Modelling Flows of Complex Fluids using the Immersed Boundary Method

Christopher Frederick Rowlatt
Supervised by Prof. T. N. Phillips

A thesis submitted for the degree of Doctor of Philosophy

20th August 2014

School of Mathematics
Cardiff University

DECLARATION

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any other degree.

Signed .....
Date .....

STATEMENT 1

This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged and explicit references given. A reference section is appended.

Signed .....
Date .....

STATEMENT 2

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed .....
Date .....

Dedication

I would like to dedicate my thesis to my Father, who unfortunately passed away during the completion of this work. My Dad has been a great influence on me during my life and has always supported everything I chose to do. He was always there, never missing an event such a sports match and he has been an incredible role model. He will be missed, not just by myself but by my entire family.

Acknowledgements

First of all I would like to thank my supervisor, Prof. Tim Phillips, for his excellent guidence, understanding, support and patience during the completion of this work. I would also like to thank him for the support and understanding he gave during the difficult period when my Dad passed away. I genuinely cannot thank you enough for everything you have done over the last few years. It has been a pleasure to work with you and I hope to be able to do so again in the future. Also, I would like to thank EPSRC for funding my PhD.

I would like to thank my examiners Prof. Simon Cox and Dr. Pierre Kerfriden for agreeing to take part in my viva and for their valuable feedback.

I would like to thank all of the members of the Applied Maths group at Cardiff for the friendly atmosphere and stimulating research environment. I would also like to thank all of the postgraduate students, past and present. In particular, I would like to thank Ross and Susanne for their helpful advice and the discussions we had about our work during the past few years. I would also like to thank all the other friends I have made over the last few years, Angelico, Leanne, Tom, Lizzie, Mike, Penny, Lisa, Julie, Iza, Dan, Jon and Cheryl, for all of the nights out, Eurovision parties and more.

I would like to thank all my friends from undergraduate - Nathan, Ciaran, Deano, Pete, Big Andy, Little Andy and Simon. Also, I would like to thank my friends from home - Alex, Rich, Parf, Peewee, James and Mandy for being such good friends over the many years we have known each other.

Finally, I would like to thank my family for all the support you have given me over the last few years. In particular, I would like to thank my Mum and my Sister for everything you have done throughout my life and for your unwavering support and belief.

Summary

In this thesis we employ the Immersed Boundary Method (IBM) to model the fluid-structure interaction problems involving elastic membranes and shells immersed in Newtonian and Oldroyd-B fluids. The IBM considers the immersed structure to be part of the fluid where the action of the fibre on the fluid is modelled as a force distribution concentrated around the immersed structure. The IBM has the advantage that only a single equation of motion needs to be solved and that both the fluid and the immersed structure can be dealt with in their natural frameworks. In the IBM, the immersed structure is automatically tracked and therefore costly re-meshing is avoided.

The IBM does have some disadvantages: the assumption of fibrous materials and constant viscosity. In this thesis, we consider the Finite Element Immersed Boundary Method (FE-IBM). This method can deal with thick immersed structures. However, it still has the disadvantage of constant viscosity. In this thesis we present an extension of the FE-IBM to allow for discontinuous viscosity for the first time. We also develop a high-order formulation of the FE-IBM, which we call the Spectral Element Immersed Boundary Method (SE-IBM).

In the case of an immersed membrane, the pressure is discontinuous across the membrane. As the discontinuity is unfitted to the fluid mesh, Gibbs phenomenon can be seen local to the discontinuity. We develop a spectral element equivalent of the $h-p$ eXtended Finite Element Method, which we term the eXtended Spectral Element Method (XSEM). The XSEM removes some of the oscillations caused by Gibbs phenomenon. We derive spectral element error estimates and discuss the inf-sup condition.

These techniques are applied to a static and oscillating closed membrane and a static and oscillating closed shell. The convergence of the approximations is studied in the static cases. Excellent convergence behaviour is obtained and compared with the results in the literature for FE-IBM.

Contents

Chapter 1

Introduction

In this thesis we consider the motion of an elastic membrane and elastic shell immersed in a Newtonian and an Oldroyd-B fluid. To model the fluid-structure interaction, we employ the Immersed Boundary Method (IBM) proposed by Charles Peskin in 1972 [56]. The IBM is both a mathematical formulation and a numerical scheme for fluid-structure interaction problems. In this chapter, we briefly introduce the IBM and a few extensions. We also briefly introduce the eXtended Spectral Element Method (XSEM).

This thesis is constructed as follows: Chapter 2 introduces a few basic ideas from continuum mechanics as well as the field equations and constitutive equations required for the modelling of fluid flow. Chapter 3 is concerned with the immersed boundary method. We discuss three key articles by Charles Peskin which introduce the ideas involved in the immersed boundary method. We also discuss the two most prominent extensions; namely the Immersed Finite Element Method (IFEM) proposed by Zhang *et al.*. [73] and the Finite Element Immersed Boundary Method (FE-IBM) proposed by Boffi and Gastaldi [14]. Then we discuss our proposed extension of the FE-IBM to allow for discontinuous viscosity. Finally, we discuss the application of the immersed boundary method to viscoelastic fluids. Chapter 4 discusses the temporal discretisations used in this thesis and gives a brief numerical investigation into the temporal stability of the immersed boundary method. Chapter 5 discusses the spatial discretisation used in this thesis, namely the Spectral Element Method and the eXtended Spectral Element Method (XSEM). We also discuss the spectral element equivalent of

the error estimates proposed by Reusken [60] and the inf-sup condition. Finally we discuss the area conservation of the immersed boundary method. In Chapter 6 we briefly investigate the properties of XSEM numerically by considering the approximation of a discontinuous function. We also discuss how we handle the quadrature of the enriched approximation in this thesis. Chapter 7 is devoted to validation for the Newtonian and Oldroyd-B solvers as well as the immersed boundary method. Chapter 8 contains results for the SE-IBM applied to a Newtonian fluid and Chapter 9 contains results for the IBM applied to an Oldroyd-B fluid. Finally, Chapter 10 is devoted to the conclusions and future work.

All of the examples considered in this thesis are in a two-dimensional setting. Any two-dimensional plots were made using the Sage Mathematical Software System (http://www.sagemath.org/) and any three-dimensional plots, as well as velocity vector plots, were made using MATLAB. Sage is an open-source equivalent of MATLAB, Maple, Mathematica and more.

1.1 Motivation

In the beginning, we were concerned with the mathematical modelling of blood flow. We quickly realised that we would like to be able to model the interaction between the blood and a surrounding vessel wall, where typically the vessel is a small blood vessel such as a capillary so that blood becomes viscoelastic. This led us to consider the immersed boundary method. The immersed boundary method considers the immersed structure to be part of the fluid and therefore only a single equation of motion needs to be solved. The use of higher-order methods within the immersed boundary method literature is rare and indeed, as far as we are aware, the spectral element method has never been considered before. Additionally, the literature on the application of the immersed boundary method to viscoelastic fluids is again quite sparse. This motivated us to apply the spectral element method to the immersed boundary method and apply the IBM to an Oldroyd-B fluid.

1.2 Immersed Boundary Method

The immersed boundary method (IBM) was proposed by Charles Peskin when studying flow patterns around heart valves [56]. Since then, it has been applied to a wide range of problems. In a classical fluid-structure interaction problem, the fluid and the structure are considered separately and then coupled together via some suitable jump conditions. In the IBM, the structure - which is usually immersed in a Newtonian fluid - is thought of as being part of the surrounding fluid. This means that only a single equation of motion needs to be solved. Additionally, in a classical context, the fluid and structure meshes are fitted to the interface between the two domains and as both domains are in general time-dependent, this entails re-meshing which can be costly computationally. The IBM allows the immersed structure to move freely over the underlying fluid mesh, which alleviates the need for re-meshing.

The IBM essentially replaces the immersed structure with a force distribution. A Lagrangian force density is *spread* to the underlying fluid using the Dirac delta distribution. The immersed structure is automatically tracked in an *interpolation* phase, where the local fluid velocity is interpolated onto the immersed structure using the Dirac delta distribution. For numerical computations, a smoothed approximation of the delta distribution is required and the same approximation must be used for both the spreading and the interpolation phases.

The original IBM proposed by Peskin [56], has a few limitations. Firstly the assumption that the immersed structure is fibrous and secondly the assumption of constant viscosity throughout the computational domain. While the first assumption may be physically realistic in certain cases, the second assumption is in general not desirable. The immersed finite element method (IFEM) proposed by Zhang *et al.* [73] used finite elements for both the fluid and the immersed structure. Using finite elements for the structure alleviates the first assumption in the original IBM and allows for a more physically realistic representation of a thick immersed structure. Additionally, IFEM used the Reproducing Kernel Particle Method (RKPM) to construct an approximation to the Dirac delta distribution. Another method which alleviates the first assumption

of the original IBM, and is the method we use in this thesis, was proposed by Boffi and Gastaldi [14] and is called the finite element immersed boundary method (FE-IBM). Like IFEM, the FE-IBM uses finite elements for both the fluid and the immersed structure, however the key difference between the two methods is that FE-IBM does not numerically approximate the Dirac delta distribution. Instead, the interaction is governed using the delta distribution's action on a test function, i.e. in the weak formulation, the sifting property of the delta function is used.

Both the IFEM and FE-IBM, still have the limitation of constant viscosity throughout the computational domain. In this thesis, we propose an extension to FE-IBM which allows for discontinuous viscosity. We also apply a high-order method to the FE-IBM, which we call the spectral element immersed boundary method (SE-IBM).

1.3 eXtended Spectral Element Method

In the immersed boundary literature, one of the simplest examples which is considered often is the case of a closed elastic membrane immersed in a Newtonian fluid. In this thesis, we also apply this example to an Oldroyd-B fluid. In such an example, the pressure is discontinuous across the membrane which is unfitted to the fluid mesh. The spectral element approximation of the pressure therefore introduces oscillations local to the membrane. This is known as Gibbs phenomenon. In this thesis, we apply a spectral element version of the extended finite element method (XFEM) which we call the eXtended Spectral Element Method (XSEM).

The XFEM was proposed by Moës *et al.* [50] to handle crack growth without remeshing and was later extended to arbitrary discontinuities by Belytschko *et al.* [6]. The method has proved to be very popular in the engineering community. The XFEM was based on the partition of unity method of Babuska and Melenk [4]. Recently XFEM has been applied to more complicated finite element situations, such as space-time finite elements [27] and convection dominated problems [1].

Spectral finite elements were first considered in the XFEM by Legay *et al.* [42] and later revisited by Cheng and Fries [26]. As far as we are aware, they are the only articles to consider spectral basis functions within the XFEM. Both articles were concerned with strong and weak discontinuities (a strong discontinuity is a discontinuity in the function itself, whereas a weak discontinuity is a discontinuity in the gradient of the function). However, the maximum degree of the polynomials used in those articles was four and the convergence analysis was performed with respect to mesh width rather than polynomial degree. In this thesis, we consider the p-type convergence of XSEM and apply it to the SE-IBM. As far as we are aware, the use of XFEM-type methods has not been considered in the IBM literature. The closest to an enriched method is the work of Boffi *et al.* [11, 12] where they added a constant to the local finite element approximation to improve the pressure solution.

In this thesis, we also apply the XSEM with an immersed membrane to an Oldroyd-B fluid. Choi *et al.* [29] applied XFEM to a viscoelastic fluid flowing past a cylinder. As far as we are aware, they are the only authors to have considered applying a XFEM-type method to viscoelastic flows. Note that throughout this thesis, the order of convergence with respect to mesh width (so-called h-type convergence) of a variable is calculated using:

$$\text{order} = -\frac{\ln\left(E\left(\frac{h}{2}\right)/E(h)\right)}{\ln(2)} \tag{1.3.1}$$

where $E(\cdot)$ is the approximation error (e.g. $E(h) = \|u - u_h\|_{L^2}$). Additionally, the order of convergence with respect to polynomial degree (so-called p-type convergence) of a variable is calculated using:

$$\text{order} = -\frac{\ln\left(E\left(2N\right)/E(N)\right)}{\ln(2)} \tag{1.3.2}$$

where $E(\cdot)$ is the approximation error (e.g. $E(N) = \|u - u_N\|_{L^2}$).

Chapter 2

Continuum Mechanics and Governing Equations

This chapter is concerned with introducing the equations which describe the flow of some material. In section §2.1 we introduce a few ideas from continuum mechanics. In section §2.2 we discuss the equations which govern the flow of a material and introduce the constitutive models used in this thesis.

2.1 Brief Introduction to Continuum Mechanics

In this section we introduce a few of the basic ideas used in continuum mechanics. Everything written in this section can be found in the book by Chadwick [25]. Following Chadwick [25] we define a body B_r to be a set of *particles* where each element of the set (or particle) can be related to points in another set B via a bijective map. The set B is known as a configuration of B_r. Thus, there must exist a function $\boldsymbol{\theta}\colon B_r \to B$ and its inverse $\boldsymbol{\Theta}\colon B \to B_r$ such that:

$$\boldsymbol{x} = \theta(\boldsymbol{s}), \qquad \boldsymbol{s} = \Theta(\boldsymbol{x}) \tag{2.1.1}$$

where we have used $\boldsymbol{x}$ to denote an element of B and $\boldsymbol{s}$ to denote an element of B_r. In general, a motion of a body causes a change in configuration with time. Throughout

this thesis, time will be denoted by the variable $t \in [0, T)$ where $T \in \mathbb{R} \cup \{+\infty\}$. Therefore, following Chadwick [25], a *motion* of B_r can be defined as the set of configurations $\{B_t \colon t \in [0, T)\}$. Thus we can define the map $\boldsymbol{X} \colon B_r \times [0, T) \to B_t$ such that

$$\boldsymbol{x} = \boldsymbol{X}(\boldsymbol{s}, t) \tag{2.1.2}$$

This implies that $B_t = \boldsymbol{X}(B_r, t)$, for any given $t \in [0, T)$. We notice, that if we fix $\boldsymbol{s}$ so that we are only *looking* at a specific particle, the particle will occupy a sequence of points which form a curve. This curve is known as a *particle path*. If we let

$$\tilde{\boldsymbol{x}}(t) = \boldsymbol{X}(\boldsymbol{s}, t)|_{\text{fixed } \boldsymbol{s}}$$

denote a given particle path in time $t \in [0, T)$ then we may define the velocity $\boldsymbol{u}$ and acceleration $\boldsymbol{a}$ of this particle as:

$$\boldsymbol{u}(\tilde{\boldsymbol{x}}(t), t) = \frac{d\tilde{\boldsymbol{x}}}{dt} \tag{2.1.3a}$$

$$\boldsymbol{a}(\tilde{\boldsymbol{x}}(t), t) = \frac{d\boldsymbol{u}(\tilde{\boldsymbol{x}}(t), t)}{dt} = \frac{d^2\tilde{\boldsymbol{x}}}{dt^2} \tag{2.1.3b}$$

Chadwick [25], makes the important distinction that the reference configuration is *not* the same as the set of *particles*. If $\mathcal{B}$ denotes the set of particles, then the elements of the set are merely labels used to denote a specific particle. A reference configuration, B_r, is then used to denote the *position* of these particles in Euclidean space. This implies the introduction of an additional map, $\boldsymbol{\phi} \colon \mathcal{B} \to B_r$. For simplicity, we have not made this distinction here and therefore assumed that $\boldsymbol{\phi}$ is the identity map.

Throughout this thesis, the variable $\boldsymbol{s}$ will be used to denote the position of a particle in reference configuration B_r, whilst the variable $\boldsymbol{x}$ will be used to denote the position of a particle in current (or spatial) configuration B_t, at a time $t \in [0, T)$. In this thesis, we freely interchange the terms reference and current with Lagrangian and Eulerian as the terms are synonymous. Following the definitions of the velocity and acceleration of a particle path, we can introduce the material derivative of a function ϕ as the rate

of change of ϕ with time following the motion of a fixed particle:

$$\frac{d\phi(\tilde{\boldsymbol{x}}(t),t)}{dt} = \frac{\partial\phi}{\partial t} + \frac{d\tilde{x}}{dt}\frac{\partial\phi}{\partial x} + \frac{d\tilde{y}}{dt}\frac{\partial\phi}{\partial y} + \frac{d\tilde{z}}{dt}\frac{\partial\phi}{\partial z} \tag{2.1.4}$$

$$= \frac{\partial\phi}{\partial t} + (\boldsymbol{u}(\tilde{\boldsymbol{x}}(t),t)\cdot\nabla)\,\phi = \left.\frac{D\phi}{Dt}\right|_{\boldsymbol{x}=\tilde{\boldsymbol{x}}(t)} \tag{2.1.5}$$

where we have used the chain rule and $d\tilde{x}/dt = u$, $d\tilde{y}/dt = v$, $d\tilde{z}/dt = w$, for $\tilde{\boldsymbol{x}} = (\tilde{x},\tilde{y},\tilde{z})$ and $\boldsymbol{u} = (u,v,w)$. The material derivative is usually denoted by $D(\cdot)/Dt = \partial(\cdot)/\partial t + (\boldsymbol{u}\cdot\nabla)(\cdot)$. Following Chadwick [25] we define a deformation to be a mapping from reference configuration to a current, deformed configuration. Therefore, we may define the deformation gradient as the rate of change of current configuration with respect to a reference configuration. Thus, the deformation gradient is defined as:

$$\mathbb{F}(\boldsymbol{s},t) = \nabla_{\boldsymbol{s}}\boldsymbol{X}(\boldsymbol{s},t) \qquad \Rightarrow \qquad \mathbb{F}_{i\alpha}(\boldsymbol{s},t) = \frac{\partial \boldsymbol{X}_i(\boldsymbol{s},t)}{\partial s_\alpha} \tag{2.1.6}$$

An infinitesimal arc $\mathrm{d}\boldsymbol{x}$ in current configuration is related to an arc $\mathrm{d}\boldsymbol{s}$ in reference configuration via: $\mathrm{d}\boldsymbol{x} = \mathbb{F}\mathrm{d}\boldsymbol{s}$. Similarly, for an infinitesimal area we have: $\boldsymbol{n}\mathrm{d}a = J\mathbb{F}^{-T}\boldsymbol{N}\mathrm{d}A$, where $\boldsymbol{n}$ and $\boldsymbol{N}$ are unit vectors normal to a surface element S_t and S_r, respectively, $J = \det(\mathbb{F})$ is the Jacobian of the transformation and $\mathbb{F}^{-T}$ is the inverse transpose of the tensor $\mathbb{F}$. Similarly, for an infinitesimal volume $\mathrm{d}v$ we have: $\mathrm{d}v = J\mathrm{d}V$. Throughout this thesis, the velocity gradient and divergence of a tensor $\boldsymbol{T}$ will be defined as:

$$(\nabla\boldsymbol{u})_{ij} = \frac{\partial u_i}{\partial x_j}, \qquad (\nabla\cdot\boldsymbol{T})_k = \sum_{i=1}^{d}\frac{\partial T_{ik}}{\partial x_i} \tag{2.1.7}$$

where d is the dimensionality of the problem. In this thesis, we are only concerned with 2D problems, therefore $d = 2$.

2.2 Field and Constitutive Equations

The governing equations which describe the flow of any material are based upon three simple principles known as conservation laws. These conservation laws are:

- Conservation of mass
- Conservation of momentum
- Conservation of energy

In this thesis we are primarily concerned with flows under isothermal conditions, therefore we do not consider the conservation of energy. The conservation of mass and momentum are known as the field equations and are discussed in section §2.2.1. The field equations hold for any material but they do not form a closed system. To close the system a relationship between an applied deformation (or strain) to a material and the resulting load (or stress) on the material is required. The mathematical description of the relationship between the stress and strain is known as a constitutive equation and the models used in this thesis are introduced in section §2.2.2.

2.2.1 Field Equations

Conservation of Mass

The conservation of mass is the mathematical statement that no mass can be created or destroyed during the flow of a material. Given a volume of material $V(t)$ at time t, the principle of conservation of mass states that the rate of change of the mass of the material occupying $V(t)$ should be zero, viz:

$$\frac{d}{dt}\int_{V(t)} \rho \, \mathrm{d}V = 0 \tag{2.2.1}$$

where ρ is the density of the material. Applying Reynolds transport theorem (see, for example, Owens and Phillips [54] - Appendix B.4.1) to the above gives:

$$\int_{V(t)} \left(\frac{D\rho}{Dt} + \rho \nabla \cdot \boldsymbol{u} \right) \, \mathrm{d}V = 0 \tag{2.2.2}$$

where $\boldsymbol{u}$ is the velocity field and the operator $D(\cdot)/Dt$ is the material derivative and is given by

$$\frac{D\boldsymbol{w}}{Dt} = \frac{\partial \boldsymbol{w}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{w} \tag{2.2.3}$$

The material derivative can be thought of as the rate of change of a quantity $\boldsymbol{w}$ (either a scalar, vector or tensor) in time following the *flow* of the material. As the material volume $V(t)$ is chosen arbitrarily, we can deduce that

$$\frac{D\rho}{Dt} + \rho \nabla \cdot \boldsymbol{u} = 0 \tag{2.2.4}$$

A material is defined to be incompressible if $D\rho/Dt = 0$ and therefore, it follows that the velocity field must be divergence-free , i.e. $\nabla \cdot \boldsymbol{u} = 0$. This is the so-called incompressibility constraint.

Conservation of Momentum

The conservation of momentum is essentially Newton's second law: force equals mass times acceleration, $\boldsymbol{F} = m\boldsymbol{a}$. The momentum of a material occupying a volume $V(t)$ is given by:

$$\int_{V(t)} \rho \boldsymbol{u} \, \mathrm{d}V \tag{2.2.5}$$

Euler's principle of linear momentum states that the rate of change of momentum of a material occupying an arbitrary region must be balanced by the total force applied to that region. In other words

$$\frac{d}{dt} \int_{V(t)} \rho \boldsymbol{u} \, \mathrm{d}V = \int_{S(t)} \boldsymbol{s}_n \, \mathrm{d}S + \int_{V(t)} \rho \boldsymbol{b} \, \mathrm{d}V \tag{2.2.6}$$

where $\boldsymbol{b}$ is a body force (such as gravity) applied to the material occupying the volume $V(t)$ and $\boldsymbol{s}_n$ is the so-called stress vector and describes the force applied on the surface $S(t)$ which bounds the volume $V(t)$. The right hand side of the above is the total force experienced by the material occupying $V(t)$. For a fluid, the surface force $\boldsymbol{s}_n$ consists of a force due to pressure and a force due to viscous and elastic stresses.

Applying Reynolds transport theorem to the left hand side and taking into account the conservation of mass, we have:

$$\int_{V(t)} \rho \frac{D\boldsymbol{u}}{Dt} \, \mathrm{d}V = \int_{S(t)} \boldsymbol{s}_n \, \mathrm{d}S + \int_{V(t)} \rho \boldsymbol{b} \, \mathrm{d}V \tag{2.2.7}$$

The conservation laws of mass and momentum applied to an incompressible material may be used to prove the existence of a symmetric stress tensor $\boldsymbol{\sigma}$ known as the Cauchy stress tensor, such that $\boldsymbol{s}_n = \boldsymbol{n} \cdot \boldsymbol{\sigma}$ where $\boldsymbol{n}$ is the outward unit normal to the surface $S(t)$, see Theorem 2.1 of Owens and Phillips [54, p. 22].

Therefore, the equations governing the flow of a material may be written as:

$$\rho \frac{D\boldsymbol{u}}{Dt} = \nabla \cdot \boldsymbol{\sigma} + \boldsymbol{f} \tag{2.2.8a}$$

$$\nabla \cdot \boldsymbol{u} = 0 \tag{2.2.8b}$$

where ρ is the density, $\boldsymbol{u}$ is the velocity field, $\boldsymbol{\sigma}$ is the Cauchy stress tensor and $\boldsymbol{f}$ is composed of body forces such as gravity. The system of equations given in (2.2.8) is not closed. To close the system, we require the Cauchy stress tensor to satisfy some constitutive law. We describe the constitutive models used in this thesis in the next section.

2.2.2 Constitutive Equations

Newtonian Fluid

We already noted that, for a fluid, the surface stress $\boldsymbol{s}_n$ is composed of a force due to pressure and a force due to viscous and elastic stresses. Therefore, we may write the Cauchy stress tensor as:

$$\boldsymbol{\sigma} = -p\boldsymbol{I} + \boldsymbol{T} \tag{2.2.9}$$

where p is the pressure and $\boldsymbol{T}$ is the deviatoric or extra-stress tensor. Newton's hypothesis states that the shear stress is linearly dependent on the rate of deformation

of the fluid. Therefore, for a Newtonian fluid, we require that the extra-stress tensor satisfies a linear stress-strain relationship:

$$\boldsymbol{T} = 2\eta_0\dot{\boldsymbol{\gamma}} \tag{2.2.10}$$

where $\dot{\boldsymbol{\gamma}} = \frac{1}{2}\left(\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T\right)$ is the rate of strain (or rate of deformation) tensor and η_0 is the viscosity of the fluid (which is a constant). Substituting the definition of the Cauchy stress tensor (2.2.9) and the extra stress tensor (2.2.10) into (2.2.8) and applying the identity $\nabla\cdot((\nabla\boldsymbol{u})^T) = \nabla(\nabla\cdot\boldsymbol{u}) = 0$, due to the incompressibility constraint, we obtain the well-known Navier-Stokes equations:

$$\rho\frac{D\boldsymbol{u}}{Dt} = -\nabla p + \eta_0\nabla^2\boldsymbol{u} + \boldsymbol{f} \tag{2.2.11a}$$

$$\nabla\cdot\boldsymbol{u} = 0 \tag{2.2.11b}$$

When the inertia in the fluid is considered to be negligible, the system of equations (2.2.11) reduce to Stokes equations and are given by:

$$-\eta_0\nabla^2\boldsymbol{u} + \nabla p = \boldsymbol{f} \tag{2.2.12a}$$

$$\nabla\cdot\boldsymbol{u} = 0 \tag{2.2.12b}$$

The Navier-Stokes equations (2.2.11) written above are in dimensional form. Most of the examples in this thesis, deal with the Navier-Stokes equations in this form. However, a couple of examples require the non-dimensional form. To non-dimensionalise the Navier-Stokes equations, we introduce the dimensionless variables:

$$\boldsymbol{t} = \boldsymbol{t}^*\frac{L}{U}, \qquad \boldsymbol{x} = \boldsymbol{x}^*L, \qquad \boldsymbol{u} = \boldsymbol{u}^*U, \qquad p = p^*\frac{\eta_0 U}{L} \tag{2.2.13}$$

where L is the characteristic length scale and U is the characteristic velocity scale. The non-dimensional Navier-Stokes equations are then written as:

$$Re\frac{D\boldsymbol{u}}{Dt} = -\nabla p + \nabla^2\boldsymbol{u} + \boldsymbol{f} \tag{2.2.14a}$$

$$\nabla\cdot\boldsymbol{u} = 0 \tag{2.2.14b}$$

where the non-dimensional group Re is called the Reynolds number. The Reynolds

number is defined as:

$$Re = \frac{\rho U L}{\eta_0} \tag{2.2.15}$$

and gives a measure of the ratio of inertial forces to viscous forces. At very low Reynolds number $Re \ll 1$, the inertial term is ignored and the system reduces to Stokes equations (2.2.12) - this is known as creeping flow. At moderate Reynolds number the flow tends to be laminar, whereas at large Reynolds number the flow becomes turbulent. The range of Reynolds number can vary quite dramatically in certain cases, e.g. in the human body, the Reynolds number can be less than 1 in the smaller blood vessels but in the region of 4000 in the heart.

Oldroyd-B Fluid

So far we have introduced the field equations, which are the mathematical descriptions of conservation of momentum and conservation of mass for the flow of any material. The field equations are not a closed system and in the previous subsection we introduced the constitutive equation for a Newtonian fluid. A Newtonian fluid is classified as any fluid that satisfies Newton's hypothesis: *the shear stress of a fluid is linearly dependent on the rate of deformation of the fluid*. Therefore, any fluid that is not a Newtonian fluid is termed a non-Newtonian fluid. The classification non-Newtonian is extremely broad and in this thesis we are only concerned with so-called viscoelastic fluids. Viscoelastic fluids, as the name suggests, are fluids that have both viscous and elastic properties. Viscoelastic fluids can behave rather differently to Newtonian fluids. An example is the well-known rod-climbing effect or Weissenberg effect. If a rod is placed in a beaker containing a Newtonian fluid and rotated, the inertia causes the fluid to dip near the rod and rise near the walls of the beaker. However, if the same experiment is performed with a viscoelastic fluid, the fluid may climb the rod. It is believed that the normal stress differences are the reason for this effect. The first and second normal stress differences are defined as:

$$N_1(\dot{\gamma}) = \sigma_{xx} - \sigma_{yy} \tag{2.2.16a}$$

$$N_2(\dot{\gamma}) = \sigma_{yy} - \sigma_{zz} \tag{2.2.16b}$$

where $\dot{\gamma}$ is the shear-rate. For a Newtonian fluid, the first and second normal stress differences are zero. However, for viscoelastic fluids, the first normal stress difference tends to be non-zero and for some constitutive models the second normal stress difference is also non-zero.

The process of deriving the Oldroyd-B constitutive equation is quite technical and so we only summarise, very briefly, the process here. Consider a distribution of dumbbells immersed in a solvent where the spring in the dumbbell satisfies a Hookean spring law. The dumbbells are a coarse-grained representation of a polymer chain and therefore the model is intended for dilute polymer solutions. An example of a dumbbell is given in Fig. 2.1. The distribution can be described via the Fokker-Planck equation:

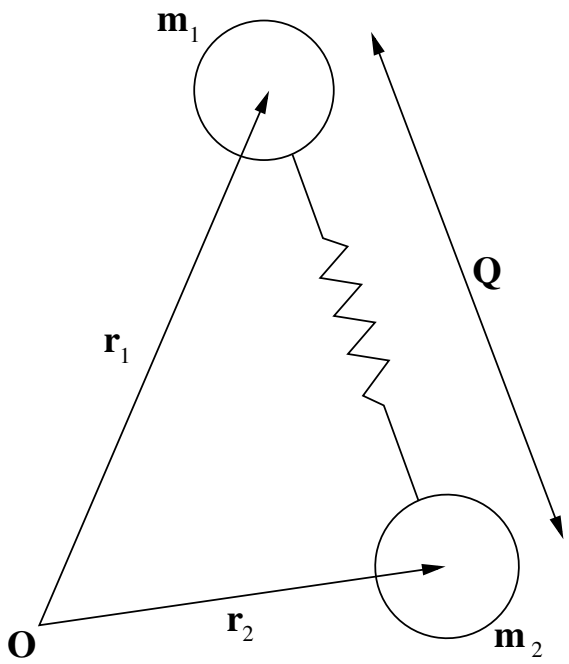


Figure 2.1: An example of a dumbbell.

$$\frac{\partial \boldsymbol{\Psi}}{\partial t} = -\frac{\partial}{\partial \boldsymbol{Q}} \cdot [(\boldsymbol{V} + (\nabla \boldsymbol{u})\boldsymbol{Q})\, \boldsymbol{\Psi}] + \frac{1}{m}\frac{\partial}{\partial \boldsymbol{V}} \cdot [(\zeta \boldsymbol{V} + 2\boldsymbol{F})\boldsymbol{\Psi}] + \frac{2kT\zeta}{m^2}\frac{\partial^2 \boldsymbol{\Psi}}{\partial \boldsymbol{V}^2} \qquad (2.2.17)$$

where $\boldsymbol{\Psi}$ is the dumbbell probability density function, $\boldsymbol{Q}$ is the end-to-end vector of the dumbbell, $\boldsymbol{V}$ is the relative velocity of the dumbbell, m is the mass of the dumbbell, $\boldsymbol{F} = H\boldsymbol{Q}$ is the spring force and ζ is the friction coefficient of the dumbbell. After some manipulation, one can arrive at the Kramers expression and Giesekus expression

for the extra-stress tensor:

$$\boldsymbol{T} = -nkT\boldsymbol{I} + \eta_s\dot{\boldsymbol{\gamma}} + nH < \boldsymbol{QQ} > \tag{2.2.18a}$$

$$\boldsymbol{T} = \eta_s\dot{\boldsymbol{\gamma}} - \frac{n\zeta}{4} < \overset{\triangledown}{\boldsymbol{Q}}\boldsymbol{Q} > \tag{2.2.18b}$$

where η_N is the solvent viscosity, $< \cdot >$ is an ensemble average and $\overset{\triangledown}{\cdot}$ is the upper-convected derivative. The upper-convective derivative is a frame invariant time derivative defined by:

$$\overset{\triangledown}{\boldsymbol{G}} = \frac{D\boldsymbol{G}}{Dt} - \boldsymbol{G}(\nabla\boldsymbol{u}) - (\nabla\boldsymbol{u})^T\boldsymbol{G} = \frac{D\boldsymbol{G}}{Dt} - \boldsymbol{E}(\boldsymbol{G},\boldsymbol{u}) \tag{2.2.19}$$

where $\boldsymbol{E}(\boldsymbol{G},\boldsymbol{u}) = \boldsymbol{G}(\nabla\boldsymbol{u}) + (\nabla\boldsymbol{u})^T\boldsymbol{G}$ is a non-linear deformation term and $D(\cdot)/Dt$ is the material derivative defined in (2.2.3). Taking the upper-convected derivative of (2.2.18a) to eliminate $< \boldsymbol{QQ} >$, gives the Oldroyd-B constitutive equation:

$$\boldsymbol{T} + \lambda_1\overset{\triangledown}{\boldsymbol{T}} = 2\eta_0\left(\dot{\boldsymbol{\gamma}} + \lambda_2\overset{\triangledown}{\dot{\boldsymbol{\gamma}}}\right) \tag{2.2.20}$$

where $\eta_0 = \eta_s + \eta_p$ is the total viscosity with η_p being the polymeric viscosity, λ_1 is the characteristic relaxation time and λ_2 is the characteristic retardation time given by:

$$\frac{\lambda_2}{\lambda_1} = \frac{\eta_s}{\eta_s + \eta_p} \tag{2.2.21}$$

The relaxation time is the time taken for the stress to relax to equilibrium and the retardation time is the time taken for the fluid to respond to a deformation. Newtonian fluids have zero relaxation and retardation times as their response is instantaneous, whereas a Hookean elastic solid has an infinite relaxation time. Although relaxation times of zero and infinity are never realised in practice, they represent idealised materials. We have omitted a large number of the steps required to derive the Oldroyd-B constitutive equation from the microscopic level of dumbbells immersed in a solvent. The full details are given in the description by Owens and Phillips [54, p. 33–37]. Alternatively, one can define the Oldroyd-B model to be a Maxwell model with an additional viscous term as was described by Van Os [68]. It is common practice to separate the solvent and polymeric contributions to the extra-stress, viz:

$$\boldsymbol{T} = 2\eta_s\dot{\boldsymbol{\gamma}} + \boldsymbol{\tau} \tag{2.2.22}$$

where $\boldsymbol{\tau}$ is the elastic or polymeric stress. Substituting the expression above into the Oldroyd-B constitutive equation gives:

$$\boldsymbol{\tau} + \lambda_1 \overset{\triangledown}{\boldsymbol{\tau}} = 2\eta_p \dot{\boldsymbol{\gamma}} \tag{2.2.23}$$

The constitutive equation for $\boldsymbol{\tau}$ given above is known as the Upper-Convected Maxwell (UCM) model and is a generalisation of the Maxwell model. The UCM model is particularly difficult to deal with numerically due to the PDE being hyperbolic and nonlinear. The additional elliptic term present in the governing equations for an Oldroyd-B fluid makes the system of PDEs numerically easier to handle. The field equations and the Oldroyd-B constitutive equation can be written as:

$$\rho \frac{D\boldsymbol{u}}{Dt} = -\nabla p + \eta_s \nabla^2 \boldsymbol{u} + \nabla \cdot \boldsymbol{\tau} + \boldsymbol{f} \tag{2.2.24a}$$

$$\nabla \cdot \boldsymbol{u} = 0 \tag{2.2.24b}$$

$$\boldsymbol{\tau} + \lambda_1 \overset{\triangledown}{\boldsymbol{\tau}} = 2\eta_p \dot{\boldsymbol{\gamma}} \tag{2.2.24c}$$

The UCM model can be obtained from the Oldroyd-B model by taking the purely elastic limit, $\eta_s \to 0$. Similar to the Newtonian case, some of the examples in this thesis require the non-dimensional form. To non-dimensionalise (2.2.24), we use the same dimensionless variables as the Newtonian case, (2.2.13) with the addition of the dimensionless stress:

$$\boldsymbol{\tau} = \boldsymbol{\tau}^* \frac{\eta_0 U}{L} \tag{2.2.25}$$

where L is the characteristic length scale and U is the characteristic velocity scale. The non-dimensional field and Oldroyd-B equations are then written as:

$$Re \frac{D\boldsymbol{u}}{Dt} = -\nabla p + \beta \nabla^2 \boldsymbol{u} + \nabla \cdot \boldsymbol{\tau} + \boldsymbol{f} \tag{2.2.26a}$$

$$\nabla \cdot \boldsymbol{u} = 0 \tag{2.2.26b}$$

$$\boldsymbol{\tau} + We \overset{\triangledown}{\boldsymbol{\tau}} = 2(1-\beta)\dot{\boldsymbol{\gamma}} \tag{2.2.26c}$$

where the non-dimensional group Re is the Reynolds number defined in (2.2.15), β is

termed the viscosity ratio and is defined by:

$$\beta = \frac{\lambda_2}{\lambda_1} = \frac{\eta_s}{\eta_s + \eta_p} \tag{2.2.27}$$

and We is called the Weissenberg number and is defined by:

$$We = \frac{\lambda_1 U}{L} \tag{2.2.28}$$

and gives a measure of the elasticity of the fluid [54]. The numerical solution of viscoelastic flows are known to suffer from stability issues at large values of the Weissenberg number - the so-called high Weissenberg number problem. The high Weissenberg number problem can be attributed to either limitations in the model or numerical approximation errors. Increasing the Weissenberg number increases the elasticity and also the hyperbolicity of the equations. Hence at large values of the Weissenberg number, standard Galerkin methods may not suitable. Additionally, numerical oscillations which propagate into the flow domain and pollute the numerical solution, can occur if the discrete problem is not well-posed. If the viscosity ratio $\beta = 1$ (and the Weissenberg number $We = 0$) then the equations reduce to a Newtonian fluid and if $\beta = 0$ then the equations reduce to the UCM model.

The Oldroyd-B model is not a physically realistic model as it may predict an unbounded stress growth. This unbounded stress growth can arise because of an infinite extensional viscosity. In a simple extensional flow, $\boldsymbol{u} = \left(\dot{\epsilon}x, -\frac{\dot{\epsilon}}{2}y, -\frac{\dot{\epsilon}}{2}z\right)$, where $\dot{\epsilon}$ is the constant extensional rate, the extensional viscosity becomes infinite at a finite extensional rate. This is due to the assumption of Hookean dumbbells which have a infinite extensibility. Assuming a finite extensibility of the dumbbells leads to the FENE model. The spring force used in a FENE model is defined as:

$$\boldsymbol{F} = \frac{H\boldsymbol{Q}}{1 - (Q^2/Q_0^2)} \tag{2.2.29}$$

where $Q^2 = tr(\boldsymbol{QQ})$ and Q_0 is some finite constant. Although in the Oldroyd-B model it is not necessary to solve the Fokker-Planck equation in order to determine the extra-stress $\boldsymbol{T}$, it is necessary in the FENE model. The reason is that it is not possible to derive a macroscopic constitutive equation for the FENE model from the

Fokker-Planck equation. Thus an approximation, known as a closure approximation, is required. An example of an approximation to the FENE model is the FENE-P model where the spring force is defined as:

$$\boldsymbol{F} = \frac{H\boldsymbol{Q}}{1 - < Q^2/Q_0^2 >} \tag{2.2.30}$$

Viscoelastic fluids may exhibit so-called shear-thinning or shear-thickening behaviour. The viscosity of a shear-thinning fluid will decrease to a value η_∞ as the shear-rate experienced by the fluid is increased, whereas the viscosity of a shear-thickening fluid will increase as the shear-rate is increased. The Oldroyd-B fluid is not a shear-thinning or shear-thickening fluid.

Chapter 3

The Immersed Boundary Method (IBM) for Fluid Structure Interaction Problems

In a classical fluid-structure interaction formulation, the fluid and structure are treated separately where the fluid is solved on a time-dependent domain and coupled to the structure equations using appropriately chosen interface conditions. The fluid-structure system of equations is then solved using either a partitioned approach or a monolithic approach. A monolithic approach involves solving a single non-linear system of equations which include both the fluid and the structure. A partitioned approach involves two systems of equations which are solved separately and then coupled together by interface conditions. There are three methods commonly used to solving fluid-structure interaction: remeshing methods, fixed-grid methods and immersed boundary methods. Strictly speaking, the immersed boundary method is also a fixed grid method. However, there is a fundamental difference between the methods which we shall discuss shortly. Here, we shall briefly discuss the remeshing and fixed-grid methods before discussing the immersed boundary method in detail.

