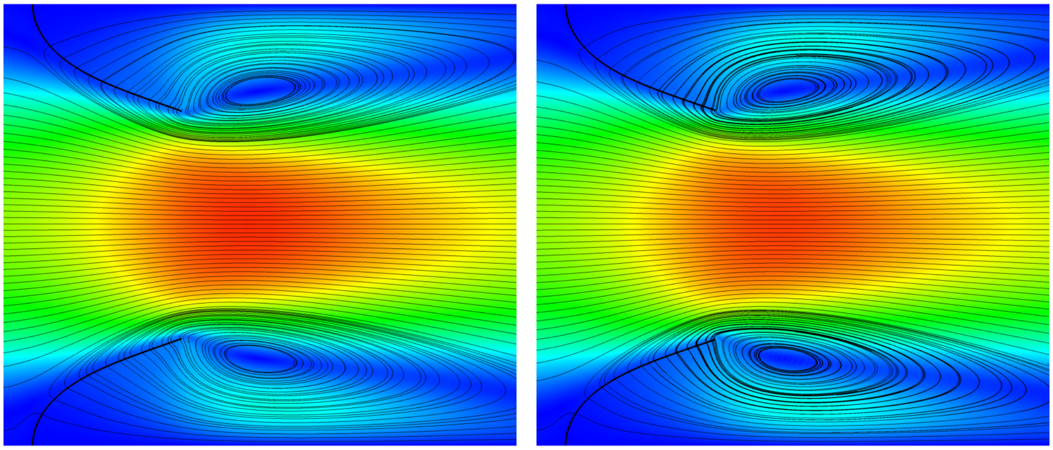


(c) Immersed M3 (d) Body-fitted reference

Figure 6.12: Pressure contours at $t = 0.5$, from immersed boundary computations on M1, M2, and M3, along with the body-fitted reference. Large pointwise pressure errors are confined to an $\mathcal{O}(h)$ neighborhood of the immersed structure, becoming increasingly localized with spatial refinement.



(a) Immersed M1 (b) Immersed M2



(c) Immersed M3 (d) Body-fitted reference

Figure 6.13: Velocity streamlines superimposed on a velocity magnitude contour plot, at $t = 0.5$, from immersogeometric computations on M1, M2, and M3, and the body-fitted reference.

region when the immersed surface is skew to the mesh lines. An alternative definition of s based on distance from the shell structure (but also accounting somehow for background mesh size) might lead to a more aesthetically-pleasing pressure field in such cases, but I have no reason to believe that this would make the solution more accurate. It is important to remember that the "pressure" plotted in Figure 6.12 corresponds to the coarse scale solution variable p^h in the semidiscrete VMS formulation. It omits the fine scale contribution $p' = \tau_{\mathrm{C}} \nabla \cdot \mathbf{u}_1$, which dominates near the beams for $s^{\text{shell}} = 10^6$. The coarse scale pressure solution p^h cannot be interpreted physically as mechanical pressure (i.e. $-\frac{1}{3}\mathrm{tr}\, \boldsymbol{\sigma}_1$) in the band of elements immediately adjacent to the immersed shell structure.

6.3 2D coapting valve

This section examines an extension of the problem studied in the previous section, wherein the valve leaflets are long enough to contact one another. I introduced this problem in [104] to empirically study the consistency–stability trade-off associated with the multiplier stabilization factor r, in a setting that is representative of practical usage. The remainder of this section is adapted from [104]. Throughout this section, the fluid subproblems are discretized using the VMS formulation of Section 3.1.1.

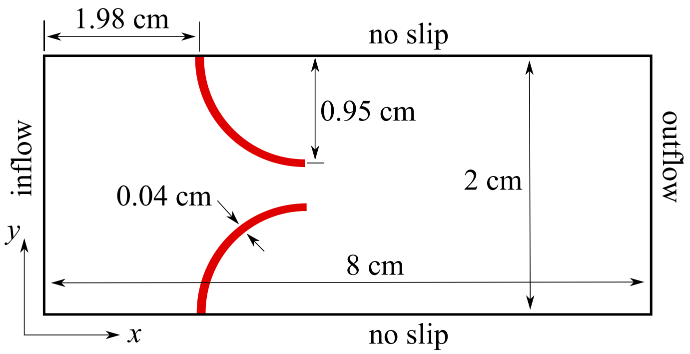


Figure 6.14: The geometry of the 2D valve problem. (Not to scale.)

6.3.1 Problem description

The 2D valve model that I study consists of two cantilevered quarter-circular beams attached to the walls of a 2 cm wide and 8 cm long channel filled with incompressible fluid. Figure 6.14 provides the complete geometry of the problem. The beams are governed by the Kirchhoff–Love shell theory for isotropic St. Venant–Kirchhoff materials described in Section 2.3, which reduces to a beam theory when deformations are constrained to two space dimensions. The fluid has density $\rho_1 = 1$ g/mL and viscosity μ =3 cP. These properties mimic those of human blood [107, 139]. I choose the thickness of the beams to be similar to that of an aortic valve leaflet, $h_{\mathrm{th}} = 0.04$ cm [122], and use a Poisson ratio of $\nu = 0.4$ to approximate incompressibility. Values of the Young's modulus vary between computations and are specified in later sections. The top and bottom of the fluid domain have no-slip and no-penetration boundary conditions. The left side of the fluid domain is nominally the inflow and the right side is nominally the outflow. These distinctions are based on the flow direction in which the valve is biased, but, when the nominal inflow and outflow faces are subjected to Neumann boundary conditions, it is possible that fluid may flow into or out of either of these sections of the boundary. There is little risk of backflow divergence at the low flow speeds considered, though, and the traction boundary stabilization factor γ is set to zero.

6.3.2 Steady flow

I first consider the case of a prescribed parabolic flow profile at the inflow boundary. This profile is given by

$$\mathbf{u}_1|_{\text{inflow}} = \left(15 \text{ ramp}(t) \left(\frac{y}{1 \text{ cm}}\right) \left(2 - \left(\frac{y}{1 \text{ cm}}\right)\right) \text{ cm/s}\right) \mathbf{e}_1 \;, \tag{6.3}$$

where

$$\text{ramp}(t) = \begin{cases} 0 & t < 0 \\ t/(1 \text{ s}) \quad , & 0 \leq t < 0.1 \text{ s} \\ 0.1 \quad , & \text{otherwise} \end{cases} \;. \tag{6.4}$$

This profile is represented exactly using the quadratic B-spline shape functions using the procedure outline in Section 6.2.2. Taking the characteristic length scale to be the width of the channel and the characteristic velocity to be the peak of the inflow profile at $y = 1$ cm and $t > 0.1$ s, the Reynolds number for this flow is 100. I deliberately select it to be much slower than ejection through a heart valve, to ensure that the fluid–structure system approaches a steady solution as $t \to \infty$, rather than developing a time-periodic or turbulent solution. I assign the beams composing the valve a Young's modulus of $E = 10^5$ dyn/cm^2. This value is selected to ensure significant structural deformation in this flow regime.

To study the effect of the stabilization parameter, r, and the mesh size, h, on this problem, I introduce a hierarchy of three fluid meshes, M1, M2, and M3, consisting of 32×128 ($h = 0.0625$ cm), 64×256 ($h = 0.03125$ cm), and 128×512 ($h = 0.015625$ cm) quadratic B-spline elements, respectively. I employ a single discretization of the beams, which divides each of the quarter-circular arcs into 64 quadratic NURBS elements. The NURBS elements are capable of exactly representing the circular geometry at any resolution [134]. As suggested by (4.8) and the low-Reynolds-number branch of (4.9), I assign the penalties τ^B_{NOR} and τ^B_{TAN} according to the formula

$$\tau^B_{\text{NOR}} = \tau^B_{\text{TAN}} = C\mu/h \ , \tag{6.5}$$

with $C = 10^2$. To ensure that the time step has a fixed, moderate proportionality to the element advective time scale (see [16, page 181]), I assign time step sizes using the formula

$$\Delta t = \frac{h}{6.25 \text{ cm/s}} \ . \tag{6.6}$$

I compute solutions for M1–M3 using the semi-implicit time integration scheme described in Section 4.2, with $r = 1$, $r = 0.1$, and $r = 0$. The generalized-α parameters in the time integration algorithm are determined by $\rho_\infty = 0.5$.

The $L^2(\Gamma_t)$-norm of λ is shown as a function of time for each case in Figures 6.15–6.17. One can see that both refinement and increased r reduce the norm at which λ reaches a steady value. While it might seem that multiplier fields differing greatly in L^2 norm would produce wildly different fluid velocity fields and/or structure deformation solutions, this is not the case. Figure 6.18 compares the structure deformation and velocity fields for $r = 1$, $r = 0.1$, and $r = 0$ at time $t = 10$ s on M1–M3. The corresponding Lagrange multiplier force fields on the structure are plotted in Figure 6.19. The difference in L^2 norm between the multiplier fields is mainly due to oscillatory modes that have very little influence on the steady-state fluid and structure solutions. As indicated by the analysis of model problems in Chapter 5, though, these modes carry potential energy, and should not be allowed to grow *ad infinitum*, as they appear to in the case of $r = 0$. While the accumulation of excess stored energy in the multiplier field has clear disadvantages, selecting r large enough to completely eliminate oscillations may result in unacceptable leakage through structures, as demonstrated in the following section. For a fixed value of r, both leakage and oscillation may be reduced through refinement, but, in practice, I recommend tuning r to obtain the best quality solution within the constraints of available time and computational resources.

The fact that $\|\lambda\|_{L^2}$ during the transient stage of the computations with $r = 0$ increases with refinement does not indicate that refinement harms the stability of the semi-implicit scheme. In fact, as approximations converge toward the exact solution, one cannot expect the $L^2(\Gamma_t)$ norm of the pressure jump to converge; for viscous incompressible flow around the edge of a thin plate, the pressure distribution on the plate is not necessarily square-integrable; we expect it, in general, to be in $H^{-1/2}(\Gamma_t)$, which is larger than

$L^2(\Gamma_t)$. A famous example of a thin plate pressure distribution less regular than L^2 is Hasimoto's solution of Stokes flow through an aperture [75, (4.8)]. The energy analysis of the regularized problems in Chapter 5 suggests that the L^2 norm of λ is a natural choice for investigating semi-discrete stability, but it should not be used in assessing convergence with spatial refinement, since the relevant energy norm (5.10) does not control λ uniformly under refinement. One can see from Figure 6.17 the rate of growth of oscillations in the steady state decreases with refinement, indicating greater stability on finer meshes.

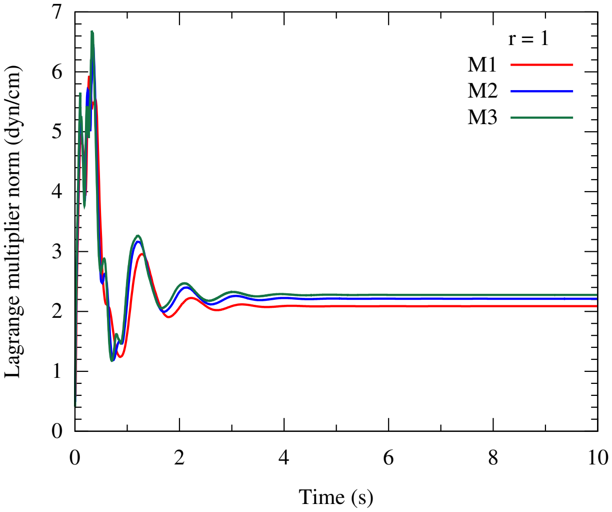


Figure 6.15: The L^2 norms of the Lagrange multiplier field as functions of time for steady flow through the 2D valve, with $r = 1$.

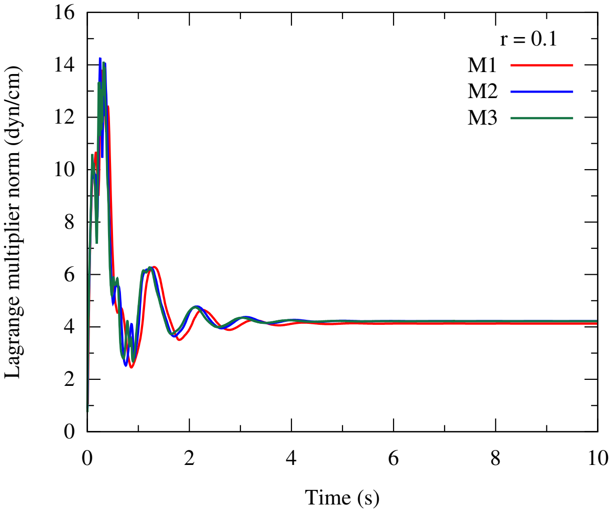


Figure 6.16: The L^2 norms of the Lagrange multiplier field as functions of time for steady flow through the 2D valve, with $r = 0.1$. Note the difference in scale from Figure 6.15.

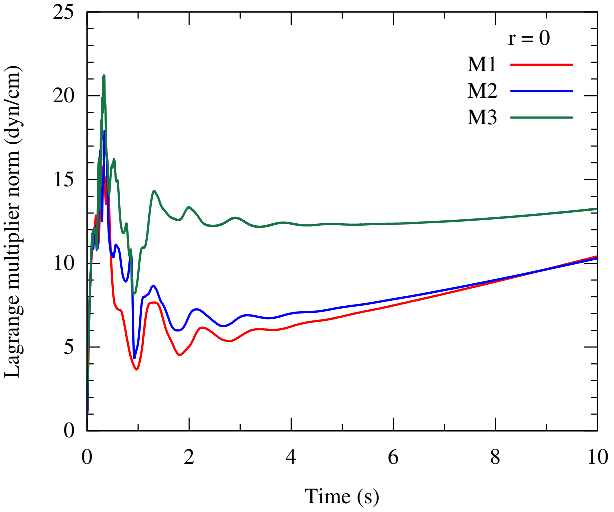


Figure 6.17: The L^2 norms of the Lagrange multiplier field as functions of time for steady flow through the 2D valve, with $r = 0$, i.e. no stabilization. Note that they do not reach steady values in this case.

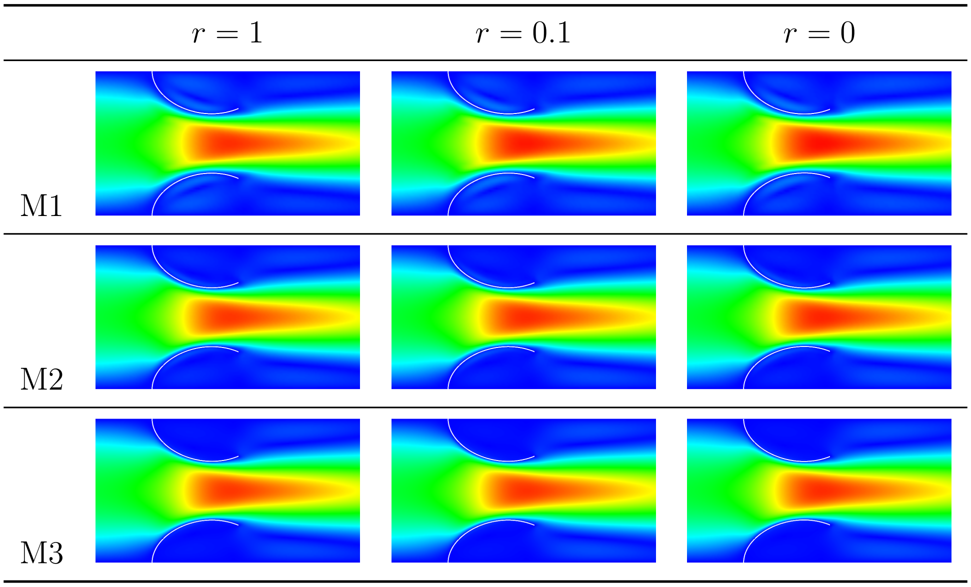


Figure 6.18: Comparisons of structure deformation and fluid velocity magnitude fields at time $t = 10$ s, computed on M3, with different stabilization parameters. Color scale: 0 (blue) to 3.4 cm/s (red).

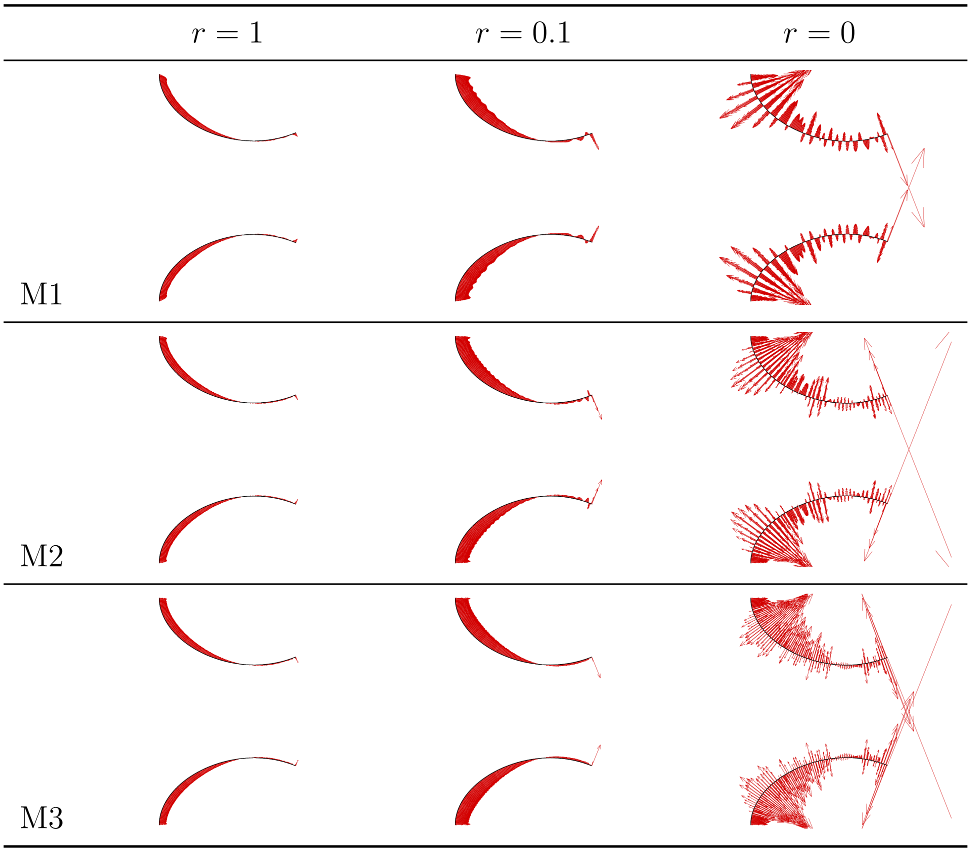


Figure 6.19: The forces on the structure due to the Lagrange multiplier field in the solutions from Figure 6.18.