One of the biggest drawbacks of the classical approach is the computational time required - remeshing is often needed as the computational domain for the fluid equations

is time-dependent. The Arbitrary-Lagrangian-Eulerian (ALE) technique was introduced to overcome the difficulties caused by the reconstruction of the mesh in time. In a fluid-structure interaction problem, the fluid is considered in an ALE formulation and the structure in a total (or pure) Lagrangian formulation, e.g. [39, 30]. The ALE formulation introduces an additional frame of reference, called the referential frame or configuration, which tracks the motion of the mesh. Let Ω_t be the Eulerian configuration of the fluid and structure at time $t \in (t_0, T)$ (where t_0 and T are the initial and final times respectively), Ω_r be the Lagrangian configuration of the fluid and structure and $\hat{\Omega}_r$ be the referential configuration for the ALE formulation (in many cases, $\hat{\Omega}_r \equiv \Omega_r$). An example of the three frames is given in Fig. 3.1. The fluid and structure domains in Eulerian configuration are given by Ω_t^f and Ω_t^s, respectively, in Lagrangian configuration they are given by Ω_r^f and Ω_r^s, respectively, and in the referential configuration they are given by $\hat{\Omega}_r^f$ and $\hat{\Omega}_r^s$, respectively. The interface between the fluid and structure in the Eulerian, Lagrangian and referential configurations are given by Γ_t, Γ_r and $\hat{\Gamma}_r$, respectively. At each time $t \in (t_0, T)$, each point of the referential configuration $\boldsymbol{Y}$ is then associated to a point $\boldsymbol{x}$ in the Eulerian configuration Ω_t using the so-called ALE map, $\mathcal{R}_t : \hat{\Omega}_r \to \Omega_t$, $\forall t \in (t_0, T)$, such that:

$$\mathcal{R}_t(\boldsymbol{Y}) = \boldsymbol{x}(\boldsymbol{Y}, t), \qquad \forall \boldsymbol{Y} \in \hat{\Omega}_r \tag{3.0.1}$$

where $\boldsymbol{Y}$ is usually called the ALE coordinate and $\boldsymbol{x}$ is the Eulerian coordinate. The ALE map can then be used to characterise the motion of the mesh using the following quantities:

$$\boldsymbol{w}(\boldsymbol{x}, t) = \left.\frac{\partial \mathcal{R}_t(\boldsymbol{Y})}{\partial t}\right|_{\boldsymbol{Y}} \tag{3.0.2a}$$

$$\frac{\hat{D} f(\boldsymbol{x}, t)}{\hat{D} t} = \left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{Y}} + (\boldsymbol{u} - \boldsymbol{w}) \cdot \nabla f \tag{3.0.2b}$$

where (3.0.2a) is the so-called mesh velocity and (3.0.2b) is the material derivative of a scalar function f with respect to the referential configuration or ALE configuration and $\boldsymbol{u}$ is the fluid velocity. By allowing the mesh to move with a velocity $\boldsymbol{w}$, the ALE formulation can avoid mesh distortion. Following [30], we write the full system

of equations for the fluid as:

$$\nabla^2 \mathcal{R}_t = 0 \qquad \text{in } \hat{\Omega}_r, \tag{3.0.3a}$$

$$\mathcal{R}_t = 0 \qquad \text{on } \partial\hat{\Omega}_r^f \setminus \hat{\Gamma}_r, \tag{3.0.3b}$$

$$\mathcal{R}_t(\hat{\Omega}_r^f) = \Omega_t^f, \tag{3.0.3c}$$

$$\rho_f \frac{\hat{D}\boldsymbol{u}}{\hat{D}t} + \nabla p = \eta \nabla^2 \boldsymbol{u} + \boldsymbol{f}_f \qquad \text{in } \Omega_t^f, \tag{3.0.3d}$$

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \text{in } \Omega_t^f, \tag{3.0.3e}$$

$$\boldsymbol{u} = \boldsymbol{u}_{in} \qquad \text{on } \Gamma_t^{in}, \tag{3.0.3f}$$

$$\boldsymbol{\sigma}_f \cdot \boldsymbol{n}_f = \boldsymbol{g}_f \qquad \text{on } \Gamma_t^{out}, \tag{3.0.3g}$$

where ρ_f is the fluid density, $\boldsymbol{u}$ is the fluid velocity, p is the pressure, η is the constant fluid viscosity, $\boldsymbol{\sigma}_f = -p\boldsymbol{I} + 2\eta\dot{\boldsymbol{\gamma}}$ (where $\boldsymbol{I}$ is the identity tensor and $\dot{\boldsymbol{\gamma}} = (\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T)/2$ is the rate-of-strain tensor) and $\boldsymbol{n}_f$ is the outward unit normal to $\partial\Omega_t^f$. The first three equations (3.0.3a)-(3.0.3c) are the governing equations for the motion of the mesh. The equations which govern the motion of the mesh can be defined using either the mesh velocity $\boldsymbol{w}$, e.g. [61], or the ALE map $\mathcal{R}_t$, e.g. [30]. Again following [30], we write the full system of equations for the structure as:

$$\rho_s \frac{\partial^2 \boldsymbol{d}}{\partial t^2} = \nabla \cdot \mathbb{P}_s - \gamma \boldsymbol{d} + \boldsymbol{f}_s \qquad \text{in } \Omega_r^s, \tag{3.0.4a}$$

$$\mathbb{P}_s \cdot \boldsymbol{n}_s = \boldsymbol{g}_s \qquad \text{on } \partial\Omega_r^s \setminus \Gamma_r, \tag{3.0.4b}$$

where ρ_s is the density of the structure, $\boldsymbol{d}$ is the displacement, γ is a coefficient accounting for possible viscoelastic effects, $\boldsymbol{n}_s$ is the outward unit normal to $\partial\Omega_r^s$ and $\mathbb{P}_s$ is the first Piola-Kirschoff stress tensor. Currently, conditions defined on the fluid-structure interface are missing. These conditions need to be defined with respect to the ALE map and therefore we quote the interface conditions from [30] verbatim:

$$\mathcal{R}_t = \lambda, \tag{3.0.5a}$$

$$\boldsymbol{u} \circ \mathcal{R}_t = \left.\frac{\partial \boldsymbol{d}}{\partial t}\right|_{\Gamma_r}, \tag{3.0.5b}$$

$$(\boldsymbol{\sigma}_f \cdot \boldsymbol{n}_f) \circ \mathcal{R}_t + \mathbb{P}_s \cdot \boldsymbol{n}_s = 0, \tag{3.0.5c}$$

where λ is an interface variable corresponding to the displacement $\boldsymbol{d}$ on Γ_r and $(f \circ$

$g)(\boldsymbol{x}) = f(g(\boldsymbol{x}))$ is the composition operator. It is evident from the above discussion that the classical approach to fluid-structure interaction problems using an ALE formulation is quite involved and complex. Additionally, an ALE formulation can be computationally expensive when large deformations or complex fluid-structure interaction systems are considered. To overcome the issues of an ALE formulation when large deformations are considered, unfitted methods called fixed-grid methods were introduced as the structure is allowed to move and deform freely.

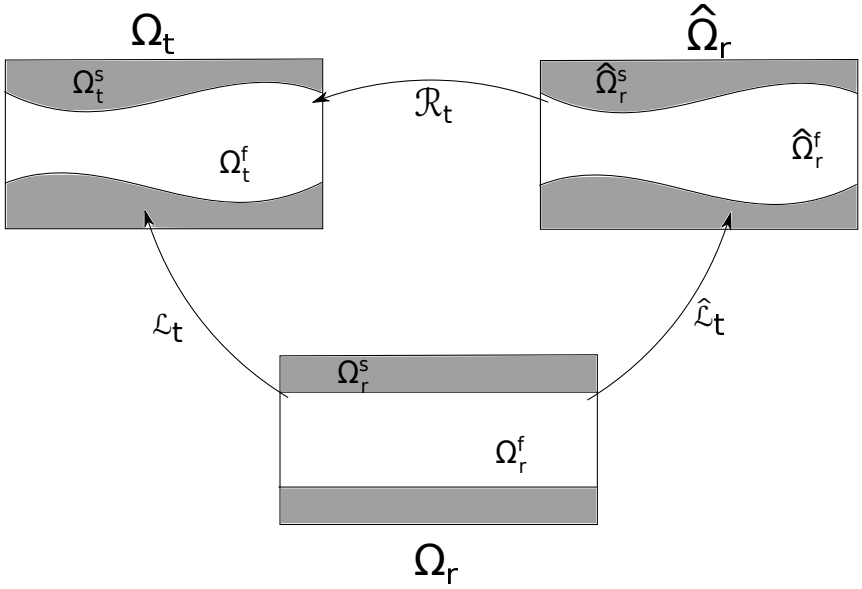


Figure 3.1: Example of the three configurations present in an ALE formulation of a fluid-structure interaction problem.

In a fixed-grid method, the fluid is solved on a fixed Eulerian grid whilst the structure is solved on a time-dependent Lagrangian grid which is allowed to move freely over the underlying fluid grid. As we mentioned previously, one of the biggest drawbacks of the classical approach (and indeed an ALE approach) is the computational time when large deformations are considered because of the fluid grid needing to be remeshed in time. A fixed-grid approach alleviates the required remeshing of the fluid and therefore, the method can deal with large deformations at a reasonable computational time. In general, fixed-grid methods proceed in a similar manner to remeshing methods: the fluid and the structure are solved separately and then coupled together via suitable interface conditions. The fluid can be solved using a Ficticious Domain approach where artificial fluid quatities are defined in the structure. Those artificial quantities should be defined so that they do not affect the physics of the problem. The biggest

drawback of fixed-grid methods is the enforcement of the interface conditions because the interface is unfitted to the underlying fluid grid. Lew and Buscaglia [43] reviewed the most popular methods of enforcing the interface conditions. The simplest approach is to force the interface to coincide with the element edges of the fluid mesh, which will obviously introduce large errors near the interface. Locally modifying the mesh (or interpolation) to coincide with the interface is another approach and allows the interface, or boundary conditions, or be enforced strongly. However, care must be taken when using such a procedure because it is possible that the numerics will not be able to obtain a solution. Finally, one can enforce the interface conditions using Lagrange multipliers. The Lagrange multiplier technique involves constructing a space of Lagrange multipliers and using that space to enforce the interface conditions weakly. However, as noted by Lew and Buscaglia [43], the method may not obtain optimal convergence if the inf-sup condition:

$$\inf_{v_h \in W_h} \sup_{\eta_h \in \Lambda_h} \frac{\int_{\Gamma_h} v_h \eta_h \, \mathrm{d}\Gamma}{h^{1/2} \left\| \eta_h \right\|_{L^2(\Gamma_h)} \left\| v_h \right\|_{H^1(\Omega_h)}} \geq \gamma > 0 \quad , \tag{3.0.6}$$

is not satisfied, where W_h is the velocity approximation space, Λ_h is the approximation space for the Lagrange multipliers, h is the mesh width and γ is a positive constant independent of h. Additionally, constructing the space of Lagrange multipliers can be complex and it is possible that inexact interface, or boundary, values are imposed. The reader is referred to the article by Lew and Buscaglia [43] for further details of the aforementioned methods. Gerstenberger and Wall [35, 36] combined a ficticious domain approach with a Lagrange multiplier technique and an extended finite element method (XFEM). The XFEM allows the approximation of the fluid variables to be *switched* off inside the ficticious domain, thus guaranteeing that the artificial fluid quantifies in the ficticious domain had no influence on the structure. The aim of Gerstenberger and Wall [35, 36] was to construct a robust algorithm for complex fluid-structure interaction systems. However, the robustness of their algorithm is debatable. Using XFEM to approximate the velocity and pressure variables changes their approximation spaces. Thus raising the question, of whether the velocity and pressure inf-sup condition is still satisifed. Additionally, using Lagrange multipliers is not necessarily that robust, due to the complexity of constructing the space of Lagrange multipliers. Not to mention, that changing the approximation space for the velocity may in fact have an impact on the inf-sup condition for the Lagrange multipliers given above (3.0.6). Lew and Buscaglia

[43] introduced a very simple and robust method for enforcing interface conditions strongly. They proposed switching from a continuous to a discontinuous approximation in the elements intersected by the interface. An alternative fixed-grid method which is common in the literature is the immersed boundary method.

The immersed boundary method was first proposed by Charles Peskin in 1972 when studying flow patterns around heart valves [56]. The idea behind the immersed boundary method is to consider the structure to be part of the fluid (which is similar to the ficticious domain approach). The key difference between the fixed-grid approach discussed above and the immersed boundary method is that the immersed structure is replaced by an Eulerian force field so that only a single system of equations needs to be solved. This means that the interface conditions are implicitly defined and therefore, do not need to be enforced. In the immersed boundary method, the interface conditions (for an immersed boundary) are given by [59]:

$$[\boldsymbol{\sigma} \cdot \boldsymbol{n}] = -\frac{\boldsymbol{F}}{|\partial \boldsymbol{X}/\partial s|} \tag{3.0.7}$$

where $\boldsymbol{F}$ is the Lagrangian force density for the immersed boundary, $\boldsymbol{X}$ is the location of the immersed boundary and s is the parameterisation of the immersed boundary. Just as with the ficticious domain approach, artificial fluid quantities are introduced into the structure and there is a strong argument that the physics of the problem is altered in the immersed boundary method. Since its introduction, the immersed boundary method has been applied to a wide range of applications, including arterial blood flow [3], modelling of the cochlea [10], modelling of red blood cells in Poiseuille flow [63] and flows involving suspended particles [32]. A more extensive list of applications can be found in [58]. The use of high-order polynomials is absent from the immersed boundary literature and hence, the aim of this thesis is to improve the accuracy of the immersed boundary method by applying a high-order approximation to the fluid variables. Additionally, the use of an enriched approximation (such as XFEM) is absent from the literature and hence, in this thesis, we consider a high-order enriched approximation.

In this chapter, we introduce Peskin's original immersed boundary method and discuss some limitations. Then we introduce a few extensions which have been proposed in the

literature before discussing the method proposed by Boffi *et al.* [17], which we use as the basis for developments in this thesis. Before we move to on to discuss the IBM, we note that throughout this thesis we synonymously interchange the terms Eulerian and Lagrangian mesh (or grid) with fluid and structure mesh (or grid) respectively. Also, we note that throughout this thesis any quantities relating to the structure will use a superscript s (e.g. Ω_r^s denotes the reference configuration of the immersed structure) and any quantities relating to the solvent or Newtonian fluid will use a subscript s (e.g. η_s denotes the solvent viscosity). Additionally, throughout this thesis, a fibre (or boundary or membrane) is defined as a one-dimensional structure which is immersed in a two-dimensional fluid, a shell is defined as a two-dimensional structure which is immersed in a two-dimensional fluid and finally, a fibrous material is a two-dimensional structure constructed from a continuum of nested fibres.

3.1 Newtonian Flows

3.1.1 Peskin's Immersed Boundary Method

The immersed boundary method was proposed by Charles Peskin in 1972 when studying flow patterns around heart valves [56]. The method is both a mathematical formulation and a numerical scheme for the study of fluid-structure interaction problems featuring an elastic boundary immersed in a fluid, where the fluid and the elastic boundary *interact* with one another. What we mean by interact, is that the elastic boundary moves with, and exerts forces on, the surrounding fluid. In a classical formulation, two systems would need to be solved, one for the fluid and one for the immersed structure. These systems can be solved either separately and coupled together via some suitable interface conditions, the so-called *partitioned* approach, or solved together in a single non-linear system, the so-called *monolithic* approach. In both approaches, re-meshing would most likely be required since the domains for the fluid and the immersed structure are time-dependent. Peskin's immersed boundary method alleviates these issues by considering the immersed structure to be part of the fluid.

The structure, which in the original method is usually considered to be incompressible,

massless and linearly elastic, is replaced by an Eulerian force field calculated from the structure configuration. The fluid experiences these forces instantaneously throughout the fluid due to the incompressibility of the immersed structure. The structure configuration is then updated using the local fluid velocity. Peskin noted that this implies that points near, on, inside or outside the IB are all treated in the same way. Whilst this is considered to be one of the key advantages of the IBM, it can also be viewed, in certain instances, as a disadvantage because in a classical context, there may be a constraint on the velocity which would need to be imposed on the interface between the structure and the fluid; namely the no-slip condition $\boldsymbol{u} = 0$. We discuss the consequences of the absence of this constraint in §5.4.2. The interaction, or coupling, is achieved using an approximation to the Dirac delta distribution $\boldsymbol{\delta}$, which we discuss below. The force field that is created is singular and is therefore sensitive to small changes in configuration. Hence, the method can be numerically unstable.

In 1972, Peskin [56] illustrated (his Fig. 2) the instabilities that can arise in the method if an explicit boundary configuration is used to calculate the force field. Therefore, Peskin introduced an implicit formulation to remove these instabilities (their Fig. 5). The implicit method may improve stability issues but it also increases the computational time as a non-linear system needs to be solved (to obtain the configuration and force field) prior to solving the linear system (to obtain the velocity and pressure fields). Additionally, these instabilities really only arise if large forcing parameters, small viscosity or large time step lengths are used. We illustrate these instabilities in §4.2. Peskin admits that the setting for the computation of the flow patterns around the heart valves may not be realistic, but the results are never-the-less impressive. There are four boundaries that are immersed in the fluid; two for the valve leaflets and two for the cushions which act to hold the leaflets in place. The cushions have restoring forces which attract them to equilibrium positions. The valve leaflets were connected to an additional point which act to pull the leaflets towards that point if they are far away and are slack if the leaflets are close. The opening and closing of the leaflets are controlled using an oscillating external force. Fluid was contained both inside and outside the cushions and this illustrates one of the limitations of the original immersed boundary method. Realistically, those cushions are most likely going to be thick viscoelastic structures (most likely with a different viscosity to that of the surrounding fluid) but the original IBM is restricted to the assumption of fibres or boundaries. Their Fig. 5 il-

lustrates the opening and closing of the valve leaflets as well as the velocity field vector.

In 1977, Peskin [57] studied the flow pattern of blood in the heart, specifically the left side of the heart during ventricular diastole and early systole. That article extended the work in the 1972 article [56]. The heart wall and valves were considered as a *single continuous* fibre which was assumed to be incompressible, neutrally buoyant and immersed in a box which contained the same fluid both inside and outside the structure. The box needed to be sufficiently large so as to minimize the effect of the external fluid on the flow patterns inside the heart. Peskin also pointed out that additional sources and sinks were required to accomodate the changes in volume during the cardiac cycle. A source was placed inside the left atrium and this simulated the inflow of blood from the lungs to the heart. A sink was placed close the boundary of the box which contained the heart. This sink accomodated the changes in volume due to the motion of the heart. The model of the heart was composed using two different models; one for the muscle wall and one for the leaflets. The leaflets were modelled using the same linearly elastic model that was used in [56]. However, the heart wall was composed of the same linearly elastic model and an additional part due to the contractile element (their Fig. 3). The length of the contractile element was time-dependent. Their Fig. 2 illustrates their approximation to the left side of the heart. It is clear from that figure that crosslinks are used connected to each side of the heart wall and also connecting the leaflets to the apex of the heart. These crosslinks act to prevent the heart walls from moving independently of one another and therefore keep the heart's shape. Peskin [57] stated that they also imitate the *atrioventricular ring* which is known to contract during systole. Additionally, Peskin stated that the crosslinks connected to the leaflets imitate the *chordae* which are present at the joint between the leaflets and the cushions which appeared in [56]. Fig. 7 of [57] illustrates the model during ventricular diastole and early systole. Those figures illustrate the vortex which forms behind the valve leaflets. This vortex aids valve closure. However, there does seem to be an additional *backflow* into the left atrium during early systole and it is unclear from [57] whether this backflow is physically realistic.

A key advantage of the immersed boundary method is that the fluid and the immersed structure are dealt with in their natural frameworks; that is Eulerian for the fluid and

Lagrangian for the structure. Additionally, due to the structure being thought of as being part of the fluid, the fluid domain can be considered to be fixed which alleviates issues associated with re-meshing.

Mathematical Formulation

In this subsection we discuss the mathematical formulation of the immersed boundary method. We mainly refer to the pioneering work of Peskin [58], as this article is primarily concerned with the mathematical formulation. We leave the discretisation and the discussion of the discrete Dirac delta function to the next subsection. In this thesis, we are only concerned with two-dimensional examples and therefore we restrict ourselves to the two-dimensional setting.

We mentioned earlier that the original IBM is restricted for use with immersed fibres or boundaries/interfaces. In his 2002 article, Peskin began by considering a higher dimensional elastic material which filled the whole space. Therefore, the original IBM is not mathematically restricted to 1D fibres but is actually an assumption. Let $\Omega \subset \mathbb{R}^2$ be our open domain containing viscoelastic continuum. Denote by the open set $\Omega^f \subset \mathbb{R}^2$ the domain containing the fluid and by the open set $\Omega^s \subset \mathbb{R}^d$ the domain containing the immersed viscoelastic structure, where $d = 1, 2$, such that $\bar{\Omega} = \bar{\Omega}^f \cup \bar{\Omega}^s$. As we are following Peskin's formulation, we assume that $d = 2$ and that $\Omega^f = \emptyset$ so that $\bar{\Omega} = \bar{\Omega}^s$. Let $\boldsymbol{s} = (q, r) \in \Omega^s_r$ be curvilinear coordinates which label a point in the reference configuration of the elastic material. Denote by $E[\boldsymbol{X}(\cdot, t)]$ the elastic energy stored in the material at time t, where $\boldsymbol{X}(\boldsymbol{s}, t)$ denotes the Cartesian position in Ω of a material point $\boldsymbol{s}$, i.e. $\boldsymbol{X} \colon \Omega^s_r \times [0, T] \to \Omega$ where T is the final time of the simulation and Ω is assumed to be fixed in time. Clearly if $\boldsymbol{s}$ is fixed, then $\boldsymbol{X}$ traces the material point in time - in other words, the particle trajectory in Ω. Therefore, the Lagrangian mesh can be thought to be made up of marker particles which move freely throughout the Eulerian mesh with no constraints at all.

Most of the models used for the elastic force density of the immersed structure are based on the tensions in the fibres or tangents to links between points $\boldsymbol{X}$, e.g. [56,

57, 10, 31, 3]. However, Peskin [58] introduced the elastic force density as the Fréchet derivative of the elastic (potential) energy. The Fréchet derivative is a generalisation of the derivative on functions defined between Banach spaces, we give the definition in Appendix A.1. Following Peskin, we let $\wp \boldsymbol{X}$ denote a perturbation in the configuration of the immersed structure, where $\wp$ is the perturbation operator. The energy associated with this perturbation can be defined as:

$$\wp E[\boldsymbol{X}(\cdot, t)] = \int_{\Omega_r^s} (-\boldsymbol{F}(\boldsymbol{s}, t)) \cdot \wp \boldsymbol{X}(\boldsymbol{s}) \, \mathrm{d}\boldsymbol{s} \tag{3.1.1}$$

where $\boldsymbol{F}$ is defined as the Fréchet derivative of the energy E. The physical interpretation is that $\boldsymbol{F}$ is the force density generated by the elasticity of the material. Peskin notes that this is essentially the principle of virtual work, which is the starting point of the formulation of the finite element immersed boundary method (FE-IBM), [17]. Peskin then gave an example where the immersed structure was assumed to be a fibre. Although the Fréchet derivative provides a rigorous mathematical framework for the definition of the elastic force density, in general it could be difficult to construct. Therefore, we believe this is the reason why the method relies on the assumption of fibrous materials.

The procedure used in [58] is to make the equations of elasticity *look* like the equations of fluid dynamics. Peskin began from the principle of least action which we state from [58] verbatim:

> Our system will evolve over the time interval $(0, T)$ in such a manner as to minimize the action S defined by:
>
> $$S = \int_0^T L(t) \, \mathrm{d}t \tag{3.1.2}$$
>
> where L is the Lagrangian of the system. Subject to the constraint of incompressibility $\partial J / \partial t = 0$ where:
>
> $$J = \det \left(\frac{\partial \boldsymbol{X}}{\partial \boldsymbol{s}} \right) \tag{3.1.3}$$
>
> as well as initial and final configurations for $\boldsymbol{X}$.

As mentioned, the approach of Peskin was to make the equations of elasticity look like the equations of fluid dynamics. This means beginning in Lagrangian variables and adding in Eulerian variables along the way. We follow the approach of Peskin [58] essentially verbatim. The Lagrangian is usually defined as the difference between the kinetic and potential energies. In Peskin's formulation it is given by:

$$L(t) = \frac{1}{2}\int_{\Omega_r^s} M(\boldsymbol{s}) \left|\frac{\partial \boldsymbol{X}}{\partial t}(\boldsymbol{s},t)\right|^2 \,\mathrm{d}\boldsymbol{s} - E[\boldsymbol{X}(\cdot,t)] \tag{3.1.4}$$

where $M(\boldsymbol{s})$ is the Lagrangian mass density such that the mass in an arbitrary region $\mathcal{Q} \subset \Omega_r^s$ is given by:

$$m(\mathcal{Q}) = \int_{\mathcal{Q}} M(\boldsymbol{s}) \,\mathrm{d}\boldsymbol{s} \tag{3.1.5}$$

Taking the perturbation of the action S above and substituting in the expression for the Lagrangian gives:

$$0 = -\wp S = \int_0^T \int_{\Omega_r^s} \left(M(\boldsymbol{s})\frac{\partial^2 \boldsymbol{X}}{\partial t^2} - \boldsymbol{F}\right) \cdot \wp \boldsymbol{X} \,\mathrm{d}\boldsymbol{s}\,\mathrm{d}t \tag{3.1.6}$$

where integration by parts has been used on the first integral and the expression for the elastic potential energy given in (3.1.1) has been used. The above expression is nothing more than Newton's second law: force equals mass times acceleration. However, Peskin noted that the perturbation $\wp \boldsymbol{X}$ is not completely arbitrary as it must be consistent with the incompressibility constraint. In other words, if $\boldsymbol{X} + \wp \boldsymbol{X}$ is substituted for $\boldsymbol{X}$ in the definition of the Jacobian J then its value doesn't change. It is at this point that Peskin introduced the Eulerian variables. As we have already mentioned, for a fixed material point $\boldsymbol{s}$, the map $\boldsymbol{X}$ traces the trajectory of the marker particle in the Eulerian domain. Therefore, as is standard in continuum mechanics, we can write:

$$\boldsymbol{u}(\boldsymbol{X}(\boldsymbol{s},t),t) = \frac{\partial \boldsymbol{X}}{\partial t}(\boldsymbol{s},t) \tag{3.1.7a}$$

$$\frac{D\boldsymbol{u}}{Dt}(\boldsymbol{X}(\boldsymbol{s},t),t) = \frac{\partial^2 \boldsymbol{X}}{\partial t^2}(\boldsymbol{s},t) \tag{3.1.7b}$$

where $D\boldsymbol{u}/Dt$ is the acceleration. Peskin also introduced a *pseudo-velocity* $\boldsymbol{v}$ that

corresponded to the perturbation $\wp \boldsymbol{X}$:

$$\boldsymbol{v}(\boldsymbol{X}(\boldsymbol{s}), t) = \wp \boldsymbol{X}(\boldsymbol{s}, t) \tag{3.1.8}$$

As the perturbation $\wp \boldsymbol{X}$ must be consistent with the incompressibility constraint, a key step in the procedure used by Peskin was to prove that the incompressibility constraint is equivalent to $\nabla \cdot \boldsymbol{v} = 0$. The proof is fairly straightforward but also quite long-winded. Therefore, we omit the proof here but we refer to the article [58] for all the details. Peskin then introduced the delta function via its sifting property. Therefore,

$$\boldsymbol{v}(\boldsymbol{X}(\boldsymbol{s}, t), t) = \wp \boldsymbol{X}(\boldsymbol{s}, t) = \int_{\Omega} \boldsymbol{v}(\boldsymbol{x}, t) \delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}\boldsymbol{x} \tag{3.1.9}$$

where $\delta(\boldsymbol{x}) = \delta(x_1)\delta(x_2)$, $\boldsymbol{x} = (x_1, x_2)$, is the two-dimensional Dirac delta distribution. Any terms which involve the delta function are called *interaction* terms as they define the relationship between the Eulerian and Lagrangian variables. Substituting (3.1.9) and (3.1.7) into (3.1.6) gives:

$$0 = \int_0^T \int_{\Omega_r^s} \int_{\Omega^s} \left(M(\boldsymbol{s}) \frac{D\boldsymbol{u}}{Dt} - \boldsymbol{F}(\boldsymbol{s}, t) \right) \cdot \boldsymbol{v}(\boldsymbol{x}, t) \delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}\boldsymbol{x} \, \mathrm{d}\boldsymbol{s} \, \mathrm{d}t \tag{3.1.10}$$

The above expression has a mixture of Eulerian and Lagrangian quantities. Taking the Dirac delta distribution inside the brackets introduces the following definitions:

$$\rho(\boldsymbol{x}, t) = \int_{\Omega_r^s} M(\boldsymbol{s}) \delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}\boldsymbol{s} \tag{3.1.11a}$$

$$\boldsymbol{f}(\boldsymbol{x}, t) = \int_{\Omega_r^s} \boldsymbol{F}(\boldsymbol{s}, t) \delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}\boldsymbol{s} \tag{3.1.11b}$$

Using the definitions of ρ and $\boldsymbol{f}$, we arrive at:

$$0 = -\wp S = \int_0^T \int_{\Omega^s} \left(\rho(\boldsymbol{x}, t) \frac{D\boldsymbol{u}}{Dt} - \boldsymbol{f}(\boldsymbol{x}, t) \right) \cdot \boldsymbol{v}(\boldsymbol{x}, t) \, \mathrm{d}\boldsymbol{x} \, \mathrm{d}t \tag{3.1.12}$$

Note that the vector field $\boldsymbol{v}$ is subject to the incompressibility constraint: $\nabla \cdot \boldsymbol{v} = 0$. Therefore, (3.1.12) is merely the Eulerian equivalent of (3.1.6). The final step is to use the Hodge decomposition to introduce the gradient of the pressure. Hodge decomposition states that an arbitrary vector field may be written as the sum of a

gradient and a divergence-free vector field. In particular:

$$\rho\frac{D\boldsymbol{u}}{Dt} - \boldsymbol{f} = -\nabla p + \boldsymbol{w} \tag{3.1.13}$$

where $\nabla \cdot \boldsymbol{w} = 0$. Peskin then proved that the vector $\boldsymbol{w} = 0$. The full system of equations is then given by:

$$\rho\frac{D\boldsymbol{u}}{Dt} + \nabla p = \eta_s \nabla^2 \boldsymbol{u} + \boldsymbol{f} \tag{3.1.14a}$$

$$\nabla \cdot \boldsymbol{u} = 0 \tag{3.1.14b}$$

$$\rho(\boldsymbol{x}, t) = \int_{\Omega_r^s} M(\boldsymbol{s})\delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t))\, \mathrm{d}\boldsymbol{s} \tag{3.1.14c}$$

$$\boldsymbol{f}(\boldsymbol{x}, t) = \int_{\Omega_r^s} \boldsymbol{F}(\boldsymbol{s}, t)\delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t))\, \mathrm{d}\boldsymbol{s} \tag{3.1.14d}$$

$$\frac{\partial \boldsymbol{X}}{\partial t} = \boldsymbol{u}(\boldsymbol{X}(\boldsymbol{s}, t), t) = \int_{\Omega} \boldsymbol{u}(\boldsymbol{x}, t)\delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t))\, \mathrm{d}\boldsymbol{x} \tag{3.1.14e}$$

$$\boldsymbol{F} = -\frac{\wp E}{\wp \boldsymbol{X}} \tag{3.1.14f}$$

Note that the viscous term $\eta_s \nabla^2 \boldsymbol{u}$ has been added to the final system. The principle of least action only applies to conservative systems and therefore the dissipative viscous term cannot be included in the derivation. Peskin [58] uses a uniform viscosity of the kind that appears in a Newtonian fluid but mentions that this could be generalised. However, later in his article, Peskin states that it is unclear how to handle the variable viscosity in his derivation. In other words, it is unclear how to introduce variable viscosity from the Lagrangian configuration. The expressions (3.1.14d) and (3.1.14e) are called the *spreading* and *interpolation* phases, respectively. They are both continuous and discrete adjoints of one another, a feature that Peskin proved in his 2002 article [58].

Note that in the case of fluid-structure interaction problems, the fluid domain $\Omega^f \neq \emptyset$. Therefore, the elastic force density $\boldsymbol{F}$ (which is found from the elastic potential energy) and the Eulerian force density $\boldsymbol{f}$ are zero at points $\boldsymbol{x}$ that are in the fluid region. Peskin noted that the formulation does not need to be changed if we choose to have a non-uniform mass density even in the fluid regions. However, as the structure is thought of as being part of the fluid, the standard procedure is to assume a uniform mass density ρ_0 which denotes the density in the fluid region. Therefore, the density in the structure

region can be defined using an excess Lagrangian mass density. Thus we can write:

$$\rho(\boldsymbol{x},t) = \rho_0 + \int_{\Omega_r^s} \tilde{M}(\boldsymbol{s})\delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s},t))\,\mathrm{d}\boldsymbol{s} \tag{3.1.15}$$

where $\tilde{M}$ denotes the excess Lagrangian mass density. In the case of an immersed boundary (the situation which gives the method its name), only a single Lagrangian variable is required. Here, this implies that the coordinate $\boldsymbol{s} = (q,r)$ reduces to q which can stand for a parameterisation if necessary. Therefore, the expressions for the density, force and interpolated velocity can be written as:

$$\rho(\boldsymbol{x},t) = \rho_0 + \int_{\Omega_r^s} M(q)\delta(\boldsymbol{x} - \boldsymbol{X}(q,t))\,\mathrm{d}q \tag{3.1.16a}$$

$$\boldsymbol{f}(\boldsymbol{x},t) = \int_{\Omega_r^s} \boldsymbol{F}(q,t)\delta(\boldsymbol{x} - \boldsymbol{X}(q,t))\,\mathrm{d}q \tag{3.1.16b}$$

$$\frac{\partial \boldsymbol{X}}{\partial t} = \boldsymbol{u}(\boldsymbol{X}(q,t),t) = \int_{\Omega} \boldsymbol{u}(\boldsymbol{x},t)\delta(\boldsymbol{x} - \boldsymbol{X}(q,t)))\,\mathrm{d}\boldsymbol{x} \tag{3.1.16c}$$

The Dirac delta distribution in (3.1.16) is still two-dimensional. Therefore, (3.1.16a) and (3.1.16b) are singular just like a one-dimensional delta function with the singularity being on the immersed boundary. Note that even though they are infinite on the immersed boundary, their integral is still finite over a finite region. However, the interpolated velocity (3.1.16c) remains unchanged because the integral is still over a two-dimensional region.

Numerical Approximation and the Discrete Delta Function

The original IBM used finite differences to numerically solve the equations of motion. Due to the mixture of Lagrangian and Eulerian variables, two independent grids are employed; one for the Lagrangian variables and one for the Eulerian variables. The Eulerian grid is the set of points $\boldsymbol{x} = \boldsymbol{j}h$, where $\boldsymbol{j} = (j_1, j_2)$ has integer components and the Lagrangian grid is the set of points $(k_q\Delta q, k_r\Delta r)$ where k_q and k_r are integers. Note that the fluid grid is uniform and the structure grid is non-uniform. In his 2002 article, Peskin employed the restriction:

$$|\boldsymbol{X}(\boldsymbol{s} + \Delta\boldsymbol{s}, t) - \boldsymbol{X}(\boldsymbol{s},t)| \leq \frac{h}{2} \tag{3.1.17}$$

to avoid leaks. This condition is well known within the immersed boundary method community. It is not a mathematical derived condition but a numerically validated condition. The spatial finite difference operators used in [58] are the central difference operator, defined by:

$$(D^0_{h,\beta})(\boldsymbol{x}) = \frac{\phi(\boldsymbol{x} + h\boldsymbol{e}_\beta) - \phi(\boldsymbol{x} - h\boldsymbol{e}_\beta)}{2h} \tag{3.1.18}$$

where β is the coordinate direction, $\boldsymbol{e}_\beta$ is the unit basis vector in the direction β and h is the mesh width, and a *tight* Laplacian, defined by:

$$(L_h\phi)(\boldsymbol{x}) = \sum_{\beta=1}^{2} \frac{\phi(\boldsymbol{x} + h\boldsymbol{e}_\beta) + \phi(\boldsymbol{x} - h\boldsymbol{e}_\beta) - 2\phi(\boldsymbol{x})}{h^2} \tag{3.1.19}$$

A key element of the spatial discretisation and the immersed boundary method in general, is the construction of the discrete delta function. The idea is to construct a smoothed approximation δ_h which tends to $\delta(\boldsymbol{x})$ as $h \to 0$; in other words, we look to construct a *delta sequence*. Here, we discuss the method proposed by Peskin [58]. Firstly, it is assumed that the discrete Delta function takes the form:

$$\delta_h(\boldsymbol{x}) = \frac{1}{h^2}\phi\left(\frac{x_1}{h}\right)\phi\left(\frac{x_2}{h}\right) \tag{3.1.20}$$

where $\boldsymbol{x} = (x_1, x_2)$. The delta function is not *technically* a function as no function can exist that satisfies all of the delta function properties and is the reason why we consider that the approximation of the delta function to be a disadvantage of the method. Therefore, one would expect a better option is to approximate its action on a test function (which is done in the extensions given in §3.1.2 below) rather than to approximate the function itself. To that end, Peskin [58] proposed various additional properties which must be satisified discretely. Following [58], we quote the properties

verbatim using r to denote either x_1/h or x_2/h:

$$\phi(r) \text{ is continuous for all real } r, \tag{3.1.21a}$$

$$\phi(r) = 0 \text{ for } |r| \geq 2, \tag{3.1.21b}$$

$$\sum_{j \text{ is even}} \phi(r-j) = \sum_{j \text{ is odd}} \phi(r-j) = \frac{1}{2} \text{ for all real } r, \tag{3.1.21c}$$

$$\sum_j (r-j)\phi(r-j) = 0 \text{ for all real } r, \tag{3.1.21d}$$

$$\sum_j (\phi(r-j))^2 = C \text{ for all real } r. \tag{3.1.21e}$$

Peskin [58] stated that the reason for each of the above properties is to hide the Eulerian lattice as much as possible. However, some of them have a continuous counterpart. The third and fourth conditions above, (3.1.21c) and (3.1.21d), are essentially discrete analogues of the conditions:

$$\int_{\mathbb{R}} \delta(r-x)\, \mathrm{d}x = 1 \tag{3.1.22a}$$

$$\int_{\mathbb{R}} (r-x)\delta(r-x)\, \mathrm{d}x = 0 \tag{3.1.22b}$$

and are required in order to show that mass, force and torque are the same whether they are evaluated in terms of Eulerian or Lagrangian variables. In other words, they are required in order to guarantee that momentum, angular momentum and energy are not spuriously created or destroyed by the interaction equations. Peskin [58] noted that this does not mean that these quantities are conserved. The second condition, (3.1.21b), which gives the support of the delta function, is required in order to *spread* the elastic force density a finite distance into the fluid. In the continuous formulation, the force is only non-zero where the immersed structure resides. Hence, if the approximation to the delta function had an infinite support (e.g. when an exponential is used), then the elastic force given to the fluid would spread to the entire domain, which is unphysical and a bad approximation to the continuous formulation. The third property, (3.1.21c), is not as obvious and we quote the reason from [58] verbatim:

> The condition (3.1.21c) is a technical issue related to the central difference operator. The null space of the gradient operator based on $\boldsymbol{D}^0$ is eight-dimensional. It contains not only the constants but any function that

is constant on each of the 8 'chains' of points $\{j_1 \text{ even}, j_2 \text{ even}, j_3 \text{ even}\}$, $\{j_1 \text{ even}, j_2 \text{ even}, j_3 \text{ odd}\}$, etc... The separate conditions, therefore, ensure that all eight chains get the same amount of force from each Lagrangian marker point, and also, that each Lagrangian marker point assigns equal weight to all eight chains when computing its interpolated velocity. This avoids oscillations from one grid point to the next that would otherwise occur, especially when localized forces are applied.

The conditions (3.1.21) are then used to construct the specific form of the approximation. For example, Peskin [58] suggested using the following for the approximation of the delta function:

$$\phi(r) = \begin{cases} 0 & r \leq -2 \\ \frac{1}{8}\left(5 + 2r - \sqrt{-7 - 12r - 4r^2}\right) & -2 \leq r \leq -1 \\ \frac{1}{8}\left(3 + 2r + \sqrt{1 - 4r - 4r^2}\right) & -1 \leq r \leq 0 \\ \frac{1}{8}\left(3 - 2r + \sqrt{1 + 4r - 4r^2}\right) & 0 \leq r \leq 1 \\ \frac{1}{8}\left(5 - 2r - \sqrt{-7 + 12r - 4r^2}\right) & 1 \leq r \leq 2 \\ 0 & 2 \leq r \end{cases} \tag{3.1.23}$$

Peskin [58] noted that the above form for ϕ can be extremely well approximated using the simple formula:

$$\phi(r) = \begin{cases} \frac{1}{4}\left(1 + \cos\left(\frac{\pi r}{2}\right)\right) & |r| \leq 2 \\ 0 & \text{otherwise} \end{cases} \tag{3.1.24}$$

For full details the reader is referred to [58]. We note that the same delta function needs to be used for the both the spreading and interpolation phases, otherwise energy is not conserved.

Limitations

There are a few limitations to the original immersed boundary method. Firstly, the assumption that the immersed structure is a boundary or interface can be quite limit-

ing, particularly when one wishes to model an immersed thick structure. As far as we are aware, within the original IBM the way to overcome this was to create a continuum of fibres or boundaries. Additionally, the viscosity was assumed to be constant throughout the whole domain Ω. This implies that the immersed structure has the same viscosity as that of the surrounding fluid and although the assumption of the structure having the same density as the fluid is valid, the assumption of their respective viscosities being the same is a bit more questionable. In 2002, Peskin [58] stated that the issue lies in how to determine the variable (or discontinuous) viscosity from the Lagrangian configuration. To our knowledge, this restriction is still present in the immersed boundary literature. However, we present an extension in §3.2 which we believe overcomes this restriction. Finally, in the literature, it is well documented that immersed boundary computations can suffer from a loss in area (2D)/volume (3D) and also stability issues at small viscosity or large forcing parameters. These issues are still not fully understood and we discuss them in more detail in §5.4.2 and §4.2, respectively.

3.1.2 Extensions of the Immersed Boundary Method

As mentioned in the previous section, there are a few limitations to the original IBM. Namely, the assumption of constant viscosity and the assumption of immersed fibres or boundaries. To apply the original IBM to the example of a thick immersed structure, the usual procedure, as far as we are aware, is to consider the thick immersed structure as a continuum of fibres, see for example [51]. In doing so, the Lagrangian variables become (r, s), for example, where s denotes the material parameter for the fibre designated by the variable r. Although this may be a reasonable approximation in some cases, such as an arterial wall, we cannot expect it be a reasonable assumption in general. In fact, even though an arterial wall is fibrous the fibres are actually of different thicknesses. Additionally, even though one could argue that a thick structure can be approximated by an infinite number of fibres, in the discrete setting this means that we need a very large number of fibres in order to minimize the *gap* between each fibre. Also, the assumption of fibres limits the deformation to the tangential direction and therefore no normal component of the stress is seen in a fibrous approximation.

Within the literature, there are several methods which attempt to overcome the limitation of the assumption of fibres. In this section we discuss the two most prominent. Namely, the Immersed Finite Element Method (IFEM) and the Finite Element Immersed Boundary Method (FE-IBM). A couple of other methods worth noting are those described in the articles [44] and [37]. In the case of a true immersed boundary, the velocity field may not necessarily be continuously differentiable - although it is always guaranteed to be continuous. In this case, the interpolation phase of the original immersed boundary method is only first-order accurate. The IIM [44] was introduced to maintain second-order accuracy in such a scenario. However, the IIM is also limited to fibres or boundaries.

Immersed Finite Element Method

The IFEM was introduced by Zhang *et al.* in 2004, [73], and is an extension of the Extended Immersed Boundary Method (EIBM) which was also introduced in 2004 by Wang and Liu [70]. Even though the IFEM is considered an extension of the EIBM, there is actually very little difference between the methods. The goal of the EIBM (and IFEM) was to allow the immersed boundary method to deal with thick immersed structures/solids which occupy a finite volume within the fluid. EIBM employed a finite difference approximation of the fluid but a finite element approximation of the immersed structure, whereas IFEM used finite elements for both the fluid and the immersed structure. The FEM approximation allows for a more realistic approximation of the stresses. An additional advantage of the EIBM (and IFEM) is the higher-order approximation of the Dirac delta distribution. This is accomplished using the reproducing kernel particle method (RKPM). Using RKPM improves the regularity of the approximation; the approximation used by Peskin [58] is only C^1-continuous, where as using RKPM gives a C^N-continuous approximation, where N is the degree of the polynomial approximation [73]. The added regularity improves the accuracy of the scheme. For the remainder of this sub-section, we work through the derivation of the IFEM formulation.

The formulation is set up similarly to §3.1.1. Let Ω denote a viscoelastic continuum containing, at time t, both an incompressible deformable structure, Ω_t^s, and an incom-

pressible fluid Ω_t^f such that:

$$\Omega_t^f \cup \Omega_t^s = \Omega \tag{3.1.25a}$$

$$\Omega_t^f \cap \Omega_t^s = \emptyset \tag{3.1.25b}$$

The reference configuration of the immersed structure is denoted by Ω_r^s. The Lagrangian variables are denoted by $\boldsymbol{s}$ and the Eulerian variables are denoted by $\boldsymbol{x}$. Just as in the previous section, the Cartesian position of an immersed structure particle in the Eulerian domain Ω is given by a mapping $\boldsymbol{X}\colon \Omega_r^s \times [0,T] \to \Omega$ such that $\boldsymbol{X}(\Omega_r^s, t) = \Omega_t^s$. Note that once again we have dropped the subscript t from the definition of Ω to illustrate the time-independence of the computational domain. The formulation of IFEM is based on the displacements of the immersed structure particles from the reference configuration. Therefore, Zhang *et al.* [73] also defined $\boldsymbol{u}^s(\boldsymbol{s},t) = \boldsymbol{X}(\boldsymbol{s},0) - \boldsymbol{X}(\boldsymbol{s},t)$ where it is assumed that the initial configuration of the material is the reference configuration. Therefore, we need to also introduce the velocity and acceleration of the immersed structure as:

$$\boldsymbol{v}^s = \frac{\partial \boldsymbol{u}^s}{\partial t} \tag{3.1.26a}$$

$$\boldsymbol{a}^s = \frac{\partial^2 \boldsymbol{u}^s}{\partial t^2} \tag{3.1.26b}$$

The starting point of the formulation is the usual conservation of momentum equation:

$$\rho \frac{D\boldsymbol{u}}{Dt} = \nabla \cdot \boldsymbol{\sigma} + \boldsymbol{f} \tag{3.1.27}$$

where ρ is the density of the material, $\boldsymbol{u}$ is the velocity field (not to confused with $\boldsymbol{u}^s$ which is the displacement of the immersed structure marker particles from reference configuration), $\boldsymbol{\sigma}$ is the Cauchy stress tensor and $\boldsymbol{f}$ contains the body forces. The above expression holds in both domains irrespective of material. Zhang *et al.* [73] assumed that the density of the immersed structure is different to that of the surrounding fluid. Therefore, they wrote the inertial force as:

$$\rho \frac{D\boldsymbol{u}}{Dt} = \begin{cases} \rho_f \dfrac{D\boldsymbol{u}}{Dt} & \text{in } \Omega_t^f \\ \rho_f \boldsymbol{a}^s + (\rho^s - \rho_f)\boldsymbol{a}^s & \text{in } \Omega_t^s \end{cases} \tag{3.1.28}$$

where ρ_f is the density of the fluid and ρ^s is the density of the immersed structure. For

simplicity, we omit any forces due to gravity, even though they are included in [73]. Zhang *et al.* defined the Cauchy stress tensor as:

$$\boldsymbol{\sigma} = \begin{cases} \boldsymbol{\sigma}_f & \text{in } \Omega_t^f \\ \boldsymbol{\sigma}_f + \boldsymbol{\sigma}^s - \boldsymbol{\sigma}_f & \text{in } \Omega_t^s \end{cases} \tag{3.1.29}$$

where $\boldsymbol{\sigma}^s$ is the Cauchy stress for the immersed structure and $\boldsymbol{\sigma}_f = -p\boldsymbol{I} + 2\eta_s\dot{\boldsymbol{\gamma}}$ is the Cauchy stress for the surrounding Newtonian fluid, where p is the pressure, η_s is the constant fluid viscosity and $\dot{\boldsymbol{\gamma}} = (\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T)/2$ is the rate-of-strain or rate-of-deformation tensor. The reader will notice that these definitions are *very* similar to the definitions given in the formulation of the FE-IBM (given in the next sub-section). Zhang *et al.* defined the force due to the immersed structure as:

$$\boldsymbol{F} = -(\rho^s - \rho_f)\boldsymbol{a}^s + \boldsymbol{\sigma}^s - \boldsymbol{\sigma}_f \qquad \text{in } \Omega_t^s \tag{3.1.30}$$

The above force term is composed of an inertial term (first) and an internal force term (second). Note that although the above expression is written in Eulerian form it is computed in Lagrangian form, which adds a level of difficulty due to the presence of the fluid stress tensor which, ideally, we would like to deal with in Eulerian form. The above force term is *spread* into the surrounding fluid using the delta function just as was done in the previous section §3.1.1.

Therefore, in strong form, we have:

$$\rho_f \frac{D\boldsymbol{u}}{Dt} = \nabla \cdot \boldsymbol{\sigma}_f + \boldsymbol{f} \qquad \text{in } \Omega \tag{3.1.31a}$$

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \text{in } \Omega \tag{3.1.31b}$$

$$\boldsymbol{f}(\boldsymbol{x}, t) = \int_{\Omega_r^s} \boldsymbol{F}(\boldsymbol{s}, t)\delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t))\, \mathrm{d}\boldsymbol{s} \tag{3.1.31c}$$

$$\boldsymbol{F} = -(\rho^s - \rho_f)\boldsymbol{a}^s + \boldsymbol{\sigma}^s - \boldsymbol{\sigma}_f \qquad \text{in } \Omega_t^s \tag{3.1.31d}$$

$$\boldsymbol{v}^s(\boldsymbol{s}, t) = \int_{\Omega} \boldsymbol{u}(\boldsymbol{x}, t)\delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t))\, \mathrm{d}\Omega \tag{3.1.31e}$$

Zhang *et al.* [73] discussed the strong form of the governing equations for both the fluid and the structure separately. A Petrov-Galerkin method was used for the weak form of the governing equations in the fluid domain and stabilized equal-order finite

elements were used for the approximation. For the governing equations in the structure domain, a standard Galerkin method was used for the weak form. Zhang *et al.* [73] employed an almost incompressible hyperelastic material satisfying the Mooney-Rivlin model to describe the immersed structure. The reason why they chose an almost incompressible model is due to the interpolated velocity field not accurately satisfying the incompressiblity constraint, which could introduce additional numerical error. We discuss this in more detail in §5.4.2. A detailed mathematical derivation and the proof of equivalence between strong and weak forms was illustrated in an article in 2007 by Wing Kam Liu *et al.* [45].

One of the main advantages of the IFEM (and EIBM) is the use of a higher-order approximation to the delta function. This is achieved by using the reproducing kernel particle method (RKPM) to *reproduce* a kernel that guarantees that polynomials of degree N are obtained exactly, i.e.:

$$x^N = \int_{-\infty}^{+\infty} y^N \phi(x-y)\,\mathrm{d}y \tag{3.1.32}$$

where ϕ is called the window function. In the EIBM article [70], the authors state that the ideal window function has an infinite support. In computational mechanics, this is impractical due to the required amount of computational time. Additionally, in the IBM, it is unphysical; analytically the force given in the fluid is only non-zero in the region where the immersed structure resides, therefore an infinite support implies that a given structure particle will impart a force on every fluid particle in the domain. To ensure that a finite support is present, a correction function C needs to be introduced such that:

$$\int_\Omega C(x;x-y)\frac{1}{r}\phi\left(\frac{x-y}{r}\right)\,\mathrm{d}\Omega = 1 \tag{3.1.33}$$

where r is a dilation parameter or refinement of the window function. In IFEM, the discretised delta function is then given by:

$$\phi_I(x) = C(x;x-y)\frac{1}{r}\phi\left(\frac{x-y}{r}\right)\Delta x_I \tag{3.1.34}$$

where the subscript I is used to denote a structure point and, as in the EIBM, ϕ is a

cubic spline. Due to the discrete delta function being zero outside of the support region, computationally IFEM constructs what is called a domain of *influence*. Essentially, the domain of influence is the set of fluid points contained in the support of the delta function centred at a structure point. Zhang *et al.* [73] found that the structure mesh width must be less than half the fluid mesh width otherwise spurious penetration of the structure can be seen. This is the same condition that was numerically validated by Peskin [56, 57, 58]. This condition is common throughout the immersed boundary method literature. The results given in the article [73] are quite impressive. They were the first authors, as far as we are aware, to use the immersed boundary method to simulate more than 1 falling sphere. The authors illustrated the application of the method to both falling rigid and flexible spheres. In order to model rigid spheres, the so called *material* modulus was increased. This would increase the stiffness of the method and therefore require much smaller timesteps than the flexible case. However, no mention of the stiffness of the method was given in the article [73]. The IFEM has been successfully applied to a range of biological systems, namely the heart, valve-fluid interaction, flow past a stent, red blood cell aggregation and more, all of which are illustrated in the article by Wing Kam Liu *et al.* [46].

Finite Element Immersed Boundary Method

From our perspective, one could say that a disadvantage of the IB method is the need to construct an approximation to the delta function. Even though the work of Zhang *et al.* [73] allows for a higher-order approximation, it still requires several additional steps; the construction of the window function, the construction of the correction function and the construction of the domain of influence. In 2003, Boffi *et al.* [14] introduced the idea of defining the delta function variationally through its action on a test function. This removed the need to construct a suitably regularised delta function and naturally allowed for a finite element type approach. This idea was later extended by the same authors to allow for thick immersed materials [17].

The formulation is set up in a similar way to the IFEM which we summarised in §3.1.2. Let $\Omega_r \in \mathbb{R}^m$ be the reference configuration of both the fluid and the immersed structure, whose domains are denoted by $\Omega_r^f \subseteq \Omega_r$ and $\Omega_r^s \subseteq \Omega_r$ respectively, such

that $\Omega_r = \Omega_r^f \cup \Omega_r^s$, where $\Omega_r^f \in \mathbb{R}^m$ and $\Omega_r^s \in \mathbb{R}^d$, $d \leq m$. In the usual continuum mechanical way, we define a sequence of configurations of our domain in time with the set $\Omega := \{\Omega_t \colon t \in [0,T]\}$ where Ω_t denotes the configuration of Ω at time t, such that $\Omega_t = \Omega_t^f \cup \Omega_t^s$ where $\Omega_t^f \subseteq \Omega_t$ and $\Omega_t^s \subseteq \Omega_t$. Define the pointwise map $\boldsymbol{X}$ from the reference configuration to a current configuration:

$$\boldsymbol{X} \colon \Omega_r \times [0,T] \to \Omega \tag{3.1.35}$$

Let $\boldsymbol{s} \in \Omega_r$ denote the reference (or Lagrangian) variable, which labels a material point in Ω_r, and $\boldsymbol{x} \in \Omega$ the Eulerian variable, as before. Therefore, for each material point $\boldsymbol{s} \in \Omega_r$ at time t, there exists an $\boldsymbol{x} = \boldsymbol{X}(\boldsymbol{s},t) \in \Omega$. The deformation gradient tensor is thus defined as:

$$\mathbb{F}(\boldsymbol{s},t) := \nabla_s \boldsymbol{X}(\boldsymbol{s},t) \tag{3.1.36}$$

whose determinant must be non-zero, as we want $\boldsymbol{X}$ to be invertible. The map $\boldsymbol{X}$ is used to denote the Cartesian position of a structure particle in the Eulerian domain.

We stress again here that the structure is considered to be part of the fluid where additional forces (and/or mass) can be seen. Boffi *et al.* assumed a Maxwell model for the immersed structure where the fluid component was assumed to be of the same form as the surrounding fluid so that the Cauchy stress tensor can be written as:

$$\boldsymbol{\sigma} := \begin{cases} \boldsymbol{\sigma}_f & \text{in } \Omega_t^f \\ \boldsymbol{\sigma}_f + \boldsymbol{\sigma}^s & \text{in } \Omega_t^s \end{cases} \tag{3.1.37}$$

where $\boldsymbol{\sigma}_f$ is the usual Cauchy stress tensor for a Newtonian fluid given by $\boldsymbol{\sigma}_f = -p\boldsymbol{I} + 2\eta_s\dot{\boldsymbol{\gamma}}$, where p is the pressure, η_s is the constant viscosity of the fluid and $\dot{\boldsymbol{\gamma}} = \frac{1}{2}(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T)$ is the rate-of-strain (or rate-of-deformation) tensor. Notice that the definition of the Cauchy stress tensor is very similar to that which is used in IFEM. The stress jump, $[\boldsymbol{\sigma}]$, across the interface between the fluid and the immersed structure

is defined as:

$$\begin{aligned}[\boldsymbol{\sigma}] &:= (\boldsymbol{n}^+ \cdot \boldsymbol{\sigma}^+) + (\boldsymbol{n}^- \cdot \boldsymbol{\sigma}^-) \\ &= (\boldsymbol{n}^+ \cdot \boldsymbol{\sigma}_f) + (\boldsymbol{n}^- \cdot \boldsymbol{\sigma}_f) + (\boldsymbol{n}^- \cdot \boldsymbol{\sigma}^s) \\ &= [\boldsymbol{\sigma}_f] + (\boldsymbol{n}^- \cdot \boldsymbol{\sigma}^s) \\ &= \boldsymbol{n}^- \cdot \boldsymbol{\sigma}^s \qquad \text{on } \partial\Omega_r^s \end{aligned} \tag{3.1.38}$$

where $\boldsymbol{n}^+$ is the outward normal to the fluid domain, $\boldsymbol{n}^-$ is the outward normal to the structure domain. Note $[\boldsymbol{\sigma}_f] = 0$ as there is no jump in fluid stress across the interface. Boffi *et al.* proceeded by applying the principle of virtual work to the momentum equation to give:

$$\int_{\mathcal{P}_t} \rho \frac{D\boldsymbol{u}}{Dt} \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x} = \int_{\mathcal{P}_t} (\nabla \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x} + \int_{\mathcal{P}_t} \boldsymbol{b} \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x} \tag{3.1.39}$$

where $\boldsymbol{u}$ is the velocity field, $\boldsymbol{\sigma}$ is the Cauchy stress tensor, $\boldsymbol{b}$ is the body force, $\boldsymbol{v}$ is the virtual displacement, $\frac{D\cdot}{Dt}$ is the material derivative, ρ is the density and $\mathcal{P}_t := \boldsymbol{X}(\mathcal{P}_r, t)$ is an arbitrary region of our current configuration $\mathcal{P}_t \subseteq \Omega_t$. Clearly, in this case, the principle of virtual work is equivalent to a variational formulation. Using integration by parts on the first integral of the right hand side and separating the region $\mathcal{P}_t$ into its fluid and structure parts, it was shown that [17]:

$$\begin{aligned}\int_{\mathcal{P}_t} (\nabla \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x} = &- \int_{\mathcal{P}_t} \boldsymbol{\sigma}_f : \nabla \boldsymbol{v} \, \mathrm{d}\boldsymbol{x} + \int_{\partial\mathcal{P}_t} (\boldsymbol{n} \cdot \boldsymbol{\sigma}_f) \cdot \boldsymbol{v} \, \mathrm{d}a \\ &- \int_{\mathcal{P}_r \cap \Omega_r^s} \mathbb{P}^s : \nabla_{\boldsymbol{s}} \boldsymbol{v}(\boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}\boldsymbol{s} + \int_{\partial\mathcal{P}_r \cap \Omega_r^s} (\boldsymbol{N} \cdot \mathbb{P}^s) \cdot \boldsymbol{v}(\boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}A \end{aligned} \tag{3.1.40}$$

where we have taken the integrals over $\mathcal{P}_t \cap \Omega_t^s$ to the reference configuration $\mathcal{P}_r \cap \Omega_r^s$, $\partial\mathcal{P}_t$ is the boundary of $\mathcal{P}_t$, $\boldsymbol{n}$ is the outward normal to the boundary of $\mathcal{P}_t$, $\boldsymbol{N}$ is the outward normal to the boundary of $\mathcal{P}_r$ defined by

$$\boldsymbol{n}\mathrm{d}a = J\mathbb{F}^{-T}\boldsymbol{N}\mathrm{d}A$$

where $\mathrm{d}a$ is an element of area in current configuration and $\mathrm{d}A$ is an element of area in reference configuration and, finally, $\mathbb{P}^s$ is the first Piola-Kirschoff stress tensor which

provides the stress with respect to the reference configuration and is required to satisfy

$$\int_{\partial\mathcal{P}_t} \boldsymbol{n}^T \cdot \boldsymbol{\sigma}^s \, \mathrm{d}a = \int_{\partial\mathcal{P}_r} \boldsymbol{N}^T \cdot \mathbb{P}^s \, \mathrm{d}A \tag{3.1.41}$$

Therefore, $\mathbb{P}^s$ is given by

$$\mathbb{P}^s(\boldsymbol{s},t) = J\boldsymbol{\sigma}^s(\boldsymbol{X}(\boldsymbol{s},t),t)\mathbb{F}^{-T}(\boldsymbol{s},t)$$

where $J = \det(\mathbb{F}(\boldsymbol{s},t))$. So far there has been no mention of the delta function. It is this step which makes the approach of Boffi *et al.* different from other formulations. The sifting property of the delta function is defined as

$$\boldsymbol{v}(\boldsymbol{X}(\boldsymbol{s},t)) = \int_{\mathcal{P}_t} \boldsymbol{v}(\boldsymbol{x})\boldsymbol{\delta}(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s},t)) \, \mathrm{d}\boldsymbol{x} \qquad\qquad \forall \boldsymbol{s} \in \mathcal{P}_r \tag{3.1.42}$$

provided $\boldsymbol{X}(\boldsymbol{s},t) \in \mathcal{P}_t$. The idea of Boffi *et al.* is to let $\boldsymbol{v}(\boldsymbol{x})$ in the sifting property above be the virtual displacement used in the principle of virtual work, thus defining the delta function variationally. The advantage of this approach is that there is no need to explicitly handle the delta function and therefore the construction of a regularised approximation, for numerical computation, is not required, but the delta function's effect is still retained by its action on the test function. Substituting the above, interchanging the order of integration and using the identities

$$\partial(\mathcal{U} \cap \mathcal{V}) = (\partial\mathcal{U} \cap \mathcal{V}) \cup (\mathcal{U} \cap \partial\mathcal{V}) \tag{3.1.43}$$

$$\int_{\mathcal{D}} (\nabla \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{v} \, \mathrm{d}\mathcal{D} = -\int_{\mathcal{D}} \boldsymbol{\sigma} \colon \nabla\boldsymbol{v} \, \mathrm{d}\mathcal{D} + \int_{\partial\mathcal{D}} (\boldsymbol{n} \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{v} \, \mathrm{d}a \tag{3.1.44}$$

yields the interaction terms found by Boffi *et al.* [17]:

$$\boldsymbol{G}(\boldsymbol{x},t) := \int_{\mathcal{P}_r} (\nabla_{\boldsymbol{s}} \cdot \mathbb{P}^s)\delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s},t)) \, \mathrm{d}\boldsymbol{s} \tag{3.1.45}$$

$$\boldsymbol{T}(\boldsymbol{x},t) := -\int_{\partial\mathcal{P}_r} (\boldsymbol{N} \cdot \mathbb{P}^s)\delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s},t)) \, \mathrm{d}A \tag{3.1.46}$$

We have only provided a brief illustration of the derivation here, for full details the

reader is referred to [17]. The full system in strong form is then given by:

$$\begin{aligned}
\rho\frac{D\boldsymbol{u}}{Dt} + \nabla p - \eta_s\nabla^2\boldsymbol{u} &= \boldsymbol{G} + \boldsymbol{T} + \boldsymbol{b} && \forall \boldsymbol{x} \in \Omega_t, t \in [0,T] \\
\nabla \cdot \boldsymbol{u} &= 0 && \forall \boldsymbol{x} \in \Omega_t, t \in [0,T] \\
\boldsymbol{G}(\boldsymbol{x},t) &= \int_{\Omega_r^s} (\nabla_{\boldsymbol{s}} \cdot \mathbb{P}^s)\delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s},t))\,\mathrm{d}\boldsymbol{s} && \forall \boldsymbol{x} \in \Omega_t, t \in [0,T] \\
\boldsymbol{T}(\boldsymbol{x},t) &= -\int_{\partial\Omega_r^s} (\boldsymbol{N} \cdot \mathbb{P}^s)\delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s},t))\,\mathrm{d}A && \forall \boldsymbol{x} \in \Omega_t, t \in [0,T] \\
\frac{\partial \boldsymbol{X}}{\partial t}(s,t) &= \boldsymbol{u}(\boldsymbol{X}(s,t),t) \\
&= \int_{\Omega_t} \boldsymbol{u}(\boldsymbol{x},t)\boldsymbol{\delta}(\boldsymbol{x} - \boldsymbol{X}(s,t))\,\mathrm{d}\Omega && \forall s \in \Omega_r^s, t \in [0,T]
\end{aligned} \tag{3.1.47}$$

Due to the ease to which this method lends itself to a finite element type approach, it becomes the natural choice for a spectral element method, and therefore is the basis on which we derive our proposed extension. In comparison to the IFEM, there is no need in the formulation of Boffi *et al.* to find the domain of influence. The domain of influence is simply an element of the Eulerian mesh which contains the immersed structure particle. Therefore, the only additional work that is required is to determine which elements of the Eulerian mesh contain the immersed structure particles.

3.2 Discontinuous/Variable Viscosity

In this section, we discuss the extensions of the formulation of Boffi *et al.* [14, 17] to allow for discontinuous or variable viscosity. First, we consider the case of discontinuous viscosity. The extension to discontinuous viscosity is fairly straightforward. We set up our formulation in the exact same way as was done in §3.1.2. Let $\Omega \subset \mathbb{R}^2$ be a viscoelastic continuum containing two sub-domains, denoted by Ω_t^f and Ω_t^s (at time t) such that:

$$\bar{\Omega}_t^f \cup \bar{\Omega}_t^s = \bar{\Omega} \tag{3.2.1}$$

$$\Omega_t^f \cap \Omega_t^s = \emptyset \tag{3.2.2}$$

where Ω_t^f denotes the region containing an incompressible fluid (assumed to be Newtonian here) and Ω_t^s denotes the region containing an incompressible viscoelastic structure. Let Γ_t denote the interface between the fluid and structure regions such that

$$\Gamma_t = \bar{\Omega}_t^f \cap \bar{\Omega}_t^s = \partial\Omega_t^f \cap \partial\Omega_t^s \tag{3.2.3}$$

Within each sub-domain, the conservation of linear and angular momentum is assumed to hold. Therefore, defining the total density ρ to be:

$$\rho = \begin{cases} \rho_f & \text{in } \Omega_t^f \\ \rho^s & \text{in } \Omega_t^s \end{cases} \tag{3.2.4}$$

and the Cauchy stress tensor as

$$\boldsymbol{\sigma} = \begin{cases} \boldsymbol{\sigma}_f & \text{in } \Omega_t^f \\ \boldsymbol{\sigma}_s & \text{in } \Omega_t^s \end{cases} \tag{3.2.5}$$

we know the momentum equation must hold in each sub-domain, i.e.:

$$\rho_a \frac{D\boldsymbol{u}}{Dt} = \nabla \cdot \boldsymbol{\sigma}_a + \rho_a \boldsymbol{b} \qquad \text{in } \Omega_t^a,\ a = f, s \tag{3.2.6}$$

Classically, these equations could be considered as a single nonlinear coupled system - the so-called *monolithic* approach - or considered separately and then coupled together via some suitable jump conditions - the so-called *partitioned* approach. In §3.1.2, both the IFEM and FE-IBM assumed that the stress jump across the interface is zero. However, if we wish to consider a discontinuous viscosity we cannot guarantee that the stress jump remains zero across the interface. Indeed, in two-phase flow, the stress jump is balanced by surface tension. Here, for simplicity, we assume that the stress jump is balanced by a, currently unknown, function $\boldsymbol{g}$. We follow the same ideas as Boffi *et al.* and consider the principle of virtual work. Before we continue, we require the fluid density and stress to be defined over the whole domain Ω as the immersed structure is considered to be part of the fluid. Following Boffi *et al.* [17, 13], define the

density as:

$$\rho = \begin{cases} \rho_f & \text{in } \Omega_t^f \\ \rho_f + \rho^s - \rho_f & \text{in } \Omega_t^s \end{cases} \tag{3.2.7}$$

and the Cauchy stress tensor as:

$$\boldsymbol{\sigma} = \begin{cases} \boldsymbol{\sigma}_f^{(1)} & \text{in } \Omega_t^f \\ \boldsymbol{\sigma}_f^{(2)} + \bar{\boldsymbol{\sigma}}^s & \text{in } \Omega_t^s \end{cases} \tag{3.2.8}$$

Note that $\boldsymbol{\sigma}_f^{(i)}$, $i = 1, 2$ denotes the Cauchy stress for a fluid with constant viscosity η_i, $i = 1, 2$ and $\bar{\boldsymbol{\sigma}}^s$ is the Cauchy stress tensor for the elastic structure. Indeed, let $\mathcal{W}_t$ denote an arbitrary region, at time t, in Ω such that:

$$\mathcal{W}_t = \mathcal{W}_t^f \cup \mathcal{W}_t^s \tag{3.2.9}$$

and let γ_t denote the portion of $\mathcal{W}_t$ which intersects the interface Γ_t, i.e. $\gamma_t = \mathcal{W}_t \cap \Gamma_t$. The principle of virtual work then gives:

$$\int_{\mathcal{W}_t} \rho \frac{D\boldsymbol{u}}{Dt} \cdot \boldsymbol{v} \, \mathrm{d}W + \int_{\mathcal{W}_t} \boldsymbol{\sigma} \colon \nabla \boldsymbol{v} \, \mathrm{d}W = \int_{\partial \mathcal{W}_t} (\boldsymbol{n} \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{v} \, \mathrm{d}S + \int_{\gamma_t} \boldsymbol{g} \cdot \boldsymbol{v} \, \mathrm{d}\Gamma + \int_{\mathcal{W}_t} \rho \boldsymbol{b} \cdot \boldsymbol{v} \, \mathrm{d}W \tag{3.2.10}$$

Separating into $\mathcal{W}_t$ into $\mathcal{W}_t^f$ and $\mathcal{W}_t^s$ respectively, and substituting the definitions of ρ and $\boldsymbol{\sigma}$ above, yields:

$$\begin{aligned} &\int_{\mathcal{W}_t} \rho_f \frac{D\boldsymbol{u}}{Dt} \cdot \boldsymbol{v} \, \mathrm{d}W + \int_{\mathcal{W}_t^f} \boldsymbol{\sigma}_f^{(1)} \colon \nabla \boldsymbol{v} \, \mathrm{d}W + \int_{\mathcal{W}_t^s} \boldsymbol{\sigma}_f^{(2)} \colon \nabla \boldsymbol{v} \, \mathrm{d}W + \int_{\mathcal{W}_t^s} \bar{\boldsymbol{\sigma}}^s \colon \nabla \boldsymbol{v} \, \mathrm{d}W = \\ &\int_{\partial \mathcal{W}_t^f} \left(\boldsymbol{n} \cdot \boldsymbol{\sigma}_f^{(1)} \right) \cdot \boldsymbol{v} \, \mathrm{d}S + \int_{\partial \mathcal{W}_t^s} \left(\boldsymbol{n} \cdot \boldsymbol{\sigma}_f^{(2)} \right) \cdot \boldsymbol{v} \, \mathrm{d}S + \int_{\partial \mathcal{W}_t^s} (\boldsymbol{n} \cdot \bar{\boldsymbol{\sigma}}^s) \cdot \boldsymbol{v} \, \mathrm{d}S - \int_{\gamma_t} [\boldsymbol{n} \cdot \boldsymbol{\sigma}] \cdot \boldsymbol{v} \, \mathrm{d}\Gamma + \\ &\int_{\gamma_t} \boldsymbol{g} \cdot \boldsymbol{v} \, \mathrm{d}\Gamma + \int_{\mathcal{W}_t} \rho_f \boldsymbol{b} \cdot \boldsymbol{v} \, \mathrm{d}W - \int_{\mathcal{W}_t^s} (\rho^s - \rho_f) \frac{D\boldsymbol{u}}{Dt} \cdot \boldsymbol{v} \, \mathrm{d}W + \int_{\mathcal{W}_t^s} (\rho^s - \rho_f) \boldsymbol{b} \cdot \boldsymbol{v} \, \mathrm{d}W \end{aligned} \tag{3.2.11}$$

Define $\boldsymbol{\sigma}_f = \boldsymbol{\sigma}_f^{(1)}$ in Ω^f and $\boldsymbol{\sigma}_f = \boldsymbol{\sigma}_f^{(2)}$ in Ω^s and using

$$\int_{\partial\mathcal{W}_t} (\boldsymbol{n}\cdot\boldsymbol{\sigma}_f)\cdot\boldsymbol{v}\,\mathrm{d}S = \int_{\partial\mathcal{W}_t^f}\left(\boldsymbol{n}\cdot\boldsymbol{\sigma}_f^{(1)}\right)\cdot\boldsymbol{v}\,\mathrm{d}S + \int_{\partial\mathcal{W}_t^s}\left(\boldsymbol{n}\cdot\boldsymbol{\sigma}_f^{(2)}\right)\cdot\boldsymbol{v}\,\mathrm{d}S - \int_{\gamma_t}[\boldsymbol{n}\cdot\boldsymbol{\sigma}_f]\cdot\boldsymbol{v}\,\mathrm{d}\Gamma \tag{3.2.12}$$

We may write, using $[\boldsymbol{n}\cdot\boldsymbol{\sigma}] = [\boldsymbol{n}\cdot\boldsymbol{\sigma}_f] + [\boldsymbol{n}\cdot\bar{\boldsymbol{\sigma}}^s]$:

$$\begin{aligned}
&\int_{\mathcal{W}_t}\rho_f\frac{D\boldsymbol{u}}{Dt}\cdot\boldsymbol{v}\,\mathrm{d}W + \int_{\mathcal{W}_t}\boldsymbol{\sigma}_f\colon\nabla\boldsymbol{v}\,\mathrm{d}W - \int_{\partial\mathcal{W}_t}(\boldsymbol{n}\cdot\boldsymbol{\sigma}_f)\cdot\boldsymbol{v}\,\mathrm{d}S = -\int_{\mathcal{W}_t^s}\bar{\boldsymbol{\sigma}}^s\colon\nabla\boldsymbol{v}\,\mathrm{d}W + \\
&\int_{\gamma_t}[\boldsymbol{n}\cdot\boldsymbol{\sigma}_f]\cdot\boldsymbol{v}\,\mathrm{d}\Gamma + \int_{\partial\mathcal{W}_t^s}(\boldsymbol{n}\cdot\bar{\boldsymbol{\sigma}}^s)\cdot\boldsymbol{v}\,\mathrm{d}S - \int_{\gamma_t}[\boldsymbol{n}\cdot\boldsymbol{\sigma}_f]\cdot\boldsymbol{v}\,\mathrm{d}\Gamma - \int_{\gamma_t}[\boldsymbol{n}\cdot\bar{\boldsymbol{\sigma}}^s]\cdot\boldsymbol{v}\,\mathrm{d}\Gamma + \\
&\int_{\gamma_t}\boldsymbol{g}\cdot\boldsymbol{v}\,\mathrm{d}\Gamma + \int_{\mathcal{W}_t}\rho_f\boldsymbol{b}\cdot\boldsymbol{v}\,\mathrm{d}W - \int_{\mathcal{W}_t^s}(\rho^s-\rho_f)\frac{D\boldsymbol{u}}{Dt}\cdot\boldsymbol{v}\,\mathrm{d}W + \int_{\mathcal{W}_t^s}(\rho^s-\rho_f)\boldsymbol{b}\cdot\boldsymbol{v}\,\mathrm{d}W
\end{aligned} \tag{3.2.13}$$

We notice that the two integrals involving the jump in fluid stress cancel and that we can apply the identity:

$$\int_{\mathcal{X}}(\nabla\cdot\boldsymbol{G})\cdot\boldsymbol{v}\,\mathrm{d}V = -\int_{\mathcal{X}}\boldsymbol{G}\colon\nabla\boldsymbol{v}\,\mathrm{d}V + \int_{\partial\mathcal{X}}(\boldsymbol{n}\cdot\boldsymbol{G})\cdot\boldsymbol{v}\,\mathrm{d}S \tag{3.2.14}$$

to the 2nd and 3rd integrals on the left hand side, as well as the 1st and 3rd integrals on the right hand side. Thus we obtain:

$$\begin{aligned}
&\int_{\mathcal{W}_t}\left(\rho_f\frac{D\boldsymbol{u}}{Dt} - \nabla\cdot\boldsymbol{\sigma}_f - \rho_f\boldsymbol{b}\right)\cdot\boldsymbol{v}\,\mathrm{d}W = \int_{\mathcal{W}_t^s}(\nabla\cdot\bar{\boldsymbol{\sigma}}^s)\cdot\boldsymbol{v}\,\mathrm{d}W - \int_{\gamma_t}(\boldsymbol{n}\cdot\bar{\boldsymbol{\sigma}}^s)\cdot\boldsymbol{v}\,\mathrm{d}\Gamma \\
&+\int_{\gamma_t}\boldsymbol{g}\cdot\boldsymbol{v}\,\mathrm{d}\Gamma - \int_{\mathcal{W}_t^s}(\rho^s-\rho_f)\frac{D\boldsymbol{u}}{Dt}\cdot\boldsymbol{v}\,\mathrm{d}W + \int_{\mathcal{W}_t^s}(\rho^s-\rho_f)\boldsymbol{b}\cdot\boldsymbol{v}\,\mathrm{d}W
\end{aligned} \tag{3.2.15}$$

We now face a rather difficult question: What is the unknown function $\boldsymbol{g}$? If we ignore the structure stress $\bar{\boldsymbol{\sigma}}^s$, then the problem reduces to an example of two-phase flow and therefore $\boldsymbol{g}$ is the surface tension term $-\tau\kappa\boldsymbol{n}$ where τ is the coefficient of surface tension and κ is the curvature. Alternatively, if we let $\boldsymbol{\sigma}_f^{(2)} = \boldsymbol{\sigma}_f^{(1)}$ then the formulation reduces to that of Boffi *et al.*, in which the interface condition was a zero stress jump. We then define our interface conditions to be a linear combination of these two so that $\boldsymbol{g} = -\tau\kappa\boldsymbol{n}$. Additionally, we note that if we let $\boldsymbol{g}$ be the surface tension force then we

have:

$$[\boldsymbol{n}\cdot\boldsymbol{\sigma}] = \boldsymbol{n}_1\cdot\boldsymbol{\sigma}_f^{(1)} + \boldsymbol{n}_2\cdot\boldsymbol{\sigma}_f^{(2)} + \boldsymbol{n}_2\cdot\bar{\boldsymbol{\sigma}}^s \tag{3.2.16}$$

$$= -\left(\boldsymbol{n}_2\cdot\boldsymbol{\sigma}_f^{(1)} - \boldsymbol{n}_2\cdot\boldsymbol{\sigma}_f^{(2)}\right) + \boldsymbol{n}_2\cdot\bar{\boldsymbol{\sigma}}^s \tag{3.2.17}$$

$$= -\left[\boldsymbol{n}_2\cdot\boldsymbol{\sigma}_f\right] + \boldsymbol{n}_2\cdot\bar{\boldsymbol{\sigma}}^s \tag{3.2.18}$$

$$\Rightarrow -2\tau\kappa\boldsymbol{n}_2 = \boldsymbol{n}_2\cdot\bar{\boldsymbol{\sigma}}^s \tag{3.2.19}$$

To be honest, we are unsure of whether the above interface condition, $[\boldsymbol{n}\cdot\boldsymbol{\sigma}] = -\tau\kappa\boldsymbol{n}_2$, is physically realistic. However, it is definitely consistent with respect to this derivation. To arrive at a strong form of our equations of motion, we proceed as was done in FE-IBM; that is, all integrals involving either Ω_t^s, or $\bar{\boldsymbol{\sigma}}^s$ are written with respect to a Lagrangian framework or reference configuration, Ω_r^s. Once this is complete we arrive at a strong form:

$$\rho_f\frac{D\boldsymbol{u}}{Dt} = \nabla\cdot\boldsymbol{\sigma}_f + \rho_f\boldsymbol{b} + \boldsymbol{d} + \boldsymbol{f} + \boldsymbol{t} \qquad \text{in } \Omega \tag{3.2.20}$$

$$\nabla\cdot\boldsymbol{u} = 0 \qquad \text{in } \Omega \tag{3.2.21}$$

$$\boldsymbol{d} = (\rho^s - \rho_f)\int_{\Omega_r^s}\left(\boldsymbol{b} - \frac{\partial^2\boldsymbol{X}}{\partial t^2}\right)\delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s},t))\,\mathrm{d}\Omega \tag{3.2.22}$$

$$\boldsymbol{f} = \int_{\Omega_r^s}\left(\nabla\cdot\bar{\mathbb{P}}^s\right)\delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s},t))\,\mathrm{d}\Omega \tag{3.2.23}$$

$$\boldsymbol{t} = -\int_{\partial\Omega_r^s}\left(\boldsymbol{N}\cdot\bar{\mathbb{P}}^s\right)\delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s},t))\,\mathrm{d}S \tag{3.2.24}$$

$$[\boldsymbol{n}\cdot\boldsymbol{\sigma}] = -\tau\kappa\boldsymbol{n}_2 \qquad \text{on } \Gamma \tag{3.2.25}$$

No additional machinery is required for the case of variable viscosity. We believe that the formulation of Boffi *et al.* is able to deal with variable viscosity, provided the variation is continuous across the interface Γ and that the assumption of zero stress jump across the interface is still valid. After all, the derivation given in §3.1.2 for FE-IBM, is general as it does not make any assumptions on the specific form of the Cauchy stress tensor. However, an issue will arise if the viscosity does not vary continuously across the interface. The biggest issue with the above derivation is that the function $\boldsymbol{g}$ is assumed to be unknown and currently, we are unaware of a general procedure to define $\boldsymbol{g}$. However, due to the assumption that a Maxwell model is used in Ω_t^s, i.e. we can separate our fluid and structure stresses, then it follows we can simply separate

the stress jump into its fluid and structure parts. This would mean that we would only need to determine the interface condition between non-Newtonian and Newtonian fluids.

3.3 Viscoelastic Flows

3.3.1 Previous Work

As far as we are aware, there are only two articles that have applied the immersed boundary method to a viscoelastic fluid; namely [66, 31]. Teran *et al.* [66], applied the immersed boundary method to an Oldroyd-B fluid flowing in a tube undergoing peristalsis with a prescribed wall motion, whilst Dillon and Zhuo [31] applied the immersed boundary method to sperm motility. Teran *et al.* defined peristalsis as *a wave of contraction passing along a fluid-bearing tube which results in net transport of the fluid in the wave direction*. Hence, peristaltic pumps work by propagating a wave along an elastic boundary which is modelled by the immersed boundary method in the article by Teran *et al.*. Peristaltic pumping is responsible for many physiological flows, e.g. rapid sperm transport from the uterus to the oviducts. Teran *et al.* noted that considerable progress has been made in regards to Newtonian fluids. However, most biological fluids are actually non-Newtonian due to the presence of suspended microstructures. The equations considered by Teran *et al.* and Dillon and Zhuo, are the Stokes Oldroyd-B (Stokes-OB) equations which are given by:

$$\nabla p - \eta_s \nabla^2 \boldsymbol{u} = \nabla \cdot \boldsymbol{T} \tag{3.3.1a}$$

$$\nabla \cdot \boldsymbol{u} = 0 \tag{3.3.1b}$$

$$\boldsymbol{T} + \lambda_1 \overset{\triangledown}{\boldsymbol{T}} = \eta_0 \left(\dot{\boldsymbol{\gamma}} + \lambda_2 \overset{\triangledown}{\dot{\boldsymbol{\gamma}}} \right) \tag{3.3.1c}$$

where η_s is the solvent viscosity, $\boldsymbol{u}$ is the velocity, p is the pressure and $\boldsymbol{T}$ is the extra-stress tensor which satisfies the Oldroyd-B constitutive equation given in (3.3.1c) where η_0 is the total viscosity and λ_1 and λ_2 are the characteristic relaxation and retardation times, respectively, which we defined earlier in §2.2.2. Teran *et al.* [66], and Dillon and Zhuo [31], incorporate the immersed boundary in the same manner as the original

immersed boundary method which we discussed in §3.1.1. Therefore, the full Stokes-OB-IB equations are given by:

$$\nabla p - \eta_s \nabla^2 \boldsymbol{u} = \nabla \cdot \boldsymbol{T} + \boldsymbol{f} \tag{3.3.2a}$$

$$\nabla \cdot \boldsymbol{u} = 0 \tag{3.3.2b}$$

$$\boldsymbol{T} + \lambda_1 \overset{\triangledown}{\boldsymbol{T}} = \eta_0 \left(\dot{\boldsymbol{\gamma}} + \lambda_2 \overset{\triangledown}{\dot{\boldsymbol{\gamma}}} \right) \tag{3.3.2c}$$

$$\boldsymbol{f}(\boldsymbol{x}, t) = \int_{\Omega_r^s} \boldsymbol{F}(\boldsymbol{s}, t) \delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}\Omega \tag{3.3.2d}$$

$$\frac{\partial \boldsymbol{X}}{\partial t} = \int_{\Omega} \boldsymbol{u}(\boldsymbol{x}, t) \delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}\Omega \tag{3.3.2e}$$

where $\boldsymbol{F}$ is the Lagrangian force density which satisfies a constitutive relation. In the article by Teran *et al.* [66], the force density $\boldsymbol{F}$ is given by:

$$\boldsymbol{F} = -\hat{\kappa} \left[\boldsymbol{X}(\boldsymbol{s}, t) - \boldsymbol{Z}(\boldsymbol{s}, t) \right] \tag{3.3.3}$$

Teran *et al.* interpreted these forces as Hookean springs with zero rest lengths connecting wall points $\boldsymbol{X}$ to so-called *tether* points $\boldsymbol{Z}$ where $\boldsymbol{Z}$ is the prescribed wall motion. A similar representation of the Lagrangian force density was employed by Dillon and Zhuo [31]. An additional method was described in the article by Dillon and Zhuo [31]. The method was first introduced by Bottino [18] and is called the immersed boundary Lagrangian mesh (IB-LM) method. The method considers a network of nodes (initially chosen randomly) that are cross-linked with viscoelastic links that could form or break. The result is that instead of a single Eulerian force density due to the immersed boundary, we have two force densities: one for the immersed elastic structure and one for the Lagrangian mesh. The delta function is used to *spread* the Lagrangian force densities for the elastic structure and the mesh to the Eulerian domain. In the article by Dillon and Zhuo [31], the viscoelastic links are described using a Jeffreys element and is formally equivalent to the Oldroyd-B constitutive equation. The main aim of the article by Dillon and Zhuo [31] was a comparison of the swimming speeds of sperm in four different fluids, namely Newtonian, Oldroyd-B (IB-OB), linear IB-LM and nonlinear IB-LM. The linear IB-LM used a Hookean spring for the elastic element in the Jeffreys model of the viscoelastic links, whereas the nonlinear IB-LM ignored the force from any viscoelastic link that stretched beyond a critical value. This suggests that the nonlinear IB-LM may be formally equivalent to the FENE model, although this

was not suggested in the article by Dillon and Zhuo [31]. Dillon and Zhuo found that the Oldroyd-B and Lagrangian mesh fluids give similar results in similar parameter regimes except that the model sperm swam faster in the IB-OB fluid than the IB-LM fluid.

Dillon and Zhuo [31] introduced the idea of an *Immersed Boundary Rheometer*. The rheometer is essentially a computational parallel-plate rheometer. The top and bottom plates are described by the immersed boundary method. Initially the IB points $\boldsymbol{X}_{top}$ are equally spaced along the top plate and connected by linear springs with stiffness S_{plate}. A constant horizontal force is applied to each of these IB points $\boldsymbol{X}_{top}$ for half the simulation and then the force is removed for the remainder of the simulation. The bottom plate is *tethered* to a set of points $\hat{\boldsymbol{X}}$ via springs of zero resting length and stiffness S_{bottom}. This gives a force of the form:

$$\boldsymbol{F}(\boldsymbol{s},t) = S_{bottom}\left(\hat{\boldsymbol{X}}(\boldsymbol{s}) - \boldsymbol{X}_{top}(\boldsymbol{s},t)\right) \tag{3.3.4}$$

Dillon and Zhuo found that the IB-OB and IB-LM models produced similar results on the immersed boundary rheometer for creeping flow. Teran *et al.* [66] illustrated a loss in reversibility for the Stokes-OB fluid. They considered a box containing a fluid and a piece of text written across the centre of the box. Two pegs inside the box are rotated in one direction (clockwise or anticlockwise) for a certain length of time T so that the text written across the centre is illegible. Then they are rotated in the opposite direction for the same length of time T so that the pegs return to their initial position. For Stokes flow of a Newtonian fluid, the text written across the centre is recovered, thus demonstrating the reversibility of Stokes flow of a Newtonian fluid. However, for Stokes flow of an Oldroyd-B fluid the text is not recovered thus demonstrating a loss of reversibility. Teran *et al.* mention that the text may be recovered for an Oldroyd-B fluid if the simulation is continued once the pegs return to their initial position. An interesting idea which was not mentioned in either the articles by Dillon and Zhuo or Teran *et al.*, is whether the immersed boundary rheometer and the reversibility simulation, can be used as immersed boundary benchmark problems. The reversibility simulation would be a particularly useful benchmark for the immersed finite element method and the finite element immersed boundary method (or indeed, any extension of the original IBM that can deal with thick structures) but we don't do this here.