6.3.3 Convergence to a hydrostatic solution during closure

I now consider the case of the 2D valve's closure when subjected to Neumann boundary conditions. Because a 2D model cannot accurately represent the mechanics of a 3D shell structure, I must make the beams significantly stiffer than the soft tissues appearing in native or bioprosthetic heart valves, to prevent the 2D valve from prolapsing when subjected to realistic pressure differences. I therefore use a Young's modulus of $E = 7 \times 10^9$ dyn/cm^2 in this example. Contact between the leaflets is an essential aspect of valve closure. For the computations in this section, I use parameters $k = 10^8$ dyn/cm^3, $c = 0.1$ cm, and $h_\mathrm{c} = 0.01$ cm in the penalty method described in Section 3.2.3. At the inflow, I apply a zero-traction boundary condition and, at the outflow, I apply the time-dependent traction $-P_\mathrm{out}(t)\mathbf{e}_1$, with

$$P_\mathrm{out}(t) = \left\{ \begin{array}{lcr} 0 & , & t \leq 0 \\ 1000\left(\frac{t}{1\text{ s}}\right) & , & 0 < t < 0.1\text{ s} \\ 100 & , & \text{otherwise} \end{array} \right\} \text{ mmHg}. \tag{6.7}$$

This approximates the pressure difference across a closed heart valve in the physiological setting [201]. Recall from the blocked plug flow model of Section 4.4.4 that, as the system approaches a steady solution, some leakage is expected for $r > 0$ that, as $r \to 0$, will be proportional to r and inversely proportional to τ^B_NOR. This is complicated somewhat in practice when using an equal-order VMS discretization of the fluid subproblem, because constraint violation is not the only source of apparent leakage in non-div-conforming discretizations: as I demonstrate in Section 6.4, approximation error in the pressure can lead to violation of the continuity equation, which leads, in turn, to "leakage" through the valve, even when the fluid and structure velocities match. For the computations in this section, I set $s^{\text{shell}} = 10^8$ to diminish this source of leakage such that it is small enough to clearly distinguish the effect of r.

I employ the fluid mesh M1 and structure meshes defined in the previous section. I reduce the time step by a factor of 10 to improve nonlinear convergence and increase penalties by a factor of 1000 to more strongly enforce the no-penetration condition. For the computations of this section,

$$\Delta t = \frac{h}{62.5 \text{ cm/s}} \qquad \text{and} \qquad \tau^B_{\text{NOR}} = \tau^B_{\text{TAN}} = C\mu/h \ , \tag{6.8}$$

with $C = 10^5$. I compute solutions with $r = \infty$ (velocity penalization), $r = 1$, $r = 0.1$, and $r = 0$. Figure 6.20 shows the volumetric flow through the outflow face as a function of time for each value of r. Approximate asymptotic values of the steady-state flow rate are given in Table 6.1, along with the crude estimates based on Eq. (4.40) from the plug flow model of Section 4.4.4. The plug flow model consistently underestimates leakage, due to the imperfect local mass conservation mentioned above. The L^2 norms of λ for each finite value of r are given as functions of time in Figure 6.21. For $r = 0$, the asymptotic value[1] of $\|\lambda\|_{L^2}$ approximately matches the rough estimate of $\sqrt{A(\Delta P)^2}$ for A equal to the surface area of the beams (with unit depth) and $\Delta P = 100$ mmHg. Figure 6.22 compares the solutions at time $t = 0.35$ s for $r = 1$ and $r = 0$. The structure deformations and fluid pressure fields are nearly indistinguishable,

---

[1] If the simulation is continued for a much longer period, it becomes clear that $\|\lambda\|^2_{L^2}$ grows, but at a much slower rate (relative to $|\lambda|$) than in the case of the open valve; on the time scale of ~ 0.5 s, it effectively flatlines.

but the solution with $r = 1$ has a more stable multiplier field at the expense of spurious flow in what should be a hydrostatic solution.

Remark 6.2. The skeptical reader may suspect that the oscillation in volumetric flow rate is a spurious phenomenon, due to the damped spring interpretation of the multiplier forcing. Such issues are dwelt on at length in the existing literature on feedback force boundary conditions (as reviewed in Section 4.4.3). However, the characteristic frequency of the fluid mass attached to the multiplier "spring" is much higher and the oscillation's presence in the penalty solution with $r = \infty$ rules out this hypothesis altogether. The oscillation is the reverberation of the water hammer on the closed elastic valve. I observed a similar effect in our simulations of a 3D valve and provided an electronic–hydraulic analogy [196] to the familiar transient response of a series RLC circuit [105, Figure 29].

Remark 6.3. Notice in Figure 6.22 that, for $r = 1$, large vortices form on the concave sides of the leaflets. These are features of the steady solution (which should be hydrostatic). A closer examination of the flow field shows that the upper vortex is counterclockwise and the lower is clockwise. These vortices are fed by flow between themselves and the walls, which is permitted by leakage through the portions of the leaflets closer to the walls. The velocity magnitude in the vortices is significantly larger than that of the flow through the leaflets, which underscores the importance of preventing unphysical leakage to obtaining reasonable solutions.

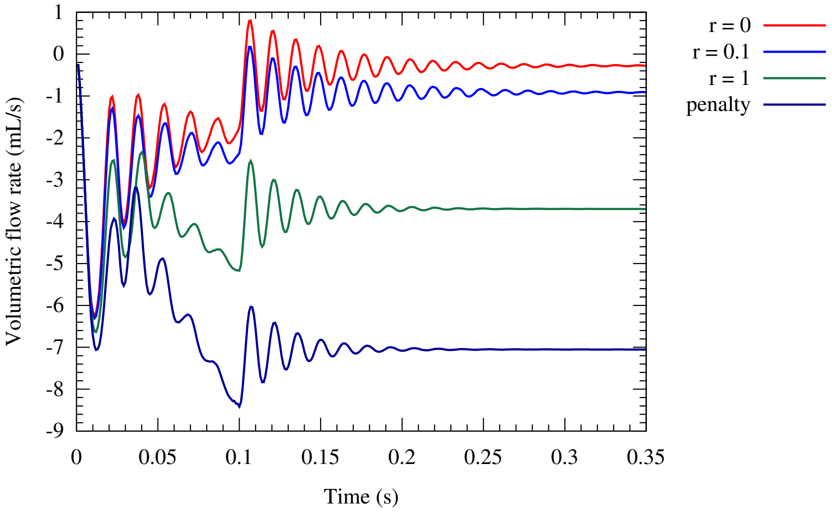


Figure 6.20: Volumetric flow rate through the closing valve as a function of time, for different values of r.

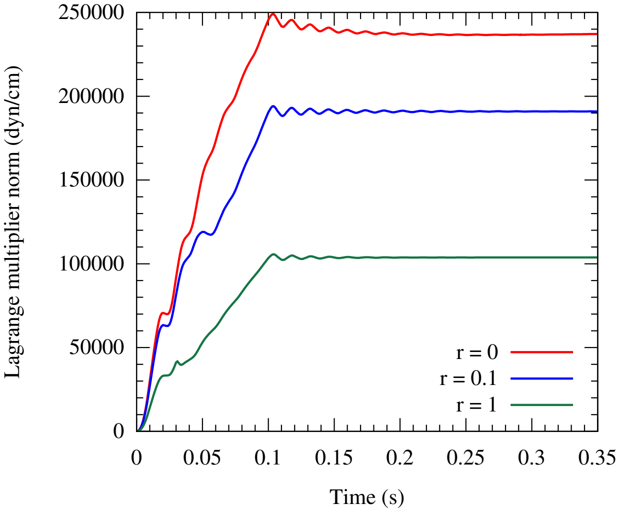


Figure 6.21: $L^2(\Gamma_t)$ norm of λ on closing valve as a function of time, for finite values of r. (For $r = \infty$, it will be zero for all t.)

r	computed leakage (mL/s)	plug flow estimate (mL/s)
∞	-7.1	-5.6
1	-3.7	-2.8
0.1	-0.9	-0.5
0	-0.3	0

Table 6.1: Steady volumetric flux (assuming 1 cm depth) through the closed 2D valve for different values of r, alongside the values estimated from the plug flow model of Section 4.4.4, which neglects leakage due to spurious volume loss.

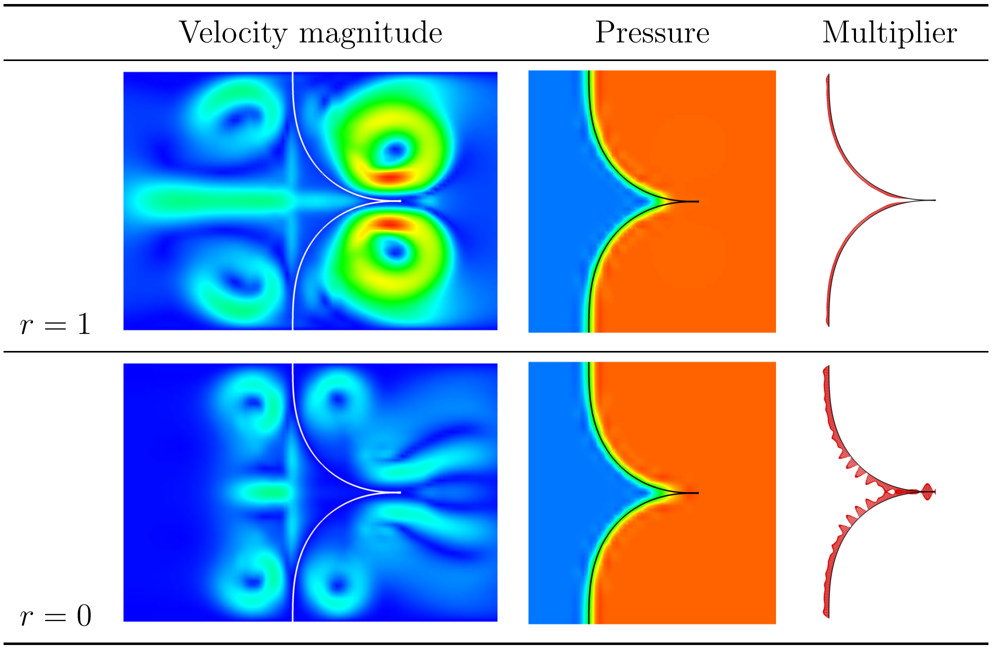


Figure 6.22: Velocity, pressure, and multiplier solutions at $t = 0.35$ s for the closed valve with $r = 1$ and $r = 0$. See Remark 6.3 for discussion on the velocity fields. Velocity color scale: 0 (blue) to 24 cm/s (red). Pressure color scale: -20000 dyn/cm^2 (blue) to 150000 dyn/cm^2 (red).

6.4 Blocked tube

This section tests the extrapolation of the analysis of Section 5.4 to the nonlinear VMS formulation of unsteady Navier–Stokes. I originally ran this experiment for [105], and have adapted [105, Section 4.4] in what follows.

6.4.1 A demonstration of the effect of pressure approximation error

Consider a simplified model of a closed valve, with fluid properties and boundary conditions similar to those found in cardiovascular applications. The fluid domain Ω_1 is an axis-aligned 2 cm × 2 cm × 2 cm cube, filled with an incompressible Newtonian fluid of density $\rho_1 = 1.0$ g/cm^3 and viscosity $\mu = 3.0 \times 10^{-2}$ g/(cm s). The vertical faces have a no-slip boundary condition, the bottom has a zero-traction outflow boundary condition, and the top has a pressure traction of 120 mmHg. The length scale, fluid properties, and pressure difference produce conditions comparable to those surrounding a closed aortic valve in diastole. The traction boundary stabilization constant is $\gamma = 0.5$. Now consider immersing a rigid, impermeable horizontal plate into this cube, blocking its entire cross section at a distance of 1.1 cm from the bottom. The exact solution for this problem should be hydrostatic, with a discontinuous pressure at the location of the plate. However, in an immersogeometric discretization, the continuity of the pressure approximation functions through the plate means that the discontinuity of the exact solution cannot be reproduced in a computation.

Remark 6.4. The plate's height of 1.1 cm is deliberately selected so that the plate will never coincide with an element boundary for any uniform division of the cube into 2^n elements in the z-direction. This may be seen by considering the fact that 0.1_{10} is a repeating fraction in binary. Even if a discontinuous pressure basis is used, the discontinuities will not be located on the structure.

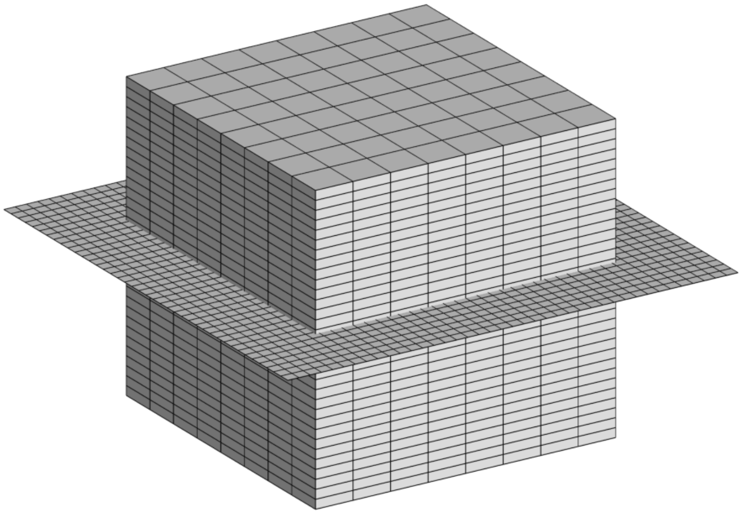


Figure 6.23: The computational mesh used for the closed-valve model problem.

I now compute a solution to this problem, starting from homogeneous initial conditions for the velocity and using Lagrange multipliers to enforce the no-penetration condition on the shell. The mesh is a trivariate C^1-continuous quadratic B-spline patch, uniformly refined into $8 \times 8 \times 32$ elements. The quadrature rule for surface integrals over the immersed plate is a sum of Gaussian quadrature rules on 40×40 quadrilaterals, evenly dividing a 3 cm × 3 cm square surface, cutting through the channel as shown in Figure 6.23. Surface quadrature points falling outside of the channel do not contribute to integrals.

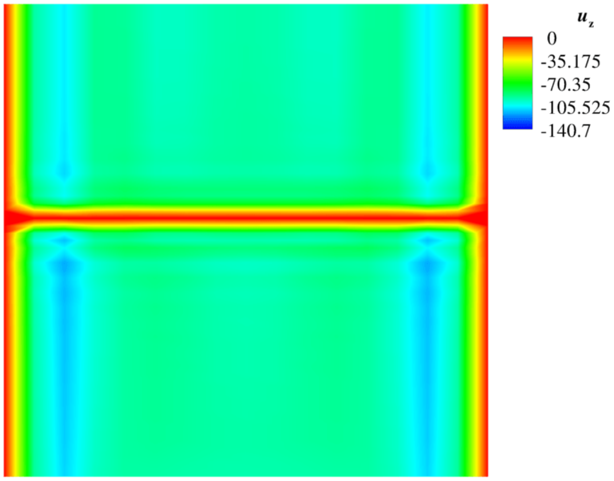


Figure 6.24: The z-component of velocity, in cm/s, for a highly unphysical steady-state flow solution through a blocked channel, as computed with $\Delta t = 10^{-4}$ s and no modifications to fluid stabilization terms. The fluid spuriously compresses to meet the velocity constraint imposed by the barrier while maintaining a large downward flow through the channel.

Computing with the time step $\Delta t = 10^{-4}$ s and converging the fluid–structure kinematic constraint implicitly using the augmented Lagrangian predictor–multi-corrector iteration suggested in Section 4.2 (which converges for $r = 0$ in this example), I obtain a highly unphysical behavior. Figure 6.24 shows the vertical velocity component on a slice of the resulting solution, after the volumetric flow rate through the top of the cube reached a steady value ($t > 0.01$ s). While the Lagrange multipliers enforce the constraint very effectively, there is still a significant flow through the top face of the cube. The steady-state volumetric flow rate is 355.2 mL/s, which is unacceptable for simulation of a valve structure that exists primarily to block flow. This would be a typical flow rate through an *open* aortic valve, during systole [186]. The flow rate varies between cross-sections of the channel, which obviously violates the incompressibility condition. The compression caused by local pressure approximation error pollutes the entire velocity solution.

6.4.2 Effect of s^{shell}

The pressure gradient is approximated especially poorly in a neighborhood of the immersed surface. It appears in the fine scale velocity of the VMS formulation, where it is scaled by τ_{M}. Locally reducing the value of τ_{M} diminishes the influence of this poorly-approximated quantity. Due to the inverse relationship between τ_{M} and τ_{C}, this will also increase the penalization of volume loss in a neighborhood of the immersed surface. Recalling the definition of the field $s(\mathbf{x})$ in Section 3.1.1, it is clear that choosing $s^{\text{shell}} > 1$ will have the desired effect of locally reducing τ_{M}. The effect of such a local scaling of stabilization parameters on an analogous linear model problem is analyzed in Section 5.4.

I now investigate the effect of s^{shell} empirically. Table 6.2 compares $s^{\text{shell}} = 1$, $s^{\text{shell}} = 10^4$, and $s^{\text{shell}} = 10^8$, showing that volumetric flow scales roughly like $\left(s^{\text{shell}}\right)^{-1/2}$, tending to zero as s^{shell} increases. An undesirable consequence of increasing s^{shell} is that the weakened stabilization near the im-

s^{shell}	Volumetric flow rate
1	355.2 mL/s
10^4	4.037 mL/s
10^8	4.048×10^{-2} mL/s

Table 6.2: The effect of s^{shell} on apparent leakage due to volume loss.

mersed surface harms the conditioning of the discrete problem. Due to the simplistic nature of the blocked tube model problem, conditioning is not a significant issue, but applying the modified stabilization terms to more complex calculations, such as BHV simulations, increases the cost of sufficient iterative solution of the linear problems to be solved in the nonlinear iteration of each time step. The development of a suitable preconditioner may avert this difficultly, but is beyond the scope of the current dissertation.

Remark 6.5. The observed scaling of leakage with respect to s^{shell} differs from the rough estimate one would get from the analysis of Section 5.4 (by assuming velocity is constant over cross-sections and applying a trace inequality). It appears, from the numerical experiment of the present section, that s^{shell} reduces leakage more quickly than anticipated. However, the empirically-observed scaling can be derived analytically in a simpler 1D model problem. The 1D analysis relies on bounding the velocity's gradient in terms of its divergence, though, which does not generalize to multiple space dimensions. However, it seems that when most of the pressure variation occurs along one space dimension, the 1D model problem provides a reasonable estimate.

6.5 Benchmark testing with div-conforming B-splines

To verify that the nonlinear fluid–thin structure interaction methodology proposed in this dissertation can practically compute accurate solutions with div-conforming B-spline discretizations of the fluid subproblem, I again use the 2D benchmark problem defined and studied using equal-order VMS discretizations of the fluid in Section 6.2. As in Section 6.2, I look at the displacement history of a material point on the structure and compare immersogeometric and boundary-fitted results. The boundary fitted reference computation is the same one described in Section 6.2.2. The content of this section is adapted from [106].

6.5.1 Div-conforming immersogeometric discretizations

To demonstrate the convergence of div-conforming immersogeometric discretizations toward the boundary-fitted reference solution, I present results from a sequence of three immersogeometric discretizations. Although the problem domain is rectangular and I could simply employ the B-spline parameter space as physical space, I demonstrate convergence with distorted fluid meshes by deforming the interior of the parametric domain while mapping it to the physical domain. For all of the div-conforming immersogeometric discretizations, the fluid domain is discretized using a B-spline patch with the knot space $\widehat{\Omega}_1 = [0,1] \times [0,1]$. A point $\mathbf{X}$ in this knot space is (in the notation of Section 3.1.2.2) mapped to the physical domain Ω_1 with the mapping

$$\phi_1 = L X_1 \ , \tag{6.9}$$

$$\phi_2 = W \left(X_2 + \frac{s}{4} \left(1 - \cos\left(2\pi N X_1\right)\right) \left(X_2^2 - X_2\right) \right) \ , \tag{6.10}$$

where $L = 8$, $W = 1.61$, $N = 5$, and $s = 1.5$. For the coarsest mesh, M1, the B-spline knot space is subdivided into 32×128 Bézier elements and div-conforming B-spline velocity and pressure spaces of degree $k' = 1$ are defined on this mesh. The meshes, M2 and M3, use 64×256 and 128×512

elements respectively. The knot lines for M1 are drawn on the depiction of Ω_1 in Figure 6.25 to indicate the mesh distortion. Because resolution in the structure problem is far from being a limiting factor in accuracy, I use the same structure mesh in both problems, dividing each beam into 128 quadratic B-spline elements.

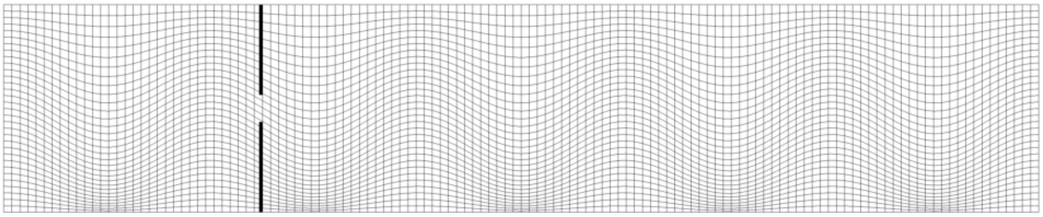


Figure 6.25: The physical image of the B-spline parameter space, showing the mesh of unique knots (thin lines) for M1 in relation to the beams (thick lines).

Normal-direction Dirichlet boundary conditions are enforced strongly, while tangential boundary conditions are enforced using the penalty method. The x_2 component of velocity at the inflow can easily be set to a nodal interpolant of the x_2 component of (6.2) when $k' = 1$, because the mapping $\boldsymbol{\phi}$ and the corresponding velocity push-forward involve only scaling by a constant factor at the inflow face of the domain. For computations on mesh $\text{M}(N+1)$, the penalty parameters are $\tau^B_{\text{NOR}} = \tau^B_{\text{TAN}} = C_{\text{no slip}} = 1000 \times 2^N$. The temporal discretization uses the backward Euler method with $\Delta t = 1.0 \times 10^{-2} \times 2^{-N}$. Six block iterations are used to couple the fluid and structure implicitly, reusing the fluid tangent from the first iteration. For this simple 2D problem, I solve for fluid increments in the block iteration using a direct solver, namely the MUltifrontal Massively Parallel sparse direct Solver (MUMPS) [1], accessed via PETSc.

6.5.2 Comparison of results

Figures 6.26 and 6.27 compare the x- and y-direction displacement histories of the tip of the upper beam in the three immersogeometric computations and the body-fitted reference described in Section 6.2.2. Refinement of the immersogeometric discretizations clearly brings this quantity of interest closer to the boundary-fitted reference curve. As in Section 6.2.4, the pressure space still struggles to approximate the discontinuous exact solution, with the discrete solution exhibiting over- and under-shoot phenomena to either side of the immersed structure, as shown on M2 in Figure 6.28. However, in the context of immersed fluid–thin structure interaction, using div-conforming B-splines and the fluid–structure coupling method described above, the quality of the pressure solution is not especially important. The pressure is not involved in the computation of fluid–structure coupling forces and discrete fluid velocities computed using div-conforming methods are immune to pressure interpolation errors [54, (6.32)]. This is perhaps a counter-intuitive statement, because it stands in stark contrast to experience with most other numerical methods for incompressible flow, in which pressure interpolation error enters into *a priori* bounds on the velocity error. (Recall Section 5.4, in which I go into more detail on exactly how this effect emerges in the stabilized finite element methods.) The robustness of div-conforming B-spline discretizations to pressure jumps is illustrated in practical setting in Section 7.2. As expected

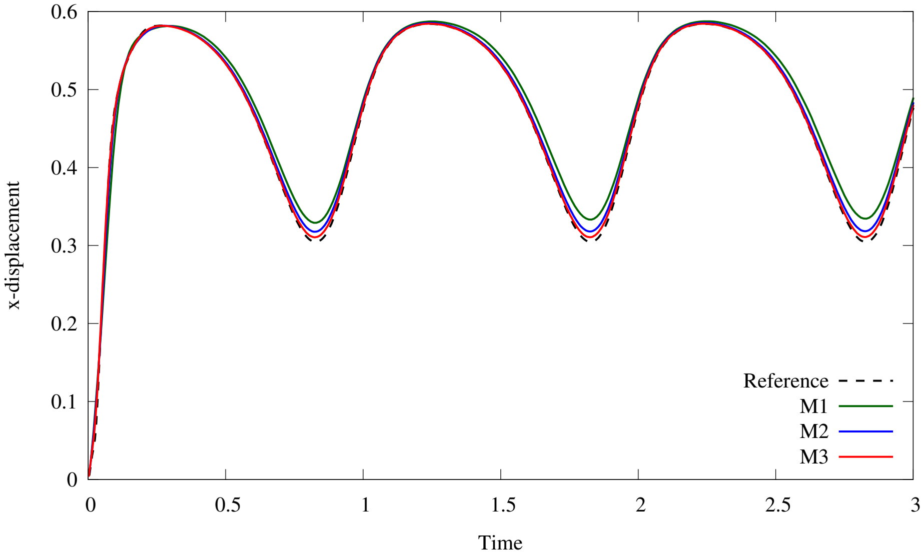


Figure 6.26: The x-direction displacement of the tip of the upper beam.

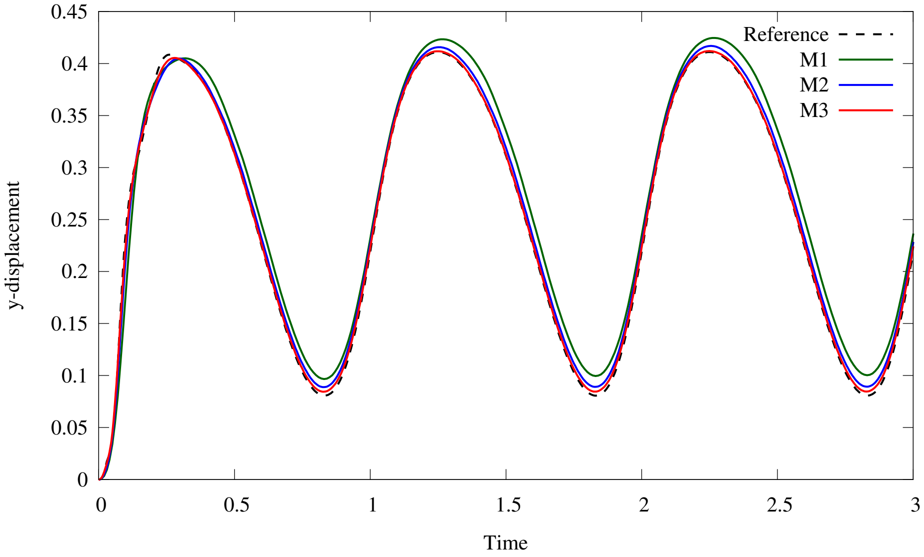


Figure 6.27: The y-direction displacement of the tip of the upper beam.

from the theory, poor pressure approximation does not appear to have any ill effect on the velocity field, which remains smooth in Figure 6.28. It is also noteworthy that there is no visible asymmetry induced in the velocity solution by the asymmetric mesh distortion. One might expect "problems" on account of the lack of momentum conservation of the div-conforming discretization on mapped domains. This can be seen easily by observing that $\mathbf{w}_1 \equiv \mathbf{e}_i$ is not in the pushed-forward velocity test space. (Recall that the push-forward applied to velocity test and trial space basis functions rotates them, so they point along the mesh lines rather than the x_1 and x_2 axes.) However, I have not noticed any artifacts attributable to spurious momentum generation on deformed meshes.

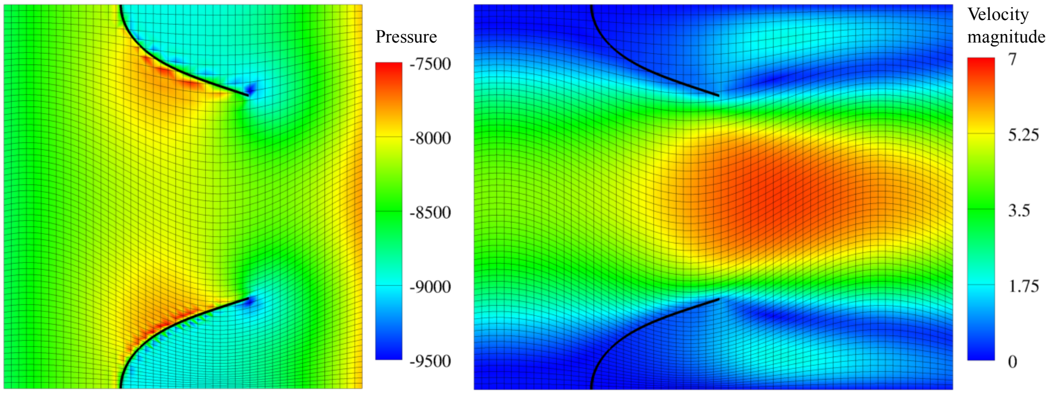


Figure 6.28: The pressure field (left) and the the velocity magnitude (right) at time $t = 0.5$ on M2.

6.5.3 Block iterative convergence

I also use this problem to observe the behavior of block iteration "in the wild", on a nonlinear FSI problem. I restart the computation on M1 from the 50th time step and look at the ℓ^2 norm of the fluid subproblem residual vector (R_f, in the notation of Section 4.3) as a function of the number of block iterations. To illustrate the effects of the fluid–structure coupling penalties τ^B_NOR and τ^B_TAN and the time step, I restart with smaller and larger values of these parameters. The results are shown in Figure 6.29. The case with 10× larger time step requires the tangent matrix A_f to be recomputed every iteration. For the other cases, A_f is assembled on the first iteration only, then reused in subsequent iterations. Comparing Figure 6.29 with the analysis of Section 5.3 and Figures 5.8 and 5.9, it is clear that the overall conclusions from the linear model for block iteration carry over to the nonlinear case: increasing the penalty parameters and/or time step causes convergence to slow down while decreasing the penalty parameters and/or time step causes convergence to speed up.

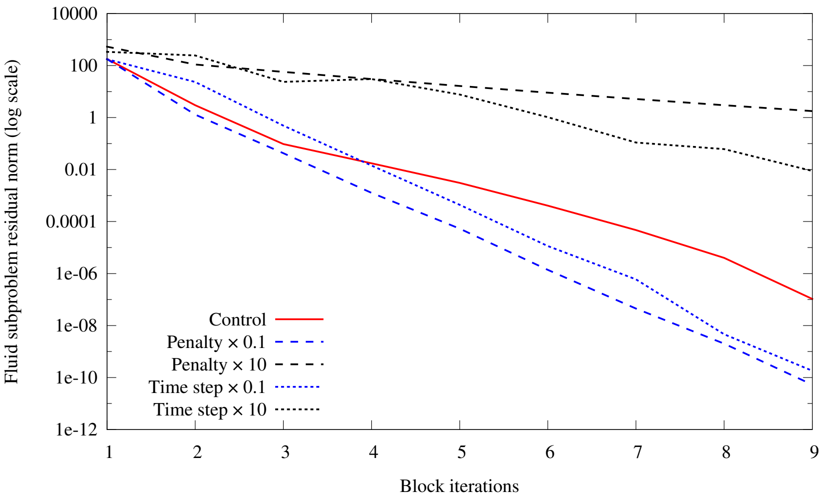


Figure 6.29: Convergence of block iteration in the 51st time step, subject to different perturbations of the parameters used in the computation on M1 (the “control” case).

Chapter 7

Application to BHV FSI analysis

A combination of the technologies described in the previous chapters can produce qualitatively-reasonable simulations of BHV FSI, without any of the usual modifications of the problem statement to make it more tractable, such as rendering the fluid compressible [35, 36, 169, 199], reducing the Reynolds number [42–44, 131, 189–191], introducing inaccurate symmetry assumptions [42–44, 189–191], prescribing the leaflet motion [173], not modeling the structure as a continuum [29, 70], or not applying physiological pressure levels to the closed valve [29, 62, 64]. I discuss some valve simulations using the analysis methods from this dissertation in Sections 7.1 and 7.2. Section 7.3 then describes an initial effort toward validating the mathematical model for BHV FSI put forward in Chapter 2.

Some of this chapter's content is derived from the following paper:

D. Kamensky, M.-C. Hsu, Y. Yu, J. A. Evans, M. S. Sacks, T. J. R. Hughes. Immersogeometric cardiovascular fluid–structure interaction analysis using divergence-conforming B-splines. *Computer Methods in Applied Mechanics and Engineering*, In review (preprint: ICES Report 16-14). D. Kamensky implemented the numerical methods, formulated and analyzed model problems, and participated in the experimental work. M.-C. Hsu provided supervision and edited the manuscript extensively. Y. Yu, J. A. Evans, and T. J. R. Hughes supervised the mathematical analysis. M. S. Sacks helped plan and supervise the laboratory experiments.

The experiment described in Section 7.3.1 of this chapter was conducted with the assistance of current and former lab members John G. Lesicko, Jordan L. Graves, Hugo Landaverde, Javier Solis, Mitchell A. Katona, Samuel Petter, and Bruno Rego.

7.1 Overview of BHV simulations

I will first review several published valve simulations that use the semi-implicit augmented Lagrangian technique developed in this dissertation. All of the computations reviewed in this section use the modified VMS discretization of the fluid subproblem described in Section 3.1.1. Some of them incorporate phenomena that are beyond the scope of the mathematical problem stated in Chapter 2, such as deforming arteries modeled by elastic solids, alternative shell structure formulations, and constitutive models not covered in this dissertation. Further, these simulations include contributions from many collaborators, as reflected by the author lists of the corresponding publications. However, these BHV simulations illustrate the versatility and practical effectiveness of the numerical techniques developed in this dissertation, so I will summarize the results while providing citations to relevant references for additional information.

I introduced the initial variant of the semi-implicit augmented Lagrangian technique (in which the stabilization parameter r is zero; cf. Section 4.1) in [105], along with the adjustments to the VMS formulation and the contact penalty needed to effectively simulate a BHV. A rather crude model of a BHV immersed in a rigid artery was used to illustrate the effectiveness of the technique, although the application of an unrealistic pinned boundary condition on the attached edges of the valve leaflets led to qualitatively incorrect deformations of the leaflets. Further, the rigid artery and resistance outflow boundary condition provided no hydraulic compliance, causing the abnormal flow rate history shown in [105, Figure 28] to emerge in response to a cyclic ventricular pumping pressure. Some snapshots of the valve deformations and velocity magnitude fields from this computation are rendered in Figure 7.1.

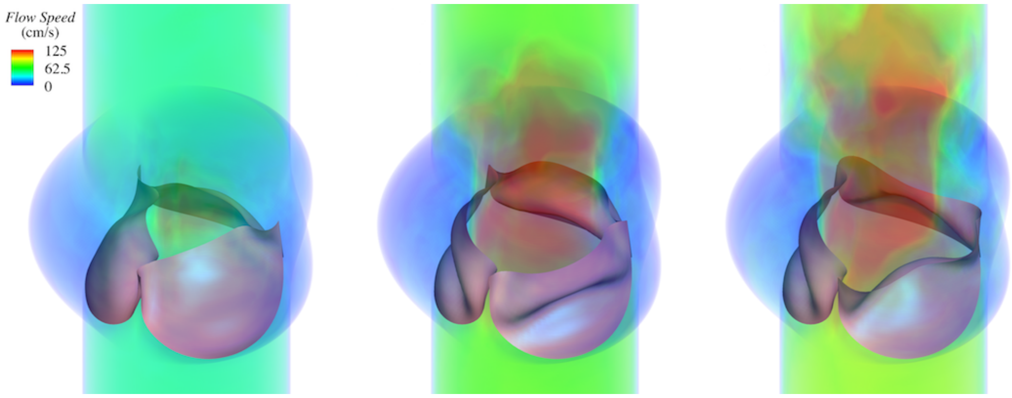


Figure 7.1: Snapshots of the valve FSI computation from [105], showing valve deformations and volume renderings of fluid velocity magnitude.