3.3.2 Integration of Viscoelasticity into Boffi *et al.* Formulation

The integration of viscoelasticity into the formulation of Boffi *et al.* [17] is actually very straightforward due to the generality of the formulation. Following the ideas of Boffi *et al.*, the Cauchy stress tensor is defined as:

$$\boldsymbol{\sigma} = \begin{cases} \boldsymbol{\sigma}_f & \text{in } \Omega_t^f \\ \boldsymbol{\sigma}_f + \boldsymbol{\sigma}^s & \text{in } \Omega_t^s \end{cases} \tag{3.3.5}$$

where the fluid stress tensor, $\boldsymbol{\sigma}_f$, is defined as: $\boldsymbol{\sigma}_f = -p\boldsymbol{I} + 2\eta_s\boldsymbol{D} + \boldsymbol{\tau}$, where p is the pressure, η_s is the solvent viscosity, $\boldsymbol{D} = \frac{1}{2}(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^T)$ is the rate of strain tensor and $\boldsymbol{\tau}$ is the polymeric stress tensor which satisfies some constitutive law. Note that for the purposes of our derivation, we have separated the solvent and polymeric parts of the extra-stress tensor. The Cauchy stress tensor can then be written:

$$\boldsymbol{\sigma} = \begin{cases} \boldsymbol{\sigma}_N + \boldsymbol{\tau} & \text{in } \Omega_t^f \\ \boldsymbol{\sigma}_N + \boldsymbol{\tau} + \boldsymbol{\sigma}^s & \text{in } \Omega_t^s \end{cases} \tag{3.3.6}$$

where $\boldsymbol{\sigma}_N = -p\boldsymbol{I} + 2\eta_s\boldsymbol{D}$ is the Newtonian part of the fluid stress tensor. The issue, once again, lies in determining the interface conditions between the viscoelastic fluid and viscoelastic structure. In order to determine the interface conditions, we always consider the classical formulation; that is, solving the equations of motion for the viscoelastic fluid and viscoelastic structure separately and then coupling them together via some coupling condition. It is this coupling condition that determines the exact form of the function $\boldsymbol{g}$ given in §3.2. If the coupling conditions require the stress jump to be zero, then the derivation proceeds exactly as was done in §3.1.2 for FE-IBM, the only difference is that we have an additional term involving the divergence of the

polymeric stress tensor. The strong form can then be written as:

$$\rho_f \frac{D\boldsymbol{u}}{Dt} = \nabla \cdot \boldsymbol{\sigma}_N + \nabla \cdot \boldsymbol{\tau} + \rho_f \boldsymbol{b} + \boldsymbol{d} + \boldsymbol{f} + \boldsymbol{t} \qquad \text{in } \Omega \tag{3.3.7}$$

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \text{in } \Omega \tag{3.3.8}$$

$$\boldsymbol{d} = (\rho^s - \rho_f) \int_{\Omega_r^s} \left(\boldsymbol{b} - \frac{\partial^2 \boldsymbol{X}}{\partial t^2} \right) \delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}\Omega \tag{3.3.9}$$

$$\boldsymbol{f} = \int_{\Omega_r^s} (\nabla \cdot \mathbb{P}^s) \, \delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}\Omega \tag{3.3.10}$$

$$\boldsymbol{t} = -\int_{\partial\Omega_r^s} (\boldsymbol{N} \cdot \mathbb{P}^s) \, \delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}S \tag{3.3.11}$$

where, in this thesis, $\boldsymbol{\tau}$ satisfies the constitutive equation (2.2.24c). However, if the coupling conditions requires that the stress jump be balanced by some function $\boldsymbol{g}$, then we must use the extension given in §3.2. In that case, the strong form would be:

$$\rho_f \frac{D\boldsymbol{u}}{Dt} = \nabla \cdot \boldsymbol{\sigma}_N + \nabla \cdot \boldsymbol{\tau} + \rho_f \boldsymbol{b} + \boldsymbol{d} + \boldsymbol{f} + \boldsymbol{t} \qquad \text{in } \Omega \tag{3.3.12}$$

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \text{in } \Omega \tag{3.3.13}$$

$$\boldsymbol{d} = (\rho^s - \rho_f) \int_{\Omega_r^s} \left(\boldsymbol{b} - \frac{\partial^2 \boldsymbol{X}}{\partial t^2} \right) \delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}\Omega \tag{3.3.14}$$

$$\boldsymbol{f} = \int_{\Omega_r^s} (\nabla \cdot \mathbb{P}^s) \, \delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}\Omega \tag{3.3.15}$$

$$\boldsymbol{t} = -\int_{\partial\Omega_r^s} (\boldsymbol{N} \cdot \mathbb{P}^s) \, \delta(\boldsymbol{x} - \boldsymbol{X}(\boldsymbol{s}, t)) \, \mathrm{d}S \tag{3.3.16}$$

$$[\boldsymbol{n} \cdot \boldsymbol{\sigma}] = \boldsymbol{g} \qquad \text{on } \Gamma \tag{3.3.17}$$

Note that in the above strong form, we argue that the solvent viscosity could be discontinuous and also that any parameters in the viscoelastic constitutive law could also be discontinuous across the interface Γ, provided we could determine the function $\boldsymbol{g}$. This shows the generality of the formulation of Boffi *et al.* which, theoretically, has the ability to deal with a wide range of situations. However, we expect that the determination of $\boldsymbol{g}$ would become increasingly more complex, as more discontinuous parameters are introduced.

Chapter 4

Temporal Discretisation

In this thesis we are concerned with both steady and transient problems. Therefore, the equations need to discretised in both time and space. In this section we discuss the temporal discretisation of the method and in §5 we discuss the spatial discretisations. This chapter is constructed as follows: We begin by discussing the temporal discretisation of the Navier-Stokes equations and, in particular, the non-linear material derivative in the momentum equation and the non-linear terms in the constitutive equation. Then we discuss the temporal discretisation of the immersed boundary terms and briefly illustrate the stability issues that are known to exist within the immersed boundary method.

4.1 Navier-Stokes Equations

Throughout this thesis, our interest lies with the governing equations of fluid flow, or simplifications thereof. Primarily, we are concerned with the equations for a Newtonian or an Oldroyd-B fluid given in (2.2.11) and (2.2.24), respectively, and are given below:

$$\rho\frac{D\boldsymbol{u}}{Dt} + \nabla p - \eta_s\nabla^2\boldsymbol{u} = \nabla\cdot\boldsymbol{\tau} + \boldsymbol{f} \tag{4.1.1a}$$

$$\nabla\cdot\boldsymbol{u} = 0 \tag{4.1.1b}$$

$$\boldsymbol{\tau} + \lambda\overset{\triangledown}{\boldsymbol{\tau}} = 2\eta_p\dot{\boldsymbol{\gamma}} \tag{4.1.1c}$$

where ρ is the fluid density, η_s is the solvent viscosity, η_p is the polymeric viscosity, λ is the characteristic relaxation time, $\boldsymbol{u}$ is the velocity, p is the pressure, $\boldsymbol{\tau}$ is the polymeric stress tensor and $\dot{\boldsymbol{\gamma}} = \frac{1}{2}\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T$ is the rate of strain (or rate of deformation) tensor. The upper-convected derivative, $\overset{\triangledown}{\bullet}$, is as defined in (2.2.19) and the material derivative $\frac{D\bullet}{Dt}$ is as defined in (2.2.3). Throughout this thesis, we assume that the polymeric stress tensor in the momentum equation (4.1.1a), is explicitly treated. This decouples the constitutive equation for the polymeric stress tensor, (4.1.1c), from the momentum equation. Below we describe the methods we use in this thesis to deal with the material derivative and the non-linear deformation term which appears above. Throughout this thesis we use the superscript n to denote either a scalar, vector or tensor quantity at time $t^n = n\Delta t$, e.g. $\boldsymbol{u}^n(\boldsymbol{x}) = \boldsymbol{u}(\boldsymbol{x}, t^n)$ denotes the velocity of the fluid at time t^n, where Δt is the timestep length.

4.1.1 Material Derivative

The material derivative, appearing in both the momentum equation and the constitutive equation, are treated using the Operator Integration Factor Splitting (OIFS) scheme of Maday *et al.* [48]. The OIFS scheme offers a good balance between efficiency, stability and simplicity whilst maintaining high-order accuracy. Additionally due to the rapid spatial convergence of spectral methods used in this thesis, maintaining a high-order temporal scheme is desired. We employ a second-order OIFS scheme, where the material derivative of a vector or tensor $\boldsymbol{G}$ is given by:

$$\frac{D\boldsymbol{G}}{Dt} = \frac{\partial\boldsymbol{G}}{\partial t} + \boldsymbol{u}\cdot\nabla\boldsymbol{G} \tag{4.1.2a}$$

$$\approx \frac{1}{2\Delta t}\left(3\boldsymbol{G}^{n+1} - 4\tilde{\boldsymbol{G}}^{n+1} + \tilde{\tilde{\boldsymbol{G}}}^{n+1}\right) \tag{4.1.2b}$$

where $\tilde{\boldsymbol{G}}$ and $\tilde{\tilde{\boldsymbol{G}}}$ satisfy the pure convection problems:

$$\frac{\partial \tilde{\boldsymbol{G}}}{\partial \tau} = -\boldsymbol{u}^* \cdot \nabla \tilde{\boldsymbol{G}} \qquad \tau \in [t^n, t^{n+1}] \tag{4.1.3a}$$

$$\frac{\partial \tilde{\tilde{\boldsymbol{G}}}}{\partial \tau} = -\boldsymbol{u}^* \cdot \nabla \tilde{\tilde{\boldsymbol{G}}} \qquad \tau \in [t^{n-1}, t^{n+1}] \tag{4.1.3b}$$

where $\tilde{\boldsymbol{G}}(\boldsymbol{x}, \tau^0) = \boldsymbol{G}^n$ and $\tilde{\tilde{\boldsymbol{G}}}(\boldsymbol{x}, \tau^0) = \boldsymbol{G}^{n-1}$ and $\tau^m = m\Delta\tau$ and $\Delta\tau = \Delta t/M$ is the inner time step length with M being the number of internal time steps. The intermediate velocity, $\boldsymbol{u}^*$, at sub-timestep τ is given by:

$$\boldsymbol{u}^*(\tau) = \frac{(\tau - t^{n-1})}{\Delta t}\boldsymbol{u}^n + \left(1 - \frac{(\tau - t^{n-1})}{\Delta t}\right)\boldsymbol{u}^{n-1} \tag{4.1.4}$$

The pure convection problems are solved in *weak form* using a fourth order Runge-Kutta scheme. Using the OIFS scheme, the semi-discrete formulation of (4.1.1a) and (4.1.1b) is given by:

$$\frac{3\rho}{2\Delta t}\boldsymbol{u}^{n+1} + \nabla p^{n+1} - \eta_s \nabla^2 \boldsymbol{u}^{n+1} = \frac{2}{\Delta t}\tilde{\boldsymbol{u}}^{n+1} - \frac{1}{2\Delta t}\tilde{\tilde{\boldsymbol{u}}}^{n+1} + \nabla \cdot \boldsymbol{\tau}^n + \boldsymbol{f}^{n+1} \tag{4.1.5a}$$

$$\nabla \cdot \boldsymbol{u}^{n+1} = 0 \tag{4.1.5b}$$

where $\tilde{\boldsymbol{u}}$ and $\tilde{\tilde{\boldsymbol{u}}}$ satisfy the pure convection problems given in (4.1.3).

Validation

We wish to validate the operator integrating factor splitting scheme before it is used in our computations. We use the same model problem that was considered by Maday *et al.* [48] as it is simple to implement. Let $\Omega = [-1, 1]^2$ contain a Newtonian fluid, whose motion over the time period $t \in [0, T]$ is governed by (4.1.1a) and (4.1.1b) with $\rho = 1$, $\eta = 1/2\pi^2$,$\boldsymbol{\tau} \equiv 0$ and $\boldsymbol{f} \equiv 0$. The initial conditions are given by:

$$u(x, y, 0) = -\cos(\pi x)\sin(\pi y) \tag{4.1.6a}$$

$$v(x, y, 0) = \sin(\pi x)\cos(\pi y) \tag{4.1.6b}$$

where $\boldsymbol{u} = (u, v)$ is the velocity of the fluid. The exact solution of this problem is given by:

$$u(x, y, t) = (-\cos(\pi x)\sin(\pi y))\, e^{-t} \tag{4.1.7a}$$

$$v(x, y, t) = (\sin(\pi x)\cos(\pi y))\, e^{-t} \tag{4.1.7b}$$

$$p(x, y, t) = \frac{-(\cos(2\pi x) + \cos(2\pi y))\, e^{-2t}}{4} \tag{4.1.7c}$$

where p is the pressure. As we are validating a second-order scheme, we use the analytical solution to impose an additional condition on the velocity at $t = \Delta t$ (i.e. after a single timestep). The reason is that in the first step, there is a very large amount of error due to $\tilde{\tilde{\boldsymbol{u}}} = 0$. The boundary conditions we employ are different to those of Maday *et al.* [48]; we employ Dirichlet (rather than periodic) boundary conditions derived from evaluating the exact solution on the domain boundary. At inflow and outflow, $x = \pm 1$, we have:

$$u(\pm 1, y, t) = (-\cos(\pm\pi)\sin(\pi y))\, e^{-t} = \sin(\pi y)\, e^{-t} \tag{4.1.8a}$$

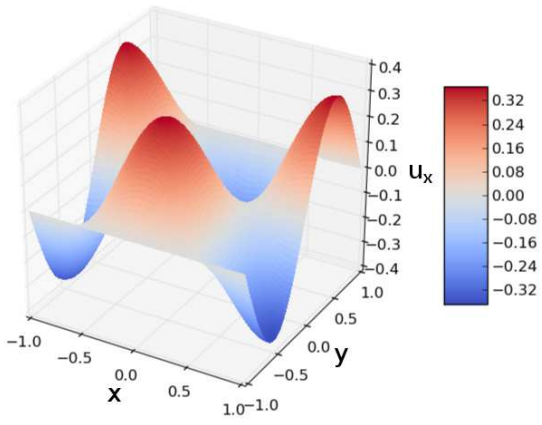
$$v(\pm 1, y, t) = (\sin(\pm\pi)\cos(\pi y))\, e^{-t} = 0 \tag{4.1.8b}$$

and at the top and bottom, $y = \pm 1$, we have:

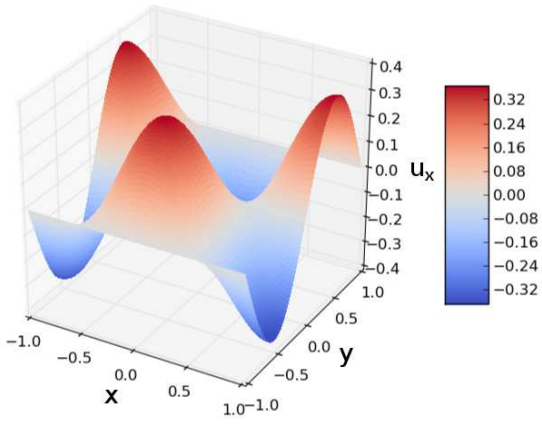
$$u(x, \pm 1, t) = (-\cos(\pi x)\sin(\pm\pi))\, e^{-t} = 0 \tag{4.1.9a}$$

$$v(x, \pm 1, t) = (\sin(\pi x)\cos(\pm\pi))\, e^{-t} = -\sin(\pi x)\, e^{-t} \tag{4.1.9b}$$

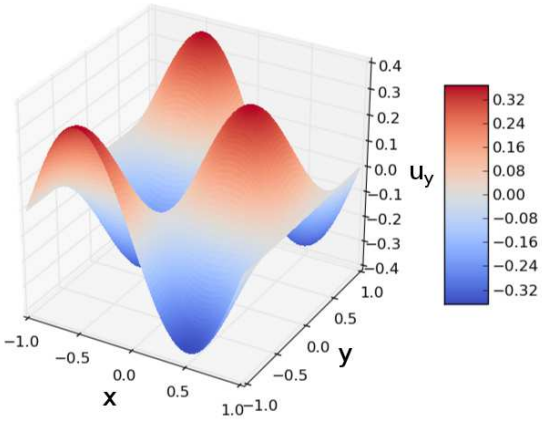
Note that when implementing these boundary conditions, we must make sure that they are calculated at time level $t^{n+1} = (n+1)\Delta t$ otherwise there is a mis-match between the boundary conditions and the computed solution. As we have specified the velocity at t^0 and t^1, the simulation begins at t^2. In Fig. 4.1, we compare the computed solution for $\boldsymbol{u}_x$, $\boldsymbol{u}_y$ and pressure against the analytical solution. It is evident from that figure, that we have good agreement between our computed solution and the analytical solution. Fig. 4.2 illustrates the H^1 norm of the velocity error at the end of the simulation, $T = 1$, for all the timesteps considered in this validation. The plot not only illustrates approximately second-order convergence rate but also gives good agreement to the results of Maday *et al.* [48] (their Fig. 2). Now that we have validated our implementation of the OIFS scheme, we move on to discuss how we handle the



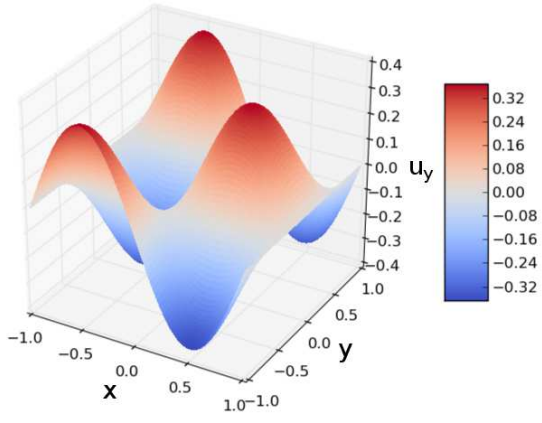
(a) x-component of the computed velocity solution.



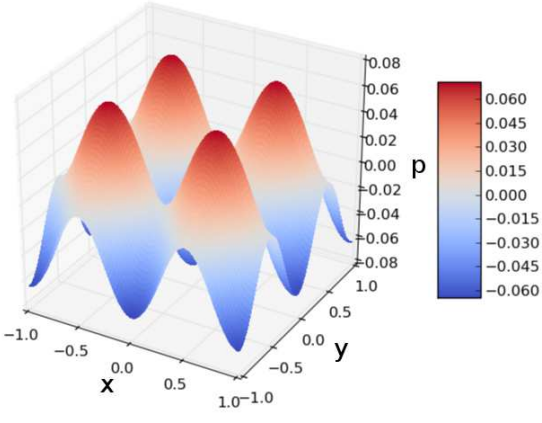
(b) x-component of the analytical velocity solution.



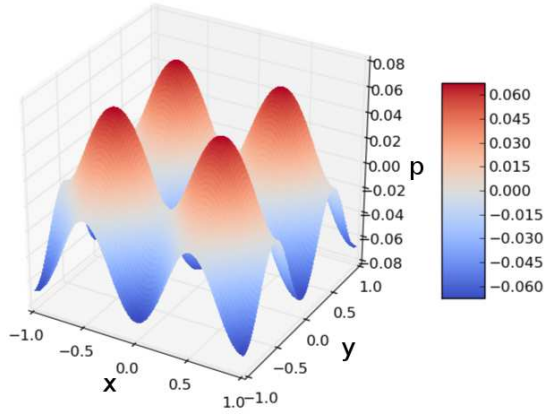
(c) y-component of the computed velocity solution.



(d) y-component of the analytical velocity solution.



(e) Computed pressure solution.



(f) Analytical pressure solution.

Figure 4.1: Velocity and pressure plots at the end of the simulation, $T = 1$, for $h_f = 1/2$, $N = 12$ and $\Delta t = 10^{-3}$. Note that the number of internal timesteps used was $M = 1$.

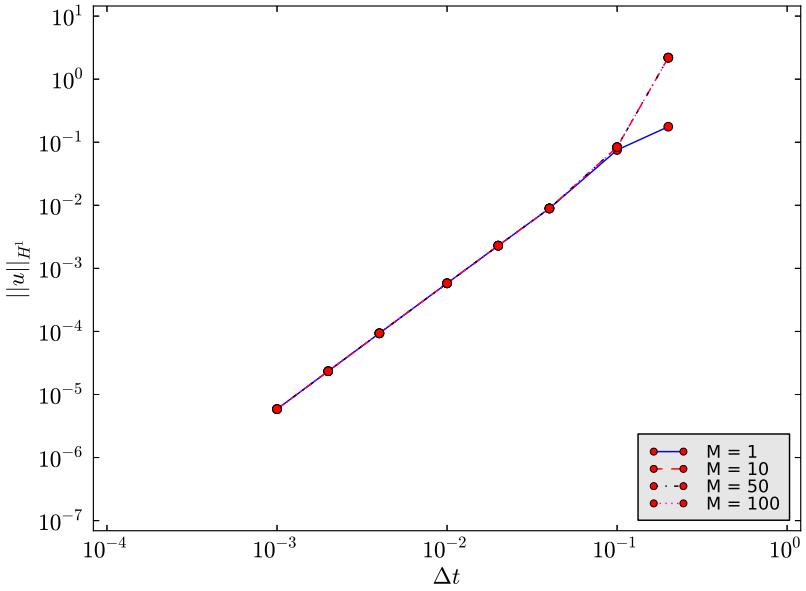


Figure 4.2: H^1 norm of the velocity error for different timesteps when $N = 12$ and $h_f = 1/2$. The parameter M is the number of internal timesteps.

non-linear terms which appear in the constitutive equation for the polymeric stress.

4.1.2 Non-linear Deformation Terms

Throughout this thesis, the non-linear deformation terms that appear in the constitutive equation for the polymeric stress, $\boldsymbol{E}(\boldsymbol{\tau}, \boldsymbol{u}) = \boldsymbol{\tau}(\nabla \boldsymbol{u}) + (\nabla \boldsymbol{u})^T \boldsymbol{\tau}$, are always treated explicitly. It is expected that as the elasticity of the fluid is increased, one may require a more implicit treatment of these terms. However, for simplicity we employ a higher-order explicit method. We will use the notation $\boldsymbol{E}^n(\boldsymbol{\tau}, \boldsymbol{u}) = \boldsymbol{E}(\boldsymbol{\tau}^n, \boldsymbol{u}^n)$ to denote the non-linear terms at time $t^n = n\Delta t$. The temporal scheme we employ is the Adams-Bashforth technique. The $(J-1)$-th order Adams-Bashforth formula is given by [40]:

$$\boldsymbol{F}^{n+1} \approx \sum_{j=1}^{J-1} \beta_j \boldsymbol{F}^{n+1-j} \tag{4.1.10}$$

where β_j, $j = 1, \ldots, J-1$, are known coefficients. Here, we choose $J = 3$ to give a second-order Adams-Bashforth method, whose coefficients are given by:

$$\beta_1 = \frac{3}{2}, \quad \beta_2 = -\frac{1}{2} \tag{4.1.11}$$

The semi-discrete form of constitutive equation for the polymeric stress tensor (4.1.1c) is given by:

$$\left(1 + \frac{3\lambda}{2\Delta t}\right)\boldsymbol{\tau}^{n+1} = \eta_p \dot{\boldsymbol{\gamma}}^{n+1} + \frac{2}{\Delta t}\tilde{\boldsymbol{\tau}}^{n+1} - \frac{1}{2\Delta t}\tilde{\tilde{\boldsymbol{\tau}}}^{n+1} + \lambda\left(\frac{3}{2}\boldsymbol{E}^n(\boldsymbol{\tau}, \boldsymbol{u}) - \frac{1}{2}\boldsymbol{E}^{n-1}(\boldsymbol{\tau}, \boldsymbol{u})\right) \tag{4.1.12}$$

where $\tilde{\boldsymbol{\tau}}$ and $\tilde{\tilde{\boldsymbol{\tau}}}$ satisfy the pure convection problems in (4.1.3). The full semi-discrete equations are then given by:

$$\frac{3\rho}{2\Delta t}\boldsymbol{u}^{n+1} + \nabla p^{n+1} - \eta_s \nabla^2 \boldsymbol{u}^{n+1} = \frac{2}{\Delta t}\tilde{\boldsymbol{u}}^{n+1} - \frac{1}{2\Delta t}\tilde{\tilde{\boldsymbol{u}}}^{n+1} + \nabla \cdot \boldsymbol{\tau}^n + \boldsymbol{f}^{n+1} \tag{4.1.13a}$$

$$\nabla \cdot \boldsymbol{u}^{n+1} = 0 \tag{4.1.13b}$$

$$\left(1 + \frac{3\lambda}{2\Delta t}\right)\boldsymbol{\tau}^{n+1} = \eta_p \dot{\boldsymbol{\gamma}}^{n+1} + \frac{2}{\Delta t}\tilde{\boldsymbol{\tau}}^{n+1} - \frac{1}{2\Delta t}\tilde{\tilde{\boldsymbol{\tau}}}^{n+1} + \lambda\left(\frac{3}{2}\boldsymbol{E}^n(\boldsymbol{\tau}, \boldsymbol{u}) - \frac{1}{2}\boldsymbol{E}^{n-1}(\boldsymbol{\tau}, \boldsymbol{u})\right) \tag{4.1.13c}$$

Note that the equation (4.1.13a) above implies that the source term is dealt with implicitly. Unfortunately, this is not true but rather it is approximated using an explicit scheme which we discuss in the next section.

4.2 Immersed Boundary Method

It has been documented throughout the IBM literature, that the IBM can be stiff and therefore a small timestep may be required in order to maintain stability of the method [15, 16, 67, 64, 65, 9, 52]. The standard approach for the immersed boundary method is the so-called Forward Euler/Backward Euler (FE/BE) method. In the FE/BE method, the Laplacian and gradient terms which appear in the momentum equation are dealt with implicitly, while the source term corresponding to the immersed boundary is dealt

with using the forward Euler method, i.e. $\boldsymbol{f}^{n+1} \approx \boldsymbol{f}^n$. The movement of the immersed boundary (3.1.14e) is then calculated using a semi-implicit Euler method given by:

$$\boldsymbol{X}^{n+1} = \boldsymbol{X}^n + \Delta t \boldsymbol{u}^{n+1}(\boldsymbol{X}^n) \tag{4.2.1}$$

It is clear that the order of the method with respect to time is at most 1. Typically the restriction on timestep is caused by a combination of diffusive and fibre effects and is therefore most severe for cases with a large force or small viscosity. Stockie and Wetton [65, 64] identified additional fibre modes, due to the presence of an immersed fibre, via a linear modal analysis. These fibre modes attribute to the additional stiffness of the method and they found that the stiffest modes arose from the class of tangential fibre oscillations. It was documented by Stockie and Wetton [64] that in the case of large fibre force or small viscosity the maximum allowable timestep can be as small as 10^{-5}. In order to increase the maximum allowable timestep, one looks to treat the immersed boundary terms more implicitly. Indeed, Peskin [57] proposed using an approximate-implicit method for the immersed boundary terms. Tu and Peskin [67] compared this approximate-implicit method against explicit and implicit approaches. It was found that the explicit method, as expected, broke down spectacularly at intermediate timesteps but that the approximate-implicit method performed almost as well as the fully implicit method. Unfortunately, the fully implicit method was deemed impractical due to the large computational times and therefore the approximate-implicit method was determined to be a viable alternative. However, as documented by Stockie and Wetton [64], the problem with such an approach is that it increases the computational time of the method, effectively destroying any advantage that would be achieved using a larger timestep; it also makes the discretisation considerably more complex.

Stockie and Wetton [64] proposed using a high-order explicit method to approximate the Laplacian, gradient and source terms. They compared this routine against the standard FE/BE method and an iterative semi-implicit approach. They found that the high-order explicit method, fourth order Runge-Kutta, performed comparably well to the standard FE/BE method but that the iterative semi-implicit approach allowed larger timesteps. However, not only was the iterative semi-implicit approach slower computationally, it did not perform as well in terms of volume loss. The problem of volume loss is not discussed here but instead is deferred until section §5.4.2.

The work by Tu and Peskin [67] and Stockie and Wetton [65, 64] was related to the original immersed boundary method. For the FE-IBM, Boffi *et al.* [15] compared the semi-implicit FE/BE method of the immersed boundary against an implicit BE/BE method and found that the BE/BE method was unconditionally stable although completely impractical. Newren *et al.* [52] also considered a fully implicit formulation and found unconditional stability. For the FE/BE method, Boffi *et al.* [16] determined a CFL condition for the FE-IBM. It is difficult to compare the work of Boffi *et al.* [15] with that of Tu and Peskin [67] or Stockie and Wetton [64] as the FE-IBM does not use the discrete delta function. Stockie and Wetton [65] considered a classical representation of the immersed boundary problem and found that the theoretical maximum allowable timestep was larger than when the discrete delta function was used in [64]. Thus the use of the discrete delta function also affects the timestep and because FE-IBM does not use the discrete delta function, it becomes difficult to do a direct comparison.

As far as we are aware, no-one has considered using a high-order method for the movement of the immersed boundary. Here, we compare the use of various methods, of varying orders, to determine the effect on the stability, temporal accuracy, timestep size. In all of the routines given below, we follow the ideas of Newren *et al.* [52] and Boffi *et al.* [15] by considering an appropriate energy for the system. Following Newren *et al.* [52], we define the energy at time $t^n = n\Delta t$ as:

$$E^n(\boldsymbol{u}, \boldsymbol{X}) = E(\boldsymbol{u}^n, \boldsymbol{X}^n) = \alpha \left\| \boldsymbol{u}^n \right\|^2_{L^2(\Omega)} + \beta \left(\boldsymbol{X}^n, -\kappa \frac{\partial^2 \boldsymbol{X}^n}{\partial s^2} \right)_{L^2(\Gamma)} \tag{4.2.2}$$

where α and β are constant coefficients which are taken to be equal to one in all cases except when the third-order backward differentiation formula is used. In an unconditionally stable scheme, the energy must decrease as the simulation progresses from time t^n to time t^{n+1}, in other words $E^{n+1}(\boldsymbol{u}, \boldsymbol{X}) - E^n(\boldsymbol{u}, \boldsymbol{X}) \leq 0$. However, for a conditionally stable scheme additional energies involving the time step length Δt can be seen. In this thesis, these additional energies can be found on the right hand side

of the energy inequality; in other words, we have situations of the following form:

$$E^{n+1}(\boldsymbol{u}, \boldsymbol{X}) - E^n(\boldsymbol{u}, \boldsymbol{X}) \leq F(\boldsymbol{u}, \boldsymbol{X}, t^n, t^{n+1}) \tag{4.2.3}$$

where the function F can depend on $\boldsymbol{u}$ and $\boldsymbol{X}$ at both times t^n and t^{n+1}. To calculate the energy in the system we follow the same procedure that was given by Newren *et al.*. As the procedure is the same for each routine, we only derive a single estimate for the Semi-Implicit Euler method. The derivation can be found in the appendix C.2. A table will be produced for each routine, illustrating the size of the timestep that causes the computation to break down. A tick (✓) will be used to identify a completed run and a dash (-) will be used to identify an incompleted run. All simulations were run until $T = 1$.

First we consider the Semi-Implicit Euler method for the immersed boundary evolution equation (4.2.1) as we will use this as a basis for the comparison against the other methods. In all the cases considered below, we linearise the momentum equation (thus removing the convection term) to give the unsteady Stokes equations:

$$\rho \frac{\partial \boldsymbol{u}}{\partial t} + \nabla p - \eta_s \nabla^2 \boldsymbol{u} = \boldsymbol{f} \tag{4.2.4a}$$

$$\nabla \cdot \boldsymbol{u} = 0 \tag{4.2.4b}$$

where $\boldsymbol{f}$ is the Eulerian force density given to the fluid by the immersed membrane. In all the cases below, we consider the same model problem. That is a closed circular membrane immersed in a Newtonian fluid. An example of the geometry is given in Fig. 4.3. Let $\Omega = [0, 1]^2$ such that $\bar{\Omega} = \bar{\Omega}(t) = \bar{\Omega}^f(t) \cup \bar{\Omega}^s(t)$ where $\Omega^f(t)$ denotes the fluid domain and $\Omega^s(t)$ denotes the immersed elastic membrane. As we are dealing with a membrane, we define $\Gamma(t) = \Omega^s(t)$. The Cartesian position of the membrane $\boldsymbol{X}$ is parameterised by arclength $s \in [0, 2\pi R]$ where $R = 0.25$ is the radius of the immersed circular membrane. Throughout this section, we let $\rho = 1$ and consider a variety of values of the viscosity, IB stiffness and timestep length. In this example, the Lagrangian force density of the immersed membrane is directed inwards towards the centre of the circular membrane. This causes a loss in area contained inside the membrane. We return to this phenomenon in §5.4.2.

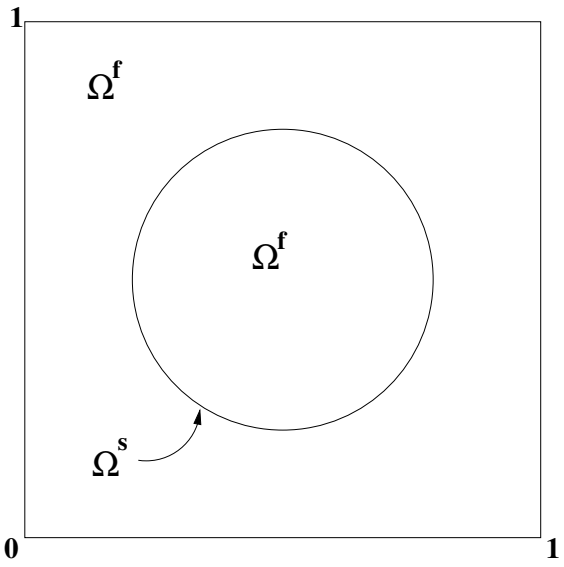


Figure 4.3: Example geometry of a static closed membrane immersed in a fluid.

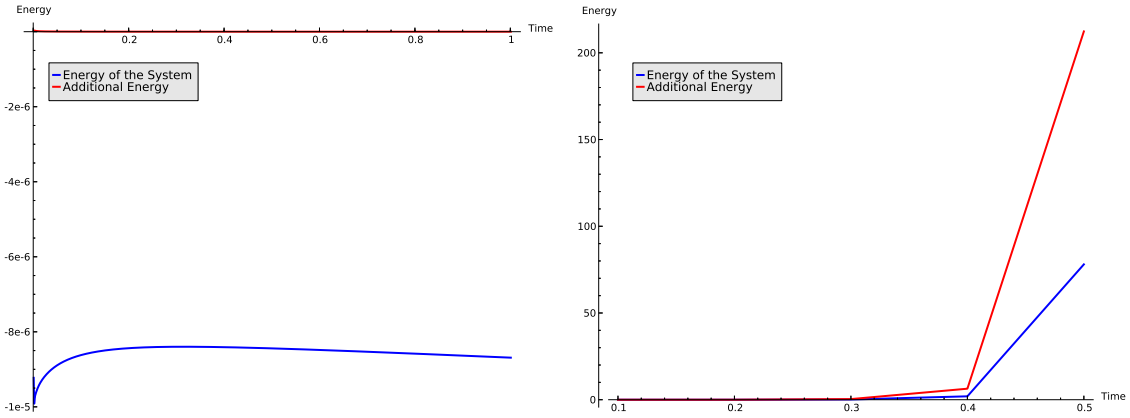
4.2.1 Semi-Implicit Euler Method

The semi-implicit Euler method (4.2.1) is the most commonly used method to evolve the location of the immersed membrane. We use the results given in this section as a basis for the comparison against other methods. The discrete energy of the system is given by:

$$\begin{aligned}
&\left\|\boldsymbol{u}_N^{n+1}\right\|_\Omega^2 + \kappa \sum_{i=1}^{K_b} \left[\frac{\left(\boldsymbol{X}_{h,i}^{n+1} - \boldsymbol{X}_{h,i-1}^{n+1}\right)^2}{\Delta s_i} \right] - \left\|\boldsymbol{u}_N^{n}\right\|_\Omega^2 - \kappa \sum_{i=1}^{K_b} \left[\frac{\left(\boldsymbol{X}_{h,i}^{n} - \boldsymbol{X}_{h,i-1}^{n}\right)^2}{\Delta s_i} \right] \\
&\leq 2\kappa\Delta t \sum_{i=1}^{K_b} \left[\frac{\boldsymbol{X}_{h,i}^{n+1} - \boldsymbol{X}_{h,i-1}^{n+1}}{\Delta s_i} - \frac{\boldsymbol{X}_{h,i}^{n} - \boldsymbol{X}_{h,i-1}^{n}}{\Delta s_i} \right] \left(\boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i}^{n}) - \boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i-1}^{n}) \right)
\end{aligned} \tag{4.2.5}$$

The above energy estimate is derived in Appendix C.2. The estimate is a theoretical bound and therefore the left hand side of the above inequality should never go beyond the right hand side. As we mentioned above, the left hand side of the above estimate is the difference between the energy at each time level and the right hand side are additional energies involving the time step length Δt present due to the temporal scheme being conditionally stable. What we hope is that the left hand side is always less than or equal to zero as this implies stability.

First of all, we consider two energy profiles. In those energy profiles, we fix $h_f = 1/8$, $N = 8$, $\eta_s = 1$ and $\kappa = 1$. Then we consider two different values for $\Delta t = 0.001, 0.1$. Fig. 4.4 illustrates these energy profiles. Clearly when $\Delta t = 0.001$, we have a stable



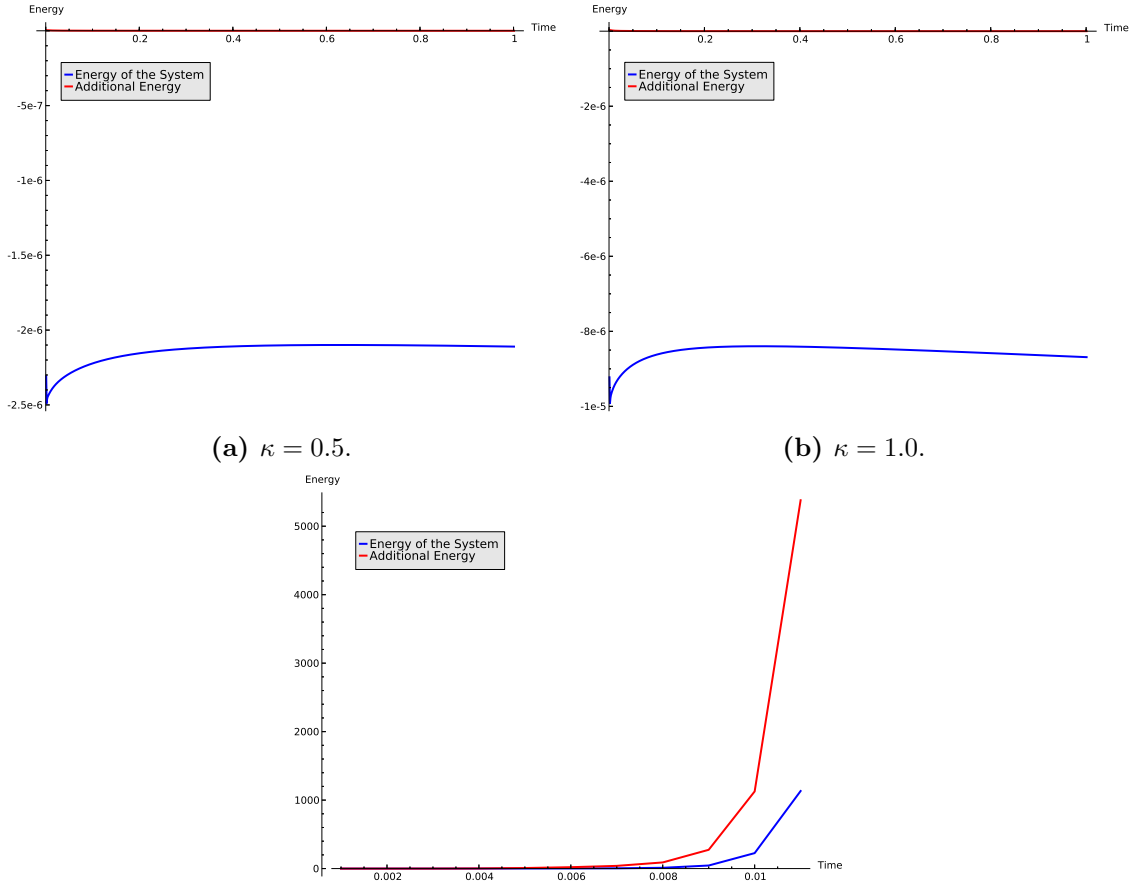
(a) $\Delta t = 0.001$. **(b)** $\Delta t = 0.1$.

Figure 4.4: Energy profiles for $h_f = 1/8$, $N = 8$, $\eta_s = 1$, $\kappa = 1$ and $\Delta t = 0.001, 0.1$. The blue line corresponds to the left hand side of the energy estimate (total energy between timesteps) and the red line is the right hand side of the energy estimate (energy due to the immersed boundary).

system. However we see that the energy starts to diverge at around $T = 0.3$ when $\Delta t = 0.1$. This illustrates that when $\Delta t = 0.1$ the system is unstable. Note that the blue line is always underneath the red line.

We are interested to see the effect of high forcing parameter (or IB stiffness) κ on such an example. We fix $\Delta t = 0.001$, $h_f = 1/8$, $N = 8$ and $\eta_s = 1$ while varying $\kappa = 0.5, 1.0, 50.0$. Fig. 4.5 illustrates the three energy profiles for the different values of κ. Clearly when $\kappa = 0.5, 1.0$ we have a stable system however when $\kappa = 50$ the system becomes unstable. Additionally, we found that when κ increased, but the timestep was chosen so that the system remains stable, there is a large loss of area. This was first noticed by Newren *et al.* [52] who suggested that the loss in area was a stabilising effect. While we don't believe that the loss in area is a cause of stability, it is a consequence of the fact that the large stiffness will produce larger spurious velocities and therefore a larger loss in area will be seen.

It is also known that the immersed boundary suffers from stability problems at small viscosities. Therefore, we fix $\Delta t = 0.001$, $h_f = 1/8$, $N = 8$ and $\kappa = 1$ while varying



(a) $\kappa = 0.5$.

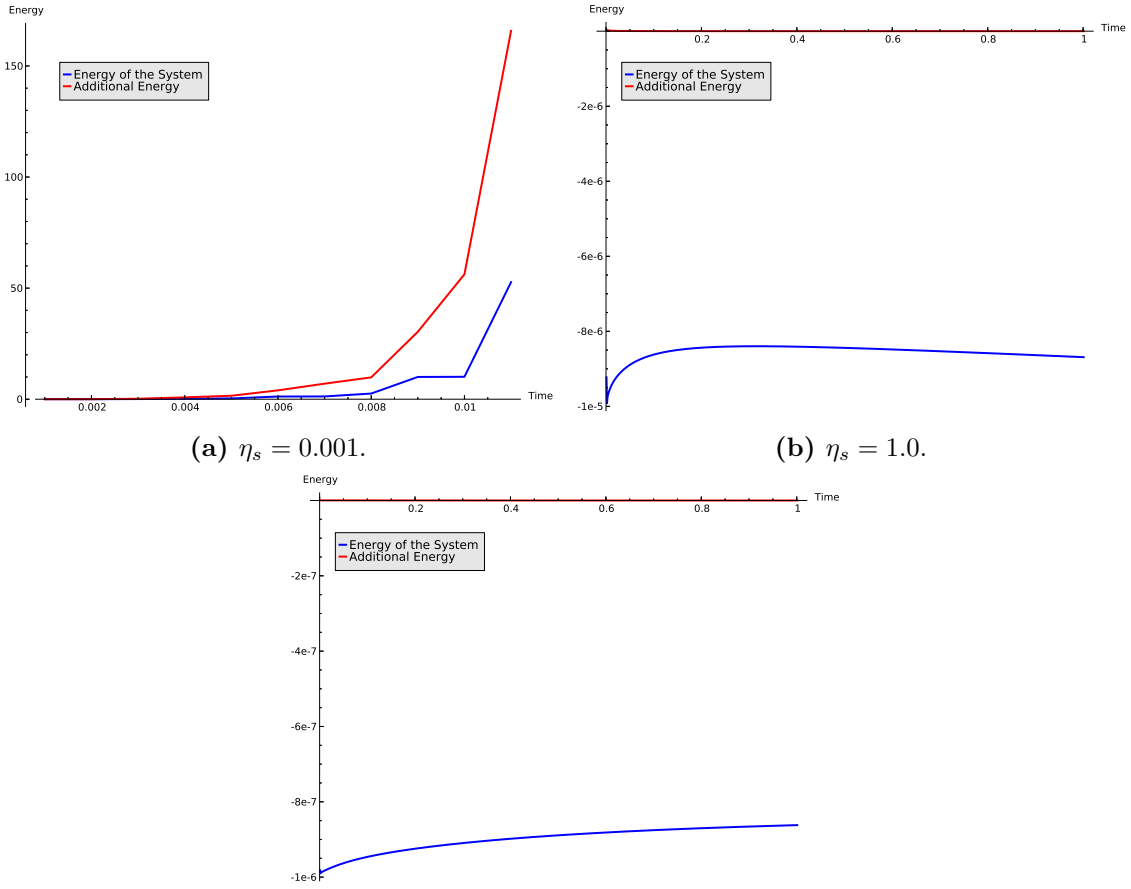
(b) $\kappa = 1.0$.

(c) $\kappa = 50.0$.

Figure 4.5: Energy profiles for $h_f = 1/8$, $N = 8$, $\eta_s = 1$, $\Delta t = 0.001$ when $\kappa = 0.5, 1, 50$. The blue line corresponds to the left hand side of the energy estimate (total energy between timesteps) and the red line is the right hand side of the energy estimate (energy due to the immersed boundary).

$\eta_s = 0.001, 1.0, 10.0$. Fig. 4.6 illustrates the three energy profiles for different values of the viscosity. Clearly we see that the energy profile becomes unstable at $\eta_s = 0.001$. This is as we expected and has been found throughout the literature. We have investigated it here so as to serve as a basis for the results given later. Unfortunately, there are too many permutations of parameters for us to give energy estimates for each permutation. Therefore, we give a table which illustrates when the numerical computation breaks down. Table 4.1 details for each permutation when the numerics break down. A tick indicates a completed run and a dash indicates that the simulation ended prematurely.

We can see from Table 4.1 that the ratio of η_s to κ is important as for the majority



(a) $\eta_s = 0.001$.

(b) $\eta_s = 1.0$.

(c) $\eta_s = 10.0$.

Figure 4.6: Energy profiles for $h_f = 1/8$, $N = 8$, $\eta_s = 0.001, 1, 10$, $\Delta t = 0.001$ when $\kappa = 1$. The blue line corresponds to the left hand side of the energy estimate (total energy between timesteps) and the red line is the right hand side of the energy estimate (energy due to the immersed boundary).

of time steps, the simulation finished when $\kappa = 50$ and $\eta_s = 10$. We can also infer from the table that the system is more unstable at smaller viscosities than the larger stiffness parameters. We can also clearly see that as the value of N is increased when $\eta_s = 0.001$ and $\kappa \leq 1$, we need to correspondingly decrease our timestep size. However, we do not need to decrease our timestep size when $\kappa = 50$ and $\eta_s \geq 1$.

η_s	Δt	$\kappa = 0.5$			$\kappa = 1.0$			$\kappa = 50.0$		
		$N = 2$	$N = 4$	$N = 8$	$N = 2$	$N = 4$	$N = 8$	$N = 2$	$N = 4$	$N = 8$
10^{-3}	10^{-1}	-	-	-	-	-	-	-	-	-
	10^{-2}	✓	-	-	✓	-	-	-	-	-
	10^{-3}	✓	✓	-	✓	✓	-	✓	-	-
	10^{-4}	✓	✓	✓	✓	✓	✓	✓	✓	-
1.0	10^{-1}	✓	✓	✓	✓	✓	-	-	-	-
	10^{-2}	✓	✓	✓	✓	✓	✓	-	-	-
	10^{-3}	✓	✓	✓	✓	✓	✓	✓	✓	-
	10^{-4}	✓	✓	✓	✓	✓	✓	✓	✓	✓
10.0	10^{-1}	✓	✓	✓	✓	✓	✓	-	-	-
	10^{-2}	✓	✓	✓	✓	✓	✓	✓	✓	-
	10^{-3}	✓	✓	✓	✓	✓	✓	✓	✓	✓
	10^{-4}	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 4.1: Illustration of when a simulation for a given set of parameters has completed or stopped for the SIME method. A tick means the simulation finished and a dash means it terminated early.

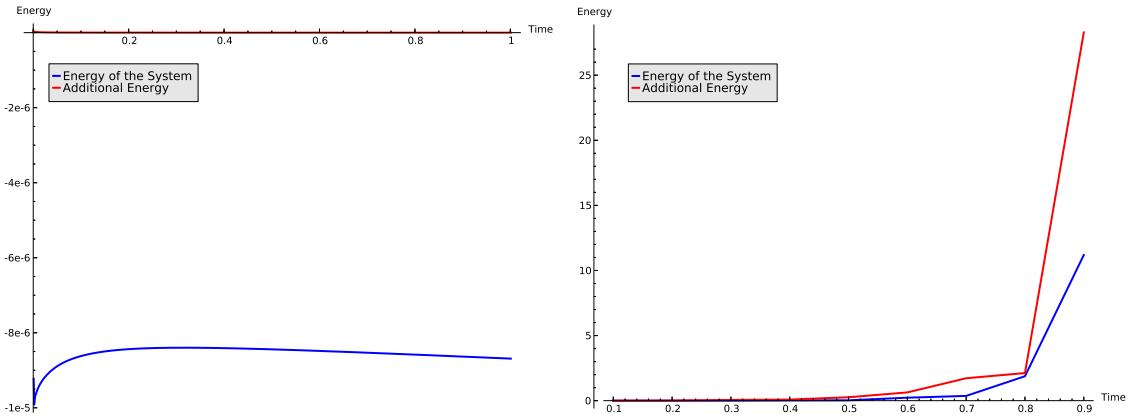
4.2.2 Backward (Implicit) Euler Method

The natural choice when considering a more implicit movement of the immersed membrane is to consider an implicit Euler method. The energy estimate for such a method is given by:

$$\begin{aligned}
&\left\|\boldsymbol{u}_N^{n+1}\right\|_\Omega^2 + \kappa \sum_{i=1}^{K_b} \left[\frac{\left(\boldsymbol{X}_{h,i}^{n+1} - \boldsymbol{X}_{h,i-1}^{n+1}\right)^2}{\Delta s_i} \right] - \left\|\boldsymbol{u}_N^{n}\right\|_\Omega^2 - \kappa \sum_{i=1}^{K_b} \left[\frac{\left(\boldsymbol{X}_{h,i}^{n} - \boldsymbol{X}_{h,i-1}^{n}\right)^2}{\Delta s_i} \right] \\
&\leq 2\kappa\Delta t \left\{ \sum_{i=1}^{K_b} \left(\frac{\boldsymbol{X}_{h,i}^{n+1} - \boldsymbol{X}_{h,i-1}^{n+1}}{\Delta s_i} \right) \left(\boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i}^{n+1}) - \boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i-1}^{n+1}) \right) - \right. \\
&\qquad \left. \sum_{i=1}^{K_b} \left(\frac{\boldsymbol{X}_{h,i}^{n} - \boldsymbol{X}_{h,i-1}^{n}}{\Delta s_i} \right) \left(\boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i}^{n}) - \boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i-1}^{n}) \right) \right\} \qquad (4.2.6)
\end{aligned}$$

The above energy estimate is derived in a similar manner to that of the semi-implicit Euler method in §4.2.1.

First of all, we consider two energy profiles. In those energy profiles, we fix $h_f = 1/8$, $N = 8$, $\eta_s = 1$ and $\kappa = 1$. Then we consider two different values for $\Delta t = 0.001, 0.1$. Fig. 4.7 illustrates these energy profiles. Clearly when $\Delta t = 0.001$, we have a stable



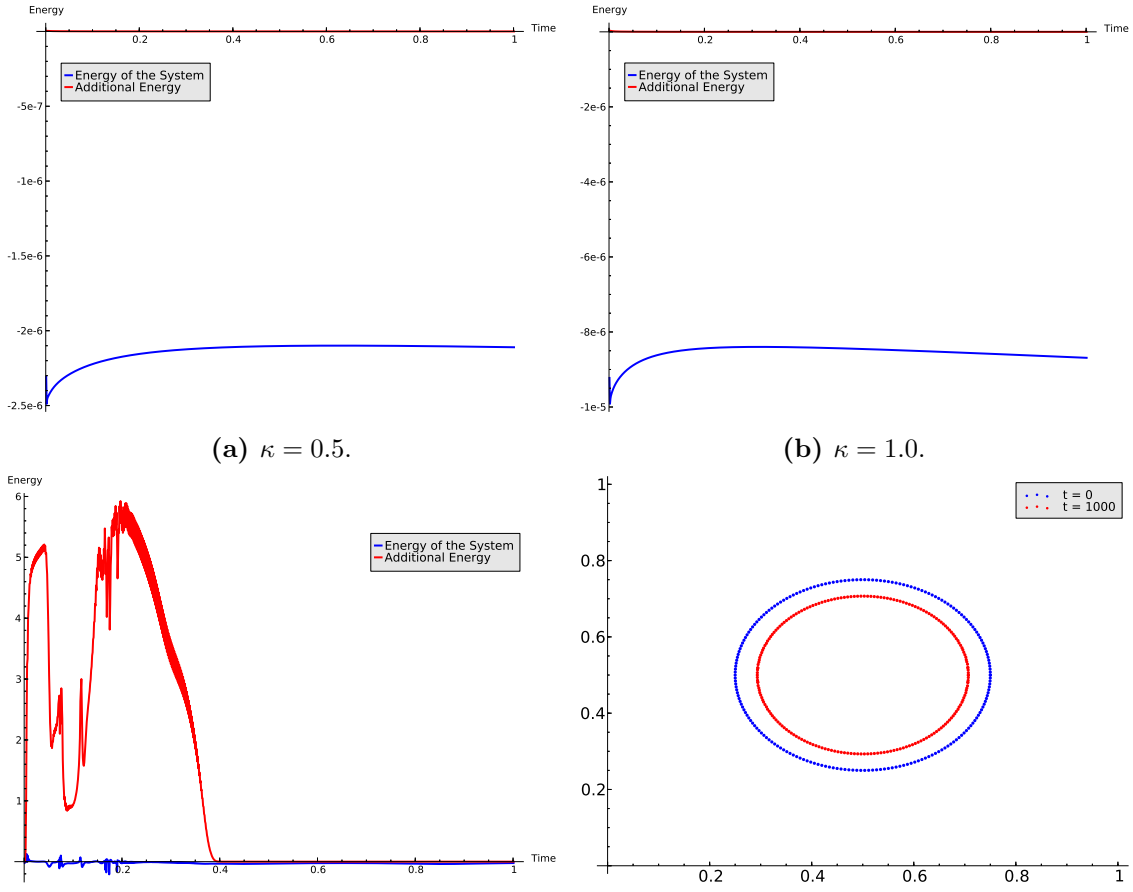
(a) $\Delta t = 0.001$. **(b)** $\Delta t = 0.1$.

Figure 4.7: Energy profiles for $h_f = 1/8$, $N = 8$, $\eta_s = 1$, $\kappa = 1$ and $\Delta t = 0.001, 0.1$. The blue line corresponds to the left hand side of the energy estimate (total energy between timesteps) and the red line is the right hand side of the energy estimate (energy due to the immersed boundary).

system. However we see that the energy starts to diverge at around $T = 0.5$ when $\Delta t = 0.1$. This illustrates that when $\Delta t = 0.1$ the system is unstable. Note that the

blue line is always underneath the red line. The point at which the energy starts to diverge, $T = 0.5$, is slightly larger for the implicit Euler than it is for the semi-implicit Euler. This suggests that making the evolution more implicit could have a stabilising effect. Indeed, the simulation crashed at $T = 0.5$ for the semi-implicit Euler, where for the implicit Euler it ran until $T = 0.9$.

We are interested to see the effect of high forcing parameter. We fix $\Delta t = 0.001$, $h_f = 1/8$, $N = 8$ and $\eta_s = 1$ while varying $\kappa = 0.5, 1.0, 50.0$. Fig. 4.8 illustrates the three energy profiles for the different values of κ. We can see that we have a stable



(a) $\kappa = 0.5$.

(b) $\kappa = 1.0$.

(c) $\kappa = 50.0$.

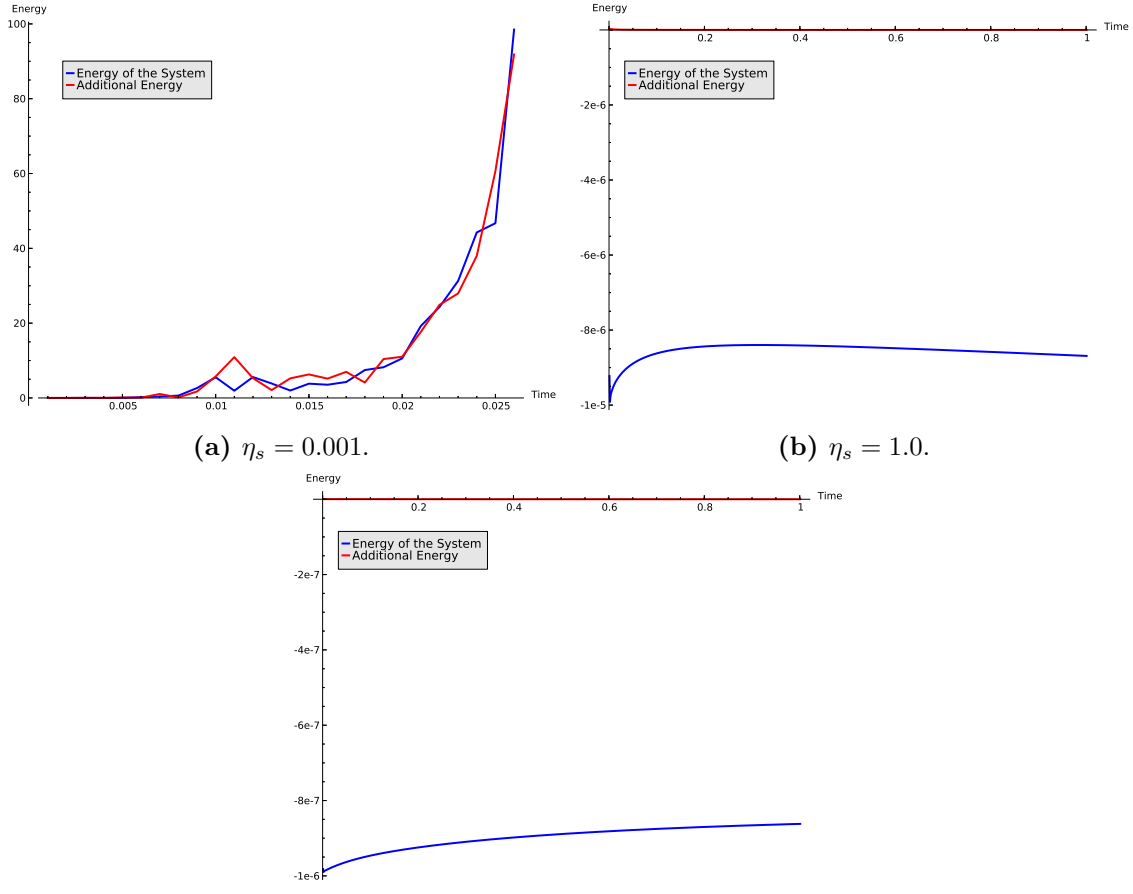
(d) Immersed boundary evolution when $\kappa = 50.0$.

Figure 4.8: Energy profiles for $h_f = 1/8$, $N = 8$, $\eta_s = 1$, $\Delta t = 0.001$ when $\kappa = 0.5, 1, 50$. The blue line corresponds to the left hand side of the energy estimate (total energy between timesteps) and the red line is the right hand side of the energy estimate (energy due to the immersed boundary). Also included is a plot of the immersed boundary evolution when $\kappa = 50$.

system for all values of κ. This is an improvement on the semi-implicit Euler results

that began to diverge when $\kappa = 50$ (see Fig. 4.5c). Fig. 4.8d illustrates the immersed boundary movement and we can clearly see that its motion is stable. There seems to be quite a bit of noise in the red curve (the immersed boundary energy) in Fig. 4.8c and we are unsure of the reason for this but it doesn't seem to adversely affect the energy in the system.

Just as before, we would like to see how the method behaves at small viscosities. Therefore, we fix $\Delta t = 0.001$, $h_f = 1/8$, $N = 8$ and $\kappa = 1$ while varying $\eta_s = 0.001, 1.0, 10.0$. Fig. 4.9 illustrates the three energy profiles for the different values of the viscosity. Again we see that the energy profile becomes unstable at $\eta_s = 0.001$. However, we



(a) $\eta_s = 0.001$.

(b) $\eta_s = 1.0$.

(c) $\eta_s = 10.0$.

Figure 4.9: Energy profiles for $h_f = 1/8$, $N = 8$, $\eta_s = 0.001, 1, 10$, $\Delta t = 0.001$ when $\kappa = 1$. The blue line corresponds to the left hand side of the energy estimate (total energy between timesteps) and the red line is the right hand side of the energy estimate (energy due to the immersed boundary).

notice that the simulation runs until $T = 0.025$ which is slightly longer than the semi-

implicit Euler method which only ran to $T = 0.015$. Comparing Fig. 4.9a to Fig. 4.8c we can infer that making the IB evolution equation more implicit does not improve the instability at smaller viscosity as well as the instability at large stiffness. Table 4.2 details for each permutation when the numerics break down. Again, a tick indicates a completed run and a dash indicates that the simulation ended early.

Comparing Table 4.2 against Table 4.1 shows that at the smaller viscosities using the implicit Euler method is more unstable when $N = 2$ for all values of κ. However, when $\kappa = 50$ and $\eta_s \geq 1$, we see that the implicit Euler method is more stable when $N = 8$ and thus allows a higher value for the timestep length. Additionally, when $\eta_s = 1$ and $\kappa = 0.5$ we see that the implicit Euler method allows a larger time step at $N = 8$. It will be interesting to see what happens with the higher-order case implicit case.

η_s	Δt	$\kappa = 0.5$			$\kappa = 1.0$			$\kappa = 50.0$		
		$N = 2$	$N = 4$	$N = 8$	$N = 2$	$N = 4$	$N = 8$	$N = 2$	$N = 4$	$N = 8$
10^{-3}	10^{-1}	-	-	-	-	-	-	-	-	-
	10^{-2}	-	-	-	-	-	-	-	-	-
	10^{-3}	✓	✓	-	✓	✓	-	-	-	-
	10^{-4}	✓	✓	✓	✓	✓	✓	✓	✓	-
1.0	10^{-1}	✓	✓	✓	✓	✓	-	-	-	-
	10^{-2}	✓	✓	✓	✓	✓	✓	-	-	-
	10^{-3}	✓	✓	✓	✓	✓	✓	✓	✓	✓
	10^{-4}	✓	✓	✓	✓	✓	✓	✓	✓	✓
10.0	10^{-1}	✓	✓	✓	✓	✓	✓	-	-	-
	10^{-2}	✓	✓	✓	✓	✓	✓	✓	✓	✓
	10^{-3}	✓	✓	✓	✓	✓	✓	✓	✓	✓
	10^{-4}	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 4.2: Illustration of when a simulation for a given set of parameters has completed or stopped for the IME method. A tick means the simulation finished and a dash means it terminated early.

4.2.3 Third-Order Backward Differentiation Formula

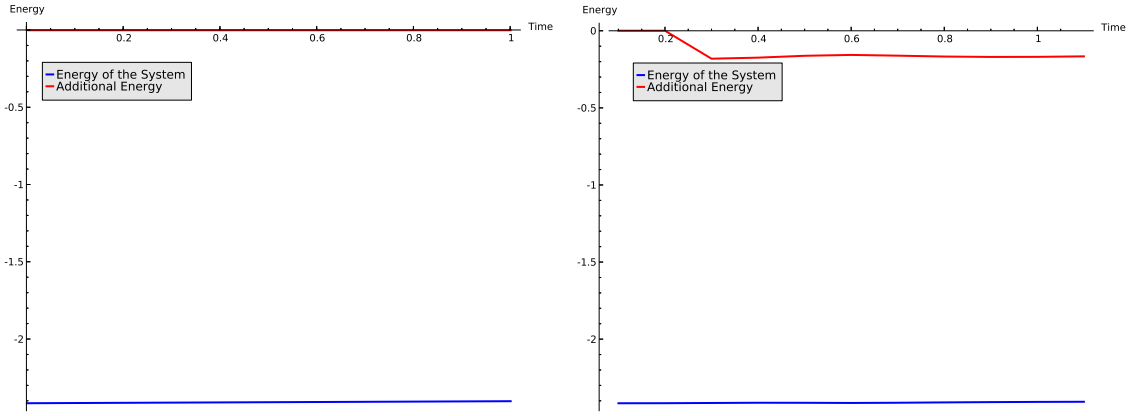
In this subsection, we consider a higher-order implicit method to move the immersed boundary to see whether we can obtain larger timestep lengths. The discrete energy estimate is given by:

$$\begin{aligned}
&\frac{1}{2}\left\|\boldsymbol{u}_N^{n+1}\right\|_\Omega^2+\frac{11}{12}\kappa\sum_{i=1}^{K_b}\left[\frac{\left(\boldsymbol{X}_{h,i}^{n+1}-\boldsymbol{X}_{h,i-1}^{n+1}\right)^2}{\Delta s_i}\right]-\frac{1}{2}\left\|\boldsymbol{u}_N^{n}\right\|_\Omega^2-\frac{27}{11}\kappa\sum_{i=1}^{K_b}\left[\frac{\left(\boldsymbol{X}_{h,i}^{n}-\boldsymbol{X}_{h,i-1}^{n}\right)^2}{\Delta s_i}\right]\\
&\leq\kappa\Delta t\left\{\sum_{i=1}^{K_b}\left(\frac{\boldsymbol{X}_{h,i}^{n+1}-\boldsymbol{X}_{h,i-1}^{n+1}}{\Delta s_i}\right)\left(\boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i}^{n+1})-\boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i-1}^{n+1})\right)-\right.\\
&\qquad\sum_{i=1}^{K_b}\left(\frac{\boldsymbol{X}_{h,i}^{n}-\boldsymbol{X}_{h,i-1}^{n}}{\Delta s_i}\right)\left(\boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i}^{n})-\boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i-1}^{n})\right)-\\
&\qquad\frac{3}{2}\sum_{i=1}^{K_b}\frac{\left(\boldsymbol{X}_{h,i}^{n-1}-\boldsymbol{X}_{h,i-1}^{n-1}\right)\left(\boldsymbol{X}_{h,i}^{n+1}-\boldsymbol{X}_{h,i-1}^{n+1}\right)}{\Delta s_i}+\\
&\qquad\left.\frac{1}{3}\sum_{i=1}^{K_b}\frac{\left(\boldsymbol{X}_{h,i}^{n-2}-\boldsymbol{X}_{h,i-1}^{n-2}\right)\left(\boldsymbol{X}_{h,i}^{n+1}-\boldsymbol{X}_{h,i-1}^{n+1}\right)}{\Delta s_i}\right\}
\end{aligned}\tag{4.2.7}$$

The above energy estimate is derived in a similar manner to the semi-implicit Euler method §4.2.1.

Following the previous subsections, we consider two energy profiles. In those energy profiles, we fix $h_f = 1/8$, $N = 8$, $\eta_s = 1$ and $\kappa = 1$. Then we consider two different values for $\Delta t = 0.001, 0.1$. Fig. 4.10 illustrates these energy profiles. Both of the figures shown in Fig. 4.10 are stable. This is an improvement on the previous two routines that we considered. The energy in the system for both the semi-implicit Euler and the implicit Euler methods diverged when $\eta_s = 1$, $\kappa = 1$ and $\Delta t = 0.1$. Thus clearly, using a higher-order implicit method has some benefits.

We are interested to see the effect of high stiffness on such an example. We fix $\Delta t = 0.001$, $h_f = 1/8$, $N = 8$ and $\eta_s = 1$ while varying $\kappa = 0.5, 1.0, 50.0$. Fig. 4.11 illustrates the three energy profiles for the different values of κ. It is evident from Fig. 4.11 that for the three values of κ considered here, with a timestep length of 0.001, the third-order backward differentiation formula is stable. This is again an



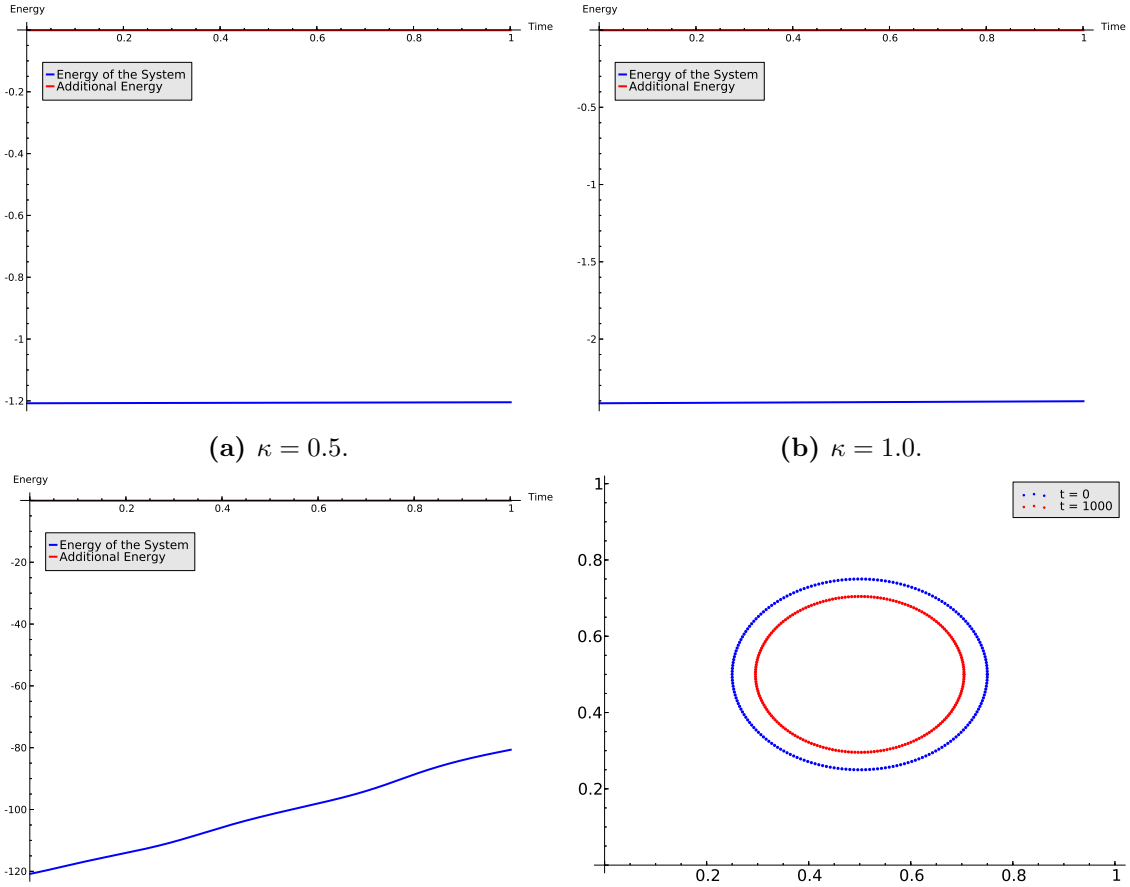
(a) $\Delta t = 0.001$. **(b)** $\Delta t = 0.1$.

Figure 4.10: Energy profiles for $h_f = 1/8$, $N = 8$, $\eta_s = 1$, $\kappa = 1$ and $\Delta t = 0.001, 0.1$. The blue line corresponds to the left hand side of the energy estimate (total energy between timesteps) and the red line is the right hand side of the energy estimate (energy due to the immersed boundary).

improvement on the semi-implicit Euler method and we notice that there is no noise at the beginning of the energy profile when $\kappa = 50$. However, the energy in the system is increasing and it is highly possible that the energy could eventually diverge if the simulation was run for longer. Fig. 4.11d illustrates that the motion of the immersed boundary is still stable when $\Delta t = 0.001$, $\kappa = 50$ and $\eta_s = 1$.

Just as before, we wish to consider small viscosities. Therefore, we fix $\Delta t = 0.001$, $h_f = 1/8$, $N = 8$ and $\kappa = 1$ while varying $\eta_s = 0.001, 1.0, 10.0$. Fig. 4.12 illustrates the three energy profiles for the different values of the viscosity. The semi-implicit Euler method and the implicit Euler method both suffered from a diverging energy profile when the viscosity was 0.001. However, Fig. 4.12a suggests that the third-order backward differentiation formula remains stable when the viscosity is 0.001. However, this is actually not the case. The energy in the system diverges to $-\infty$. This is a rather strange manifestation of instability as we require the energy in the system to be negative in order for the system to remain stable. However, the simulation definitely terminated early. Table 4.3 details, for each permutation, when the numerics break down. A tick indicates that a completed run and a dash indicates that the simulation ended early.

Comparing Table 4.3 with Table 4.1 and 4.2 shows that for most of the time the third-



(a) $\kappa = 0.5$.

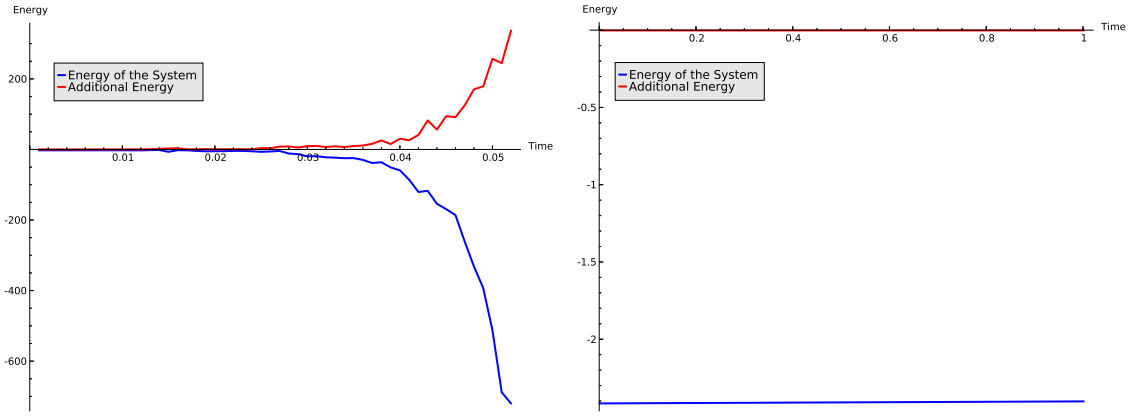
(b) $\kappa = 1.0$.

(c) $\kappa = 50.0$.

(d) Immersed boundary evolution when $\kappa = 50.0$.

Figure 4.11: Energy profiles for $h_f = 1/8$, $N = 8$, $\eta_s = 1$, $\Delta t = 0.001$ when $\kappa = 0.5, 1, 50$. The blue line corresponds to the left hand side of the energy estimate (total energy between timesteps) and the red line is the right hand side of the energy estimate (energy due to the immersed boundary).

order BDF behaves similarly to the first-order implicit Euler method, except for a few exceptions such as $\eta_s = 0.001$, $\kappa = 50$ and $\Delta t = 0.01$. It appears that the use of an implicit method to move the immersed membrane does stabilise the method slightly and allows larger Δt at the larger values of the stiffness. However, at smaller viscosities, the implicit method does not behave any better than the semi-implicit method.



(a) $\eta_s = 0.001$.

(b) $\eta_s = 1.0$.

(c) $\eta_s = 10.0$.

Figure 4.12: Energy profiles for $h_f = 1/8$, $N = 8$, $\eta_s = 0.001, 1, 10$, $\Delta t = 0.001$ when $\kappa = 1$. The blue line corresponds to the left hand side of the energy estimate (total energy between timesteps) and the red line is the right hand side of the energy estimate (energy due to the immersed boundary).

η_s	Δt	$\kappa = 0.5$			$\kappa = 1.0$			$\kappa = 50.0$		
		$N = 2$	$N = 4$	$N = 8$	$N = 2$	$N = 4$	$N = 8$	$N = 2$	$N = 4$	$N = 8$
10^{-3}	10^{-1}	-	-	-	-	-	-	-	-	-
	10^{-2}	-	-	-	-	-	-	-	-	-
	10^{-3}	✓	✓	-	✓	✓	-	✓	-	-
	10^{-4}	✓	✓	✓	✓	✓	✓	✓	✓	-
1.0	10^{-1}	✓	✓	✓	✓	✓	✓	-	-	-
	10^{-2}	✓	✓	✓	✓	✓	✓	✓	-	-
	10^{-3}	✓	✓	✓	✓	✓	✓	✓	✓	✓
	10^{-4}	✓	✓	✓	✓	✓	✓	✓	✓	✓
10.0	10^{-1}	✓	✓	✓	✓	✓	✓	✓	-	-
	10^{-2}	✓	✓	✓	✓	✓	✓	✓	✓	✓
	10^{-3}	✓	✓	✓	✓	✓	✓	✓	✓	✓
	10^{-4}	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 4.3: Illustration of when a simulation for a given set of parameters has completed or stopped for the third-order BDF. A tick means the simulation finished and a dash means it terminated early.

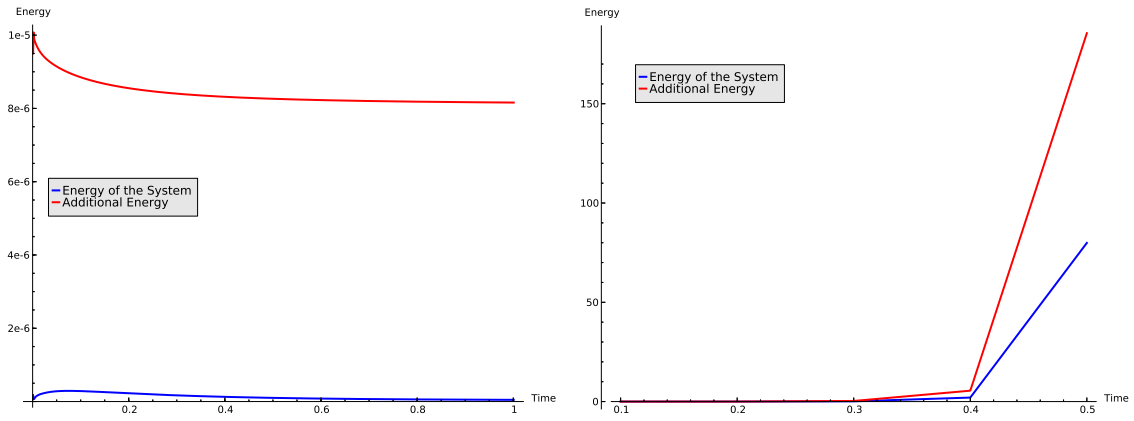
4.2.4 Velocity Correction Scheme

The velocity correction scheme (VCS) was introduced to preserve the area inside the membrane up to the order $O((\Delta t)^2)$. The VCS, which we derive in appendix C.1, is based on the semi-implicit Euler method and therefore, we don't expect to see that much of a difference here. The energy estimate is given by:

$$\begin{aligned}
&\left\|\boldsymbol{u}_N^{n+1}\right\|_\Omega^2 + \kappa \sum_{i=1}^{K_b} \left[\frac{\left(\boldsymbol{X}_{h,i}^{n+1} - \boldsymbol{X}_{h,i-1}^{n+1}\right)^2}{\Delta s_i} \right] - \left\|\boldsymbol{u}_N^{n}\right\|_\Omega^2 - \kappa \sum_{i=1}^{K_b} \left[\frac{\left(\boldsymbol{X}_{h,i}^{n} - \boldsymbol{X}_{h,i-1}^{n}\right)^2}{\Delta s_i} \right] \\
&\leq 2\kappa \Delta t \left\{ \sum_{i=1}^{K_b} \left(\frac{\boldsymbol{X}_{h,i}^{n+1} - \boldsymbol{X}_{h,i-1}^{n+1}}{\Delta s_i} \right) \left(\hat{\boldsymbol{u}}(\boldsymbol{X}_{h,i}^{n}) - \hat{\boldsymbol{u}}(\boldsymbol{X}_{h,i-1}^{n}) \right) - \right. \\
&\qquad \left. \sum_{i=1}^{K_b} \left(\frac{\boldsymbol{X}_{h,i}^{n} - \boldsymbol{X}_{h,i-1}^{n}}{\Delta s_i} \right) \left(\boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i}^{n}) - \boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i-1}^{n}) \right) \right\}
\end{aligned} \tag{4.2.8}$$

The above energy estimate is derived in a similar manner to the semi-implicit Euler method §4.2.1.

First of all, we consider two energy profiles. In those energy profiles, we fix $h_f = 1/8$, $N = 8$, $\eta_s = 1$ and $\kappa = 1$. Then we consider two different values for $\Delta t = 0.001, 0.1$. Fig. 4.13 illustrates these energy profiles. Clearly when $\Delta t = 0.001$, we have a stable



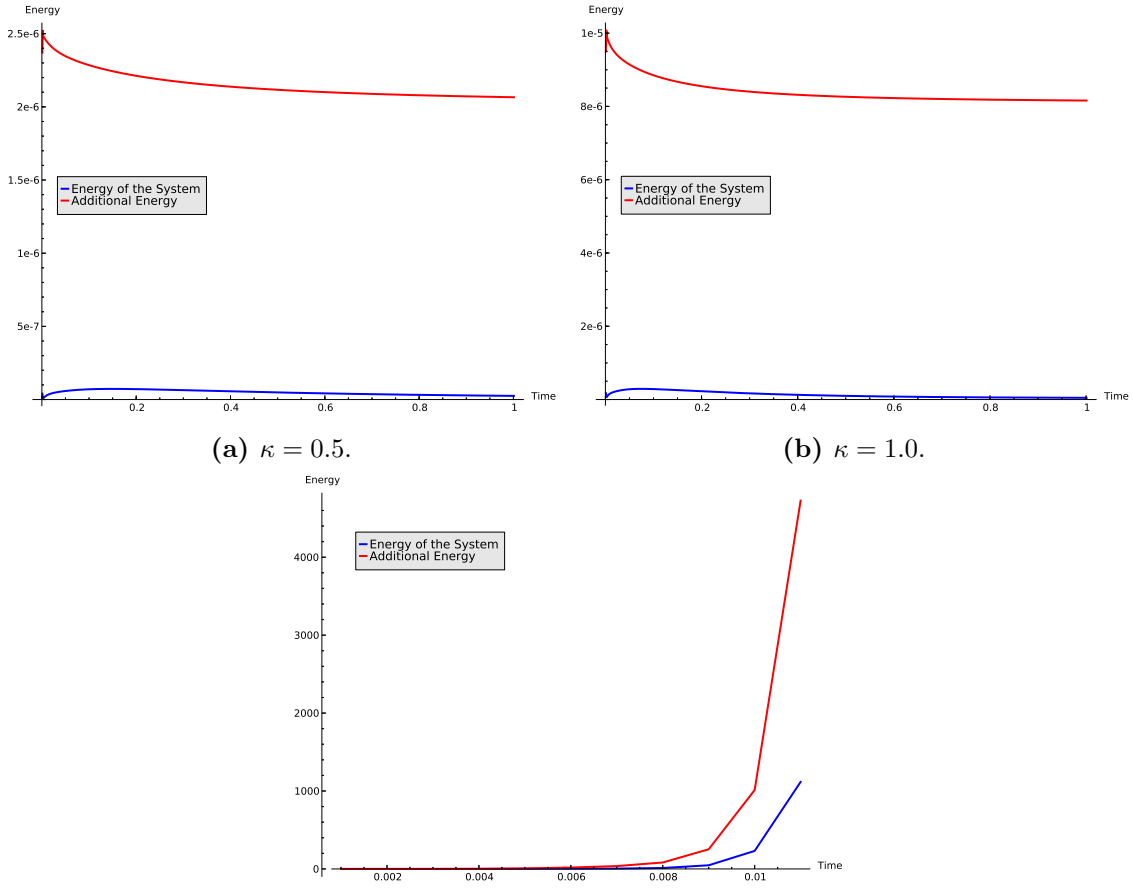
(a) $\Delta t = 0.001$. **(b)** $\Delta t = 0.1$.

Figure 4.13: Energy profiles for $h_f = 1/8$, $N = 8$, $\eta_s = 1$, $\kappa = 1$ and $\Delta t = 0.001, 0.1$. The blue line corresponds to the left hand side of the energy estimate (total energy between timesteps) and the red line is the right hand side of the energy estimate (energy due to the immersed boundary).

system. However we see that the energy starts to diverge at around $T = 0.3$ when

$\Delta t = 0.1$. This illustrates that when $\Delta t = 0.1$ the system is unstable. Note that the blue line is always underneath the red line. This is very similar to the semi-implicit Euler case, which isn't too surprising considering we used the semi-implicit Euler method in the derivation of the velocity correction scheme.

We are interested to see the effect of high stiffness on such an example. We fix $\Delta t = 0.001$, $h_f = 1/8$, $N = 8$ and $\eta_s = 1$ while varying $\kappa = 0.5, 1.0, 50.0$. Fig. 4.14 illustrates the three energy profiles for the different values of κ. Clearly when



(a) $\kappa = 0.5$.

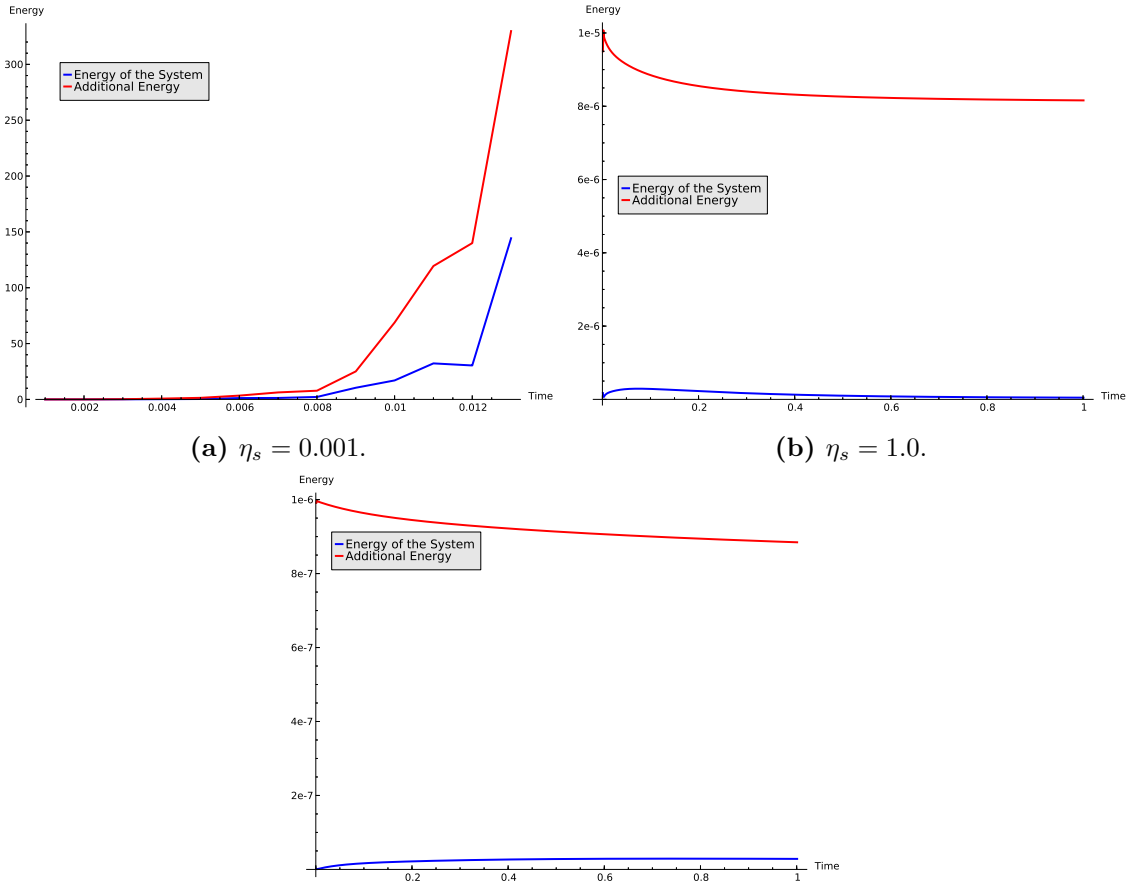
(b) $\kappa = 1.0$.

(c) $\kappa = 50.0$.

Figure 4.14: Energy profiles for $h_f = 1/8$, $N = 8$, $\eta_s = 1$, $\Delta t = 0.001$ when $\kappa = 0.5, 1, 50$. The blue line corresponds to the left hand side of the energy estimate (total energy between timesteps) and the red line is the right hand side of the energy estimate (energy due to the immersed boundary).

$\kappa = 0.5, 1.0$ we have a stable system however when $\kappa = 50$ the system becomes unstable. The results given in Fig. 4.14 are very similar to the results given in Fig. 4.5 and when $\kappa = 50$ the simulation stops at a very similar time.

Just as before, we consider small viscosities. Therefore, we fix $\Delta t = 0.001$, $h_f = 1/8$, $N = 8$ and $\kappa = 1$ while varying $\eta_s = 0.001, 1.0, 10.0$. Fig. 4.15 illustrates the three energy profiles for the different values of the viscosity. Clearly we see that the energy



(a) $\eta_s = 0.001$.

(b) $\eta_s = 1.0$.

(c) $\eta_s = 10.0$.

Figure 4.15: Energy profiles for $h_f = 1/8$, $N = 8$, $\eta_s = 0.001, 1, 10$, $\Delta t = 0.001$ when $\kappa = 1$. The blue line corresponds to the left hand side of the energy estimate (total energy between timesteps) and the red line is the right hand side of the energy estimate (energy due to the immersed boundary).

profile becomes unstable at $\eta_s = 0.001$. Again the results given in Fig. 4.15 are very similar to the results given for the semi-implicit Euler case in Fig. 4.6. Table 4.4 once again details for each permutation when the numerics break down. A tick indicates that a completed run and a dash indicates that the simulation ended early.