The BHV model of [105] was augmented with some hydraulic compliance in a follow-up publication [84], by modeling the artery wall as an elastic solid. Unlike the immersed valve leaflets, the fluid–artery interface was discretized in a boundary-fitted manner, and the mesh of the arterial lumen was allowed to deform. This deformation of the fluid mesh enters into the fluid subproblem of Chapter 2 as $\hat{\mathbf{u}} \neq \mathbf{0}$. (The solid elastic artery is outside the scope of the problem stated in Chapter 2.) The model with an elastic artery led to more realistic flow rates than the model with a rigid artery. The results of [84] highlight the fact that the discrete formulation detailed in Chapters 3 and 4 does not rely on a Cartesian, structured, or even stationary grid covering the fluid domain, in contrast to many immersed boundary methods in the literature. Reference [84] also illustrates how immersed-boundary and boundary-fitted approaches can be combined, to take advantage of the efficiency of fitted discretizations for boundaries that deform only mildly.

Reference [85] realized the potential of immersogeometric FSI analysis to streamline the design-through-analysis process for biomedical FSI systems. The parametric design-through-analysis framework [86] developed by the group of M.-C. Hsu at Iowa State University was used to generate an analysis-suitable T-spline [17] model of a BHV. The techniques developed in this dissertation then allowed for the BHV design geometry to be directly immersed into an unfitted discretization of an artery and lumen. The BHV model incorporated a realistic stent geometry, clamped boundary conditions representative of typical industrial BHVs (cf. patent illustrations in [95]), and a soft tissue constitutive model, which enhanced realism relative to the models of [105] and [84]. A snapshot of the resulting BHV FSI simulation is shown in Figure 7.2. The results of [85] emphasize how the computational FSI techniques that I have developed in this dissertation facilitate straightforward connection of design and analysis without tedious and labor-intensive conversion of design geometries into computational models.

7.2 Div-conforming BHV simulation

The remainder of this chapter focuses on BHV simulations using div-conforming B-splines to discretize the fluid subproblem. This technology has, so far, been less thoroughly studied and documented by my collaborators and I.

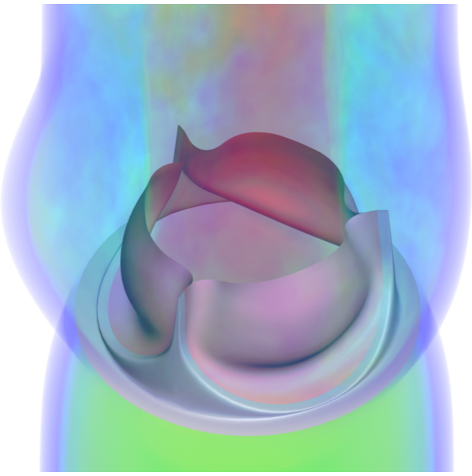


Figure 7.2: Snapshot of the valve FSI computation from [85], showing valve deformation and volume rendering of fluid velocity magnitude. Clamped leaflet boundary conditions, an elastic artery, and unstructured T-spline geometry modeling of the valve enhance realism relative to results from [105], shown in Figure 7.1.

A capability that is not verified by the div-conforming FSI benchmark testing in Section 6.5 is the effective simulation of closing heart valves. Recall that my motivation for using div-conforming B-splines for the fluid discretization is to avoid the *ad hoc* scaling of stabilization parameters that is needed to improve mass conservation in VMS discretizations. In principle, div-conforming B-splines should prevent mass loss altogether, but, in practice, for 3D problems, one generally does not solve the discrete algebraic problem exactly, as would be required for (3.11)–(3.14) to remain valid. This section demonstrates the feasibility of using div-conforming B-splines as a fluid discretization for BHV FSI with inexact iterative solution of the discrete problem. The content of this section is adapted from [106].

7.2.1 Test problem definition

The test problem that I use to illustrate the div-conforming discretization's potential for practical valve simulation is as follows. A variant of the BHV geometry constructed in [105, Section 5.1] is immersed in a cylindrical fluid domain of radius 1.25 cm and height 3 cm. Rigid extensions are added to the leaflets, as in [105], to block flow passing around the attached boundaries of the leaflets. The fluid subproblem posed on the cylindrical domain has traction boundary conditions on the ends and no-slip and no-penetration boundary conditions on the sides. The bottom face of the cylinder is subject to a time-dependent traction $\mathbf{h}_1 = P(t)\mathbf{e}_3$, where $P(t)$ is given by

$$P(t) = \begin{cases} P_1 & t < T_1 \\ at + b & T_1 \leq t \leq T_2 \\ P_2 & t > T_2 \end{cases} . \tag{7.1}$$

$P_1 = 2 \times 10^4$ dyn/cm^2 is the opening pressure applied before $T_1 = 0.05$ s, $P_2 = -10^5$ dyn/cm^2 is the closing pressure applied after $T_2 = 0.1$ s, and $a = (P_2 - P_1)/(T_2 - T_1)$ and $b = P_1 - aT_1$ are selected to continuously interpolate between the two states. The top face is subject to the traction boundary condition $\mathbf{h}_1 = \mathbf{0}$. The traction boundary stabilization scaling factor is set to $\gamma = 1$ on both sides. The properties of the fluid are $\rho_1 = 1$ g/cm^3 and $\mu = 4$ cP. The valve is modeled as an incompressible neo-Hookean material with shear modulus $\mu_s = 600$ kPa and density $\rho_2 = 1$ g/cm^3. The shell thickness is $h_{\text{th}} = 0.04$ cm. The attached edges of the valve leaflets are subject to a clamped boundary condition. The fluid and structure are initially at rest at time $t = 0$. This problem is not intended to be a physiologically realistic FSI model of a BHV in an artery. It is constructed to exhibit the same challenging flow conditions, for the purpose of demonstrating that div-conforming B-splines can easily circumvent difficulties encountered by methods that produce only weakly divergence-free velocity solutions.

7.2.2 Discretization

The cylindrical fluid domain is discretized using a B-spline knot space $\widehat{\Omega}_1 = [-1,1]\times[-1,1]\times[-1,2]$. A point $\mathbf{X}$ in this knot space is (in the notation of Section 3.1.2.2) mapped to the physical domain Ω_1 with the mapping

$$\phi_1 = RX_1\sqrt{1-\frac{1}{2}X_2^2} \tag{7.2}$$

$$\phi_2 = RX_2\sqrt{1-\frac{1}{2}X_1^2} \tag{7.3}$$

$$\phi_3 = LX_3 \tag{7.4}$$

with $R = 1.25$ cm and $L = 1$ cm. This mapping, illustrated in Figure 7.3, becomes singular at the corners of the parametric domain. A robust inverse of $\boldsymbol{\phi}(\mathbf{X})$ (as required for the immersed surface integration described in Section 3.3) is implemented by using Newton iteration with an exception to short-circuit the iteration and map $\boldsymbol{\phi}^{-1}(\mathbf{x})$ to a sentinel value outside of the knot space if $\mathbf{x}$ falls outside of the physical cylinder of radius R. (Otherwise, the singular nature of the mapping can prevent convergence.) The knot space is evenly subdivided into $40 \times 40 \times 40$ knot spans and div-conforming B-spline velocity and pressure spaces of degree $k' = 1$ are defined on this mesh. The no-penetration constraint on the sides of the cylinder is enforced strongly and

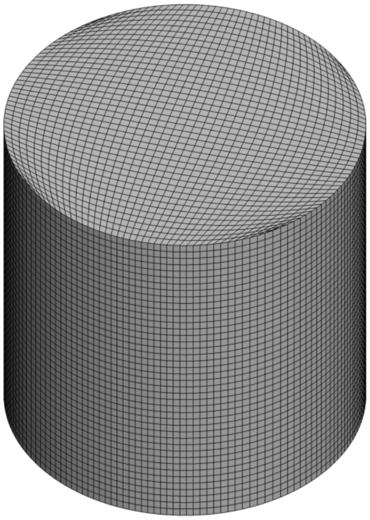


Figure 7.3: The physical image of the B-spline parameter space, showing the mesh of unique knots (black lines).

the no-slip condition is enforced weakly by velocity penalization with penalty parameter $C_{\text{no slip}} = 10$ dyn/cm^2/(cm/s). The FSI coupling parameters are $\tau^B_{\text{NOR}} = 1000$ dyn/cm^2/(cm/s), $\tau^B_{\text{TAN}} = 10$ dyn/cm^2/(cm/s), and $r = 0$. The contact parameters in the discrete structure subproblem are $k = 10^7$ dyn/cm^2/cm, $h_{\text{c}} = 0.005$ cm, and $c = 0.1$ cm. The temporal discretization uses the backward Euler method with $\Delta t = 5.0 \times 10^{-4}$ s. Six block iterations are used in each time step. The formulation is under-integrated, using a reduced quadrature rule with $(k' + 1)^d$ points in each Bézier element. Typically $(k' + 2)^d$ points per element are needed to obtain optimal convergence rates with smooth solutions,[1] but, in the presence of immersed boundaries,

---

[1] That is, if one ignores the possibility of more efficient quadrature rules for IGA spline spaces, e.g. [94].

convergence rates are limited by regularity of the exact solution.

7.2.3 Results

The opening of the valve is illustrated by several snapshots in Figure 7.4. The closed state at time $t = 0.197$ s is shown in Figure 7.5. The history of volumetric flow rate through the bottom of the cylinder is given in Figure 7.6, which indicates that the valve is able to block flow without the spurious apparent leakage that spoils solutions computed with unmodified stabilized formulations (cf. Section 6.4). These results illustrate the basic soundness of using div-conforming B-splines as a fluid discretization for BHV FSI simulations. I now take a closer look at the mass conservation in the computed solutions. Because I use an iterative solver to approximate the fluid increments in the block iteration (Section 4.3), $\nabla \cdot \mathbf{u}_1^h$ is not exactly zero. For the results presented above, I solve for fluid increments with the default

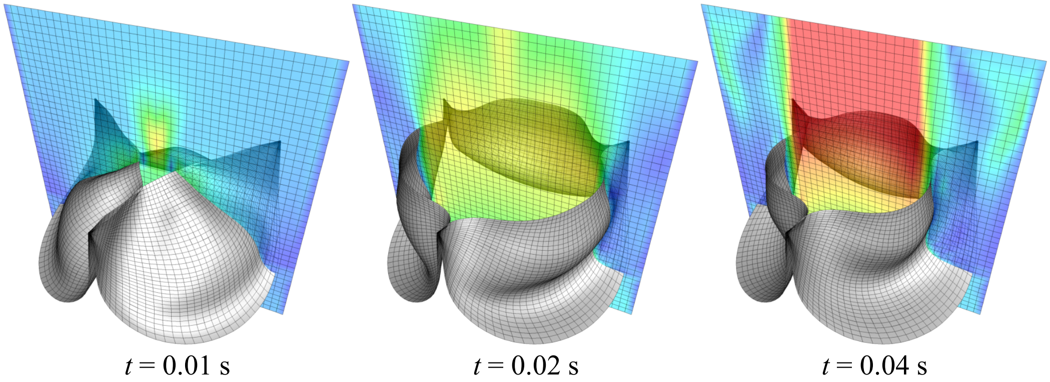


Figure 7.4: Snapshots of the opening process. Velocity magnitude is plotted on a slice, using a color scale ranging from 0 (blue) to ≥ 200 cm/s (red).

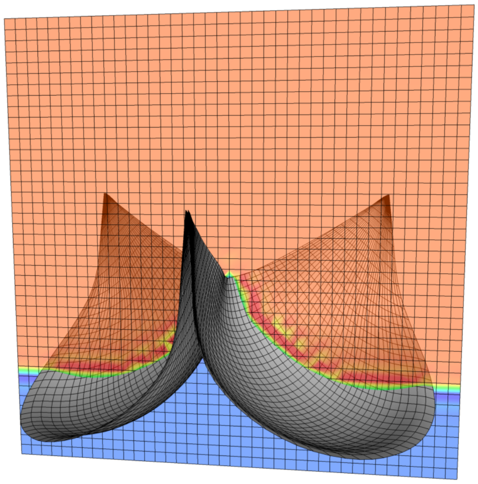


Figure 7.5: The closed valve at time $t = 0.16$ s. Pressure is plotted on a slice, using a color scale ranging from $\leq -1.1 \times 10^5$ dyn/cm^2 (blue) to $\geq 10^4$ dyn/cm^2 (red). Some over- and under-shoot is evident near the immersed structure.

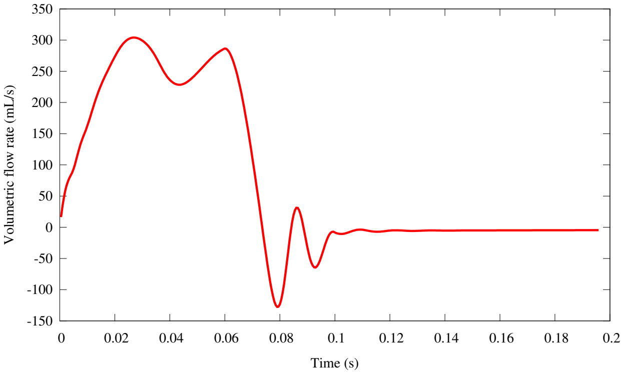


Figure 7.6: The volumetric flow rate through the cylinder.

Krylov method of PETSc (namely, GMRES(30) with a simple preconditioner)[2] to a relative tolerance of 10^{-2} for convergence of the preconditioned residual.

---

[2] This naive solver does not scale well with spatial refinement, which is entirely expected when solving saddle point problems, but the development of optimal linear solution strategies is beyond the scope of the present dissertation.

Even with this loose tolerance, there is no disastrous mass loss. I now recompute step 391 (at time $t = 0.1955$ s, when the valve is closed, under a large pressure jump) with a range of relative tolerances. For this experiment, I use the *un*preconditioned residual to measure convergence, so that results generalize more readily to other iterative solvers. The residual is assembled in centimeter–gram–second (CGS) units, without any scaling to compensate for the difference in units between entries of the momentum and continuity equation residuals. The velocity divergence L^2 norms of the solutions to this time step are collected in Table 7.1. As expected, the velocity divergence approaches zero as the algebraic solution accuracy improves.

Table 7.1: The effect of relative tolerance in the approximate inversion of A_f (Section 4.3) on mass conservation.

Solver tolerance	$\|\nabla \cdot \mathbf{u}_1\|_{L^2(\Omega_1)}$ (CGS units)
10^{-1}	3.9×10^{-5}
10^{-2}	1.2×10^{-5}
10^{-3}	3.0×10^{-7}
10^{-4}	2.0×10^{-8}
10^{-5}	1.2×10^{-9}
10^{-6}	2.4×10^{-10}
10^{-7}	4.3×10^{-11}

7.3 Simulating an *in vitro* experiment

The content of this dissertation so far has focused on verifying that the FSI analysis techniques described in Chapters 3 and 4 can accurately approximate solutions to the mathematical model stated in Chapter 2. However, this model is not guaranteed *a priori* to describe the dynamics of a heart valve immersed in fluid. The present section serves both to further illustrate the application of div-conforming B-splines to realistic problems and to argue that the modeling assumptions from Chapter 2 can represent the dynamics of an artificial heart valve immersed in fluid. The content of this section is adapted from [106].

This section provides preliminary experimental validation of the model from Chapter 2 by qualitatively comparing FSI simulation outputs with the results of *in vitro* experiments using a latex valve in a device called a flow loop. A flow loop is an artificial hydraulic system comprising a series of fluid-carrying tubes connecting several components in a closed loop. Typically one component, the pump, drives fluid through the loop. This might be accomplished in a continuous manner by, e.g., a centrifugal pump, but, to construct *in vitro* models of cardiovascular systems, the pump is usually fashioned to mimic the action of a cardiac ventricle: a time-varying pressure is applied to a fluid-filled chamber, with valves upstream and downstream to ensure that this pressure induces a unidirectional flow through the loop. Additional components in the loop can tune the response of the flow to this pumping action by providing viscous drag, hydrostatic pressure differences (from changes in elevation), or pressure in proportion to stored fluid, known as hydraulic compliance. In the experiment described in this section, the measurements that are collected from the flow loop experiment are photographic images of the deforming valve and volumetric flow rate through it. The flow rate is used as data for the model we construct, while valve deformation is the feature of the physical system that the model is used to predict. I assess the validity of the model in the context of predicting valve deformation by qualitatively comparing the computed and photographed deformations.

Remark 7.1. The purpose of this section is emphatically *not* to experimentally validate the *numerical methods* described in this Chapters 3 and 4. Recall the terminological conventions that I commit to in Section 1.4. Numerical methods approximate mathematical problems. The verification of numerical methods and their implementations (i.e. "solving the equations right") is a separate concern from validation of mathematical models (i.e. "solving the right equations") [9, Section 4, Rule 5]. Attempts to experimentally validate numerical methods or computer programs (rather than mathematical models) reflect confusion over the distinction between verification and validation.

7.3.1 Description of the experiment

The preliminary validation experiment consists of a latex valve in an acrylic tube. Volumetric flow rate through the tube is measured using an ultrasonic flow meter and images of the valve are collected using a borescope. Water is pumped through the tube using a flow loop system similar to the bioreactor detailed in [80]. The flow loop is shown in Figure 7.7, with annotations indicating the locations of different components and the prevailing flow direction permitted by the valves. I use the FSI analysis techniques developed in this dissertation to simulate only the segment of tubing between the pump and the flow meter, containing the artificial aortic valve. The cyclical

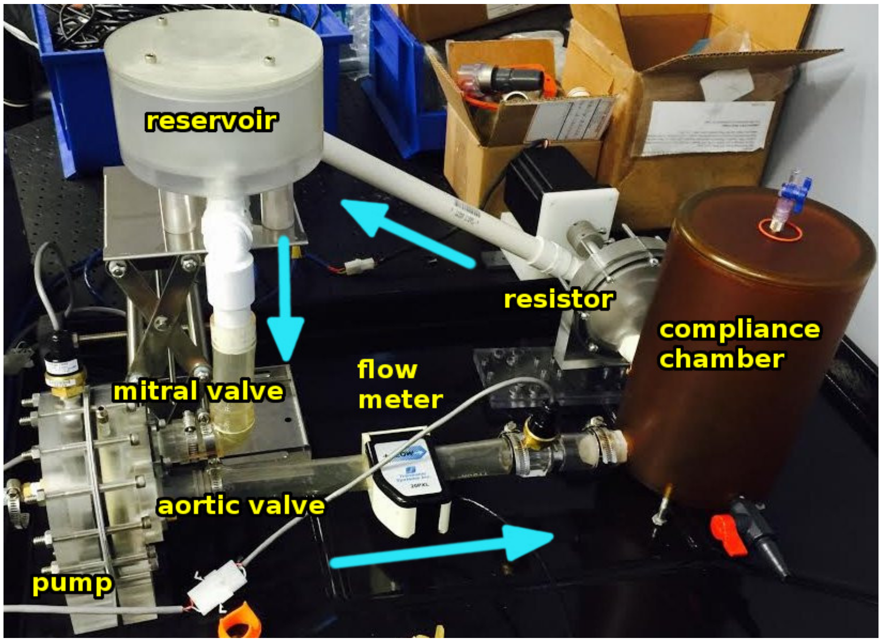


Figure 7.7: An annotated photograph of the flow loop. The blue arrows indicate the direction of flow permitted by the valves.

response of flow rate to pressure produced in the pump is highly sensitive to the precise configuration of the compliance chamber and resistor, along with the inertia of fluid in the loop, and additional resistance and compliance associated with other components outside of the simulated domain. To avoid complications associated with experimentally controlling and mathematically modeling the interaction of the valve and adjacent fluid with the hydraulic components upstream and downstream of the valve location, I focus on the phase of the flow cycle during which the valve opens. In this limited context, the net effect of the components upstream and downstream of the valve can be seen as a black box determining the volumetric flow rate through the valve. Any configuration leading to the same flow rate would lead to essentially equivalent deformations of the valve. This is clearly *not* a reasonable assumption when studying valve closure, where the deformation of the valve is largely determined by the pressure difference across it, which can be altered without changing the (lack of) flow through the valve. In the case of a closing valve, the dynamics of the valve and adjacent fluid are inseparably coupled to the compliance, resistance, and fluid inertia upstream and downstream of the valve. Experimental reproducibility—which is an essential prerequisite to representing an experiment with a mathematical model [9, Section 4, Rule 8]—would depend on meticulous control over and documentation of the entire system, which is beyond the scope of the present validation effort.

Remark 7.2. The ability to accommodate valve closure under physiological pressure levels is one of the numerical method's distinguishing successes. However, the formulation of realistic boundary conditions to model specific animal circulatory systems or artificial fluidic devices (rather than merely obtaining Reynolds numbers and pressure differences of the right general magnitude) remains an open problem. The details of modeling a specific *in vitro* experiment will not likely carry over directly to the *in vivo* setting or even to other *in vitro* experiments. The scientific value of formulating, calibrating, and validating a sophisticated boundary condition to model the specific experiment described here is therefore questionable, as it would not clearly inform any future applications.

7.3.1.1 The acrylic tube

In this experiment, the valve is placed in a straight tube, illustrated in Figures 7.8 and 7.9. The inner diameter varies between 2 and 3 cm along the length of the tube, as shown in Figure 7.9, and is roughly the size of a typical human ascending aorta. To capture images of the valve, a hole is included in the side of the tube, to permit insertion of a borescope. Using a mirror attachment to the end of the borescope's optical relay, this allows for a view of the valve from the aortic side as illustrated schematically in Figure 7.10.

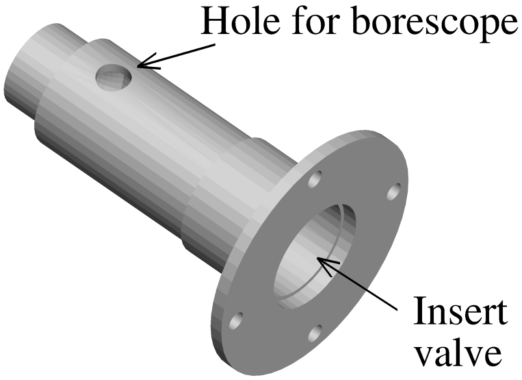


Figure 7.8: A 3D rendering of a CAD model of the acrylic tube.

7.3.1.2 The valve

I constructed the valve by attaching latex leaflets to an aluminum stent with superglue. The valve is shown in Figure 7.11. I cut each leaflet from a flat sheet of latex with thickness 0.054 cm in such a way that the free edge is straight in the flattened configuration and the attached edge matches the geometry of the stent if the latex sheet is deformed into a cylinder without

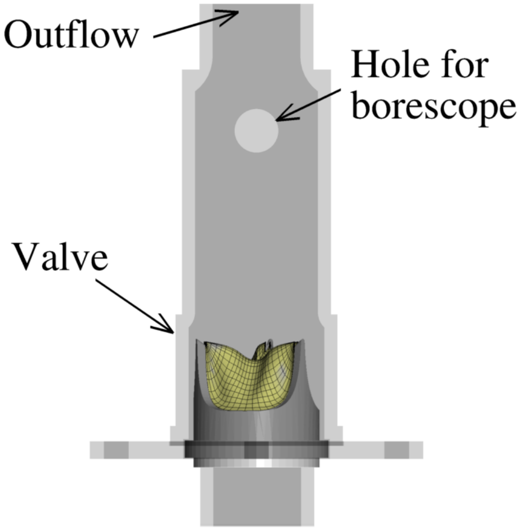


Figure 7.9: A 2D to-scale view of the tube, showing its relation to the valve and stent. The inflow and outflow have inner diameters of 2 cm.

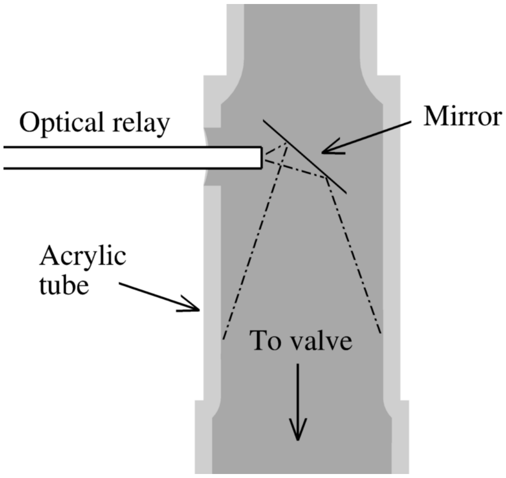


Figure 7.10: A schematic illustration of how images of the valve are captured.

stretching. Cutting leaflets out from the latex sheet can be done to a high degree of precision (~ 0.05 cm), but the difficulty of manually gluing leaflets onto the stent in a consistent manner is a major source of uncertainty in the experiment. In my estimation, this is the largest source of uncertainty affecting leaflet displacements during the opening phase of the flow cycle. The level of uncertainty in leaflet displacement introduced by inconsistencies in leaflet attachment can be roughly estimated by looking at the deviations from trefoil symmetry of the leaflets in their static equilibrium configuration shown in Figure 7.11.

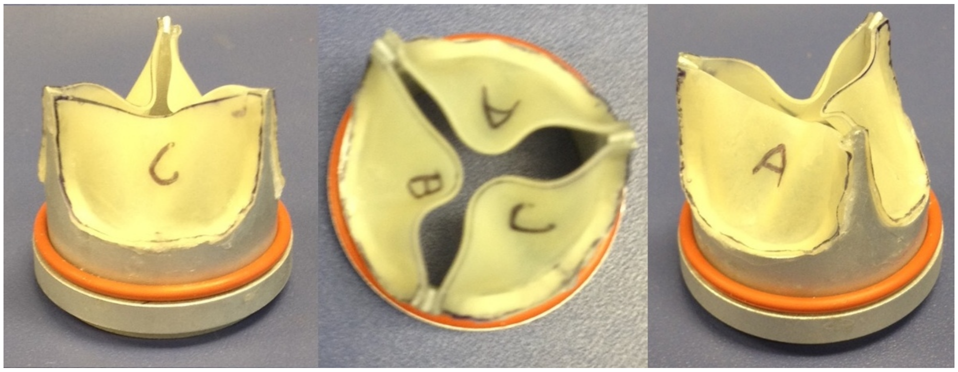


Figure 7.11: The physical valve used in the validation experiment.

7.3.2 Mathematical model of the experiment

This section specifies an instance of the mathematical problem stated in Chapger 2 that is designed to resemble the experiment described in Section 7.3.1.

7.3.2.1 Fluid subproblem

The mathematical model includes some deliberate simplifications of the geometry of the region occupied by fluid. Ω_1 consists of the image of a parametric space $\widehat{\Omega}_1 = (-1,1) \times (-1,1) \times (-1,4.5) \subset \mathbb{R}^3$ under the mapping

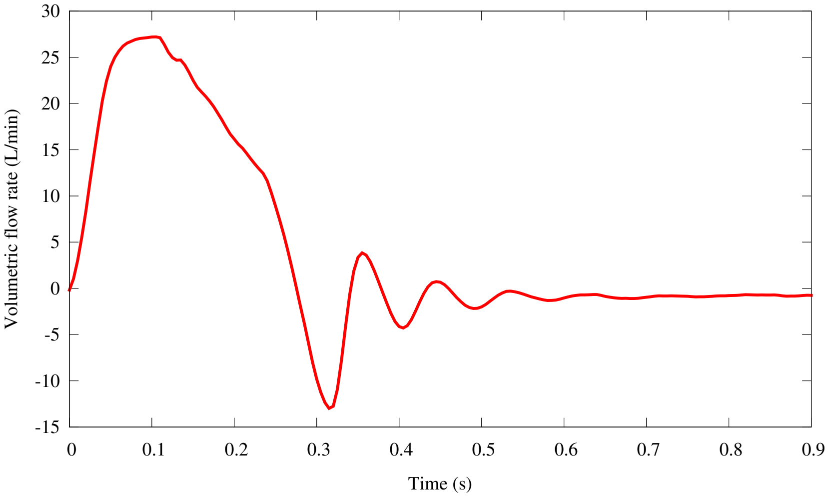


Figure 7.12: The measured volumetric flow rate used to set a Dirichlet boundary condition in the mathematical model.

$\boldsymbol{\phi}(\mathbf{X})$, which is defined by

$$\phi_1 = R(X_3)X_1\sqrt{1-\frac{1}{2}X_2^2}\ , \tag{7.5}$$

$$\phi_2 = R(X_3)X_2\sqrt{1-\frac{1}{2}X_1^2}\ , \tag{7.6}$$

$$\phi_3 = LX_3\ , \tag{7.7}$$

where $L = 1$ cm and $R(X_3)$ is defined by

$$R(X_3) = \begin{cases} R_{\text{in}} & X_3 < z_1 \\ R_{\text{out}} & X_3 > z_2 \\ (R_{\text{out}} - R_{\text{in}})\sin^2\left(\frac{\pi(X_3-z_1)}{2(z_2-z_1)}\right) + R_{\text{in}} & \text{otherwise} \end{cases}\ , \tag{7.8}$$

with $z_1 = -0.45$ cm, $z_2 = 0$, $R_{\text{in}} = 1$ cm, and $R_{\text{out}} = 1.4025$ cm. The shape of Ω_1 is shown in Figure 7.13. This is an admittedly crude approximation of the connection between the two tubes, but it is convenient from the standpoint of computing with a div-conforming B-spline space defined on a single patch.

Figure 7.13: The shape of the fluid subproblem domain, Ω_1, defined by applying the transformation (7.5)–(7.7) to a trivariate B-spline parameter space.

The lateral sides of Ω_1 are subject to a no-slip and no-penetration condition. The inflow face of the domain is subject to a time-dependent plug flow boundary condition with the volumetric flow rate history shown in Figure 7.12. This flow rate was directly measured from the flow loop. The outflow face of the domain is subject to a traction-free boundary condition, including backflow stabilization with $\gamma = 1$. The fluid velocity initial condition is $\mathbf{u}_1^0 \equiv \mathbf{0}$. To model water, the viscosity of the fluid is set to $\mu = 1$ cP and the density is set to $\rho_1 = 1.0$ g/cm^3.

7.3.2.2 Structure subproblem

The latex is modeled as an incompressible neo-Hookean material with shear modulus $\mu_s = 8.7 \times 10^6$ dyn/cm^2 (based on uniaxial stretching experiments). Numerical experiments indicate that the leaflet opening kinematics are insensitive to this number, so long as it is the right order of magnitude, but the details of these numerical experiments have been omitted for brevity. I remark, however, that the strain distribution in closed leaflets (cf. [85, Figure 5]) or the opening kinematics in a *pressure*-driven flow would be more sensitive to leaflet material properties.

The geometry of the stress-free reference configuration Γ_0 is specified by manually selecting B-spline control points to generate the configuration shown in Figure 7.14. The leaflets are flat in Γ_0, based on the fact that the physical leaflets are cut out of a flat latex sheet. These leaflets are deformed into a static equilibrium configuration Γ_0', (a discrete approximation of) which is also shown in Figure 7.14. The boundary corresponding to the attached edge is subject to a strongly-enforced clamped boundary condition, in which displacement and derivatives of displacement are fixed to equal their values in Γ_0'. The stent is assumed to be rigid and its principle effect on the fluid is presumed to be merely preventing flow from passing between the wall of the tube and the attached leaflet edges. The stent is therefore modeled crudely in the FSI problem, as a rigid extension of the leaflets, closing the gap between the attached edge and the boundary of Ω_1. This extension is shown in relation to the leaflets in Figure 7.16. In a slight abuse of the notation introduced in Chapter 2, the leaflets are considered to be initially at rest in the deformed configuration Γ_0' (rather than the stress-free configuration Γ_0 which is used as a reference configuration in (2.8)). Figure 7.15 compares the model to the physical valve.

The partially-closed equilibrium configuration shown in Figure 7.15 is

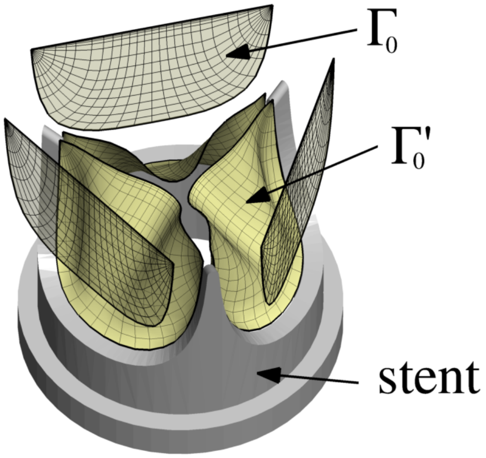


Figure 7.14: The reference (Γ_0) and initial (Γ_0') configurations of the valve model, shown in relation to a CAD model of the aluminum stent.

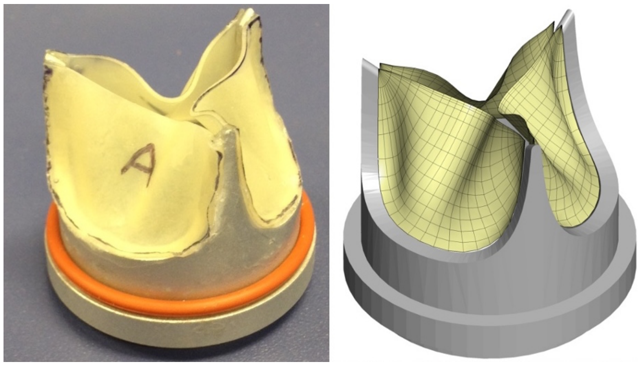


Figure 7.15: A visual comparison of the physical valve and its model, in the configuration Γ_0'.

not the unique static equilibrium configuration of the valve. Each leaflet can be snapped through to a stable open configuration. I found this necessary to ensure reproducible behavior in the experiment. (Otherwise, subtle, uncontrolled variations in the closed leaflet geometry lead to large differences in behavior between leaflets.) This also provides a simple analytical explana-

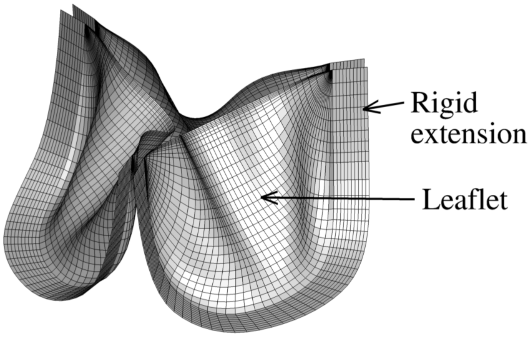


Figure 7.16: The rigid extensions closing the gap between the attached edges of the leaflets and the boundary of Ω_1.

tion for the insensitivity of opening kinematics to shear modulus. Analytical elimination of the pressure Lagrange multiplier from the 2^{nd} Piola–Kirchhoff stress shows that the entire internal work term of (2.8) is proportional to μ_s, which means that a static equilibrium configuration with strongly enforced kinematic boundary conditions and no external loading is independent of μ_s. The (unloaded) closed and open configurations of the thin shell are therefore dictated by *geometry*, which perhaps partly explains the extreme improvements in performance that can be obtained from isogeometric discretizations of shell structures; see the comparison in [127] for a practical example in the context of heart valve structural analysis. Loosely speaking, fluid flow drives the opening valve over a material-dependent energy barrier separating these two material-independent equilibrium configurations. The prescription of a Dirichlet boundary condition at the inflow causes the force driving the fluid to adjust to the height of this energy barrier, lessening the apparent dependence of the system's dynamics on μ_s (relative to pressure-driven flow).

7.3.3 Discretization of the mathematical model

The fluid subproblem parametric domain $\widehat{\Omega}_1$ is split evenly into $64 \times 64 \times 99$ Bézier elements, used to define div-conforming B-spline spaces of degree $k' = 1$. The no-slip and inflow Dirichlet boundary conditions are enforced by velocity penalization, with penalty-constants of $C_{\text{no slip}} = 10$ dyn/cm^2/(cm/s) and $C_{\text{inflow}} = 1000$ dyn/cm^2/(cm/s) respectively, while the no-penetration condition on the lateral sides of the flow domain is enforced strongly. The renderings of the structure subproblem model in Figures 7.14 and 7.15 show the isogeometric discrete model, which consists of a 936-element quadratic B-spline mesh (with the element count excluding the rigid extensions shown in Figure 7.16). The equilibrium configuration Γ_0' is approximated in the computational model by driving a structural dynamics simulation with mass damping from Γ_0 to a steady solution with the attached edges of the leaflets clamped into the configuration shown in Figure 7.14. The values of the contact penalty parameters are $k = 10^8$ (dyn/cm^2)/cm, $h_{\text{c}} = 0.04$ cm, and $c = 0.1$ cm. The FSI penalty parameters are $\tau^B_{\text{NOR}} = 1000$ dyn/cm^2/(cm/s) and $\tau^B_{\text{TAN}} = 10$ dyn/cm^2/(cm/s). The multiplier stabilization parameter r is set to zero. The backward Euler time integration method is used with $\Delta t = 2.5 \times 10^{-4}$ s. Seven block iterations, reusing A_{f} from the first, are used to converge the implicit phase of each time step. A_{f} is inverted approximately, using GMRES(300) (via PETSc, with the default preconditioning options) and a relative tolerance of 10^{-3} for the unpreconditioned residual.