It is clear from Table 4.4 that the velocity correction scheme is not well suited for problems with a small viscosity as it mostly failed to converge for all values of κ and

all values of N with the odd exception. Comparing Table 4.4 to Table 4.1, we notice that for $\kappa = 50$ and $\eta_s \geq 1$ the velocity correction scheme behaves exactly the same as the semi-implicit Euler method. This is not surprising as the VCS is based on the semi-implicit Euler method.

In this section, we investigated the stiffness of the immersed boundary method by comparing the standard semi-implicit temporal discretisation of the immersed boundary evolution equation against a first-order and third-order implicit method as well as a velocity correction scheme. We found that using an implicit method does improve the stiffness, in other words larger time step lengths can be used. However, the improvement is not as substantial as we would have liked. Using a higher-order implicit method allowed larger time step lengths at higher values of polynomial degree, however the improvement is again not as substantial as we would have liked. As a result, we do not make use of the implicit schemes in the results presented in Chapters 7,8 and 9. Instead, we only use the semi-implicit discretisation of the immersed boundary evolution equation.

η_s	Δt	$\kappa = 0.5$			$\kappa = 1.0$			$\kappa = 50.0$		
		$N = 2$	$N = 4$	$N = 8$	$N = 2$	$N = 4$	$N = 8$	$N = 2$	$N = 4$	$N = 8$
10^{-3}	10^{-1}	-	-	-	-	-	-	-	-	-
	10^{-2}	-	-	-	-	-	-	-	-	-
	10^{-3}	✓	-	-	-	-	-	-	-	-
	10^{-4}	-	✓	✓	-	-	✓	-	-	-
1.0	10^{-1}	✓	✓	-	✓	✓	-	-	-	-
	10^{-2}	✓	✓	✓	✓	✓	✓	-	-	-
	10^{-3}	✓	✓	✓	✓	✓	✓	✓	✓	-
	10^{-4}	✓	✓	✓	✓	✓	✓	✓	✓	✓
10.0	10^{-1}	✓	✓	✓	✓	✓	✓	-	-	-
	10^{-2}	✓	✓	✓	✓	✓	✓	✓	✓	-
	10^{-3}	✓	✓	✓	✓	✓	✓	✓	✓	✓
	10^{-4}	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 4.4: Illustration of when a simulation for a given set of parameters has completed or stopped for the VCS. A tick means the simulation finished and a dash means it terminated early.

Chapter 5

Spatial Discretisation

Throughout the majority of the IB literature, the finite difference method has been the discretisation method of choice. However, it was well known that the original IBM was limited to the assumption of immersed fibres or membranes. In order to consider thick immersed structures, it is natural to turn to finite element type discretisations, which lend themselves to more complex geometries. The extended immersed boundary method (EIBM) proposed by Wang *et al.* [70] used finite elements for the immersed structure only. In a follow up paper, the immersed finite element method (IFEM) [73] used finite elements for both the fluid and the immersed structure. In the majority of the immersed boundary literature, a regularised approximation of the delta function is used to govern the interaction between the fluid and the elastic structure, and this approximation can be considered a disadvantage. Boffi *et al.* [14, 17] proposed a variational treatment of the immersed boundary which naturally lends itself to finite element type methods. In this thesis, we employ a spectral element discretisation of the fluid whilst maintaining the linear finite element discretisation of the immersed structure. Our hope is that the spectral accuracy of the fluid variables will improve the accuracy of the *spreading* and *interpolation* phases of the IBM.

This chapter is constructed as follows: We will discuss the spectral meshes which we use throughout this thesis in §5.1 followed by the spectral element discretisation of the fluid variables in §5.2 where, additionally, we illustrate some problems which can

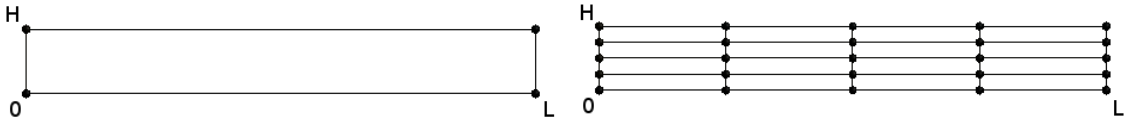
occur with unfitted discontinuities. In §5.3 we will introduce the eXtended Spectral Element Method (XSEM) which was first proposed by Legay *et al.* [42] and present some theoretical error estimates as well as a discussion relating to the inf-sup, or LBB, condition. In §5.4 we discuss the spatial discretisation of the immersed boundary, which for the sake of simplicity will always be assumed to be a linear approximation, and discuss the volume/area conservation problem which is known to exist in immersed boundary computations.

5.1 Spectral Meshes

The numerical solution of the partial differential equations by finite element type discretisations require the decomposition of a physical domain. The resulting decomposition is known as a *mesh*. In this section, we discuss the meshes that are used for the transient Poiseuille flow validation considered in §7.1 and the meshes that are used in all of the immersed boundary method simulations.

5.1.1 Transient Poiseuille Flow

In §7.1, we consider the transient Poiseuille flow of both a Newtonian fluid and an Oldroyd-B fluid. In those examples, the domain is taken to be a channel of length L and height H (which for now we leave undefined). Fig. 5.1 illustrates the two meshes (without the Gauss-Lobatto-Legendre (GLL) grid in each spectral element) that we use for both Newtonian and Oldroyd-B fluids. In both cases, the domain is decomposed into K_f quadrilateral spectral elements as illustrated in the figure.

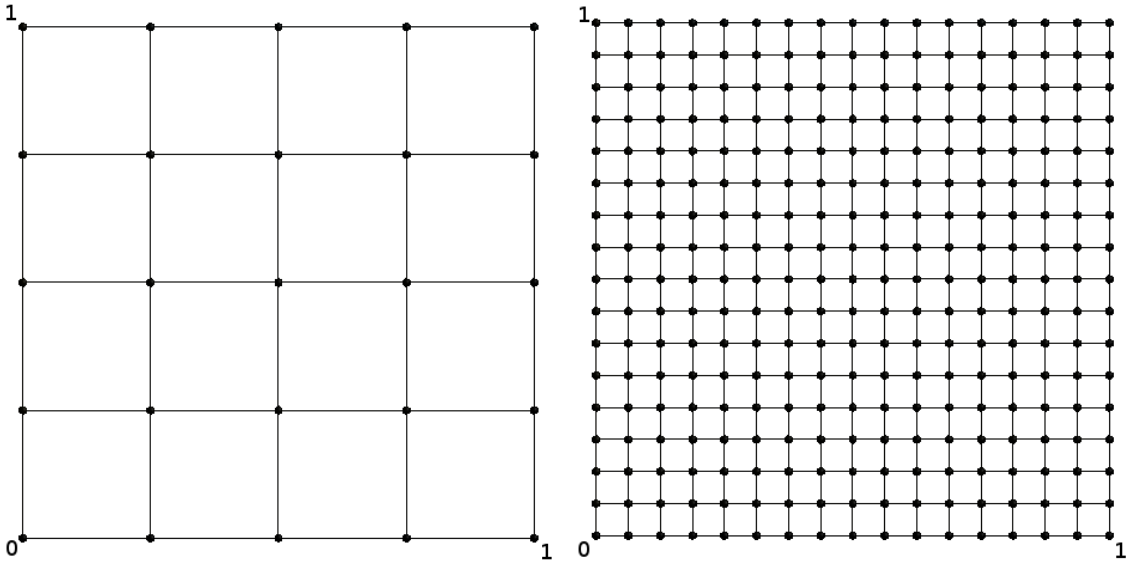


(a) $K_f = 1$ **(b)** $K_f = 16$

Figure 5.1: Sample meshes for $K_f = 1$ and $K_f = 16$ used for transient Poiseuille flow through a channel of length L and height H.

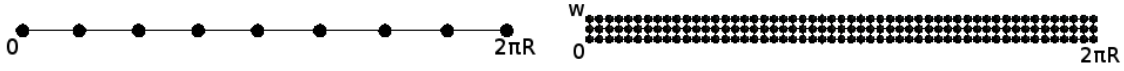
5.1.2 Immersed Boundary Method

In §7.2, §8 and §9 the spectral element immersed boundary method (SE-IBM) is used to model the fluid-structure interaction of an elastic membrane and shell immersed in both Newtonian and Oldroyd-B fluids. In all those examples, the fluid domain is the unit square and the structure domain is either a line of length $2\pi R$ or a rectangular domain of dimensions $[0, w] \times [0, 2\pi R]$. Fig. 5.2 illustrates two fluid meshes (without the Gauss-Lobatto-Legendre (GLL) grid) for mesh widths $h_f = 1/4$ and $h_f = 1/16$. Fig. 5.3 illustrates the two meshes that are used to model an immersed elastic membrane and an elastic shell.



(a) $h_f = 1/4$ **(b)** $h_f = 1/16$

Figure 5.2: Sample meshes for the fluid with mesh width $h_f = 1/4$ and $h_f = 1/16$ used for IBM computations.



(a) $h^s = 2\pi R/8$ **(b)** $h^s \approx 2\pi R/52$

Figure 5.3: Sample meshes for the structure with mesh width $h^s = 2\pi R/8$ and $h^s \approx 2\pi R/52$.

5.1.3 Gauss-Lobatto-Legendre Grid

In the previous subsections, we have illustrated the fluid meshes, and Lagrangian meshes, used throughout this thesis. Each element in a given mesh is mapped to

a parent domain $[-1,1] \times [-1,1]$ by a transfinite mapping defined by (5.2.9). Lagrange interpolants based on the Gauss-Lobatto-Legendre grid (which we define in (5.2.15)) are then used to approximate the local variables. Fig. 5.4 illustrates the Gauss-Lobatto-Legendre grid on the parent domain.

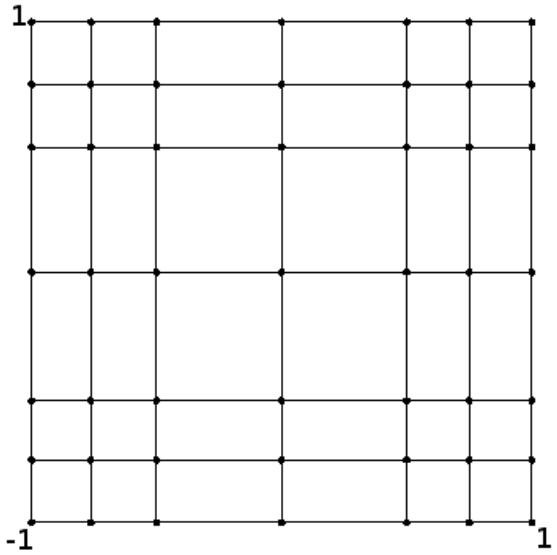


Figure 5.4: Example Gauss-Lobatto-Legendre grid when $N = 6$.

5.2 Spectral Element Method (SEM)

The spectral element method was first proposed by Patera [55] in an attempt to alleviate the issues found when spectral methods are applied to complex geometries. Indeed, SEM can be considered to be a cross between a finite element method and spectral method, so that SEM has the geometric flexibility of a finite element method but the accuracy of a spectral method and in principle is similar to *hp* - FEM. As mentioned above, our hope is that applying the SEM to the fluid variables in the IBM, will improve the accuracy of the spreading and interpolation phases. It is well known that the SEM *should* perform better than traditional finite elements both in terms of accuracy and efficiency provided the solution is sufficiently regular and that the accepted error level is taken to be sufficiently small. However, if the regularity of the solution is low then the spectral element method will only perform as well as finite elements.

5.2.1 Weak Formulation

The equations governing Newtonian and viscoelastic fluid flow were given in §2.2.2. Those equations are represented in *strong* form. For finite element type methods, such as spectral elements, one aims to solve these equations in their equivalent weak form. The weak formulation is constructed by multiplying the strong form of the equations by a *suitable* test function selected from a *suitable* function space, then integrating the whole equation over the entire domain. The advantage of such a procedure is that the order of the differential equation can be reduced using integration by parts enabling one to find a solution in a larger space.

Before we write out the weak formulation, we must define *suitable* function spaces for the dependent variables. In the equations governing *viscoelastic* fluid flow, the dependent variables are the velocity $\boldsymbol{u}$, the pressure p and the polymeric stress $\boldsymbol{\tau}$. Let $\Omega \subset \mathbb{R}^2$ and assume integrability, then one can assume the velocity $\boldsymbol{u} \in [H^1(\Omega)]^2$. For the pressure, one can assume $p \in L^2(\Omega)$. Similarly for the polymeric stress tensor, $\boldsymbol{\tau} \in [L^2(\Omega)]^{2\times 2}$ where $[L^2(\Omega)]^{2\times 2}$ is the space of 2×2 tensors whose components are in $L^2(\Omega)$. Our *suitable* function spaces are then chosen to be subspaces of these. We define:

$$\mathcal{V} := [H_0^1(\Omega)]^2 = \left\{ \boldsymbol{v} \in [H^1(\Omega)]^2 \colon \boldsymbol{v} = 0 \text{ on } \partial\Omega \right\} \subset [H^1(\Omega)]^2 \quad (5.2.1a)$$

$$\mathcal{Q} := L_0^2(\Omega) = \left\{ q \in L^2(\Omega) \colon \int_\Omega q \, \mathrm{d}\Omega = 0 \right\} \subset L^2(\Omega) \quad (5.2.1b)$$

$$\mathcal{S} := [L^2(\Omega)]_s^{2\times 2} = \left\{ \boldsymbol{s} \in [L^2(\Omega)]^{2\times 2} \colon \boldsymbol{s} \text{ is a symmetric tensor } \right\} \subset [L^2(\Omega)]^{2\times 2} \quad (5.2.1c)$$

where $\mathcal{V}$ is the velocity space, $\mathcal{Q}$ is the pressure space and $\mathcal{S}$ is the polymeric stress space. The zero integral pressure condition present in the definition of $\mathcal{Q}$ is required in order to remove any indeterminacy in the pressure.

The weak formulation of the semi-discrete equations, in the absence of the immersed

boundary, given in §4.1 is then: find $(\boldsymbol{u}^{n+1}, p^{n+1}, \boldsymbol{\tau}^{n+1}) \in \mathcal{V} \times \mathcal{Q} \times \mathcal{S}$ such that

$$\begin{aligned}
\frac{3\rho}{2\Delta t}(\boldsymbol{u}^{n+1}, \boldsymbol{v})_{L^2} + \eta_s a(\boldsymbol{u}^{n+1}, \boldsymbol{v}) + b(\boldsymbol{v}, p^{n+1}) &= c(\boldsymbol{\tau}^n, \boldsymbol{v}) \\
&+ \frac{\rho}{2\Delta t}(4\tilde{\boldsymbol{u}}^{n+1} - \tilde{\tilde{\boldsymbol{u}}}^{n+1}, \boldsymbol{v})_{L^2} + L(\boldsymbol{v}) \qquad \forall \boldsymbol{v} \in \mathcal{V} \quad (5.2.2a) \\
b(\boldsymbol{u}^{n+1}, q) &= 0 \qquad \forall q \in \mathcal{Q} \quad (5.2.2b) \\
\left(1 + \frac{3\lambda}{2\Delta t}\right)(\boldsymbol{\tau}^{n+1}, \boldsymbol{s})_{L^2} - \lambda(\boldsymbol{E}^{n+1}, \boldsymbol{s})_{L^2} & \\
= \eta_p(\dot{\boldsymbol{\gamma}}^{n+1}, \boldsymbol{s})_{L^2} + \frac{\lambda}{2\Delta t}(4\tilde{\boldsymbol{\tau}}^{n+1} - \tilde{\tilde{\boldsymbol{\tau}}}^{n+1}, \boldsymbol{s})_{L^2} &\quad \forall \boldsymbol{s} \in \mathcal{S} \quad (5.2.2c)
\end{aligned}$$

where η_s is the solvent viscosity, η_p is the polymeric viscosity, λ is the characteristic relaxation time and $\boldsymbol{E}^{n+1} = \boldsymbol{E}^{n+1}(\boldsymbol{\tau}, \boldsymbol{u})$ is the non-linear deformation term which appears in the upper-convected derivative (2.2.19). We have used a second-order OIFS scheme, (4.1.2), for the material derivative which appears in both the momentum equation and in the upper-convective derivative in the polymeric constitutive equation. The terms $\tilde{\boldsymbol{u}}$, $\tilde{\tilde{\boldsymbol{u}}}$, $\tilde{\boldsymbol{\tau}}$ and $\tilde{\tilde{\boldsymbol{\tau}}}$ are found by solving the pure advection equations given in (4.1.3).

The bilinear forms $a(\cdot,\cdot)\colon \mathcal{V} \times \mathcal{V} \to \mathbb{R}$, $b(\cdot,\cdot)\colon \mathcal{V} \times \mathcal{Q} \to \mathbb{R}$ and $c(\cdot,\cdot)\colon \mathcal{S} \times \mathcal{V} \to \mathbb{R}$ are defined as:

$$a(\boldsymbol{u}, \boldsymbol{v}) = \int_\Omega \nabla \boldsymbol{u} \colon \nabla \boldsymbol{v} \, \mathrm{d}\Omega \tag{5.2.3a}$$

$$b(\boldsymbol{v}, q) = \int_\Omega q \nabla \cdot \boldsymbol{v} \, \mathrm{d}\Omega \tag{5.2.3b}$$

$$c(\boldsymbol{s}, \boldsymbol{v}) = \int_\Omega \boldsymbol{s} \colon \nabla \boldsymbol{v} \, \mathrm{d}\Omega \tag{5.2.3c}$$

We define the L^2-inner product $(\cdot,\cdot)_{L^2}$ between two vectors and two tensors, respectively, by:

$$(\boldsymbol{u}, \boldsymbol{v})_{L^2(\Omega)} = \int_\Omega \boldsymbol{u} \cdot \boldsymbol{v} \, \mathrm{d}\Omega \tag{5.2.4a}$$

$$(\boldsymbol{\tau}, \boldsymbol{s})_{L^2(\Omega)} = \int_\Omega \boldsymbol{\tau} \colon \boldsymbol{s} \, \mathrm{d}\Omega \tag{5.2.4b}$$

and the linear form $L(\cdot)\colon \mathcal{V} \to \mathbb{R}$ is defined as:

$$L(\boldsymbol{v}) = \left\langle \boldsymbol{f}^{n+1}, \boldsymbol{v} \right\rangle = \int_{\Omega} \boldsymbol{f}^{n+1} \cdot \boldsymbol{v} \, \mathrm{d}\Omega \tag{5.2.5}$$

where $\langle \cdot, \cdot \rangle$ denotes a duality pairing between $\mathcal{V}'$ and $\mathcal{V}$ where $\mathcal{V}'$ is the dual of $\mathcal{V}$. The function $\boldsymbol{f}$ is simply assumed, at this point, to be a source term which is always treated explicitly in this thesis, i.e. $\boldsymbol{f}^{n+1} \approx \boldsymbol{f}^{n}$. This source term may contain body forces such as gravity and any forces due to an immersed structure via the IBM - the IBM term will be considered in §5.4. Note that the weak formulation (5.2.2) implies that the non-linear terms in the constitutive equation are dealt with implicitly. This is not the case. In this thesis, we employ a second-order Adams-Bashforth technique to approximate these non-linear terms as we discussed in §4.1.2.

5.2.2 Spatial Discretisation

In the spatial discretisation of the semi-discrete problem (5.2.2) using the spectral element method, it is necessary to choose conforming discrete subspaces of the velocity, pressure and stress spaces. Let $\mathcal{V}_N \subset \mathcal{V}$, $\mathcal{Q}_N \subset \mathcal{Q}$ and $\mathcal{S}_N \subset \mathcal{S}$ denote these subspaces, respectively. The domain Ω is divided into K uniform, non-overlapping, quadrilateral spectral elements Ω_e, $e = 1, \ldots, K$, such that

$$\bar{\Omega} = \bigcup_{e=1}^{K} \bar{\Omega}_e \tag{5.2.6}$$

Let $P_N(\Omega_e)$ denote the space of all polynomials on Ω_e of degree less than or equal to N and define:

$$\mathrm{P}_N(\Omega) := \left\{ \phi\colon \ \phi|_{\Omega_e} \in P_N(\Omega_e) \right\} \tag{5.2.7}$$

The velocity, pressure and stress approximation spaces may then be defined as:

$$\mathcal{V}_N := \mathcal{V} \cap [\mathrm{P}_N(\Omega)]^2 \tag{5.2.8a}$$

$$\mathcal{Q}_N := \mathcal{Q} \cap \mathrm{P}_{N-2}(\Omega) \tag{5.2.8b}$$

$$\mathcal{S}_N := \mathcal{S} \cap [\mathrm{P}_N(\Omega)]^{2\times 2} \tag{5.2.8c}$$

where it is understood that $[\mathrm{P}_N(\Omega)]^{2\times 2}$ defines each component of a tensor to be a member of $\mathrm{P}_N(\Omega)$. It was shown by Maday *et al.* [49, 47] that the pressure approximation space must be degree $N-2$ in order for the velocity and pressure spaces to be compatible and hence the discrete problem to be well-posed. We will return to this compatibility condition in §5.3.3. It was also shown by Gerritsma and Phillips [34] that the polymeric stress approximation space is compatible with the velocity space if polynomials of degree N are used.

Each spectral element is mapped to the parent domain $D = [-1,1] \times [-1,1]$, where for each point $\boldsymbol{\xi} = (\xi, \eta) \in D$ there exists a point $\boldsymbol{x} = (x(\xi,\eta), y(\xi,\eta)) \in \Omega_e$, using the transfinite mapping, $\boldsymbol{F}$, of Schneidesch and Deville [62] such that $\boldsymbol{x} = \boldsymbol{F}(\boldsymbol{\xi})$. The mapping $\boldsymbol{F}$ is defined by:

$$\begin{aligned}\boldsymbol{F}(\xi,\eta) =& \boldsymbol{\gamma}_1(\xi)\phi_1(\eta) + \boldsymbol{\gamma}_2(\eta)\phi_2(\xi) + \boldsymbol{\gamma}_3(\xi)\phi_2(\eta) + \boldsymbol{\gamma}_4(\eta)\phi_1(\xi) \\ &- \boldsymbol{x}_1\phi_1(\eta)\phi_1(\xi) - \boldsymbol{x}_2\phi_1(\eta)\phi_2(\xi) - \boldsymbol{x}_3\phi_2(\eta)\phi_2(\xi) - \boldsymbol{x}_4\phi_2(\eta)\phi_1(\xi) \end{aligned} \tag{5.2.9}$$

where the parameterisations $\boldsymbol{\gamma}_i$ map the parent element boundaries $\widehat{\Gamma}_i$ onto the corresponding physical element boundaries Γ_i, $\boldsymbol{\gamma}_i \colon \widehat{\Gamma}_i \to \Gamma_i$ and are given by:

$$\boldsymbol{\gamma}_1(\xi) = \boldsymbol{x}(\xi,-1) \qquad \boldsymbol{\gamma}_2(\eta) = \boldsymbol{x}(1,\eta) \tag{5.2.10}$$

$$\boldsymbol{\gamma}_3(\xi) = \boldsymbol{x}(\xi,1) \qquad \boldsymbol{\gamma}_4(\eta) = \boldsymbol{x}(-1,\eta) \tag{5.2.11}$$

Each $\boldsymbol{\gamma}_i$, $i = 1,\ldots,4$, is defined by:

$$\boldsymbol{\gamma}_i(\xi) = \frac{\boldsymbol{\gamma}_i(1) - \boldsymbol{\gamma}_i(-1)}{2}\left[\xi + \frac{\boldsymbol{\gamma}_i(1) + \boldsymbol{\gamma}_i(-1)}{\boldsymbol{\gamma}_i(1) - \boldsymbol{\gamma}_i(-1)}\right] \tag{5.2.12}$$

so that the corners of our physical element are given by $\boldsymbol{x}_1 = \boldsymbol{\gamma}_1(-1) = \boldsymbol{\gamma}_4(-1)$, etc. The so-called blending functions ϕ_i are given by:

$$\phi_1(\xi) = \frac{1-\xi}{2} \qquad \phi_2(\xi) = \frac{1+\xi}{2} \tag{5.2.13}$$

The velocity approximation at time $t^n = n\Delta t$ on the element Ω_e is then given by:

$$\boldsymbol{u}_N^{e,n}(\xi,\eta) = \sum_{i=0}^{N}\sum_{j=0}^{N} \boldsymbol{u}_{i,j}^{e,n} h_i(\xi) h_j(\eta) \tag{5.2.14}$$

where $h_i(\xi)$, $i = 0,\ldots,N$, are the Lagrange interpolants defined on the parent interval $\xi \in [-1,1]$ by:

$$h_i(\xi) = -\frac{(1-\xi^2)L_N'(\xi)}{N(N+1)L_N(\xi_i)(\xi-\xi_i)} \tag{5.2.15}$$

where the points ξ_i, $i = 0,\ldots,N$, are the collocation points on the Gauss-Lobatto Legendre grid. Similarly, the approximation of the polymeric stress at time t^n on the element Ω_e is defined as:

$$\boldsymbol{\tau}_N^{e,n}(\xi,\eta) = \sum_{i=0}^{N}\sum_{j=0}^{N} \boldsymbol{\tau}_{i,j}^{e,n} h_i(\xi) h_j(\eta) \tag{5.2.16}$$

The pressure approximation is defined as:

$$p_N^{e,n}(\xi,\eta) = \sum_{i=1}^{N-1}\sum_{j=1}^{N-1} p_{i,j}^{e,n} \tilde{h}_i(\xi) \tilde{h}_j(\eta) \tag{5.2.17}$$

where $\tilde{h}_j(\xi)$, $j = 1,\ldots,N-1$, are the Lagrange interpolants based on the interior Gauss-Lobatto Legendre points, and are thus defined:

$$\tilde{h}_i(\xi) = -\frac{(1-\xi_i^2)L_N'(\xi)}{N(N+1)L_N(\xi_i)(\xi-\xi_i)} \tag{5.2.18}$$

Note that the pressure is allowed to be discontinuous across element interfaces but the velocity and stress are assumed to be continuous between elements. Gerritsma and Phillips [33] suggested using a discontinuous stress approximation for the velocity-pressure-stress formulation of Stokes flow and found the method to be more stable for the stick-slip benchmark problem. For all of the examples used in this thesis, the polymeric stress will be assumed to be continuous between element edges. The full discrete weak formulation of (5.2.2) is: find $(\boldsymbol{u}_N^{n+1}, p_N^{n+1}, \boldsymbol{\tau}_N^{n+1}) \in \mathcal{V}_N \times \mathcal{Q}_N \times \mathcal{S}_N$ such

that

$$\frac{3\rho}{2\Delta t}(\boldsymbol{u}_N^{n+1}, \boldsymbol{v}_N)_{L^2} + \eta_s a(\boldsymbol{u}_N^{n+1}, \boldsymbol{v}_N) + b(\boldsymbol{v}_N, p_N^{n+1}) = c(\boldsymbol{\tau}_N^n, \boldsymbol{v}_N) + \frac{\rho}{2\Delta t}(4\tilde{\boldsymbol{u}}_N^{n+1} - \tilde{\tilde{\boldsymbol{u}}}_N^{n+1}, \boldsymbol{v}_N)_{L^2} + L(\boldsymbol{v}_N) \qquad \forall \boldsymbol{v}_N \in \mathcal{V}_N \tag{5.2.19a}$$

$$b(\boldsymbol{u}_N^{n+1}, q_N) = 0 \qquad \forall q_N \in \mathcal{Q}_N \tag{5.2.19b}$$

$$\left(1 + \frac{3\lambda}{2\Delta t}\right)(\boldsymbol{\tau}_N^{n+1}, \boldsymbol{s}_N)_{L^2} - \lambda(\boldsymbol{E}_N^{n+1}, \boldsymbol{s}_N)_{L^2} = \eta_p(\dot{\boldsymbol{\gamma}}_N^{n+1}, \boldsymbol{s}_N)_{L^2} + \frac{\lambda}{2\Delta t}(4\tilde{\boldsymbol{\tau}}_N^{n+1} - \tilde{\tilde{\boldsymbol{\tau}}}_N^{n+1}, \boldsymbol{s}_N)_{L^2} \quad \forall \boldsymbol{s}_N \in \mathcal{S}_N \tag{5.2.19c}$$

where the integration in the bilinear forms (5.2.3) is computed using Gauss-Lobatto Legendre quadrature - see Appendix B.

5.2.3 Problems with Unfitted Discontinuities

The spectral element method described in §5.2.2 is a high order method based on polynomial interpolation using the Legendre polynomials. Therefore, within an element Ω_e, the approximations of $\boldsymbol{u}, p$ and $\boldsymbol{\tau}$ are all continuous. If the function being approximated has a discontinuity, which we assume is unfitted to the mesh, spurious oscillations can be seen in the approximation. This phenomenon is well known and is called the Gibbs phenomenon, see e.g. [11, 12]. Gibbs phenomenon can be, formally, classified as the inability to approximate a discontinuity using continuous functions. To illustrate this phenomenon we consider the spectral element interpolation of a discontinuous function on a grid of uniformly spaced points. For simplicity, we will assume that our domain $\Omega \subset \mathbb{R}$ and that we only have a *single* spectral element. Define the domain $\Omega = [-1, 1]$ and assume that the function to be interpolated is piecewise constant:

$$f(x) = \begin{cases} 0 & \forall x \in [-1, 10^{-4}) \\ 1 & \forall x \in [10^{-4}, 1] \end{cases} \tag{5.2.20}$$

Note that the point 10^{-4} is chosen as the intersection between the boundaries of the two subdomains because 0 is a member of the Gauss-Lobatto Legendre grid. We construct

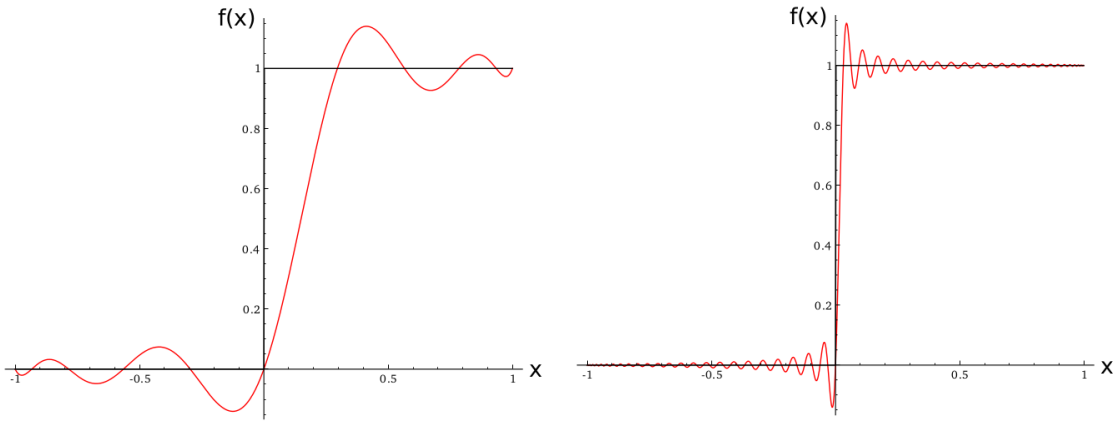
the spectral element interpolant of this function on a uniformly spaced grid using a single element. The uniformly spaced grid is denoted:

$$D_u = \{x_j \in \Omega \colon x_j - x_{j-1} = C, j = 1, \ldots, M\}$$

where $C = \Delta x = 2/M$ is the constant mesh width, $x_0 = -1$ and $x_M = 1$ and $M+1$ is the total number of uniformly spaced points. The interpolant is then:

$$f_N(x_k) = \sum_{i=0}^{N} f_i h_i(x_k) \qquad \forall x_k \in D_u \tag{5.2.21}$$

where the polynomials h_i, $i = 0, \ldots, N$, are the Lagrange interpolants defined in (5.2.15). Figure 5.5 shows the spurious oscillations present around the discontinuity when $M = 1000$ and $N = 10$ and $N = 100$. As N increases from 10 to 100 we see that the frequency of the oscillations increases local to the discontinuity. Away from the discontinuity, we see the amplitude of the oscillations decreases as N increases. Thus, the Gibbs phenomenon becomes more and more local as N increases. This implies that if we let $N \to \infty$ we would eventually converge to the solution.



(a) $N = 10$. **(b)** $N = 100$.

Figure 5.5: Spectral element interpolation of a discontinuous function on a grid of uniformly spaced points for $M = 1000$. The black line is the function f and the red line is the interpolant.

This example, although very simple, illustrates the problems which occur when using spectral methods to approximate a discontinuous function. The spurious oscillations can pollute the other variables in the computation. This phenomenon is also seen in finite elements. However, it is not as severe as it is in spectral methods due to the lower

order polynomial. For viscoelastic fluids, the hyperbolicity of the constitutive equation could cause these oscillations to grow rapidly and cause the numerical computation to break down, even if only a single dependent variable has a discontinuity and all other variables are continuous. One could argue that there is a need to fit the discontinuity to the mesh. However, doing so would greatly increase the computational time, particularly if the discontinuity was allowed to move freely within the computational domain. In the next section we discuss a method to alleviate these oscillations and we return to this example later.

5.3 eXtended Spectral Element Method (XSEM)

The previous subsection §5.2.3, illustrated that the approximation of a discontinuity, that is unfitted to the computational mesh, using continuous polynomials can lead to spurious oscillations local to the discontinuity. These oscillations can propagate throughout the computational domain and pollute other variables. If the discontinuity is assumed to move freely within the domain, fitting the mesh to the discontinuity becomes computationally very expensive. In the case of finite elements, a method known as the eXtended Finite Element Method (XFEM) was proposed by Moës *et al.* [50, 6] in an attempt to alleviate this issue. In the case of a strong discontinuity (the type of discontinuity considered in this thesis), the general idea behind XFEM, in a very formal description, is to *enrich* the original finite element space of admissible functions by *something* discontinuous. This allows the numerics to capture the discontinuity and achieve optimal order of convergence for functions with a lower regularity. This enrichment is achieved by adding to the original finite element space, a space which is spanned by discontinuous basis functions. In this thesis, we apply the method to spectral elements and hence name it, the eXtended Spectral Element Method (XSEM).

The XSEM was first proposed by Legay *et al.* [42] when studying strong and weak discontinuities using, what they called, *spectral finite elements*. In that article, the authors note that additional considerations, such as careful design of the blending elements, are required when higher-order elements are considered. In that article, the union of the elements which contain the discontinuity, either strong or weak, was denoted Ω_{LPU}

and the union of the blending elements (the elements which share an edge or node with the elements of Ω_{LPU}) was denoted Ω_B. The global enriched approximation was given by:

$$u(\boldsymbol{x}) = \sum_{I \in \mathbb{S}} N_I^P(\boldsymbol{x}) u_I + \sum_{J \in \mathbb{S}_P} \phi_J(\boldsymbol{x}) \psi(\boldsymbol{x}) q_J$$

where N_I^P are the spectral basis functions of order P, ψ is an enrichment function, u_I are nodal values, q_J are additional degrees of freedom, $\mathbb{S}$ is the set of nodes in the model and $\mathbb{S}_P \subset \mathbb{S}$ is the set of nodes in Ω_{LPU}. The functions ϕ_J form a local partition of unity over Ω_{LPU}. For the case of strong discontinuities, the functions ϕ_J are constructed from linear spectral basis functions N_J^1 whilst for the case of weak discontinuities, or gradient discontinuities, the functions ϕ_J are constructed from spectral basis functions of one order lower than the standard approximation, i.e. from N_J^{P-1}. The spectral interpolant employed by Legay *et al.* is based on Chebyshev polynomials and the Chebyshev-Gauss grid. Legay *et al.* found that for the case of strong discontinuities no additional considerations were necessary in the blending elements. However, for the case of weak discontinuities, Legay *et al.* found that higher-order terms appear in the blending elements which need to be removed. They noted that using polynomials of degree $P-1$ for the enrichment was sufficient to remove the higher-order terms in the blending elements. However, when $P = 1$, the assumed strain method [28] was required to deal with the blending problems. Even though spectral basis functions were considered by Legay *et al.* [42], the authors approached XSEM from the perspective of high-order FEM. Therefore, the maximum order the authors considered was 4 and h-type convergence (in other words, convergence with respect to mesh width) of the method was studied. Legay *et al.* found that, for weak straight discontinuities, the method obtained *nearly* optimal order of convergence. However, they found that for weak curved discontinuities, the method obtained suboptimal order of convergence. The suboptimal order of convergence was attributed to approximations in the quadrature scheme. The quadrature scheme employed by Legay *et al.* involved subdividing the spectral element containing the discontinuity into smaller elements. If one of these smaller elements contains the discontinuity the smaller element is again subdivided into triangles so that a linear approximation of the discontinuity takes place within the element.

Cheng and Fries [26] studied strong and weak discontinuities with the aim of improv-

ing the suboptimal order of convergence found when higher-order XFEM is applied to curved discontinuities. As the approximations in the quadrature were the source of the suboptimal order of convergence found by Legay *et al.*, Cheng and Fries [26] proposed a different quadrature scheme where they subdivided an element containing the discontinuity into smaller elements where one side of the subelement is curved. This means that there are more nodes on the curved side of the element than the straight sides. This adds additional complexity to the quadrature. In this thesis, we use a different scheme for the quadrature which we discuss in §6.2. Cheng and Fries [26] also proposed using a corrected higher-order XFEM. The standard higher-order XFEM did not obtain optimal rates of convergence for curved weak discontinuities. However, Cheng and Fries found that the corrected higher-order XFEM did obtain optimal rates of convergence when equal order basis functions are used for the standard and extended parts of the enriched approximation. They also considered a modified abs-enrichment (where the enrichment function is an absolute value function) and found that suboptimal order of convergence is seen for curved discontinuities. Again all rates of convergence considered were h-type, that is convergence with respect to mesh width and the maximum polynomial degree considered was again 4.

In this thesis, we approach XSEM from the angle of spectral methods and our aim is to obtain optimal order of convergence with respect to polynomial degree, from which we can then infer spectral accuracy on functions with discontinuities. In this section we summarise the spatial discretisation of XSEM and we introduce the framework of Reusken [60] which we use to prove XSEM approximation results and discuss the inf-sup condition.

For the purposes of this section, we let $\Omega \subset \mathbb{R}^2$ contain two subdomains Ω_1 and Ω_2 such that $\bar{\Omega} = \bar{\Omega}_1 \cup \bar{\Omega}_2$ and $\Omega_1 \cap \Omega_2 = \emptyset$. Let Γ denote the interface (and discontinuity) between the two regions such that $\Gamma = \bar{\Omega}_1 \cap \bar{\Omega}_2 = \partial\Omega_1 \cap \partial\Omega_2$. Note that we always assume that the interface is Lipschitz. Lipschitz boundaries are important for the study of the regularity of solutions for boundary value problems as they describe the smoothness of the domain boundary. We do not make explicit use of the Lipschitz conditions here. Later in this thesis, the interface Γ will be represented by an immersed elastic membrane. In the majority of the examples considered in Chapter 7, 8 and 9,

the pressure variable will be discontinuous but the polymeric stress and velocity will both be continuous across the interface.

5.3.1 Spatial Discretisation

We begin, as in §5.2.1, by defining *suitable* function spaces for the dependent variables of the equations governing viscoelastic fluid flow. As the pressure variable is the only variable being enriched - due its discontinuous nature across the interface - we keep the function spaces for velocity and polymeric stress the same as in (5.2.1a) and (5.2.1c), respectively. Fortunately, as the pressure is assumed to be fairly irregular we do *not* consider changing it either. However, for brevity, we denote the space $\mathcal{Q}^\Gamma = L_0^2(\Omega)$, where the superscript Γ is used to denote an enriched space or an enriched function and $L_0^2(\Omega)$ is as defined in (5.2.1b). Note that in general, for an arbitrary discontinuity, the enriched function space $\mathcal{W}^\Gamma$ will be larger than the one used in the standard approximation, $\mathcal{W}$, as the idea of XSEM is to enrich the original space. The particular choice of $\mathcal{W}^\Gamma$ is problem dependent as it depends on the *kind* of discontinuity, e.g. strong or weak. For the approximation results given later, in §5.3.2, the space of functions will be denoted $\mathcal{V}^\Gamma$ and be defined as a *broken* Sobolev space. The weak form of the semi-discrete equations, in the absence of the immersed boundary, is exactly as in (5.2.2) except that we replace $\mathcal{Q}$, p^{n+1} and q with $\mathcal{Q}^\Gamma$, $p^{\Gamma,n+1}$ and q^Γ, respectively.

As before, we decompose our domain into K uniform, non-overlapping, quadrilateral spectral elements Ω_e, $e = 1, \ldots, K$, which are *not* fitted to Γ, such that (5.2.6) is satisfied. In §5.2.2, the discrete approximation subspaces $\mathcal{V}_N$, $\mathcal{Q}_N$ and $\mathcal{S}_N$ were then defined, in (5.2.8), by an intersection with the polynomial space, defined by (5.2.7). While the approximation spaces $\mathcal{V}_N$ and $\mathcal{S}_N$ are still suitable for the velocity and polymeric stress approximations, the approximation space $\mathcal{Q}_N$ is not suitable for the pressure approximation due to the function being discontinuous inside an element Ω_e. Therefore we need a suitable approximation space for the pressure. In this subsection, we introduce a suitable approximation space for the enriched pressure approximation (which we also introduce) and then illustrate the effect the enrichment has on the approximation of a discontinuous function. Following Groß and Reusken [38], we introduce the set of

elements containing the discontinuity Γ:

$$\Omega^{\Gamma} := \{\Omega_e \colon \Omega_e \cap \Gamma \neq \emptyset,\, e = 1, \ldots, K\} \tag{5.3.1}$$

Let $\{\Psi_i,\, i \in \mathcal{I}\}$ denote the global basis functions of $\mathcal{Q}_N$, with $\mathcal{I} = \{1, \ldots, N_Q\}$ where N_Q denotes the dimension of $\mathcal{Q}_N$, and define the space

$$\mathcal{P}(\Omega^{\Gamma}) := \text{ span } \left\{\Psi_i \Phi_i,\, i \in \mathcal{I}^{\Gamma}\right\} \tag{5.3.2}$$

where $\mathcal{I}^{\Gamma} \subset \mathcal{I}$ is a subset of nodal points which we require to be *enriched* and Φ_i are global enrichment functions which for the moment we leave undefined. The enriched approximation space is then defined as:

$$\mathcal{Q}_N^{\Gamma} := \mathcal{Q}_N \oplus \mathcal{P}(\Omega^{\Gamma}) \tag{5.3.3}$$

Therefore, the global XSEM approximation of a function $p^{\Gamma} \in \mathcal{Q}^{\Gamma}$ is:

$$\begin{aligned} \mathcal{Q}_N^{\Gamma} \ni p_N^{\Gamma}(x,y) &= p_N(x,y) + p_N^X(x,y) \\ &= \sum_{i \in \mathcal{I}} p_i \Psi_i(x,y) + \sum_{j \in \mathcal{I}^{\Gamma}} \alpha_j \Psi_j(x,y) \Phi_j(x,y) \end{aligned} \tag{5.3.4}$$

where p_N^X is termed the extended part of the enriched approximation p_N^{Γ} and the α_j's are additional degrees of freedom at the nodal points which have been enriched. The global XSEM approximation is very similar to its XFEM counterpart, [38]. Each spectral element is mapped on to the parent domain $D = \{(\xi, \eta) \in [-1,1] \times [-1,1]\}$ via the transfinite map $\boldsymbol{F}$ defined in (5.2.9). Note that over an element $\Omega_e \notin \Omega^{\Gamma}$, the local enriched approximation must satisfy $p_N^{\Gamma}(\xi,\eta) = p_N(\xi,\eta)$ so that on elements without a discontinuity, the method reduces to standard SEM. If we restrict (5.3.4) to an element $\Omega_e \in \Omega^{\Gamma}$, then the local enriched approximation is:

$$\begin{aligned} p_N^{\Gamma}(\xi,\eta) &= p_N(\xi,\eta) + p_N^X(\xi,\eta) \\ &= \sum_{i,j=1}^{N-1} p_{i,j} \tilde{h}_i(\xi) \tilde{h}_j(\eta) + \sum_{k,l=1}^{N-1} \alpha_{k,l} \tilde{h}_k(\xi) \tilde{h}_l(\eta) \phi_{k,l}(\xi,\eta) \end{aligned} \tag{5.3.5}$$

where $\tilde{h}_i$ are the Lagrange interpolants based on the interior Gauss-Lobatto Legendre grid and are defined in (5.2.18). The function $\phi_{k,l}$ is a local version of the enrichment

function Φ_j present in the extended part of the global enriched approximation. This function, obviously, depends on the type of discontinuity, or singularity, being enriched. For the some of examples considered in Chapter 8 and 9, we are interested in functions with a strong discontinuity across the interface Γ. Therefore we use the same global enrichment function as one defined in [38], i.e. $\Phi_i(x,y) = H(x,y) - H(x_i,y_i)$, where $H(x,y)$ is the Heaviside function defined by:

$$H(x,y) = \begin{cases} 0 & \boldsymbol{x} = (x,y) \in \Omega_1 \\ 1 & \boldsymbol{x} = (x,y) \in \Omega_2 \end{cases} \tag{5.3.6}$$

Therefore, through the transfinite map $\boldsymbol{F}$, it makes sense for us to define our local version $\phi_{k,l}$ as $\phi_{k,l}(\xi,\eta) = H(\xi,\eta) - H(\xi_k,\eta_l)$ where the Heaviside function is now defined as:

$$H(\xi,\eta) = \begin{cases} 0 & \boldsymbol{x} = \boldsymbol{F}(\xi,\eta) \in \Omega_1 \\ 1 & \boldsymbol{x} = \boldsymbol{F}(\xi,\eta) \in \Omega_2 \end{cases} \tag{5.3.7}$$

To illustrate the eXtended Spectral Element Method, we consider the same example that was used in §5.2.3; that is the interpolation of a piecewise constant function $f(x)$ defined by (5.2.20). Again we construct the interpolant, over a single element, of the function on a uniformly spaced grid defined by the set D_u. The XSEM interpolant is then

$$f_N^{\Gamma}(x_k) = \sum_{i=0}^{N} f_i h_i(x_k) + \sum_{j=0}^{N} \alpha_j h_j(x_k)\phi_j(x_k) \qquad \forall x_k \in D_u \tag{5.3.8}$$

where h_j are the Lagrange interpolants defined by (5.2.15). The coefficients α_j are completely unknown. In order to calculate them, we assume that $f_N^{\Gamma}(x_k) \equiv f(x_k)$, $\forall x_k \in D_u$. The α_j's are then found from the residual of the standard SEM approximation:

$$\sum_{j=0}^{N} M_{k,j}\alpha_j = f(x_k) - \sum_{i=0}^{N} f_i h_i(x_k) \qquad \forall x_k \in D_u \tag{5.3.9}$$

where the entries of the matrix are given by $M_{k,j} = h_j(x_k)\phi_j(x_k)$. Note that the number of uniformly spaced points will in general be larger than the degree of the

polynomial, thus the matrix is not square. Therefore it is inverted by multiplying by its transpose to produce a square matrix which is then inverted. Figure 5.6 shows the XSEM interpolant. It is evident that it has captured the discontinuity *exactly*. Unfortunately, we are unable to take $N \geq 13$ because the Choleski factorisation (of $M^T M$) we have implemented fails to converge to a solution, as the diagonal entries of the lower triangular matrix, L, either become close to zero, or contain complex numbers even though the matrix M always contains real values. As N increases the standard SEM approximation tends to the exact solution, therefore the right hand side of (5.3.9) will tend to zero. This results in the matrix M becoming increasingly singular as N increases. Intuitively, we can interpret this as a decrease in the *amount* of enrichment required. This idea is revisited in our discussion of the inf-sup condition in §5.3.3 later.