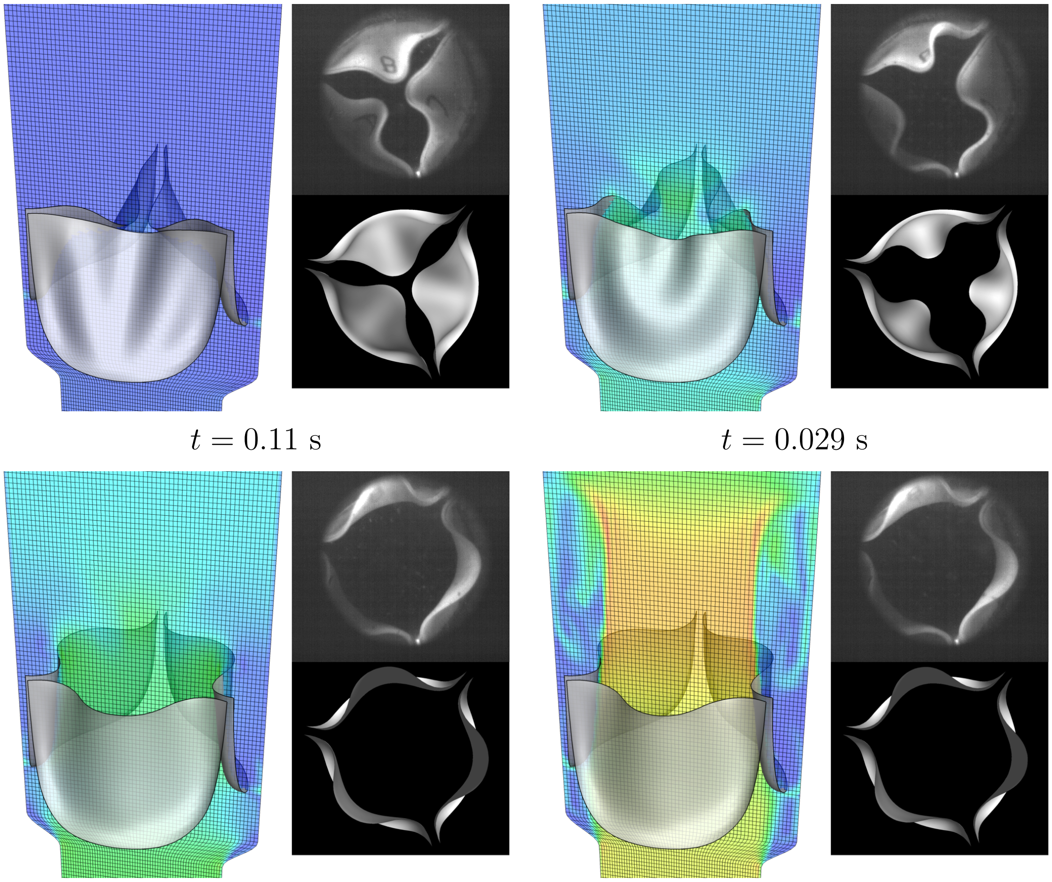
7.3.4 Comparison of results

This section qualitatively compares the computational and experimental results. The experimental results consist of a flow rate history and a sequence of images taken through the borescope, as illustrated in Figure 7.10. The flow rate history is used as an input to the mathematical model, so it is vacuous to compare the flow rate measurements with the flow rates in the model. The measurements that remain for comparison are the sequence of images of the valve. Due to limitations of measurement equipment, I do not have information on when images were recorded in relation to the time axis of Figure 7.12. (The images and flow rate history are most likely from different cycles altogether, but the flow rate was very nearly periodic, with cycle-to-cycle differences too small to affect the conclusions drawn from this work.) To associate images with values on the time axis of the flow rate plot in Figure 7.12, I first select an image of the valve in which it appears, subjectively, to be starting to open. Next, I assume that this corresponds to the time value at which the flow rate first becomes positive. Then I assign time values to subsequent images by assuming that they are captured at a constant frame rate. By counting the total number of frames and comparing with the number of times the valve opens, I estimate the frame rate to be 220 frames per second. I use this estimated rate, along with the subjectively-identified frame corresponding to $t = 0$, to assign time values to the experimental images. I estimate that this introduces several milliseconds of uncertainty into the temporal alignment of the images with the flow rate.

Figure 7.17 compares the computed valve deformations at several time points with images collected in the experiment. The time values given in this figure are synchronized with the time axis of the flow rate plot in Figure 7.12. The relative alignment in time of the photographic images is only accurate to within several milliseconds. For direct comparison with experimental images, the computed leaflet deformations are rendered using perspective (i.e. a pinhole camera model) from a vantage point that is positioned relative to the valve and stent in a way that corresponds to the location of the tip of the borescope the experiment.

Remark 7.3. The use of perspective and appropriate viewer position are critical to obtaining a qualitative correspondence in the results. When the leaflets are rendered using isometric perspective (i.e. the assumption that the scene is viewed from an infinite distance, which is applied by default in many visualization programs), the ventricular sides of the leaflets are not visible when the valve is fully open.

The main qualitative difference between these sets of images is in the degree of symmetry of the leaflet deformations during the transition to the fully open state. This difference is not unexpected given that the initial condition to the computer simulation is symmetrical while the physical valve is not. The degree of asymmetry in the valve is evident from the photographs in Figure 7.11. As explained in Section 7.3.1.2, this asymmetry is mainly due to differences in the stresses introduced by manually gluing each initially-flat



$t = 0.11$ s

$t = 0.029$ s

$t = 0.039$ s

$t = 0.084$ s

Figure 7.17: Several snapshots of the computed solution, compared with experimental images. At each time instant, the computed solution is shown in the left-hand frame and at the bottom of the right-hand frame. The experimental results are shown in the top of the right-hand frame. Colors indicate fluid velocity magnitude on a slice. Color scale: 0 (blue) to ≥200 cm/s (red).

leaflet into the stent. The physical valve assembled for this experiment is *nominally* symmetric (as per the stated experimental procedure), so the difference in behavior between leaflets is indicative of the degree of experimental reproducibility. Differences between the deformations of the three physical leaflets therefore put a lower bound on meaningful differences between computational and experimental results. Figure 7.17 also shows the computational results from a different view, with contour plots of fluid velocity magnitude on slices cutting through Ω_1. This illustrates the ability of computer simulations to provide additional information about the flow field and the full 3D deformation of the leaflets that would be difficult to measure experimentally.

The qualitative resemblance of computed leaflet deformations to the observed deformations indicates that the modeling assumptions of Chapter 2 are not wildly inappropriate for predicting the deformations of heart valve leaflets immersed in physiological flow fields and may be able to predict quantities of interest related to deformation (such as strain) with practically-useful accuracy. The computed results demonstrate agreement with qualitative features of artificial valve leaflet deformations observed in other *in vitro* experiments as well. The computed solution at time $t = 0.029$ s shows the opening process, as characterized by reversal of leaflet curvature, beginning primarily near the attached edge, in the so-called belly region of the leaflet. This is in agreement with the observations of Iyengar et al. [96] who used images captured from multiple vantage points to reconstruct 3D deformations of valve leaflets *in vitro*. As demonstrated in [85], this behavior is *not* captured by simulations using structural dynamics alone, which underscores the importance of accounting for FSI in heart valve modeling. These preliminary results are therefore sufficient to justify the nontrivial expenses associated with more rigorous experimental validation in the future.

Chapter 8

Conclusions and further work

In this dissertation, I develop, analyze, and test a novel numerical method for computational analysis of thin structures with spline-based geometries immersed in viscous incompressible fluids. I find that this method is sufficiently robust to survive application to FSI analysis of BHVs functioning under physiological conditions and that it can be shown *a priori* to converge when applied to simplified linear model problems. I then use this numerical method to demonstrate that the underlying mathematical model of FSI that it approximates is sufficient to reproduce qualitative behaviors of artificial valves from *in vitro* experiments.

An uncommon feature of the technique I have developed is that it is immediately compatible with stabilized finite element discretizations of the fluid subproblem. Such discretizations form the backbone of the well-developed open source vascular FSI platforms SimVascular [120] and CRIMSON [108]. Related numerical methods also underpin recent pioneering efforts by HeartFlow Inc. [202] to commercialize vascular CFD. This suggests a practical path toward connecting the BHV FSI technology developed in this dissertation to rich platforms for segmenting and meshing patient-specific geometries and applying sophisticated physiological boundary conditions to the fluid subproblem.

The computational FSI method developed in this dissertation is not limited to BHV simulation. I am currently collaborating with the research group of M.-C. Hsu to apply immersogeometric fluid–thin structure interaction analysis to simulation of hydraulic arresting gears that help dissipate the kinetic energy of fixed-wing aircraft landing on short runways. Initial results, published in [198], compare favorably with earlier body-fitted simulations of such devices [195]. The flexibility provided by immersogeometric FSI analysis allowed for automated optimization of the device geometry.

Despite its successful application to BHV FSI and other problems, the numerical method developed in this dissertation can be improved. In its current form, there are a number of free parameters. The present guidelines for selecting these are based on imprecise dimensional analysis. More precise and rational selection of parameters will likely stem from further numerical analysis of linear model problems, building on the initial work presented in Chapter 5. As mentioned in Remark 5.2, Y. Yu has already derived several improvements to the analysis of Chapter 5. Another undesirable aspect of the method presented in this dissertation is the trade-off between conservation and stability parameterized by the stabilization coefficient r (introduced in Section 4.1). A possible improvement that I am currently investigating with J. A. Evans is to apply the inconsistent stabilization following from $r > 0$ only to fine scales of the fluid–structure interface Lagrange multiplier while retaining strong consistency on coarse scales. This may result in a method that is both stable and conservative. Visualizations from a preliminary implementation of this approach applied to the problem of Section 6.3 are shown in Figure 8.1; both the flow field and interface Lagrange multiplier are qualitatively-reasonable.

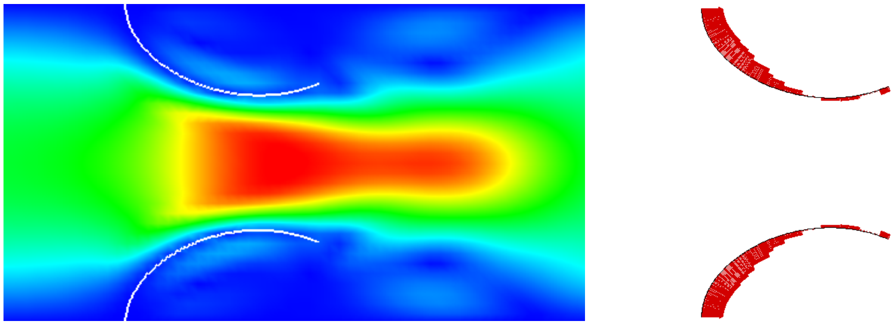


Figure 8.1: Example of an FSI solution with interface Lagrange multiplier stabilization applied only to fine scales. Left: velocity magnitude. Right: Lagrange multiplier field (visualized as in Section 6.3).

Lastly, the promising initial results of immersogeometric FSI analysis using div-conforming B-spline discretizations of the fluid subproblem indicate that div-conforming B-splines merit further investigation. The ideas of immersogeometric FSI analysis and div-conforming B-spline flow discretizations appear to enjoy a symbiotic connection, in that the strong mass conservation of structure preserving flow discretizations improves the quality of immersogeometric FSI solutions, while the application of div-conforming B-splines to increasingly complicated and realistic problems motivates the development of more powerful implementations.

Bibliography

[1] MUMPS: a MUltifrontal Massively Parallel sparse direct Solver. `http://mumps.enseeiht.fr/`. Accessed 24 April 2016.

[2] R. A. Adams. *Sobolev Spaces.* Academic Press, New York, 1975.

[3] I. Akkerman, Y. Bazilevs, V. M. Calo, T. J. R. Hughes, and S. Hulshoff. The role of continuity in residual-based variational multiscale modeling of turbulence. *Computational Mechanics*, 41:371–378, 2008.

[4] D. N. Arnold, R. S. Falk, and R. Winther. Finite element exterior calculus, homological techniques, and applications. *Acta Numer.*, 15:1–155, 2006.

[5] D. N. Arnold and X. B. Liu. Local error estimates for finite element discretizations of the Stokes equations. *RAIRO Modél. Math. Anal. Numér.*, 29(3):367–389, 1995.

[6] M. Astorino, J.-F. Gerbeau, O. Pantz, and K.-F. Traoré. Fluid–structure interaction and multi-body contact: Application to aortic valves. *Computer Methods in Applied Mechanics and Engineering*, 198:3603–3612, 2009.

[7] F. Auricchio, M. Conti, A. Ferrara, S. Morganti, and A. Reali. Patient-specific simulation of a stentless aortic valve implant: the impact of

fibres on leaflet performance. *Computer Methods in Biomechanics and Biomedical Engineering*, 17(3):277–285, 2014.

[8] F. P. T. Baaijens. A fictitious domain/mortar element method for fluid–structure interaction. *International Journal for Numerical Methods in Fluids*, 35(7):743–761, 2001.

[9] I. Babuška and J. T. Oden. Verification and validation in computational engineering and science: basic concepts. *Computer Methods in Applied Mechanics and Engineering*, 193(36–38):4057–4066, 2004.

[10] C. Bacuta. A unified approach for Uzawa algorithms. *SIAM Journal on Numerical Analysis*, 44(6):2633–2649, 2006.

[11] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, S. Zampini, and H. Zhang. PETSc users manual. Technical Report ANL-95/11 - Revision 3.6, Argonne National Laboratory, 2015.

[12] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, S. Zampini, and H. Zhang. PETSc Web page. `http://www.mcs.anl.gov/petsc`, 2015.

[13] S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith. Efficient management of parallelism in object oriented numerical software libraries.

In E. Arge, A. M. Bruaset, and H. P. Langtangen, editors, *Modern Software Tools in Scientific Computing*, pages 163–202. Birkhäuser Press, 1997.

[14] H. J. C. Barbosa and T. J. R. Hughes. The finite element method with Lagrange multipliers on the boundary: circumventing the Babuška-Brezzi condition. *Computer Methods in Applied Mechanics and Engineering*, 85(1):109–128, 1991.

[15] Y. Bazilevs and I. Akkerman. Large eddy simulation of turbulent Taylor–Couette flow using isogeometric analysis and the residual–based variational multiscale method. *Journal of Computational Physics*, 229:3402–3414, 2010.

[16] Y. Bazilevs, V. M. Calo, J. A. Cottrel, T. J. R. Hughes, A. Reali, and G. Scovazzi. Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 197:173–201, 2007.

[17] Y. Bazilevs, V. M. Calo, J. A. Cottrell, J. A. Evans, T. J. R. Hughes, S. Lipton, M. A. Scott, and T. W. Sederberg. Isogeometric analysis using T-splines. *Computer Methods in Applied Mechanics and Engineering*, 199:229–263, 2010.

[18] Y. Bazilevs, V. M. Calo, T. J. R. Hughes, and Y. Zhang. Isogeometric fluid–structure interaction: theory, algorithms, and computations. *Computational Mechanics*, 43:3–37, 2008.

[19] Y. Bazilevs, M.-C. Hsu, and M. A. Scott. Isogeometric fluid–structure interaction analysis with emphasis on non-matching discretizations, and with application to wind turbines. *Computer Methods in Applied Mechanics and Engineering*, 249–252:28–41, 2012.

[20] Y. Bazilevs and T. J. R. Hughes. Weak imposition of Dirichlet boundary conditions in fluid mechanics. *Computers and Fluids*, 36:12–26, 2007.

[21] Y. Bazilevs, C. Michler, V. M. Calo, and T. J. R. Hughes. Weak Dirichlet boundary conditions for wall-bounded turbulent flows. *Computer Methods in Applied Mechanics and Engineering*, 196:4853–4862, 2007.

[22] Y. Bazilevs, C. Michler, V. M. Calo, and T. J. R. Hughes. Isogeometric variational multiscale modeling of wall-bounded turbulent flows with weakly enforced boundary conditions on unstretched meshes. *Computer Methods in Applied Mechanics and Engineering*, 199:780–790, 2010.

[23] Y. Bazilevs, K. Takizawa, and T. E. Tezduyar. *Computational Fluid–Structure Interaction: Methods and Applications.* Wiley, Chichester, 2013.

[24] S. Bertoluzza, A. Decoene, L. Lacouture, and S. Martin. Local error estimates of the finite element method for an elliptic problem with a Dirac source term. `http://arxiv.org/abs/1505.03032`, 2015.

[25] S. Bertoluzza, M. Ismail, and B. Maury. Analysis of the fully discrete fat boundary method. *Numerische Mathematik*, 118(1):49–77, 2010.

[26] M. Bischoff, W. A. Wall, K. -U. Bletzinger, and E. Ramm. Models and finite elements for thin-walled structures. In E. Stein, R. de Borst, and T. J. R. Hughes, editors, *Encyclopedia of Computational Mechanics*, Volume 3: Solids and Structures, chapter 3. John Wiley & Sons, 2004.

[27] D. Boffi, F. Brezzi, and M. Fortin. Finite elements for the Stokes problem. In D. Boffi and L. Gastaldi, editors, *Mixed Finite Elements, Compatibility Conditions, and Applications*, Lecture Notes in Mathematics, pages 45–100. Springer-Verlag Berlin Heidelberg, 2008.

[28] A. E. J. Bogaers, S. Kok, B. D. Reddy, and T. Franz. Quasi-Newton methods for implicit black-box FSI coupling. *Computer Methods in Applied Mechanics and Engineering*, 279(0):113–132, 2014.

[29] I. Borazjani. Fluid–structure interaction, immersed boundary-finite element method simulations of bio-prosthetic heart valves. *Computer Methods in Applied Mechanics and Engineering*, 257:103–116, 2013.

[30] I. Borazjani, L. Ge, and F. Sotiropoulos. Curvilinear immersed boundary method for simulating fluid structure interaction with complex 3D rigid bodies. *Journal of Computational Physics*, 227(16):7587–7620, 2008.

[31] S. C. Brenner and L. R. Scott. *The Mathematical Theory of Finite Element Methods, 3rd ed.* Springer, 2008.

[32] A. N. Brooks and T. J. R. Hughes. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 32:199–259, 1982.

[33] A. Buffa, J. Rivas, G. Sangalli, and R. Vásquez. Isogeometric discrete differential forms in three dimensions. *SIAM Journal on Numerical Analysis*, 49(2):814–844, 2011.

[34] A. Buffa, G. Sangalli, and R. Vázquez. Isogeometric analysis in electromagnetics: B-splines approximation. *Computer Methods in Applied Mechanics and Engineering*, 199(17–20):1143–1152, 2010.

[35] C. J. Carmody, G. Burriesci, I. C. Howard, and E. A. Patterson. An approach to the simulation of fluid–structure interaction in the aortic valve. *Journal of Biomechanics*, 39:158–169, 2006.

[36] G. G. Chew, I. C. Howard, and E. A. Patterson. Simulation of damage in a porcine prosthetic heart valve. *J Med Eng Technol*, 23(5):178–189, 1999.

[37] A. J. Chorin. A numerical method for solving incompressible viscous flow problems. *Journal of Computational Physics*, 135(2):118–125, 1967.

[38] J. Chung and G. M. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-α method. *Journal of Applied Mechanics*, 60:371–75, 1993.

[39] LS-DYNA Finite Element Software: Livermore Software Technology Corp. http://www.lstc.com/products/ls-dyna. Accessed 29 March 2012.

[40] R. Courant, K. Friedrichs, and H. Lewy. Über die partiellen Differenzengleichungen der mathematischen Physik. *Mathematische Annalen*, 100(1):32–74, 1928.

[41] R. Courant, K. Friedrichs, and H. Lewy. On the partial difference equations of mathematical physics. *IBM J. Res. Develop.*, 11:215–234, 1967.

[42] J. de Hart. *Fluid–Structure Interaction in the Aortic Heart Valve: a three-dimensional computational analysis.* Ph.D. thesis, Technische Universiteit Eindhoven, Eindhoven, Netherlands, 2002.

[43] J. De Hart, F. P. T. Baaijens, G. W. M. Peters, and P. J. G. Schreurs. A computational fluid–structure interaction analysis of a fiber-reinforced stentless aortic valve. *Journal of Biomechanics*, 36:699–712, 2003.

[44] J. De Hart, G. W. M. Peters, P. J. G. Schreurs, and F. P. T. Baaijens. A three-dimensional computational analysis of fluid–structure interaction in the aortic valve. *Journal of Biomechanics*, 36:103–112, 2003.

[45] L. De Lorenzis, İ. Temizer, P. Wriggers, and G. Zavarise. A large deformation frictional contact formulation using NURBS-based isogeometric

analysis. *International Journal for Numerical Methods in Engineering*, 87:1278–1300, 2011.

[46] J. Donea, S. Giuliani, and J. P. Halleux. An arbitrary Lagrangian–Eulerian finite element method for transient dynamic fluid–structure interactions. *Computer Methods in Applied Mechanics and Engineering*, 33:689–723, 1982.

[47] J. Donea, A. Huerta, J.-P. Ponthot, and A. Rodriguez-Ferran. Arbitrary Lagrangian–Eulerian methods. In *Encyclopedia of Computational Mechanics*, Volume 3: Fluids, chapter 14. John Wiley & Sons, 2004.

[48] C. J. Doolittle, S. D. Drews, and D. B. Goldstein. Near-field flow structures about subcritical surface roughness. *Physics of Fluids*, 26:124106–1–124106–18, 2014.

[49] A. Düster, J. Parvizian, Z. Yang, and E. Rank. The finite cell method for three-dimensional problems of solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 197(45–48):3768–3782, 2008.

[50] M. Esmaily-Moghadam, Y. Bazilevs, T.-Y. Hsia, I. E. Vignon-Clementel, A. L. Marsden, and Modeling of Congenital Hearts Alliance (MOCHA). A comparison of outlet boundary treatments for prevention of backflow divergence with relevance to blood flow simulations. *Computational Mechanics*, 48:277–291, 2011.

[51] J. A. Evans. *Divergence-free B-spline Discretizations for Viscous Incompressible Flows.* Ph.D. thesis, University of Texas at Austin, Austin, Texas, United States, 2011.

[52] J. A. Evans, Y. Bazilevs, I. Babuška, and T. J. R. Hughes. n-Widths, sup-infs, and optimality ratios for the k-version of the isogeometric finite element method. *Computer Methods in Applied Mechanics and Engineering*, 198:1726–1741, 2009.

[53] J. A. Evans and T. J. R. Hughes. Explicit trace inequalities for isogeometric analysis and parametric hexahedral finite elements. *Numerische Mathematik*, 123:259–290, 2013.

[54] J. A. Evans and T. J. R. Hughes. Isogeometric divergence-conforming B-splines for the steady NavierStokes equations. *Mathematical Models and Methods in Applied Sciences*, 23(08):1421–1478, 2013.

[55] J. A. Evans and T. J. R. Hughes. Isogeometric divergence-conforming B-splines for the unsteady Navier–Stokes equations. *Journal of Computational Physics*, 241:141–167, 2013.

[56] L. C. Evans. *Partial Differential Equations (Graduate Studies in Mathematics, Vol. 19).* American Mathematical Society, Providence, RI, 2002.

[57] R. Fan and M. S. Sacks. Simulation of planar soft tissues using a structural constitutive model: Finite element implementation and validation.

Journal of Biomechanics, 47(9):2043–2054, 2014. Functional Tissue Engineering.

[58] L. P. Franca, S. L. Frey, and T. J. R. Hughes. Stabilized finite element methods: I. Application to the advective-diffusive model. *Computer Methods in Applied Mechanics and Engineering*, 95:253–276, 1992.

[59] K. J. Galvin, A. Linke, L. G. Rebholz, and N. E. Wilson. Stabilizing poor mass conservation in incompressible flow problems with large irrotational forcing and application to thermal convection. *Computer Methods in Applied Mechanics and Engineering*, 237–240:166–176, 2012.

[60] L. Ge and F. Sotiropoulos. A numerical method for solving the 3D unsteady incompressible Navier–Stokes equations in curvilinear domains with complex immersed boundaries. *Journal of Computational Physics*, 225(2):1782–1809, 2007.

[61] A.J. Gil, A. Arranz Carreño, J. Bonet, and O. Hassan. An enhanced immersed structural potential method for fluid–structure interaction. *Journal of Computational Physics*, 250:178–205, 2013.

[62] A. Gilmanov, T. B. Le, and F. Sotiropoulos. A numerical approach for simulating fluid structure interaction of flexible thin shells undergoing arbitrarily large deformations in complex domains. *Journal of Computational Physics*, 300:814–843, 2015.

[63] A. Gilmanov and F. Sotiropoulos. A hybrid cartesian/immersed boundary method for simulating flows with 3d, geometrically complex, moving bodies. *Journal of Computational Physics*, 207(2):457–492, 2005.

[64] A. Gilmanov and F. Sotiropoulos. Comparative hemodynamics in an aorta with bicuspid and trileaflet valves. *Theoretical and Computational Fluid Dynamics*, pages 1–19, 2015.

[65] R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Mon. Not. R. astr. Soc.*, 181:375–389, 1977.

[66] D. Goldstein, R. Handler, and L. Sirovich. Modeling a no-slip flow boundary with an external force field. *Journal of Computational Physics*, 105(2):354–366, 1993.

[67] D. B. Goldstein. DNS for new applications of surface textures and MEMS actuators for turbulent boundary layer control - FINAL REPORT. Technical Report AFRL-SR-AR-TR-07-0363, AFSOR, 2006.

[68] D. B. Goldstein. Transient boundary layer disturbance growth and bypass transition due to realistic roughness and continued study of transition over riblets. Technical Report AFRL-OSR-VA-TR-2012-0288, AFSOR, 2011.

[69] D. B. Goldstein and T.-C. Tuan. Secondary flow induced by riblets. *Journal of Fluid Mechanics*, 363:115–151, 1998.

[70] B. E. Griffith. Immersed boundary model of aortic heart valve dynamics with physiological driving and loading conditions. *International Journal for Numerical Methods in Biomedical Engineering*, 28(3):317–345, 2012.

[71] J.-L. Guermond and P. Minev. High-order time stepping for the incompressible Navier–Stokes equations. *SIAM Journal on Scientific Computing*, 37(6):A2656–A2681, 2015.

[72] J. L. Guermond and J. Shen. Velocity-correction projection methods for incompressible flows. *SIAM Journal on Numerical Analysis*, 41(1):112–134, 2003.

[73] P. Hansbo and A. Szepessy. A velocity-pressure streamline diffusion finite element method for the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 84:175–192, 1990.

[74] F. Hartmann. The discrete Babuška-Brezzi condition. *Ingenieur-Archiv*, 56(3):221–228, 1986.

[75] H. Hasimoto. On the flow of a viscous fluid past a thin screen at small Reynolds numbers. *Journal of the Physical Society of Japan*, 13(6):633–639, 1958.

[76] A. Haufe, K. Weimar, and U. Göhner. Advanced airbagsimulation using fluid-structure-interaction and the Eluerian method in LS-DYNA. In *Proceedings of the LS-DYNA Anwenderforum*, 2004.

[77] C. Hesch, A. J. Gil, A. Arranz Carreño, and J. Bonet. On continuum immersed strategies for fluid-structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 247-248:51–64, 2012.

[78] M. R. Hestenes. Multiplier and gradient methods. *Journal of Optimization Theory and Applications*, 4(5):303–320, 1969.

[79] Magnus R. Hestenes and Eduard Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards*, 49(6):409–436, December 1952.

[80] D. K. Hildebrand. Design and evaluation of a novel pulsatile bioreactor for biologically active heart valves. Master's thesis, University of Pittsburgh, Pittsburgh, United States, 2003.

[81] G. A. Holzapfel. *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*. Wiley, Chichester, 2000.

[82] M.-C. Hsu, I. Akkerman, and Y. Bazilevs. High-performance computing of wind turbine aerodynamics using isogeometric analysis. *Computers & Fluids*, 49:93–100, 2011.

[83] M.-C. Hsu, I. Akkerman, and Y. Bazilevs. Wind turbine aerodynamics using ALE–VMS: Validation and the role of weakly enforced boundary conditions. *Computational Mechanics*, 50:499–511, 2012.

[84] M.-C. Hsu, D. Kamensky, Y. Bazilevs, M. S. Sacks, and T. J. R. Hughes. Fluid–structure interaction analysis of bioprosthetic heart valves: signif-

icance of arterial wall deformation. *Computational Mechanics*, 54:1055–1071, 2014.

[85] M.-C. Hsu, D. Kamensky, F. Xu, J. Kiendl, C. Wang, M. C. H. Wu, J. Mineroff, A. Reali, Y. Bazilevs, and M. S. Sacks. Dynamic and fluid–structure interaction simulations of bioprosthetic heart valves using parametric design with T-splines and Fung-type material models. *Computational Mechanics*, pages 1–15, 2015.

[86] M.-C. Hsu, C. Wang, A. J. Herrema, D. Schillinger, A. Ghoshal, and Y. Bazilevs. An interactive geometry modeling and parametric design platform for isogeometric analysis. *Computers & Mathematics with Applications*, 70(7):1481–1500, 2015. High-Order Finite Element and Isogeometric Methods.

[87] M.-C. Hsu, C. Wang, F. Xu, A. J. Herrema, and A. Krishnamurthy. Direct immersogeometric fluid flow analysis using B-rep CAD models. *Computer Aided Geometric Design*, pages –, 2016.

[88] W.-X. Huang, S. J. Shin, and H. J. Sung. Simulation of flexible filaments in a uniform flow by the immersed boundary method. *Journal of Computational Physics*, 226(2):2206–2228, 2007.

[89] W.-X. Huang and H. J. Sung. An immersed boundary method for fluid–flexible structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 198(33–36):2650–2661, 2009.

[90] T. J. R. Hughes, J. A. Cottrell, and Y. Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry, and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194:4135–4195, 2005.

[91] T. J. R. Hughes, G. R. Feijóo, L. Mazzei, and J. B. Quincy. The variational multiscale method–A paradigm for computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 166:3–24, 1998.

[92] T. J. R. Hughes, L. P. Franca, and M. Balestra. A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška–Brezzi condition: A stable Petrov–Galerkin formulation of the Stokes problem accommodating equal-order interpolations. *Computer Methods in Applied Mechanics and Engineering*, 59:85–99, 1986.

[93] T. J. R. Hughes, W. K. Liu, and T. K. Zimmermann. Lagrangian–Eulerian finite element formulation for incompressible viscous flows. *Computer Methods in Applied Mechanics and Engineering*, 29:329–349, 1981.

[94] T. J. R. Hughes, A. Reali, and G. Sangalli. Efficient quadrature for NURBS-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 199:301–313, 2010.

[95] V. L. Huynh, T. Nguyen, H. L. Lam, X. G. Guo, and R. Kafesjian. Cloth-covered stents for tissue heart valves, 2003. US Patent 6,585,766.

[96] A. K. S. Iyengar, H. Sugimoto, D. B. Smith, and M. S. Sacks. Dynamic in vitro quantification of bioprosthetic heart valve leaflet motion using structured light projection. *Annals of Biomedical Engineering*, 29(11):963–973, 2001.

[97] K. E. Jansen, C. H. Whiting, and G. M. Hulbert. A generalized-α method for integrating the filtered Navier-Stokes equations with a stabilized finite element method. *Computer Methods in Applied Mechanics and Engineering*, 190:305–319, 2000.

[98] A. A. Johnson and T. E. Tezduyar. Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces. *Computer Methods in Applied Mechanics and Engineering*, 119:73–94, 1994.

[99] A. A. Johnson and T. E. Tezduyar. 3D simulation of fluid-particle interactions with the number of particles reaching 100. *Computer Methods in Applied Mechanics and Engineering*, 145:301–321, 1997.

[100] A. A. Johnson and T. E. Tezduyar. Parallel computation of incompressible flows with complex geometries. *International Journal for Numerical Methods in Fluids*, 24:1321–1340, 1997.

[101] A. A. Johnson and T. E. Tezduyar. Advanced mesh generation and update methods for 3D flow simulations. *Computational Mechanics*, 23:130–143, 1999.

[102] C. Kadapa, W. G. Dettmer, and D. Perić. A fictitious domain/distributed Lagrange multiplier based fluid–structure interaction scheme with hierarchical B-spline grids. *Computer Methods in Applied Mechanics and Engineering*, 301:1–27, 2016.

[103] B. Kallemov, A. P. S. Bhalla, B. E. Griffith, and A. Donev. An immersed boundary method for rigid bodies. *Communications in Applied Mathematics and Computational Science*, 11(1):79–141, 2016.

[104] D. Kamensky, J. A. Evans, and M.-C. Hsu. Stability and conservation properties of collocated constraints in immersogeometric fluid-thin structure interaction analysis. *Communications in Computational Physics*, 18:1147–1180, 10 2015.

[105] D. Kamensky, M.-C. Hsu, D. Schillinger, J. A. Evans, A. Aggarwal, Y. Bazilevs, M. S. Sacks, and T. J. R. Hughes. An immersogeometric variational framework for fluid–structure interaction: Application to bioprosthetic heart valves. *Computer Methods in Applied Mechanics and Engineering*, 284:1005–1053, 2015.

[106] D. Kamensky, M.-C. Hsu, Y. Yu, J. A. Evans, M. S. Sacks, and T. J. R. Hughes. Immersogeometric cardiovascular fluid–structure interaction analysis using divergence-conforming B-splines. *Computer Methods in Applied Mechanics and Engineering*, 2016. In review. Also appeared as ICES Report 16-14.

[107] T. Kenner. The measurement of blood density and its meaning. *Basic Research in Cardiology*, 84(2):111–124, 1989.

[108] R. Khlebnikov and C. A. Figueroa. CRIMSON: Towards a software environment for patient-specific blood flow simulation for diagnosis and treatment. In C. Oyarzun Laura, R. Shekhar, S. Wesarg, M. A. González Ballester, K. Drechsler, Y. Sato, M. Erdt, and G. M. Linguraru, editors, *Clinical Image-Based Procedures: Translational Research in Medical Imaging*, pages 10–18. Springer International Publishing, Switzerland, 2016.

[109] J. Kiendl. *Isogeometric Analysis and Shape Optimal Design of Shell Structures*. PhD thesis, Lehrstuhl für Statik, Technische Universität München, 2011.

[110] J. Kiendl, K.-U. Bletzinger, J. Linhard, and R. Wüchner. Isogeometric shell analysis with Kirchhoff–Love elements. *Computer Methods in Applied Mechanics and Engineering*, 198:3902–3914, 2009.

[111] J. Kiendl, M.-C. Hsu, M. C. H. Wu, and A. Reali. Isogeometric Kirchhoff–Love shell formulations for general hyperelastic materials. *Computer Methods in Applied Mechanics and Engineering*, 291(0):280–303, 2015.

[112] H. Kim, J. Lu, M. S. Sacks, and K. B. Chandran. Dynamic simulation of bioprosthetic heart valves using a stress resultant shell model. *Annals of Biomedical Engineering*, 36(2):262–275, 2008.

[113] J. Kim, P. Moin, and R. Moser. Turbulence statistics in fully developed channel flow at low Reynolds number. *Journal of Fluid Mechanics*, 177:133–166, 1987.

[114] T. Köppl and B. Wohlmuth. Optimal a priori error estimates for an elliptic problem with Dirac right-hand side. *SIAM Journal on Numerical Analysis*, 52(4):1753–1769, 2014.

[115] J. Korelc. Automation of primal and sensitivity analysis of transient coupled problems. *Computational Mechanics*, 44(5):631–649, 2009.

[116] J. Korelc. *Recent Developments and Innovative Applications in Computational Mechanics*, chapter Automatic Differentiation Based Formulation of Computational Models, pages 79–86. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[117] A. Krishnamurthy, R. Khardekar, S. McMains, K. Haller, and G. Elber. Performing efficient NURBS modeling operations on the GPU. *Visualization and Computer Graphics, IEEE Transactions on*, 15(4):530–543, 2009.

[118] D. Kwak, C. Kiris, and J. Dacles-Mariani. *Sixteenth International Conference on Numerical Methods in Fluid Dynamics: Proceedings of the Conference Held in Arcachon, France, 6 – 10 July 1998*, chapter An assessment of artificial compressibility and pressure projection methods for incompressible flow simulations, pages 177–182. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.

[119] L. Lacouture. Local error analysis for the stokes equations with a singular source term. `https://hal.archives-ouvertes.fr/hal-01230999/document`, 2016.