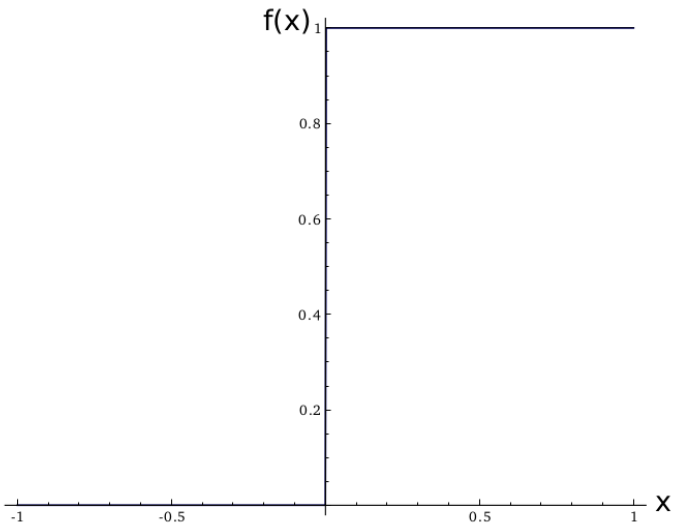


Figure 5.6: XSEM interpolation of a discontinuous function on a grid of uniformly spaced points for $M = 1000$ and $N = 10$. The black line is the function f, the blue line (underneath the black line) is the XSEM interpolant.

5.3.2 Approximation Results

To prove the approximation results in this section, we use the framework of Reusken [60]. Most of the difficulty in analysing the method comes from the dependence of the approximation on the enrichment function Φ. Depending on the *kind* of enrichment which is required, this function can vary quite dramatically. Even in the case of strong and weak discontinuities at an interface, the enrichment function is completely different. Therefore, it would be difficult to analyse the method in a unified manner when

an enrichment function is considered. The framework of Reusken provides a unified treatment of the XFEM/XSEM approximation, for functions with strong or weak discontinuities, by removing the enrichment function. Therefore, the approach allows for a relatively straightforward analysis. The reader will notice that the proofs given here are very similar to their XFEM counterparts given in [60] but are included for completeness. Before we prove our results, we briefly summarize the Reusken framework. Let the space of functions $\mathcal{V}^\Gamma$ be defined as:

$$\mathcal{V}^\Gamma = H^m(\Omega_1 \cup \Omega_2) = \left\{u \in L^2(\Omega) \colon u \in H^m(\Omega_i),\ i = 1, 2\right\} \tag{5.3.10}$$

where $H^m(\Omega_1 \cup \Omega_2)$ is a broken Sobolev space of order m equipped with the norm

$$\|u\|^2_{H^1(\Omega_1 \cup \Omega_2)} = \sum_{i=1}^{2} \|u\|^2_{H^1(\Omega_i)} \tag{5.3.11}$$

where $m \geq 1$ is an integer. Note that if we allow $m = 0$, then we can define:

$$H^0(\Omega_1 \cup \Omega_2) = L^2(\Omega_1 \cup \Omega_2) = \left\{u \in L^2(\Omega) \colon u \in L^2(\Omega_i),\ i = 1, 2\right\} = L^2(\Omega) \tag{5.3.12}$$

Note that in the definition of the broken Sobolev space (5.3.10), we have defined a function u to be in L^2. Buffa and Ortner [21] used L^1 functions in their definition of a broken Sobolev space. The reason for choosing a broken Sobolev space is that we lower the regularity defined on the interface, thus allowing a greater class of functions to be considered. As was done in (5.2.8), we would like to construct the approximation space as $\mathcal{V}^\Gamma_N = \mathcal{V}^\Gamma \cap \mathrm{P}_N(\Omega)$ where $\mathrm{P}_N(\Omega)$ is defined in (5.2.7). However, for an element Ω_e this implies that approximation is continuous and hence is not suitable for XSEM. If such a space is used, in the case of finite elements, we cannot expect a better bound than, [60]:

$$\inf_{u_h \in \mathcal{V}^\Gamma_h} \|u - u_h\|_{L^2(\Omega)} \leq C\sqrt{h}\, \|u\|_{H^1(\Omega_1 \cup \Omega_2)} \tag{5.3.13}$$

We expect a similar bound for spectral elements as when the regularity of the solution decays, spectral elements will not necessarily perform better than finite elements. However, we are unaware of any such bound for spectral elements. In all the XFEM literature, as far as we are aware, the enriched approximation is based on the original approximation and this framework is no exception. Note that for the purposes of these

proofs, we consider only a *single* spectral element. Let N_V denote the dimension of $\mathcal{V}_N$ and let $\{\Psi_k\}_{k\in\mathcal{I}}$, where $\mathcal{I} = \{1,\ldots,N_V\}$, denote the global basis functions spanning $\mathcal{V}_N$. Let $\mathcal{X} = \{\boldsymbol{x}_k\}_{k\in\mathcal{I}}$ be the set of all nodal points. Reusken's idea was to define the enriched function space as a restriction of the original approximation space to each side of the interface Γ - this is the reason why the framework is only suited to strong or weak discontinuities. The restriction operator, $R_i\colon L^2(\Omega) \to L^2(\Omega)$, $i = 1,2$, is defined as:

$$R_i u = \begin{cases} u|_{\Omega_i} & \text{in } \Omega_i \\ 0 & \text{otherwise} \end{cases} \tag{5.3.14}$$

Hence the enriched approximation space is defined as: $\mathcal{V}_N^\Gamma = R_1\mathcal{V}_N \oplus R_2\mathcal{V}_N$. In Theorem 2 of [60], Reusken showed that

$$\mathcal{V}_N^\Gamma = R_1\mathcal{V}_N \oplus R_2\mathcal{V}_N = \mathcal{V}_N \oplus \mathcal{V}_N^{\Gamma,1} \oplus \mathcal{V}_N^{\Gamma,2}$$

where

$$\mathcal{V}_N^{\Gamma,i} = \text{ span } \left\{R_i\Psi_j\colon j \in \mathcal{I}_i^\Gamma\right\} \qquad i = 1,2$$

and

$$\mathcal{I}_1^\Gamma = \{j \in \mathcal{I}\colon \boldsymbol{x}_j \in \Omega_2 \quad \underline{\text{and}} \quad \text{supp}(\Psi_j) \cap \Gamma \neq \emptyset\} \tag{5.3.15}$$

$$\mathcal{I}_2^\Gamma = \{j \in \mathcal{I}\colon \boldsymbol{x}_j \in \Omega_1 \quad \underline{\text{and}} \quad \text{supp}(\Psi_j) \cap \Gamma \neq \emptyset\} \tag{5.3.16}$$

This representation is similar to how XFEM/XSEM is implemented in practice. Additionally, one may write:

$$\mathcal{V}_N^\Gamma \ni u_N^\Gamma = u_N + \sum_{k\in\mathcal{I}_1^\Gamma} \beta_k^{(1)} R_1\Psi_k + \sum_{k\in\mathcal{I}_2^\Gamma} \beta_k^{(2)} R_2\Psi_k \tag{5.3.17}$$

where $u_N \in \mathcal{V}_N$ is the standard approximation and $\beta_k^{(i)}$ are additional degrees of freedom - for full details of their derivation the reader is referred to [60]. We are now in a position to present the XSEM approximation results in the case of a single element.

Theorem 1:
Let $\Omega = [-1,1]^d$ with $d \in \mathbb{Z}^+$. Define $\mathcal{E}_i^m\colon H^m(\Omega_i) \to H^m(\Omega)$ to be an extension

operator such that

$$(\mathcal{E}_i^m u)|_{\Omega_i} = u \qquad \textit{and} \qquad \|\mathcal{E}_i^m u\|_{H^m(\Omega)} \leq c\,\|u\|_{H^m(\Omega_i)} \qquad \forall u \in H^m(\Omega_i)$$

Let $\pi_N^m \colon H^m(\Omega) \to \mathcal{V}_N := H^m(\Omega) \cap P_N(\Omega)$ *be a projection operator satisfying*

$$\|u - \pi_N^m u\|_{L^2(\Omega)} \leq C_1 N^{-m}\,\|u\|_{H^m(\Omega)} \qquad (m \geq 0) \tag{5.3.18}$$

$$|u - \pi_N^m u|_{H^1(\Omega)} \leq C_2 N^{1-m}\,\|u\|_{H^m(\Omega)} \qquad (m \geq 2) \tag{5.3.19}$$

$\forall u \in H^m(\Omega)$, *[22, p. 314]. Then*

$$\inf_{u_N^\Gamma \in \mathcal{V}_N^\Gamma} \left\|u - u_N^\Gamma\right\|_{L^2(\Omega_1 \cup \Omega_2)} \leq C N^{-m}\,\|u\|_{H^m(\Omega_1 \cup \Omega_2)} \qquad (m \geq 0) \tag{5.3.20}$$

$$\inf_{u_N^\Gamma \in \mathcal{V}_N^\Gamma} \left\|u - u_N^\Gamma\right\|_{H^1(\Omega_1 \cup \Omega_2)} \leq C N^{1-m}\,\|u\|_{H^m(\Omega_1 \cup \Omega_2)} \qquad (m \geq 2) \tag{5.3.21}$$

Proof. Let $u \in H^m(\Omega_1 \cup \Omega_2)$. Let $u_N^\Gamma \in \mathcal{V}_N^\Gamma$ be defined by

$$u_N^\Gamma = R_1 \pi_N^m \mathcal{E}_1^m R_1 u + R_2 \pi_N^m \mathcal{E}_2^m R_2 u \tag{5.3.22}$$

Then

$$\begin{aligned}
\left\|u - u_N^{\Gamma}\right\|^2_{L^2(\Omega_1\cup\Omega_2)} &= \sum_{i=1}^{2} \left\|u - u_N^{\Gamma}\right\|^2_{L^2(\Omega_i)} \\
&= \sum_{i=1}^{2} \left\|u - \pi_N^m \mathcal{E}_i^m R_i u\right\|^2_{L^2(\Omega_i)} \\
&= \sum_{i=1}^{2} \left\|\mathcal{E}_i^m R_i u - \pi_N^m \mathcal{E}_i^m R_i u\right\|^2_{L^2(\Omega_i)} \\
&\leq \sum_{i=1}^{2} \left\|\mathcal{E}_i^m R_i u - \pi_N^m \mathcal{E}_i^m R_i u\right\|^2_{L^2(\Omega)} \\
&\leq C_1^2 N^{-2m} \sum_{i=1}^{2} \left\|\mathcal{E}_i^m R_i u\right\|^2_{H^m(\Omega)} \\
&\leq C^2 N^{-2m} \sum_{i=1}^{2} \left\|R_i u\right\|^2_{H^m(\Omega_i)} \\
&= C^2 N^{-2m} \sum_{i=1}^{2} \left\|u\right\|^2_{H^m(\Omega_i)} \\
&= C^2 N^{-2m} \left\|u\right\|^2_{H^m(\Omega_1\cup\Omega_2)} \qquad (5.3.23)
\end{aligned}$$

Now for the H^1-norm. Let $\|u\|^2_{H^1(\Omega)} = \|u\|^2_{L^2(\Omega)} + |u|^2_{H^1(\Omega)}$ where the H^1 semi-norm, $|u|^2_{H^1(\Omega)}$, of a function $u \in H^1(\Omega)$, $\Omega \subset \mathbb{R}^2$, is given by:

$$|u|^2_{H^1(\Omega)} = \int_\Omega \left\{ \left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 \right\} \, \mathrm{d}\Omega \qquad (5.3.24)$$

Then

$$
\begin{aligned}
\left\|u - u_N^\Gamma\right\|^2_{H^1(\Omega_1\cup\Omega_2)} &= \sum_{i=1}^{2} \left\|u - u_N^\Gamma\right\|^2_{H^1(\Omega_i)} \\
&= \sum_{i=1}^{2} \left[\left\|u - u_N^\Gamma\right\|^2_{L^2(\Omega_i)} + \left|u - u_N^\Gamma\right|^2_{H^1(\Omega_i)}\right] \\
&= \left\|u - u_N^\Gamma\right\|^2_{L^2(\Omega_1\cup\Omega_2)} + \sum_{i=1}^{2} \left|u - u_N^\Gamma\right|^2_{H^1(\Omega_i)} \\
&= \left\|u - u_N^\Gamma\right\|^2_{L^2(\Omega_1\cup\Omega_2)} + \sum_{i=1}^{2} \left|u - \pi_N^m \mathcal{E}_i^m R_i u\right|^2_{H^1(\Omega_i)} \\
&= \left\|u - u_N^\Gamma\right\|^2_{L^2(\Omega_1\cup\Omega_2)} + \sum_{i=1}^{2} \left|\mathcal{E}_i^m R_i u - \pi_N^m \mathcal{E}_i^m R_i u\right|^2_{H^1(\Omega_i)} \\
&\leq \left\|u - u_N^\Gamma\right\|^2_{L^2(\Omega_1\cup\Omega_2)} + \sum_{i=1}^{2} \left|\mathcal{E}_i^m R_i u - \pi_N^m \mathcal{E}_i^m R_i u\right|^2_{H^1(\Omega)} \\
&\leq C^2 N^{-2m} \left\|u\right\|^2_{H^m(\Omega_1\cup\Omega_2)} + C_2^2 N^{2(1-m)} \sum_{i=1}^{2} \left\|\mathcal{E}_i^m R_i u\right\|^2_{H^m(\Omega)} \\
&\leq C^2 N^{-2m} \left\|u\right\|^2_{H^m(\Omega_1\cup\Omega_2)} + \hat{C}^2 N^{2(1-m)} \sum_{i=1}^{2} \left\|R_i u\right\|^2_{H^m(\Omega_i)} \\
&\leq C^2 N^2 N^{-2m} \left\|u\right\|^2_{H^m(\Omega_1\cup\Omega_2)} + \hat{C}^2 N^{2(1-m)} \sum_{i=1}^{2} \left\|R_i u\right\|^2_{H^m(\Omega_i)} \\
&= C N^{2(1-m)} \left\|u\right\|^2_{H^m(\Omega_1\cup\Omega_2)} \qquad (5.3.25)
\end{aligned}
$$

□

We note that the theorem given above is essentially the same as the one given by Reusken [60]. Indeed, the only difference between the XSEM result above and the XFEM result of Reusken [60], is the projection operator π_N^m and the use of the H^1-semi-norm. Also, the steps in the proof given above are exactly the same as in [60] and are included for completeness. We also note that we can use either a projection operator or an interpolation operator in this case because we are using Legendre polynomials based on the Gauss-Lobatto-Legendre grid and it is well known that, in that instance, the interpolation error asymptotically behaves the same as the projection error [22]. We have proved that the XSEM approximation of a function u from a broken Sobolev

space should indeed be optimal with respect to N in both the L^2 and H^1 norms. This result is numerically validated in §6.1.

5.3.3 Inf-Sup Condition

In this section we concern ourselves only with a discussion of the inf-sup condition. We begin with the continuous inf-sup condition and recall some results of Maday *et al.* [49, 47] for the discrete inf-sup condition for the standard SEM approximation, before finally discussing the discrete inf-sup condition for the XSEM approximation. Note that for simplicity, we will assume that our domain is $\Omega = [-1, 1]^2$ and that we only have a single element. The governing equations given in (2.2.24) are expressed in terms of velocity, pressure and extra-stress the so-called mixed formulation or velocity-pressure-stress formulation. The compatibility of the velocity-pressure-stress formulation was considered by Gerritsma and Phillips [34], where their analysis introduced a doubly-constrained minimisation problem. For the time being, we will assume that there is no extra stress so that we can consider the velocity-pressure formulation of the Stokes problem (2.2.12); the velocity-pressure-stress formulation is left as future work.

The weak formulation of Stokes problem is: find $(\boldsymbol{u}, p) \in \mathcal{V} \times \mathcal{Q}$ such that

$$\eta_s a(\boldsymbol{u}, \boldsymbol{v}) + b(\boldsymbol{v}, p) = \langle \boldsymbol{f}, \boldsymbol{v} \rangle \qquad \forall \boldsymbol{v} \in \mathcal{V} \tag{5.3.26a}$$

$$b(\boldsymbol{u}, q) = 0 \qquad \forall q \in \mathcal{Q} \tag{5.3.26b}$$

where $\mathcal{V}$ and $\mathcal{Q}$ are the function spaces for the velocity and pressure, respectively. Note that we let $\eta_s = 1$. The bilinear forms $a(\cdot, \cdot)$ and $b(\cdot, \cdot)$ induce continuous linear operators $A \colon \mathcal{V} \to \mathcal{V}'$ and $B \colon \mathcal{V} \to \mathcal{Q}'$, respectively, such that

$$\langle A\boldsymbol{u}, \boldsymbol{v} \rangle_{\mathcal{V}' \times \mathcal{V}} = a(\boldsymbol{u}, \boldsymbol{v}) \qquad \forall \boldsymbol{u} \in \mathcal{V},\ \forall \boldsymbol{v} \in \mathcal{V} \tag{5.3.27}$$

$$\langle B\boldsymbol{v}, q \rangle_{\mathcal{Q}' \times \mathcal{Q}} = \langle \boldsymbol{v}, B^* q \rangle_{\mathcal{V} \times \mathcal{V}'} = b(\boldsymbol{u}, q) \qquad \forall \boldsymbol{v} \in \mathcal{V},\ \forall q \in \mathcal{Q} \tag{5.3.28}$$

where $B^* \colon \mathcal{Q} \to \mathcal{V}'$ is the transpose, or adjoint, of the operator B. These operators

allow us to write (5.3.26) in an equivalent dual formulation:

$$A\boldsymbol{u} + B^* p = \boldsymbol{f} \qquad \text{in } \mathcal{V}' \tag{5.3.29a}$$

$$B\boldsymbol{u} = 0 \qquad \text{in } \mathcal{Q}' \tag{5.3.29b}$$

The inf-sup condition arises from consideration of the existence and uniqueness of the solution to (5.3.26) and is dependent on the properties of the operator B - specifically its range and kernel, $\mathrm{Rg}(B)$ and $\ker(B)$ respectively. Thus, we state Theorem 1.1 of Brezzi and Fortin [20] verbatim:

Theorem 2:
Let the bilinear forms $a(\cdot,\cdot)$ and $b(\cdot,\cdot)$ be continuous on $\mathcal{V}\times\mathcal{V}$ and $\mathcal{V}\times\mathcal{Q}$, respectively. Let the range of B, denoted $\mathrm{Rg}(B)$, be closed in $\mathcal{Q}'$; that is, there exists a $k_0 > 0$ such that

$$\sup_{\boldsymbol{v}\in\mathcal{V}} \frac{b(\boldsymbol{v}, q)}{\|\boldsymbol{v}\|_{\mathcal{V}}} \geq k_0 \|q\|_{\mathcal{Q}/\ker(B^*)} \tag{5.3.30}$$

Moreover, let the bilinear form $a(\cdot,\cdot)$ be coercive on $\ker(B)$; that is, there exists α_0 such that

$$a(\boldsymbol{v}_0, \boldsymbol{v}_0) \geq \alpha_0 \|\boldsymbol{v}_0\|_{\mathcal{V}}^2 \qquad \forall \boldsymbol{v}_0 \in \ker(B) \tag{5.3.31}$$

Then there exists a unique solution $(\boldsymbol{u}, p) \in \mathcal{V}\times\mathcal{Q}/\ker(B^*)$ to (5.3.26) for any $\boldsymbol{f}\in\mathcal{V}'$ provided $0 \in \mathrm{Rg}(B)$.

Note that Theorem 1.1 of Brezzi and Fortin [20] requires that the bilinear form $a(\cdot,\cdot)$ only be invertible on $\ker(B)$ rather than coercive as they state that coercivity is too strong a condition. Infact, the coercivity on $\ker(B)$ is only a sufficient condition, whereas the weaker condition that requires $a(\cdot,\cdot)$ to only be invertible (nonsingular) is both a sufficient and necessary condition. However, for the Stokes problem considered here, the bilinear form $a(\cdot,\cdot)$ is coercive. The statement (5.3.30) is known as the continuous inf-sup condition and mathematically describes compatibility between the velocity and pressure spaces.

We now turn to the finite-dimensional setting of the discrete problem: find $(\boldsymbol{u}_N, p_N) \in \mathcal{V}_N \times \mathcal{Q}_N$ such that

$$a(\boldsymbol{u}_N, \boldsymbol{v}_N) + b(\boldsymbol{v}_N, p_N) = \langle \boldsymbol{f}_N, \boldsymbol{v}_N \rangle \qquad \forall \boldsymbol{v}_N \in \mathcal{V}_N \tag{5.3.32a}$$

$$b(\boldsymbol{u}_N, q_N) = 0 \qquad \forall q_N \in \mathcal{Q}_N \tag{5.3.32b}$$

where $\mathcal{V}_N \subset \mathcal{V}$ and $\mathcal{Q}_N \subset \mathcal{Q}$ are the approximation spaces for the velocity and pressure, respectively. Once again the bilinear forms $a(\cdot,\cdot)$ and $b(\cdot,\cdot)$ induce linear operators A_N and B_N, respectively. The corresponding discrete dual formulation can now be expressed as:

$$A_N \boldsymbol{u}_N + B_N^* p_N = \boldsymbol{f}_N \qquad \text{in } \mathcal{V}_N' \tag{5.3.33a}$$

$$B_N \boldsymbol{u}_N = 0 \qquad \text{in } \mathcal{Q}_N' \tag{5.3.33b}$$

The results of Theorem 2 above can be used to ensure that the discrete problem has a unique solution $(\boldsymbol{u}_N, p_N) \in \mathcal{V}_N \times \mathcal{Q}_N / \ker(B_N^*)$ (Proposition 2.1 of Brezzi and Fortin [20]). In Theorem 2, the pressure solution is uniquely determined up to an element of $\ker(B^*)$ (a constant), whereas for the discrete problem, the pressure solution is uniquely determined up to an element of $\ker(B_N^*)$. Therefore we require that $\ker(B_N^*) \subseteq \ker(B^*)$, otherwise spurious pressure modes of the discrete problem may be seen. We begin by considering the standard spectral method based on Legendre polynomials over the domain $[-1,1]^2$ before considering an extended spectral method.

The first step in analysing an inf-sup condition, is to guarantee that there does not exist any spurious modes. Let $\mathcal{V}_N$ and $\mathcal{Q}_N$ be defined as in (5.2.8a) and (5.2.8b), respectively. Also let $\mathcal{M}_N = \mathcal{Q} \cap \mathrm{P}_N(\Omega)$, where $\Omega = [-1,1]^2$, and $\mathcal{D}_N$ be the range of $\mathcal{V}_N$ by the divergence operator. Denote by $\mathcal{Z}_N$ the set of all polynomials of $\mathrm{P}_N(\Omega)$ that are orthogonal to $\mathcal{D}_N$ with respect to the $L^2(\Omega)$ inner product. Then according to Lemma 3.3 of Maday *et al.* [49], and Proposition 4.1 of Bernardi and Maday [7, p. 126], the set $\mathcal{Z}_N$ is given by:

$$\begin{aligned} \mathcal{Z}_N = \text{ span } \big\{ & 1, L_N(x), L_N(y), L_N(x)L_N(y), L'_{N+1}(x)L'_{N+1}(y), \\ & L'_N(x)L'_{N+1}(y), L'_{N+1}(x)L'_N(y), L'_N(x)L'_N(y) \big\} \end{aligned} \tag{5.3.34}$$

If we let $p_N \in \mathcal{M}_N$, then $\mathcal{Z}_N$ is the set of spurious modes because the addition of any element of $\mathcal{Z}_N$, or linear combination thereof, to the pressure approximation will not affect the velocity solution. The approximation space $\mathcal{Q}_N$ is then chosen such that $\mathcal{M}_N = \mathcal{Q}_N \oplus \mathcal{Z}_N$, in other words the pressure *must* be a polynomial of degree $N-2$, as defined in (5.2.8b).

Remark: *Actually, the pressure doesn't* have *to be a polynomial of degree* $N-2$. *It was shown by Bernardi and Maday [8] that the pressure can be a polynomial of degree* m *where*

- $m = N - \lambda$ *for a fixed* $\lambda \geq 2$,
- $m = N - \lambda N^{\alpha}$ *for two real numbers* $\lambda > 0$ *and* $0 < \alpha < 1$,
- $m = \lambda\sqrt{N}$ *for a positive and small enough real number* λ,
- $m = \lambda N$ *for* $0 < \lambda < 1$.

It was shown in [8] that when $m = \lambda N$ *the discrete inf-sup parameter is independent of* N.

This is the so-called $P_N \times P_{N-2}$ method of Maday *et al.* [49] and guarantees that $\ker(B_N^*) \subseteq \ker(B^*)$. From this we can deduce that the velocity and pressure approximation spaces are indeed compatible. According to Remark 2.10 of Brezzi and Fortin [20], if $\ker(B_N^*) \subseteq \ker(B^*)$ and the continuous inf-sup condition (5.3.30) holds then we can guarantee that the discrete inf-sup condition holds; that is there exists a $k_N > 0$, in general dependent on N, such that:

$$\sup_{\boldsymbol{v}_N \in \mathcal{V}_N} \frac{b(\boldsymbol{v}_N, q_N)}{\|\boldsymbol{v}_N\|_{\mathcal{V}}} \geq k_N \|q_N\|_{\mathcal{Q}/\ker(B_N^*)} \tag{5.3.35}$$

The above inequality is the mathematical statement that the velocity and pressure approximation spaces are compatible. This, together with the assumption that the bilinear forms $a(\cdot,\cdot)$ and $b(\cdot,\cdot)$ are continuous on $\mathcal{V} \times \mathcal{V}$ and $\mathcal{V} \times \mathcal{Q}$, respectively, and that $a(\cdot,\cdot)$ is coercive on $\ker(B_N)$ is sufficient to prove that the problem (5.3.32) is well-posed and has a unique solution $(\boldsymbol{u}_N, p_N) \in \mathcal{V}_N \times \mathcal{Q}_N/\ker(B_N^*)$. The next step

is to attempt to obtain the parameter k_N as it is crucial in determining *a-priori* error estimates. The method used in determining k_N is *very* technical and can be found in Bernardi and Maday [7] for 2D and Maday *et al.* [49] for 3D and is omitted here due its technicality. The discrete inf-sup condition was found, in both articles, to be:

$$\sup_{\boldsymbol{v}_N \in \mathcal{V}_N} \frac{b(\boldsymbol{v}_N, q_N)}{\|\boldsymbol{v}_N\|_{\mathcal{V}}} \geq c N^{\frac{1-d}{2}} \|q_N\|_{\mathcal{Q}/\ker(B_N^*)} \tag{5.3.36}$$

where $d = 2, 3$. The fact that k_N is dependent on N is a consequence of the pressure being approximated by polynomials of degree $N - 2$. The factor $N^{\frac{1-d}{2}}$ impairs the order of convergence for the pressure approximation. We can now move to discuss the inf-sup analysis for XSEM.

Just as before, we wish to prove that $\ker(B_N^*) \subseteq \ker(B^*)$. Let Ω contain two subdomains Ω_1 and Ω_2, respectively, such that $\bar{\Omega} = \bar{\Omega}_1 \cup \bar{\Omega}_2$. Let $\mathcal{Q}_N^\Gamma$ be the approximation space for the extended pressure. In XSEM, additional basis functions are added to the original approximation; typically we would have for $p_N^\Gamma \in \mathcal{Q}_N^\Gamma$:

$$p_N^\Gamma(x, y) = \sum_{i=0}^{N} \sum_{j=0}^{N} \hat{p}_{i,j} L_i(x) L_j(y) + \sum_{i=0}^{N} \sum_{j=0}^{N} \hat{\alpha}_{i,j} L_i(x) L_j(y) \Phi(x, y) \tag{5.3.37}$$

where L_i, $i = 0, \ldots, N$, are the Legendre polynomials up to degree N, the tensor product of which form a basis for $\mathrm{P}_N(\Omega)$, and $\Phi(x, y)$ is an enrichment function which we leave undefined for the moment. Therefore we wish to consider:

$$b(\boldsymbol{u}_N, q_N^\Gamma) = \int_\Omega (\nabla \cdot \boldsymbol{u}_N) L_k(x) L_l(y) \, \mathrm{d}\Omega + \int_\Omega (\nabla \cdot \boldsymbol{u}_N) L_k(x) L_l(y) \Phi(x, y) \, \mathrm{d}\Omega \tag{5.3.38}$$

where $k, l = 0, \ldots, N$. Proceeding as before, we let $\mathcal{D}_N$ be the range of $\mathcal{V}_N$ by the divergence operator and denote by $\mathcal{Z}_N^\Gamma$ the set of all polynomials of $\mathrm{P}_N(\Omega) \oplus \mathcal{P}(\Omega^\Gamma)$ that are orthogonal to $\mathcal{D}_N$ with respect to the L^2 inner product, where

$$\mathcal{P}(\Omega^\Gamma) = \text{ span } \{L_i(x) L_j(y) \Phi(x, y) \colon i, j = 0, \ldots, N\}$$

The question one may then ask is: Is the set $\mathcal{Z}_N^\Gamma$ non-empty? The answer is clearly YES as $0 \in \mathrm{P}_N(\Omega) \oplus \mathcal{P}(\Omega^\Gamma)$. However, the other elements in $\mathcal{Z}_N^\Gamma$ are, in general, unknown as it is completely dependent on the enrichment function; for example, if we let $\Phi = 1$

then $\mathcal{Z}_N^\Gamma \equiv \mathcal{Z}_N$, whereas if $\Phi(x,y) = H(x,y)$, where $H(x,y)$ is the Heaviside function defined in (5.3.6), then $\mathcal{Z}_N^\Gamma \not\equiv \mathcal{Z}_N$. For the some of examples considered in this thesis, there will be a strong discontinuity in the pressure. Therefore, for the purposes of our analysis, $\Phi(x,y)$ will be defined as: $\Phi(x,y) = H(x,y)$. Substituting this into the form $b(\cdot,\cdot)$ gives:

$$b(\boldsymbol{u}_N, q_N^\Gamma) = \int_\Omega (\nabla \cdot \boldsymbol{u}_N)\, L_k(x) L_l(y)\, \mathrm{d}\Omega + \int_\Omega (\nabla \cdot \boldsymbol{u}_N)\, L_k(x) L_l(y) H(x,y)\, \mathrm{d}\Omega \quad (5.3.39)$$

$$= \int_\Omega (\nabla \cdot \boldsymbol{u}_N)\, L_k(x) L_l(y)\, \mathrm{d}\Omega + \int_{\Omega_2} (\nabla \cdot \boldsymbol{u}_N)\, L_k(x) L_l(y)\, \mathrm{d}\Omega \quad (5.3.40)$$

where $k, l = 0, \ldots, N$.

We can see that if k, l are chosen such that $L_k(x)L_l(y) \in \mathcal{Z}_N$, (where $\mathcal{Z}_N$ is as defined in (5.3.34)), then the first integral is zero. Thus the original set $\mathcal{Z}_N$ is still a set of spurious modes for the XSEM approximation. However, we cannot guarantee that it is the only set and in fact, in general, we can only guarantee $\mathcal{Z}_N \subset \mathcal{Z}_N^\Gamma$. The next simplest step, would be to consider $L_k(x)L_l(y)\Phi(x,y)$ as a possible spurious mode. However, the second integral is not necessarily zero because the integration is over a subset of $\Omega = [-1,1]^2$. Having said that, we note that if the second integral above is small in comparison to the first integral, or very close to zero, then one could infer a spurious mode as the *amount* of enrichment is small. This is quite surprising as it implies that the enrichment could potentially result in a spurious pressure mode. However, this idea matches the work of Groß and Reusken [38], who (in the case of finite elements) found better inf-sup stability if they removed some of the enrichment corresponding to regions of small support. Hence one may, possibly, deduce $\mathcal{Z}_N^\Gamma = \mathcal{Z}_N \oplus \mathcal{Z}_N^X$ where $\mathcal{Z}_N^X$ contains any terms involving the enrichment function. Therefore, it is reasonable for us to assume that using polynomials of degree $N-2$ for the pressure will remove the majority of the spurious modes. However, removing $\mathcal{Z}_N^X$ is not as simple. This is a subject of future research and requires a much more indepth analysis than what is covered here.

The next step, just as before, is to determine the value of the parameter k_N. Its existence is not guaranteed currently, as we cannot say with absolute certainty that $\ker(B_N^*) \subseteq \ker(B^*)$ due to the potential existence of spurious modes caused by the

enrichment term. However, we will proceed never the less. The XSEM approximation of the pressure is defined as in (5.3.4):

$$\begin{aligned} p_N^\Gamma(x,y) &= p_N(x,y) + p_N^X(x,y) \\ &= \sum_{i\in\mathcal{I}} p_i \Psi_i(x,y) + \sum_{j\in\mathcal{I}^\Gamma} \alpha_j \Psi_j(x,y)\Phi_j(x,y) \end{aligned} \tag{5.3.41}$$

where $\mathcal{I} = \{1,\ldots,N_Q\}$, with N_Q being the dimension of $\mathcal{Q}_N$, $\{\Psi_k\}_{k\in\mathcal{I}}$ are the global basis functions spanning $\mathcal{Q}_N$ and $\mathcal{I}^\Gamma \subset \mathcal{I}$. The enrichment function Φ_j, $j \in \mathcal{I}^\Gamma$, once again gives some added difficulty. Being problem dependent, the enrichment function is very difficult to deal with in a general context. In this thesis we are concerned with strong discontinuities and therefore look to the framework of Reusken [60] as a means of removing the dependence on the enrichment function. Using the framework of Reusken [60], the enriched pressure approximation becomes:

$$p_N^\Gamma = \sum_{i\in\mathcal{I}} p_i \Psi_i(x,y) + \sum_{k\in\mathcal{I}_1^\Gamma} \beta_k^{(1)} E_1^0[R_1\Psi_k(x,y)] + \sum_{k\in\mathcal{I}_2^\Gamma} \beta_k^{(2)} E_2^0[R_2\Psi_k(x,y)] \tag{5.3.42}$$

where the sets $\mathcal{I}_i^\Gamma$, $i = 1,2$, are given by:

$$\mathcal{I}_1^\Gamma = \{i \in \mathcal{I} \colon \boldsymbol{x}_i \in \Omega_2 \quad \underline{\text{and}} \quad \text{supp}(\Psi_i(x,y)) \cap \Gamma \neq \emptyset\} \tag{5.3.43}$$

$$\mathcal{I}_2^\Gamma = \{i \in \mathcal{I} \colon \boldsymbol{x}_i \in \Omega_1 \quad \underline{\text{and}} \quad \text{supp}(\Psi_i(x,y)) \cap \Gamma \neq \emptyset\} \tag{5.3.44}$$

with $\mathcal{I} = \{1,\ldots,N_Q\}$. The expression (5.3.42) can be written as: $q_N^\Gamma = q_N + q_{N,1}^X + q_{N,2}^X$ where $q_N \in \mathcal{Q}_N$ is the standard SEM approximation and $q_{N,i}^X \in \mathcal{Q}_{N,i}^X$, $i = 1,2$, are the extended parts, where

$$\mathcal{Q}_{N,i}^X = \text{ span } \left\{E_i^0[R_i\Psi_k(x,y)], k \in \mathcal{I}_i^\Gamma\right\} \qquad i = 1,2 \tag{5.3.45}$$

are defined as the *extended* spaces. Note that the expression (5.3.42) above is slightly different to (5.3.17) due to the restriction operator, R_i, $i = 1,2$, above being defined slightly differently here compared to (5.3.14). Let $R_i \colon L^2(\Omega) \to L^2(\Omega_i)$, $i = 1,2$, be the restriction operator and let $E_i \colon L^2(\Omega_i) \to L^2(\Omega)$ denote an extension operator, which acts as an inverse so that

$$E_i[R_i u] = u \qquad i = 1,2,\ \forall u \in L^2(\Omega) \tag{5.3.46}$$

Finally, denote by $E_i^0 \colon L^2(\Omega_i) \to L^2(\Omega)$, $i = 1, 2$, the zero extension operator such that

$$E_i^0[w] = \begin{cases} w(x,y) & \text{in } \Omega_i \\ 0 & \text{in } \Omega \setminus \Omega_i \end{cases} \qquad i = 1, 2,\ \forall w \in L^2(\Omega_i) \tag{5.3.47}$$

Thus the restriction operator defined in (5.3.14) is equivalent to $E_i^0[R_i u]$ defined above. Clearly, we have

$$\|R_i u\|_{L^2(\Omega_i)} \leq \|u\|_{L^2(\Omega)} = \|E_i[R_i u]\|_{L^2(\Omega)} \tag{5.3.48}$$

Additionally, we have $\|E_i^0[w]\|_{L^2(\Omega)} = \|w\|_{L^2(\Omega_i)}$ because, $\forall w \in L^2(\Omega_i)$

$$\begin{aligned} \left\|E_i^0[w]\right\|_{L^2(\Omega)}^2 &= \int_\Omega \left|E_i^0[w]\right|^2 \, \mathrm{d}\Omega = \int_{\Omega_i} \left|E_i^0[w]\right|^2 \, \mathrm{d}\Omega + \int_{\Omega \setminus \Omega_i} \left|E_i^0[w]\right|^2 \, \mathrm{d}\Omega \\ &= \int_{\Omega_i} |w|^2 \, \mathrm{d}\Omega = \|w\|_{L^2(\Omega_i)}^2 \end{aligned} \tag{5.3.49}$$

In particular, we have $\|E_i^0[R_i u]\|_{L^2(\Omega)} = \|R_i u\|_{L^2(\Omega_i)}$, $i = 1, 2$, $\forall u \in L^2(\Omega)$. We wish to show that there exists a positive constant k_N such that

$$\sup_{\boldsymbol{v}_N \in \mathcal{V}_N} \frac{b(\boldsymbol{v}_N, q_N^\Gamma)}{\|\boldsymbol{v}_N\|_{\mathcal{V}}} \geq k_N \left\|q_N^\Gamma\right\|_{\mathcal{Q}/\ker(B_N^*)} \tag{5.3.50}$$

or equivalently,

$$b(\boldsymbol{v}_N, q_N^\Gamma) \geq k_N \|\boldsymbol{v}_N\|_{\mathcal{V}} \left\|q_N^\Gamma\right\|_{\mathcal{Q}/\ker(B_N^*)} \tag{5.3.51}$$

Note that the L^2-norm is a valid norm on $\mathcal{Q}/\ker(B_N^*)$ and the H^1-norm is a valid norm on $\mathcal{V}$. Consider the form $b(\cdot, \cdot)$:

$$b(\boldsymbol{v}_N, q_N^\Gamma) = b(\boldsymbol{v}_N, q_N) + b(\boldsymbol{v}_N, q_{N,1}^X) + b(\boldsymbol{v}_N, q_{N,2}^X) \tag{5.3.52}$$

$$\geq CN^{-\frac{1}{2}} \|\boldsymbol{v}_N\|_{H^1(\Omega)^2} \|q_N\|_{L^2(\Omega)} + b(\boldsymbol{v}_N, q_{N,1}^X) + b(\boldsymbol{v}_N, q_{N,2}^X) \tag{5.3.53}$$

where we have substituted the discrete inf-sup condition, (5.3.36), derived by Maday *et al.* [49, 7]. The question then is: What can we do about the two extended parts? If we proceed in a manner analagous to Maday *et al.*, then we would require a projection operator onto P_{N-2} with respect to $L^2(\Omega_i)$, or weighted L^2 norms, which could get a bit tricky. Instead, we use the restriction operator and extension operators (5.3.46)

and (5.3.47) defined above. If we let $q_N \in \mathrm{P}_{N-2}(\Omega)$ denote any polynomial of degree less than or equal to $N-2$, then we know, for $i=1,2$, that $E_i[R_i q_N] = q_N \in \mathcal{Q}_N$ and therefore, in particular, $E_i[R_i q_N]$, $i=1,2$, satisfies the discrete inf-sup condition, i.e.:

$$b(\boldsymbol{v}_N, E_i[R_i q_N]) \geq CN^{-\frac{1}{2}} \|\boldsymbol{v}_N\|_{H^1(\Omega)^2} \|E_i[R_i q_N]\|_{L^2(\Omega)} \tag{5.3.54}$$

$$\geq CN^{-\frac{1}{2}} \|\boldsymbol{v}_N\|_{H^1(\Omega)^2} \|R_i q_N\|_{L^2(\Omega_i)} \qquad i=1,2 \tag{5.3.55}$$

where we have used (5.3.48). To finish the proof, we make use of the fact that:

$$b(\boldsymbol{v}_N, E_1^0[R_1 q_N]) = b(\boldsymbol{v}_N, E_2[R_2 q_N]) - b(\boldsymbol{v}_N, E_2^0[R_2 q_N]) \tag{5.3.56}$$

$$b(\boldsymbol{v}_N, E_2^0[R_2 q_N]) = b(\boldsymbol{v}_N, E_1[R_1 q_N]) - b(\boldsymbol{v}_N, E_1^0[R_1 q_N]) \tag{5.3.57}$$

and that $b(\boldsymbol{v}_N, q_{N,i}^X) = b(\boldsymbol{v}_N, E_i^0[R_i q_N])$. We may write:

$$b(\boldsymbol{v}_N, E_1^0[R_1 q_N]) + b(\boldsymbol{v}_N, E_2^0[R_2 q_N]) = \frac{1}{2}\left(b(\boldsymbol{v}_N, E_1[R_1 q_N]) + b(\boldsymbol{v}_N, E_2[R_2 q_N])\right) \tag{5.3.58}$$

$$\geq \frac{1}{2} CN^{-\frac{1}{2}} \|\boldsymbol{v}_N\|_{H^1(\Omega)^2} \left(\|R_1 q_N\|_{L^2(\Omega_1)} + \|R_2 q_N\|_{L^2(\Omega_2)}\right) \tag{5.3.59}$$

Thus we have

$$b(\boldsymbol{v}_N, q_N^\Gamma) = b(\boldsymbol{v}_N, q_N) + b(\boldsymbol{v}_N, q_{N,1}^X) + b(\boldsymbol{v}_N, q_{N,2}^X) \tag{5.3.60}$$

$$\geq CN^{-\frac{1}{2}} \|\boldsymbol{v}_N\|_{H^1(\Omega)^2} \|q_N\|_{L^2(\Omega)} + \frac{1}{2} CN^{-\frac{1}{2}} \|\boldsymbol{v}_N\|_{H^1(\Omega)^2} \left(\|R_1 q_N\|_{L^2(\Omega_1)} + \|R_2 q_N\|_{L^2(\Omega_2)}\right) \tag{5.3.61}$$

$$\geq \frac{1}{2} CN^{-\frac{1}{2}} \|\boldsymbol{v}_N\|_{H^1(\Omega)^2} \left(\|q_N\|_{L^2(\Omega)} + \|R_1 q_N\|_{L^2(\Omega_1)} + \|R_2 q_N\|_{L^2(\Omega_2)}\right) \tag{5.3.62}$$

Since $\|E_i^0[R_i q_N]\|_{L^2(\Omega)} = \|R_i q_N\|_{L^2(\Omega_i)}$, $i=1,2$, due to (5.3.49) we have

$$b(\boldsymbol{v}_N, q_N^\Gamma) \geq \frac{1}{2} CN^{-\frac{1}{2}} \|\boldsymbol{v}_N\|_{H^1(\Omega)^2} \left(\|q_N\|_{L^2(\Omega)} + \left\|E_1^0[R_1 q_N]\right\|_{L^2(\Omega)} + \left\|E_2^0[R_2 q_N]\right\|_{L^2(\Omega)}\right) \tag{5.3.63}$$

Then using Minkowski's inequality (or triangle inequality), given in the Appendix A.2,

we obtain:

$$b(\boldsymbol{v}_N, q_N^\Gamma) \geq \frac{1}{2} C N^{-\frac{1}{2}} \|\boldsymbol{v}_N\|_{H^1(\Omega)^2} \left(\left\| q_N + E_1^0[R_1 q_N] + E_2^0[R_2 q_N] \right\|_{L^2(\Omega)} \right) \tag{5.3.64}$$

$$= \frac{1}{2} C N^{-\frac{1}{2}} \|\boldsymbol{v}_N\|_{H^1(\Omega)^2} \left(\left\| q_N + q_{N,1}^X + q_{N,2}^X \right\|_{L^2(\Omega)} \right) \tag{5.3.65}$$

$$= \frac{1}{2} C N^{-\frac{1}{2}} \|\boldsymbol{v}_N\|_{H^1(\Omega)^2} \left\| q_N^\Gamma \right\|_{L^2(\Omega)} \tag{5.3.66}$$

This result is slightly disappointing (although not surprising since we assumed $q_N \in \mathrm{P}_{N-2}(\Omega)$), because the inf-sup parameter, k_N, can be seen to be dependent on N. In fact, its dependence on N is identical to the inf-sup condition for the SEM approximation (5.3.36). However, we do not believe that the bound given above is the sharpest bound. Perhaps following the approach of Maday *et al.* [49, 7] with weighted L^2 norms would yield a better answer. As far as we are aware, for standard XFEM, the only authors to have considered the inf-sup condition are Groß and Reusken [38] and Reusken [60]. Both articles were concerned with the numerical inf-sup condition and no analysis was given. Therefore the inf-sup condition for XFEM still remains unsolved. For Nitsche's XFEM Burman *et al.* [5], found that a stabilisation term, known as the *ghost penalty* term, was required for the inf-sup condition to be satisfied.