[120] H. Lan, J. Merkow, A. Updegrove, D. Schiavazzi, N. Wilson, S. Shadden, and A. Marsden. Simvascular 2.0: an integrated open source pipeline for image-based cardiovascular modeling and simulation. In *Proceedings of the 68th Annual Meeting of the APS Division of Fluid Dynamics*. American Physical Society, 2015.

[121] C. Lee. Stability characteristics of the virtual boundary method in three-dimensional applications. *Journal of Computational Physics*, 184(2):559–591, 2003.

[122] J. Li, X. Y. Luo, and Z. B. Kuang. A nonlinear anisotropic model for porcine aortic heart valves. *Journal of Biomechanics*, 34(10):1279–1289, 2001.

[123] S. Lipton, J. A. Evans, Y. Bazilevs, T. Elguedj, and T. J. R. Hughes. Robustness of isogeometric structural discretizations under severe mesh distortion. *Computer Methods in Applied Mechanics and Engineering*, 199:357–373, 2010.

[124] V. B. Makhijani, H. Q. Yang, P. J. Dionne, and M. J. Thubrikar. Three-dimensional coupled fluid–structure simulation of pericardial bioprosthetic aortic valve function. *ASAIO Journal*, 43:M387–M392, 1997.

[125] C. Michler, H. van Brummelen, and R. de Borst. An investigation of interface-GMRES(R) for fluid–structure interaction problems with flutter and divergence. *Computational Mechanics*, 47(1):17–29, 2011.

[126] R. Mittal and G. Iaccarino. Immersed boundary methods. *Annual Review of Fluid Mechanics*, 37:239–261, 2005.

[127] S. Morganti, F. Auricchio, D. J. Benson, F. I. Gambarin, S. Hartmann, T. J. R. Hughes, and A. Reali. Patient-specific isogeometric structural analysis of aortic valve closure. *Computer Methods in Applied Mechanics and Engineering*, 284:508–520, 2015.

[128] N. Nguyen-Thanh, J. Kiendl, H. Nguyen-Xuan, R. Wüchner, K.U. Bletzinger, Y. Bazilevs, and T. Rabczuk. Rotation-free isogeometric thin shell analysis using PHT-splines. *Computer Methods in Applied Mechanics and Engineering*, 200:3410–3424, 2011.

[129] J. Nitsche. Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 36:9–15, 1971.

[130] J. A. Nitsche and A. H. Schatz. Interior estimates for Ritz–Galerkin methods. *Mathematics of Computation*, 28(128):937–958, 1974.

[131] C. S. Peskin. Flow patterns around heart valves: A numerical method. *Journal of Computational Physics*, 10(2):252–271, 1972.

[132] C. S. Peskin and B. F. Printz. Improved volume conservation in the computation of flows with immersed elastic boundaries. *Journal of Computational Physics*, 105(1):33–46, 1993.

[133] P. Pibarot and J. G. Dumesnil. Prosthetic heart valves: selection of the optimal prosthesis and long-term management. *Circulation*, 119(7):1034–1048, 2009.

[134] L. Piegl and W. Tiller. *The NURBS Book (Monographs in Visual Communication), 2nd ed.* Springer-Verlag, New York, 1997.

[135] M. J. D. Powell. A method for nonlinear constraints in minimization problems. In R. Fletcher, editor, *Optimization*, pages 283–298. Academic Press, New York, 1969.

[136] I. Ramière. Convergence analysis of the Q1-finite element method for elliptic problems with non-boundary-fitted meshes. *International Journal for Numerical Methods in Engineering*, 75(9):1007–1052, 2008.

[137] E. Rank, M. Ruess, S. Kollmannsberger, D. Schillinger, and A. Düster. Geometric modeling, isogeometric analysis and the finite cell method. *Computer Methods in Applied Mechanics and Engineering*, 249-252:104–115, 2012.

[138] S. E. Rogers, D. Kwak, and U. Kaul. On the accuracy of the pseudocompressibility method in solving the incompressible Navier-Stokes equations. *Applied Mathematical Modelling*, 11(1):35–44, 1987.

[139] R. Rosencranz and S. A. Bogen. Clinical laboratory measurement of serum, plasma, and blood viscosity. *American Journal of Clinical Pathology*, 125:S78–S86, 2006.

[140] M. Ruess, D. Schillinger, Y. Bazilevs, V. Varduhn, and E. Rank. Weakly enforced essential boundary conditions for NURBS-embedded and trimmed NURBS geometries on the basis of the finite cell method. *International Journal for Numerical Methods in Engineering*, 95:811–846, 2013.

[141] M. Ruess, D. Schillinger, A. I. Özcan, and E. Rank. Weak coupling for isogeometric analysis of non-matching and trimmed multi-patch geometries. *Computer Methods in Applied Mechanics and Engineering*, 269:46–731, 2014.

[142] R. Rukmani and R. Usha. Arbitrary squeeze flow between two disks. *International Journal of Mathematics and Mathematical Sciences*, 17:779–782, 1994.

[143] J. Ryu, S. G. Park, B. Kim, and H. J. Sung. Flapping dynamics of an inverted flag in a uniform flow. *Journal of Fluids and Structures*, 57:159–169, 2015.

[144] Y. Saad and M. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal of Scientific and Statistical Computing*, 7:856–869, 1986.

[145] E. M. Saiki and S. Biringen. Numerical simulation of a cylinder in uniform flow: Application of a virtual boundary method. *Journal of Computational Physics*, 123(2):450–465, 1996.

[146] E. M. Saiki and S. Biringen. Spatial numerical simulation of boundary layer transition: effects of a spherical particle. *Journal of Fluid Mechanics*, 345:133–164, 1997.

[147] Roger A. Sauer and Laura De Lorenzis. A computational contact formulation based on surface potentials. *Computer Methods in Applied Mechanics and Engineering*, 253(0):369–395, 2013.

[148] D. Schillinger, L. Dedè, M. A. Scott, J. A. Evans, M. J. Borden, E. Rank, and T. J. R. Hughes. An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline CAD surfaces. *Computer Methods in Applied Mechanics and Engineering*, 249–252:116–150, 2012.

[149] D. Schillinger, I. Harari, M.-C. Hsu, D. Kamensky, K.F.S. Stoter, Y. Yu, and Y. Zhao. The non-symmetric Nitsche method for the parameter-free coupling imposition of weak boundary and coupling conditions in immersed finite elements. *Computer Methods in Applied Mechanics and Engineering*, 2016. Submitted.

[150] D. Schillinger and M. Ruess. The Finite Cell Method: A review in the context of higher-order structural analysis of CAD and image-based

geometric models. *Archives of Computational Methods in Engineering*, 2014. `http://dx.doi.org/10.1007/s11831-014-9115-y`.

[151] D. Schillinger, M. Ruess, N. Zander, Y. Bazilevs, A. Düster, and E. Rank. Small and large deformation analysis with the p- and B-spline versions of the Finite Cell Method. *Computational Mechanics*, 50(4):445–478, 2012.

[152] F. J. Schoen and R. J. Levy. Calcification of tissue heart valve substitutes: progress toward understanding and prevention. *Ann. Thorac. Surg.*, 79(3):1072–1080, 2005.

[153] M. A. Scott. *T-splines as a Design-Through-Analysis Technology*. PhD thesis, The University of Texas at Austin, August 2011.

[154] T. W. Sederberg, D.L. Cardon, G.T. Finnigan, N.S. North, J. Zheng, and T. Lyche. T-spline simplification and local refinement. *ACM Transactions on Graphics*, 23(3):276–283, 2004.

[155] T.W. Sederberg, J. Zheng, A. Bakenov, and A. Nasri. T-splines and T-NURCCS. *ACM Transactions on Graphics*, 22(3):477–484, 2003.

[156] J. Shen. On error estimates of the penalty method for unsteady Navier–Stokes equations. *SIAM Journal on Numerical Analysis*, 32(2):386–403, 1995.

[157] S. J. Shin, W.-X. Huang, and H. J. Sung. Assessment of regularized delta functions and feedback forcing schemes for an immersed bound-

ary method. *International Journal for Numerical Methods in Fluids*, 58(3):263–286, 2008.

[158] R. F. Siddiqui, J. R. Abraham, and J. Butany. Bioprosthetic heart valves: modes of failure. *Histopathology*, 55:135–144, 2009.

[159] J. C. Simo, P. Wriggers, and R. L. Taylor. A perturbed Lagrangian formulation for the finite element solution of contact problems. *Computer Methods in Applied Mechanics and Engineering*, 50(2):163–180, 1985.

[160] F. Sotiropoulos and X. Yang. Immersed boundary methods for simulating fluid–structure interaction. *Progress in Aerospace Sciences*, 65:1–21, 2014.

[161] M. Souli, N. Capron, and U. Khan. Fluid structure interaction and airbag ALE for out of position. In *Proceedings of the ASME Pressure Vessels and Piping Conference*. AMSE, 2005.

[162] M. Souli, Y. Sofiane, and L. Olovsson. ALE and fluid/structure interaction in LS-DYNA. In *Proceedings of Emerging Technology in Fluids, Structures, and Fluid–Structure Interactions*. ASME, 2004.

[163] M. Souli, J. Wang, I. Do, and C. Hao. ALE and fluid structure interaction in LS-DYNA. In *Proceedings of the 8th International LS-DYNA Users Conference*, 2011.

[164] A. Stavrev. The role of higher-order geometry approximation and accurate quadrature in nurbs based immersed boundary methods. Master's thesis, Technische Universität München, Munich, Germany, 2012.

[165] K. Stein, T. Tezduyar, and R. Benney. Mesh moving techniques for fluid–structure interactions with large displacements. *Journal of Applied Mechanics*, 70:58–63, 2003.

[166] K. Stein, T. E. Tezduyar, and R. Benney. Automatic mesh update with the solid-extension mesh moving technique. *Computer Methods in Applied Mechanics and Engineering*, 193:2019–2032, 2004.

[167] J. S. Strand and D. B. Goldstein. Direct numerical simulations of riblets to constrain the growth of turbulent spots. *Journal of Fluid Mechanics*, 668:267–292, 2011.

[168] G. Strang and G. J. Fix. *An Analysis of the Finite Element Method*. Prentice-Hall, Englewood Cliffs, NJ, 1973.

[169] F. Sturla, E. Votta, M. Stevanella, C. A. Conti, and A. Redaelli. Impact of modeling fluid–structure interaction in the computational analysis of aortic root biomechanics. *Medical Engineering and Physics*, 35:1721–1730, 2013.

[170] W. Sun, A. Abad, and M. S. Sacks. Simulated bioprosthetic heart valve deformation under quasi-static loading. *Journal of Biomechanical Engineering*, 127(6):905–914, 2005.

[171] K. Takizawa, C. Moorman, S. Wright, J. Christopher, and T. E. Tezduyar. Wall shear stress calculations in space–time finite element computation of arterial fluid–structure interactions. *Computational Mechanics*, 46:31–41, 2010.

[172] K. Takizawa, T. E. Tezduyar, A. Buscher, and S. Asada. Space–time interface-tracking with topology change (ST-TC). *Computational Mechanics*, 2013. doi:10.1007/s00466-013-0935-7.

[173] Kenji Takizawa, TayfunE. Tezduyar, Austin Buscher, and Shohei Asada. Spacetime fluid mechanics computation of heart valve models. *Computational Mechanics*, pages 1–14, 2014.

[174] C. A. Taylor, T. J. R. Hughes, and C. K. Zarins. Finite element modeling of blood flow in arteries. *Computer Methods in Applied Mechanics and Engineering*, 158:155–196, 1998.

[175] Adrián Buganza Tepole, Hardik Kabaria, Kai-Uwe Bletzinger, and Ellen Kuhl. Isogeometric Kirchhoff–Love shell formulations for biological membranes. *Computer Methods in Applied Mechanics and Engineering*, (0):–, 2015.

[176] T. Tezduyar, S. Aliabadi, M. Behr, A. Johnson, and S. Mittal. Parallel finite-element computation of 3D flows. *Computer*, 26(10):27–36, 1993.

[177] T. Tezduyar, S. Aliabadi, M. Behr, A. Johnson, and S. Mittal. Massively parallel finite element computation of 3D flows – mesh update strate-

gies in computation of moving boundaries and interfaces. In A. Ecer, J. Hauser, P. Leca, and J. Periaux, editors, *Parallel Computational Fluid Dynamics – New Trends and Advances*, pages 21–30. Elsevier, 1995.

[178] T. E. Tezduyar. Computation of moving boundaries and interfaces and stabilization parameters. *International Journal for Numerical Methods in Fluids*, 43:555–575, 2003.

[179] T. E. Tezduyar, M. Behr, and J. Liou. A new strategy for finite element computations involving moving boundaries and interfaces – the deforming-spatial-domain/space–time procedure: I. The concept and the preliminary numerical tests. *Computer Methods in Applied Mechanics and Engineering*, 94(3):339–351, 1992.

[180] T. E. Tezduyar, M. Behr, S. Mittal, and J. Liou. A new strategy for finite element computations involving moving boundaries and interfaces – the deforming-spatial-domain/space–time procedure: II. Computation of free-surface flows, two-liquid flows, and flows with drifting cylinders. *Computer Methods in Applied Mechanics and Engineering*, 94(3):353–371, 1992.

[181] T. E. Tezduyar, S. Sathe, R. Keedy, and K. Stein. Space–time finite element techniques for computation of fluid–structure interactions. *Computer Methods in Applied Mechanics and Engineering*, 195:2002–2027, 2006.

[182] M. J. Thubrikar, J. D. Deck, J. Aouad, and S. P. Nolan. Role of mechanical stress in calcification of aortic bioprosthetic valves. *J. Thorac. Cardiovasc. Surg.*, 86(1):115–125, Jul 1983.

[183] M. Toma, M. Ø. Jensen, D. R. Einstein, A. P. Yoganathan, R. P. Cochran, and K. S. Kunzelman. Fluid–structure interaction analysis of papillary muscle forces using a comprehensive mitral valve model with 3D chordal structure. *Annals of Biomedical Engineering*, pages 1–12, 2015.

[184] T.-C. Tuan and D. B. Goldstein. Direct numerical simulation of arrays of microjets to manipulate near wall turbulence. Technical Report CAR-96-3, U. T. Austin Center for Aerodynamics Research, 2011.

[185] E. Uddin, W.-X. Huang, and H. J. Sung. Actively flapping tandem flexible flags in a viscous flow. *Journal of Fluid Mechanics*, 780:120–142, 10 2015.

[186] J. B. Uther, K. L. Peterson, R. Shabetai, and E. Braunwald. Measurement of ascending aortic flow patterns in man. *Journal of Applied Physiology*, 34(4):513–518, 1973.

[187] H. Uzawa and K. J. Arrow. Iterative methods for concave programming. In *Preference, production, and capital*, pages 135–148. Cambridge University Press, 1989. Cambridge Books Online.

[188] E. H. van Brummelen. Added mass effects of compressible and incompressible flows in fluid–structure interaction. *Journal of Applied Mechanics*, 76:021206, 2009.

[189] R. van Loon. *A 3D method for modelling the fluid–structure interaction of heart valves*. Ph.D. thesis, Technische Universiteit Eindhoven, Eindhoven, Netherlands, 2005.

[190] R. van Loon. Towards computational modelling of aortic stenosis. *International Journal for Numerical Methods in Biomedical Engineering*, 26:405–420, 2010.

[191] R. van Loon, P. D. Anderson, and F. N. van de Vosse. A fluid–structure interaction method with solid-rigid contact for heart valve dynamics. *Journal of Computational Physics*, 217:806–823, 2006.

[192] T. M. van Opstal. Finite-element/boundary-element coupling for inflatables: effective contact resolution. *ArXiv e-prints*, June 2015.

[193] T. M. van Opstal, J. Yan, C. Coley, J. A. Evans, T. Kvamsdal, and Y. Bazilevs. Isogeometric divergence-conforming variational multiscale formulation of incompressible turbulent flows. *Computer Methods in Applied Mechanics and Engineering*, 2016. Submitted.

[194] L. B. Wahlbin. Local behavior in finite element methods. In P. G. Ciarlet and J. L. Lions, editors, *Finite Element Methods (Part 1)*, volume 2 of *Handbook of Numerical Analysis*, pages 353–522. North-Holland, 1991.

[195] C. Wang, M. C. H. Wu, F. Xu, M.-C. Hsu, and Y. Bazilevs. Modeling of a hydraulic arresting gear using fluid-structure interaction and isogeometric analysis. *Computers & Fluids*, 2016. `http://dx.doi.org/10.1016/j.compfluid.2015.12.004`.

[196] N. Westerhof, J.-W. Lankhaar, and B. E. Westerhof. The arterial Windkessel. *Medical & Biological Engineering & Computing*, 47:131–141, 2009.

[197] T. Wick. Flapping and contact FSI computations with the fluid–solid interface-tracking/interface-capturing technique and mesh adaptivity. *Computational Mechanics*, 53(1):29–43, 2014.

[198] M. C. H. Wu, D. Kamensky, C. Wang, A. J. Herrema, F. Xu, M. S. Pigazzini, A. Verma, A. L. Marsden, Y. Bazilevs, and M.-C. Hsu. Optimizing fluid–structure interaction systems with immersogeometric analysis and surrogate modeling: Application to a hydraulic arresting gear. *Computer Methods in Applied Mechanics and Engineering*, 2016. In review.

[199] W. Wu, D. Pott, B. Mazza, T. Sironi, E. Dordoni, C. Chiastra, L. Petrini, G. Pennati, G. Dubini, U. Steinseifer, S. Sonntag, M. Kuetting, and F. Migliavacca. Fluid-structure interaction model of a percutaneous aortic valve: Comparison with an in vitro test and feasibility study in a patient-specific case. *Ann Biomed Eng*, Aug 2015.

[200] F. Xu, D. Schillinger, D. Kamensky, V. Varduhn, C. Wang, and M.-C. Hsu. The tetrahedral finite cell method for fluids: Immersogeometric analysis of turbulent flow around complex geometries. *Computers & Fluids*, pages –, 2015.

[201] C. H. Yap, N. Saikrishnan, G. Tamilselvan, and A. P. Yoganathan. Experimental technique of measuring dynamic fluid shear stress on the aortic surface of the aortic valve leaflet. *Journal of Biomechanical Engineering*, 133(6):061007, 2011.

[202] C. K. Zarins, C. A. Taylor, and J. K. Min. Computed fractional flow reserve (FFTCT) derived from coronary CT angiography. *Journal of Cardiovascular Translational Research*, 6(5):708–714, 2013.