Computationally, the XSEM is implemented by separating the extended part of the approximation in the incompressibility constraint, i.e., numerically we enforce:

$$b(\boldsymbol{u}_N, q_N) = 0 \qquad \text{and} \qquad b(\boldsymbol{u}, q_N^X) = 0 \tag{5.3.67}$$

One could argue that this redefines the problem to be a double-constrained minimisation problem. A doubly-constrained minimisation was considered by Gerritsma and Phillips [34] for the inf-sup condition of the velocity-pressure-stress formulation of Stokes flow. It is possible that a similar approach can be considered for the XSEM.

5.4 Immersed Boundary

Predominantly, throughout the IB literature, the discretisation method of choice was finite differences. In the IFEM of Zhang *et al.* [73] and the FE-IBM of Boffi *et al.* [14, 17], the fluid and the immersed structure are both discretised using finite elements. In this thesis, we have chosen to discretise the fluid using spectral elements as mentioned in the previous section. However, in this thesis, we use a piecewise linear finite element approximation of an immersed elastic fibre (just as was done by Boffi *et al.*) and a linear spectral element approximation of an immersed structure, which we summarise below. Finally, we discuss the volume/area loss problem found in immersed boundary computations.

5.4.1 Spatial Discretisation

Analagous to §5.2, we require the definition of suitable function spaces for the dependent variables. For the immersed boundary terms, there is only a single dependent variable: $\boldsymbol{X}$, which gives the Cartesian position of the immersed structure. The function $\boldsymbol{X}$ is a map from the reference (Lagrangian) configuration of the immersed structure, Ω_r^s, to its current (Eulerian) configuration, $\Omega^s \subseteq \Omega$; i.e. $\boldsymbol{X} : \Omega_r^s \to \Omega$.

For the cases where we consider a one-dimensional elastic curve immersed in a two-dimensional fluid, we employ a linear finite element discretisation. Following the ideas of Boffi *et al.* [14, 17] we assume that $\boldsymbol{X}$ is Lipschitz continuous, which is a stronger condition than regular continuity. Therefore, we define the function space to be:

$$\mathcal{X} = W^{1,\infty}(\Omega_r^s)^d = \left\{ \boldsymbol{Y} \in L^\infty(\Omega_r^s)^d : \nabla_{\boldsymbol{s}} \boldsymbol{Y} \in L^\infty(\Omega_r^s)^{d\times d} \right\} \tag{5.4.1}$$

where it is understood that the Lebesgue space $L^\infty(\Omega_r^s)$, is the space of all functions that are essentially bounded on Ω_r^s. As our domain Ω_r^s is convex, the space $W^{1,\infty}(\Omega_r^s)$ is equivalent to the space of Lipschitz continuous functions on Ω_r^s (Brenner and Scott [19, p. 43]). Just as before, we then consider a finite dimensional subspace $\mathcal{X}_h \subset \mathcal{X}$. Note we have used the subscript h here to emphasize the use of finite elements rather than spectral elements. Let $\Omega_{r,h}^s$ be a subdivision of Ω_r^s into segments $[s_{j-1}, s_j]$, $j =$

$1, \ldots, K_b$, known as the triangulation. Denote by s_j, $j = 1, \ldots, K_v$, the vertices of the triangulation and by $\Omega_k^s = [s_{k-1}, s_k]$, $k = 1, \ldots, K_b$, the elements of the triangulation. We follow the ideas of Boffi *et al.* [14, 17] and let $\mathcal{X}_h$ denote the finite element space of piecewise linear functions on $\Omega_{r,h}^s$ as follows:

$$\mathcal{X}_h = \left\{ \boldsymbol{Y} \in C^0(\Omega_{r,h}^s;\, \Omega)^d \colon\ \boldsymbol{Y}|_{\Omega_k^s} \in P_1(\Omega_k^s)^d,\ k = 1, \ldots, K_b \right\} \tag{5.4.2}$$

where $P_1(\Omega_k^s)$ denotes the set of affine polynomials of degree ≤ 1 on Ω_k^s, $k = 1, \ldots, K_b$. The finite element approximation, $\boldsymbol{X}_h$, of a function $\boldsymbol{X}$ at time $t^n = n\Delta t$ is then given by:

$$\boldsymbol{X}_h^n = \sum_{j=1}^{K_v} \boldsymbol{X}_j^n N_j(x, y) \tag{5.4.3}$$

where N_j are the usual finite element shape functions: i.e. hat functions. Note that $\operatorname{span}\{N_j \colon j = 1, \ldots, K_v\}$ form a basis of $\mathcal{X}_h$ and that the Kronecker delta property is satisfied: $N_j(x_i, y_i) = \delta_{ij}$.

For the cases where we consider a two-dimensional elastic structure immersed in a two-dimensional fluid, we use linear spectral elements. Define the function space $\mathcal{X}$ as in (5.4.1). Following §5.2.2, it is necessary to choose a conforming discrete subspace of $\mathcal{X}$. Let $\mathcal{X}_N \subset \mathcal{X}$ denote such a subspace. Note we have used the subscript N to denote a spectral element discretisation rather than finite element. The domain Ω_r^s is divided into K_b uniform, non-overlapping, quadrilateral spectral elements, Ω_e^s, $e = 1, \ldots, K_b$, such that

$$\bar{\Omega}_r^s = \bigcup_{e=1}^{K_b} \bar{\Omega}_e^s$$

Let $P_1(\Omega_e^s)$ denote the space of all linear polynomials on Ω_e^s and let $\mathrm{P}_1(\Omega_r^s)$ be defined as in (5.2.7) taking $N = 1$. Following §5.2.2, the approximation space for the IB position is defined as:

$$\mathcal{X}_N := \mathcal{X} \cap [\mathrm{P}_1(\Omega_r^s)]^d \tag{5.4.4}$$

Each spectral element is mapped to the parent domain $D = [-1, 1] \times [-1, 1]$, using the transfinite map of Schneidesch and Deville [62] which was introduced in §5.2.2. The

approximation of the IB position at time $t^n = n\Delta t$ on the element Ω_e^s is then given by:

$$\boldsymbol{X}_N^{e,n}(\xi,\eta) = \sum_{i=0}^{N}\sum_{j=0}^{N} \boldsymbol{X}_{i,j}^{e,n} h_i(\xi) h_j(\eta) \tag{5.4.5}$$

where $h_i(\xi)$, $i = 0, \ldots, N$, are the Lagrange interpolants defined on the parent interval $\xi \in [-1, 1]$ by (5.2.15).

For the case of a one-dimensional elastic structure immersed in a two-dimensional fluid, the integration in the IB source term does not require the use of quadrature. However, for two-dimensional immersed structures, quadrature rules are required. Therefore, the reason for choosing spectral elements, rather than linear finite elements, for the approximation of the two-dimensional immersed structure is that the integration involved in the calculation of the IB source term, can be done simply and accurately using Gauss-Lobatto-Legendre quadrature.

5.4.2 Area Conservation

The immersed boundary method is known to suffer from volume or area loss problems [59, 17, 71, 12, 52]. Peskin [56, 57, 58] noted that if a compatibility condition between the Eulerian and Lagrangian mesh widths was not satisifed then a large amount of numerical diffusion takes place across the interface between the fluid and the structure. The compatibility condition was given by: $h^s \leq h_f$, where h^s is the Lagrangian mesh width and h_f the Eulerian mesh width. Boffi *et al.* [15] found that the optimal compatibility condition for FE-IBM was $h^s \leq h_f/2$. However, even if this condition is satisifed, numerical diffusion can still be seen to take place across the interface. In the literature, it is believed that this is primarily due to the velocity field not being divergence-free and thus the incompressibility condition *not* being accurately satisifed. Newren *et al.* [52], suggested that the cause of the volume/area loss was because the interpolated velocity field, although discretely divergence-free, is not continuously divergence-free.

In the immersed boundary method, the location of the interface is automatically tracked by:

$$\frac{\partial \boldsymbol{X}}{\partial t} = \boldsymbol{u}(\boldsymbol{X}(\boldsymbol{s}, t), t) \tag{5.4.6}$$

which enforces the no-slip condition at every point $\boldsymbol{X}$ by assuming the interface moves with the local fluid velocity. Clearly, if the velocity field, on the approximation to the immersed boundary, is zero then there will be no movement. However, in order to accomplish this one would require the no-slip boundary condition, $\boldsymbol{u}|_{\Gamma} = 0$, being enforced *strongly* on the immersed boundary. Due to the immersed boundary being unfitted, this is a non-trivial task. The article by Lew and Buscaglia [43] introduced a discontinuous Galerkin approach which allowed boundary conditions, on an unfitted interface, to be enforced *strongly*. We believe that this approach would alleviate the problems with volume/area conservation. However, we have not yet implemented such a method and this is therefore left for future work. Also, whether we wish to enforce Dirichlet boundary conditions on the interface is problem dependent.

For the standard approach used in IB computations, the movement of the IB is decoupled from the momentum equation and incompressibility constraint. Thus, the velocity field *should* be discretely divergence-free if the incompressibility constraint is solved implicitly. However, the interpolated velocity is unlikely to be *exactly* discretely divergence-free due to the immersed boundary being unfitted. Additionally, the velocity field is most likely going to be non-zero local to the interface, due to the non-zero force which is *spread* to the surrounding fluid and the no-slip boundary conditions not being enforced strongly. Indeed, in the simple example considered below - and by many in the literature [52, 67, 17] - the action of the forcing term is to pull the elastic curve inwards. Therefore, without an additional constraint on the velocity on the interface, the immersed boundary is freely allowed to move inwards. Peskin [59] considered a modified delta function, and difference operators, to improve the volume/area conservation but at an additional computational cost to the *spreading* and *interpolation* phases. It was shown by Owens *et al.* [53] that increasing the number of points inside the support of the delta function, gave improved volume/area conservation.

In this section, we investigate the volume/area loss of the SE-IBM considered in this thesis. We consider the standard model problem, given in §7.2, of a circular static membrane immersed in a Newtonian fluid. Let $\Omega = [0,1] \times [0,1]$ denote our domain of interest such that $\bar{\Omega} = \bar{\Omega}(t) = \bar{\Omega}^f(t) \cup \bar{\Omega}^s(t)$ where $\Omega^f(t) \subset \Omega$ denotes the fluid domain and $\Omega^s(t) \subset \Omega$ denotes the immersed elastic membrane satisfying $\Omega^f(t) \cap \Omega^s(t) = \emptyset$, $\forall t \in [0,T]$. Additionally we define the interface between the fluid and immersed membrane as $\Gamma = \partial\Omega^f \cap \partial\Omega^s$. In this example, as we are dealing with a one-dimensional structure immersed in a two-dimensional fluid, we denote $\Omega^s \equiv \Gamma$. An example of the geometry is given in Fig. 4.3.

In this geometry, the Cartesian position of the membrane, $\boldsymbol{X}$ is parameterised by arclength, $s \in [0, 2\pi R]$ where $R = 0.25$ is the radius of the immersed circular membrane. For the purposes of this section, we use an explicit Euler method to approximate the IB source term, i.e. $\boldsymbol{f}^{n+1} \approx \boldsymbol{f}^n$ and a semi-implicit Euler method to approximate the movement of the immersed boundary:

$$\frac{\boldsymbol{X}^{n+1} - \boldsymbol{X}^n}{\Delta t} = \boldsymbol{u}^{n+1}(\boldsymbol{X}^n) \tag{5.4.7}$$

This is the standard approach used in the immersed boundary literature. For reference, the density of fluid and the structure was taken to be the same, $\rho = 1$, the viscosity of the fluid $\eta_s = 1$ and the coefficient of elasticity for the structure was taken to be $\kappa = 1$. As we are studying the effect of the spatial discretisation on the area conservation, we have taken the timestep to be sufficiently small so as to reduce the amount of temporal error, $\Delta t = 10^{-4}$. We note for reference that we performed 1000 timesteps and the analytical value of the area contained inside the membrane is given by 0.19635 (to 5 decimal places).

In the spatial discretisation of the immersed boundary position $\boldsymbol{X}$, the domain for the arclength parameter is divided into K_b segments, $[s_{i-1}, s_i]$, $i = 1, \dots K_b$, each of constant mesh width $h^s = 2\pi R/K_b$. There are two ways in which we can define the number of Lagrangian segments, K_b. The first is the standard way which is done in

the FE-IBM, we take:

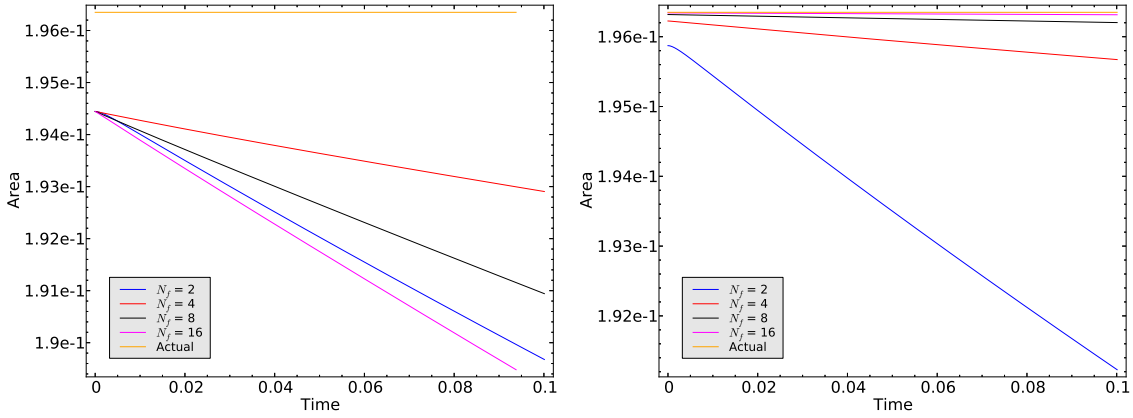
$$h^s = \frac{2\pi R}{K_b} \leq \frac{h_f}{2} = \frac{1}{2K_f}$$
$$\Rightarrow \quad K_b \geq \frac{4\pi R}{h_f} = 4\pi R K_f \tag{5.4.8}$$

where $h_f = 1/K_f$ is the constant fluid mesh width and K_f is the number of fluid elements in the x-direction. (We are dealing with a uniform discretisation of a square domain. Therefore the number of elements is the same in both x and y directions.). The above condition, shows that the number of points on the immersed boundary is defined as a function of mesh width h_f. As we are dealing with a spectral method, we can also increase the order of our polynomial. Therefore, we can define the number of immersed boundary points as a function of N. Indeed:

$$h^s = \frac{2\pi R}{K_b} \leq \frac{h_f}{2} = \frac{1}{2NK_f}$$
$$\Rightarrow \quad K_b \geq \frac{4\pi R}{h_f} = 4\pi R N K_f \tag{5.4.9}$$

where we define $h_f = 1/NK_f$ to be the fluid mesh width. (This is equivalent to assuming we have NK_f uniformly spaced elements in the x-direction). As K_b is required to be an integer, we take the nearest *even* number which satisfies the condition above. It is obvious that (5.4.9) also satisifies $h^s \leq h_f/2$ as all the expression does is increase the number of IB points. However, making the number of IB points dependent on N makes a lot more sense.

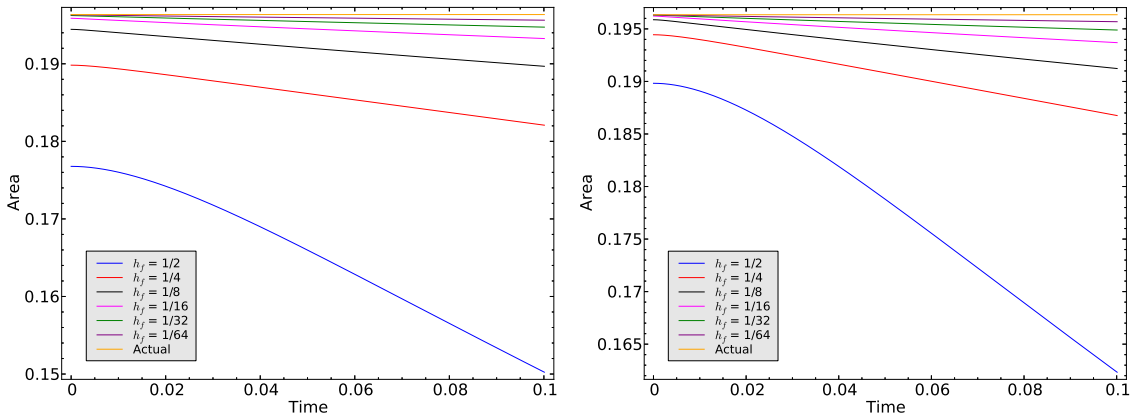
Fig. 5.7 shows a comparison of area loss between $K_b = K_b(h_f)$ and $K_b = K_b(N)$ where we fix $h_f = 1/8$ so that we have an 8 by 8 grid of elements. In Fig. 5.7a, it can be seen that for $K_b = K_b(h_f)$, increasing N actually increases the area loss inside the circular membrane. This may seem to be an error, however, increasing the degree of the polynomial N, improves the resolution of the fluid grid. Therefore, it makes sense for the area loss to increase as we haven't correspondingly improved the resolution of the Lagrangian grid. Fig. 5.7b illustrates that we do see better area conservation as we increase N provided $K_b = K_b(N)$.



(a) $K_b = K_b(h_f)$ **(b)** $K_b = K_b(N)$

Figure 5.7: Comparison of the area conservation for different values of K_b, for increasing N when $h_f = 1/8$.

Fig. 5.8 show a comparison of area loss between $K_b = K_b(h_f)$ and $K_b = K_b(N_f)$ where we fix $N = 2$ and consider h-type convergence. As expected, as the mesh width de-



(a) $K_b = K_b(h_f)$ **(b)** $K_b = K_b(N)$

Figure 5.8: Comparison of the area conservation for different values of K_b, for decreasing h_f when $N = 2$.

creases better area conservation is seen and is due, primarily, to the better resolution of the interface. We note that using $K_b = K_b(N)$ does improve the area conservation but not drastically as it does for p-type convergence.

Comparing Fig. 5.8 against Fig. 5.7 suggests that the area conservation for a fixed value of $h_f = 1/8$ and increasing N, in other words p-type convergence, has a better order of convergence than the h-type convergence. It is clear that the larger values of

N give better area conservation. The orders of convergence with respect to mesh width at the end of the simulation are given in Table 5.1 when $K_b = K_b(h_f)$ and $K_b = K_b(N)$. The order of convergence of the area with respect to mesh width is approximately 1

$K_b = K_b(h_f)$		$K_b = K_b(N)$	
h_f	Order	h_f	Order
1/2	-	1/2	-
1/4	1.69273	1/4	1.82584
1/8	1.09619	1/8	0.90613
1/16	1.10877	1/16	0.94604
1/32	0.93103	1/32	0.86730
1/64	1.15752	1/64	1.12957

Table 5.1: Order of convergence of the area with respect to h_f for both $K_b(h_f)$ and $K_b(N)$.

for both $K_b(h_f)$ and $K_b(N)$. This is good but we expect better order of convergence when we increase the polynomial degree.

Table 5.2 details the order of convergence when $h_f = 1/2$ for increasing N for both $K_b = K_b(h_f)$ and $K_b(N)$. The order of convergence with respect to N is clearly not

$K_b = K_b(h_f)$		$K_b = K_b(N)$	
N	Order	N	Order
2	-	2	-
4	0.77820	4	2.26199
8	-0.14894	8	2.24401
16	-0.18563	16	2.11087
32	-0.20212	32	2.06047

Table 5.2: Order of convergence with respect to N for both $K_b = K_b(h_f)$ and $K_b(N)$.

very good when $K_b = K_b(h_f)$, however it is approximately 2 when $K_b = K_b(N)$ which

is an improvement on the h-type convergence considered in Table 5.1.

Table 5.3 details the order of convergence for the area when $h_f = 1/8$ for increasing N for both $K_b = K_b(h_f)$ and $K_b(N)$. Note that we were unable to deal with $N = 16$ because the simulation terminated early due to limits in computational time. Clearly

$K_b = K_b(h_f)$		$K_b = K_b(N)$	
N	Order	N	Order
2	-	2	-
4	0.95412	4	2.91851
8	-0.65082	8	2.20305

Table 5.3: Order of convergence with respect to N for both $K_b = K_b(h_f)$ and $K_b(N)$.

we see the same problems with the order when $K_b = K_b(h_f)$ and the same order of 2 when $K_b = K_b(N)$ as we saw in Table 5.2. The question is: Why do we see better order of convergence with respect to polynomial degree than mesh width?

It was shown by Owens *et al.* [53] that the area loss can be improved by increasing the number of points inside the support of the delta function. In the FE-IBM, and therefore SE-IBM, the support of the delta function is just an element of the Eulerian mesh which contains the immersed structure. Therefore, increasing the value of N, increases the number of points inside an element of the mesh and hence, the support of the delta function. So not only do we see better area conservation due to the improved resolution of the interface (K_b is dependent on N) but also additional area conservation due to the increased number of points inside the support of the delta function. On top of this, we believe that increasing the polynomial degree improves the incompressibility constraint better than refining the mesh; i.e. $\boldsymbol{u}_N(\boldsymbol{X}_h)$ becomes more discretely divergence-free as N is increased.

Table 5.4 details the value of the area at the end of the simulation for $K_b = K_b(N)$ when $h_f = 1/64$ with $N = 2$, $h_f = 1/8$ with $N = 8$ and $h_f = 1/2$ with $N = 32$.

From Table 5.4, it is clear that there is a better final value for the area for $h_f = 1/2$

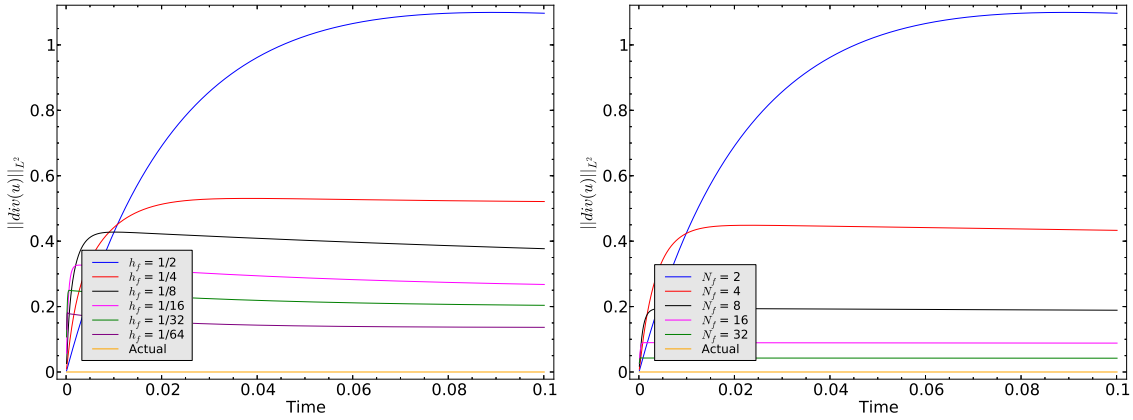
Parameters	Final Area	Computational Time
$h_f = 1/64$, $N = 2$	0.19568	42 hours
$h_f = 1/8$, $N = 8$	0.19620	3 hours
$h_f = 1/2$, $N = 32$	0.19627	87 hours

Table 5.4: Value of the area at the end of the simulation for different parameter regimes.

with $N = 32$ than the others and that $h_f = 1/8$ with $N = 8$ has a better value than $h_f = 1/64$ with $N = 2$. However, the improvement in the value of the area is not all that great. It is clear from Table 5.4, that the computational time required when $h_f = 1/64$ and $N = 2$ was a lot larger than when $h_f = 1/8$ and $N = 8$. So not only does $h_f = 1/8$ with $N = 8$ give better area conservation than when $h_f = 1/64$ but it also takes a fraction of the computational time. However, the largest computational time occured when $h_f = 1/2$ and $N = 32$. Therefore, using a very coarse mesh with a very high order polynomial, although gives good area conservation, takes far too long to complete. This suggests that using a relatively fine mesh with a medium polynomial degree is the best option. This lands in the grey area between finite elements and spectral elements. On a very fine mesh with a low order polynomial, finite elements are considered to be the best option; on a coarse mesh with a high order polynomial, spectral elements are considered to be the best option. For the grey area that lies between the two, a possible option is the Spectral/hp method of Karniadakis and Sherwin [40].

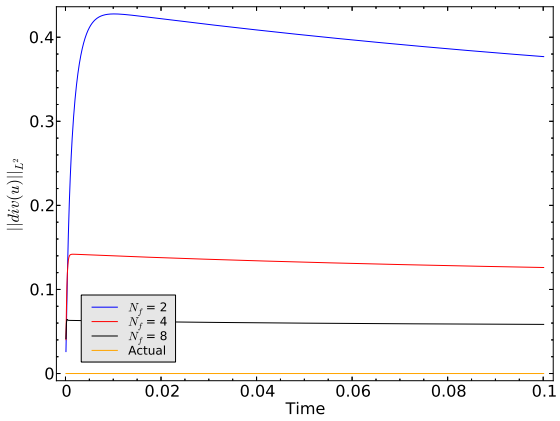
As the divergence-free constraint plays an important role in the area conservation, Fig. 5.9 illustrates the convergence of the L^2 norm of the divergence of the velocity field with respect to h_f and N when $K_b = K_b(N)$. Clearly, smaller values for the divergence of the velocity can be obtained when higher polynomial degrees are used. Thus reinforcing that the better area conservation can be seen with SE-IBM compared to FE-IBM.

Table 5.5 details the order of convergence for the L^2 norm of the divergence of the



(a) h-type convergence for $N = 2$.

(b) p-type convergence for $h_f = 1/2$.



(c) p-type convergence for $h_f = 1/8$.

Figure 5.9: Comparison of the $\|\nabla \cdot \boldsymbol{u}_N\|_{L^2(\Omega)}$.

velocity for both h and p types of convergence. Fig. 5.9 clearly suggests that the order of convergence with respect to N should be higher than convergence with respect to h_f. Indeed, this is confirmed by Table 5.5. The order of convergence with respect to mesh width is approximately 0.5, whereas it is approximately 1 with respect to N for both $h_f = 1/2$ and $1/8$.

We mentioned at the beginning of this section, that the immersed boundary force acts to pull the immersed boundary inwards and, because there is no additional constraint on the velocity on the interface, the immersed boundary moves freely with the non-zero local velocity. We wish to investigate what happens to the velocity field when we increase the degree of our polynomial and also compare this against the usual finite element approach of decreasing the mesh width.

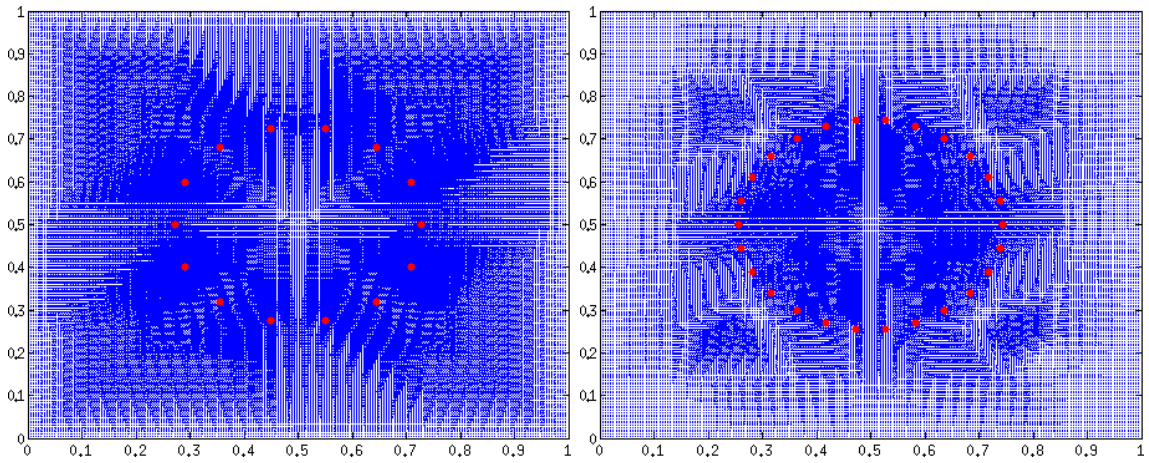
$\|\nabla \cdot \boldsymbol{u}_N\|_{L^2(\Omega)}$					
$N = 2$		$h_f = 1/2$		$h_f = 1/8$	
h_f	Order	N	Order	N	Order
1/2	-	2	-	2	-
1/4	1.07363	4	1.34078	4	1.57980
1/8	0.46723	8	1.19718	8	1.10968
1/16	0.49433	16	1.09679		
1/32	0.39247	32	1.06528		
1/64	0.57819				

Table 5.5: Comparison of the orders of convergence with respect to h_f and N.

Fig. 5.10 illustrates the velocity field for $N = 2$ and $h_f = 1/2, \ldots 1/64$ when $K_b = K_b(N)$. What we expect is that the velocity field will become more local as the mesh is refined. Fig. 5.10 shows that as the mesh width decreases, the velocity field does indeed become more local. This is physically realistic as analytically, there shouldn't be any spurious velocities except in the region where the structure resides. We notice that the general *pattern* does not change (this is most clearly seen between in Figs. 5.10d, 5.10e and 5.10f). In other words, the velocity is not necessarily becoming more *resolved* it is merely becoming more local or more *refined*.

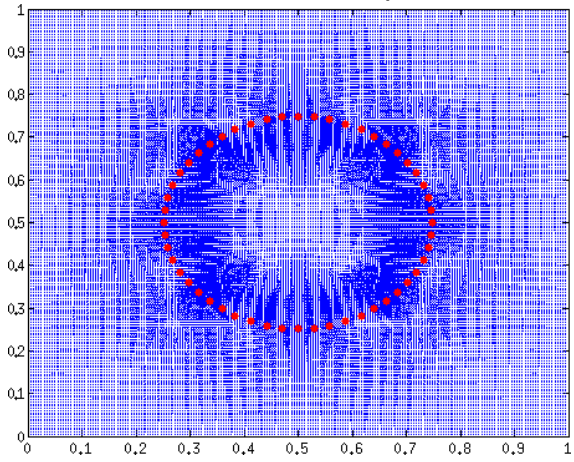
Correspondingly, Fig. 5.11 illustrates the velocity field for $h_f = 1/2$ with $N = 2, \ldots 32$ when $K_b = K_b(N)$. Clearly as the value of N is increased, the velocity field is not becoming more local, instead the magnitude is decreasing and a distinct *pattern* begins to emerge (this is clearly seen between Figs. 5.11c, 5.11d and 5.11e). This is the difference between refinement and resolution. As N is increased, the velocity field is becoming more *resolved* whilst when the mesh width is decreased the velocity field becomes more local and hence more *refined*.

Fig. 5.12 illustrates the velocity field for $h_f = 1/8$ with $N = 2, 4, 8$ and $K_b = K_b(N)$.

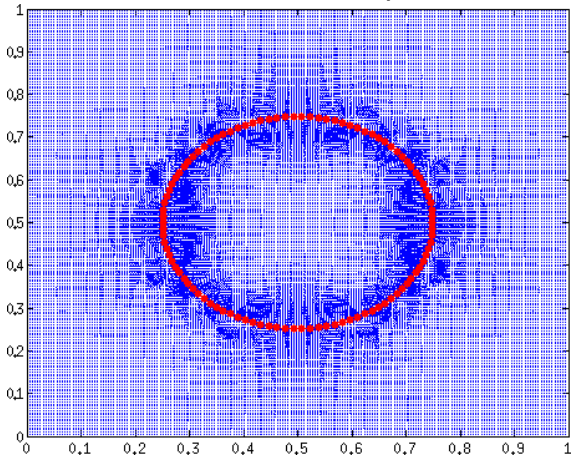


(a) Velocity vector for $h_f = 1/2$.

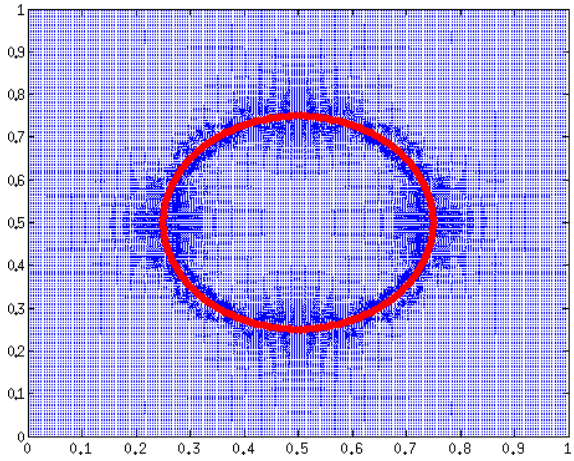
(b) Velocity vector for $h_f = 1/4$.



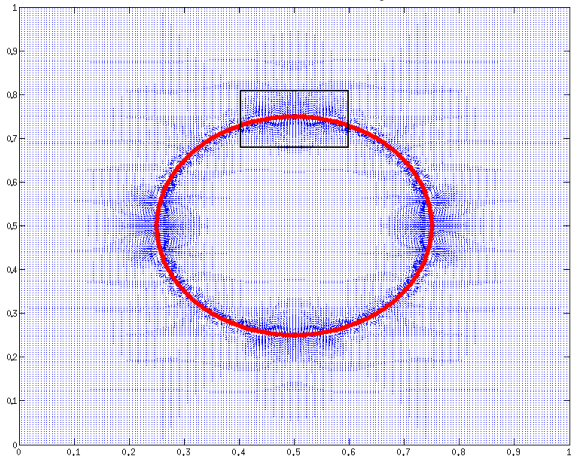
(c) Velocity vector for $h_f = 1/8$.



(d) Velocity vector for $h_f = 1/16$.



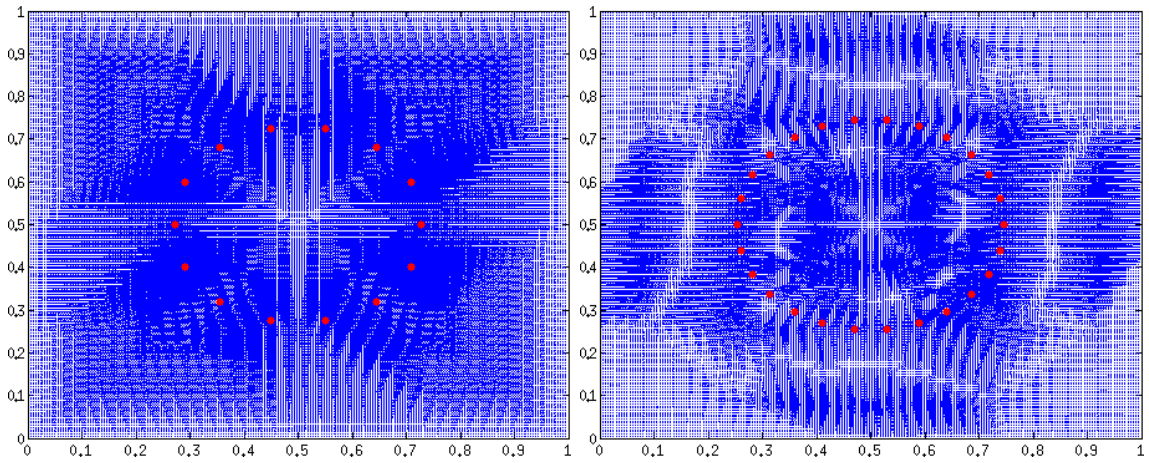
(e) Velocity vector for $h_f = 1/32$.



(f) Velocity vector for $h_f = 1/64$.

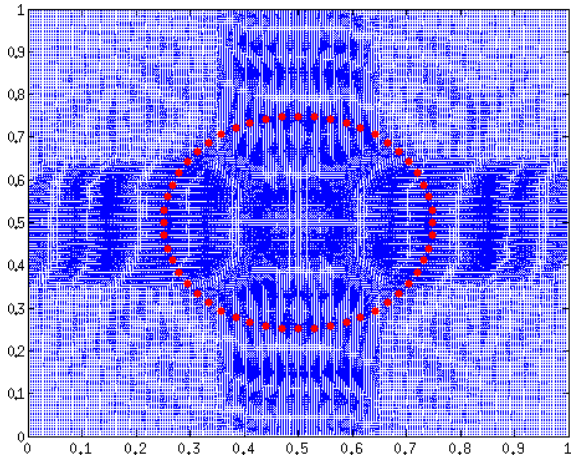
Figure 5.10: Illustration of the velocity vector for decreasing mesh width when $N = 2$ and $K_b = K_b(N)$.

Once again as N is increased, a distinct *pattern* is formed in the velocity field - a thick blue square. Comparing Figs. 5.10f, 5.11e and 5.12c shows three distinct patterns for

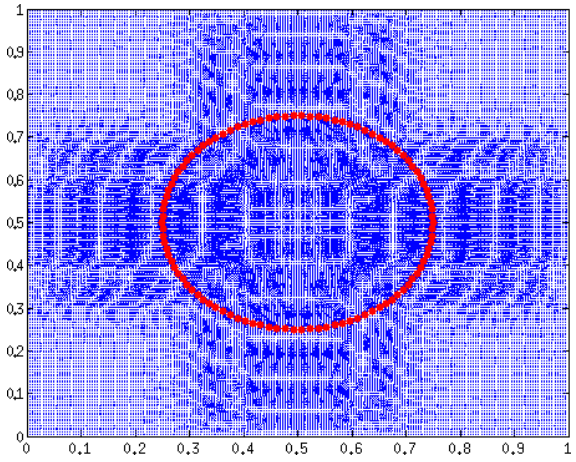


(a) Velocity vector for $N = 2$.

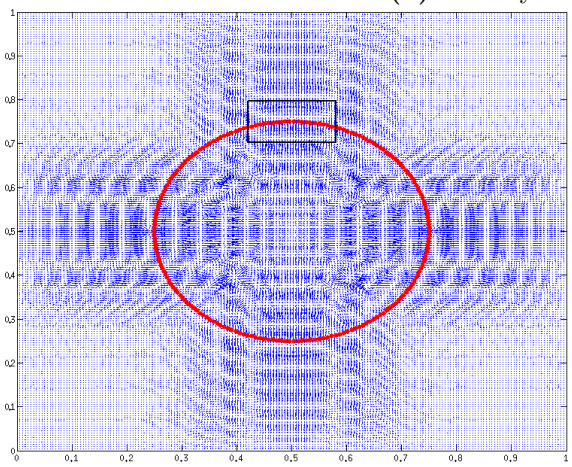
(b) Velocity vector for $N = 4$.



(c) Velocity vector for $N = 8$.



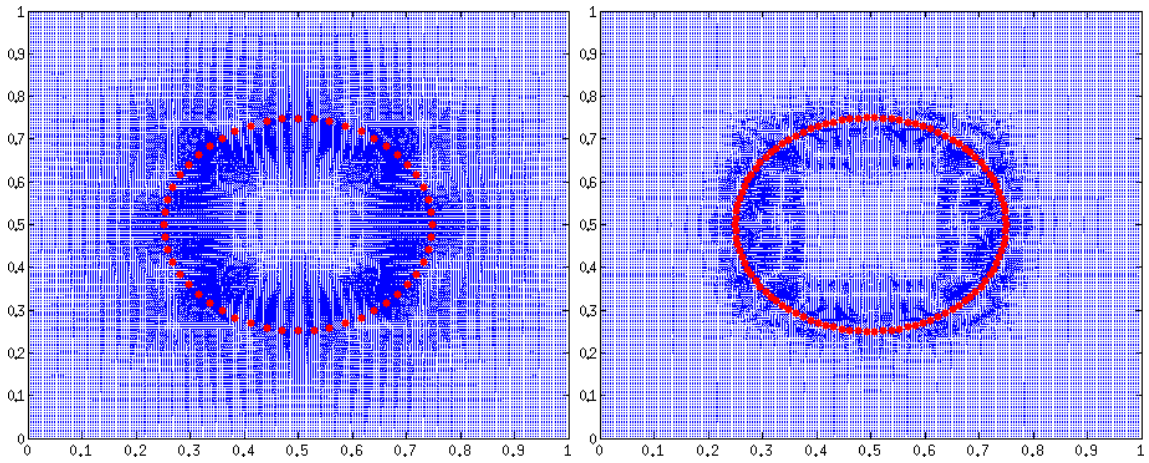
(d) Velocity vector for $N = 16$.



(e) Velocity vector for $N = 32$.

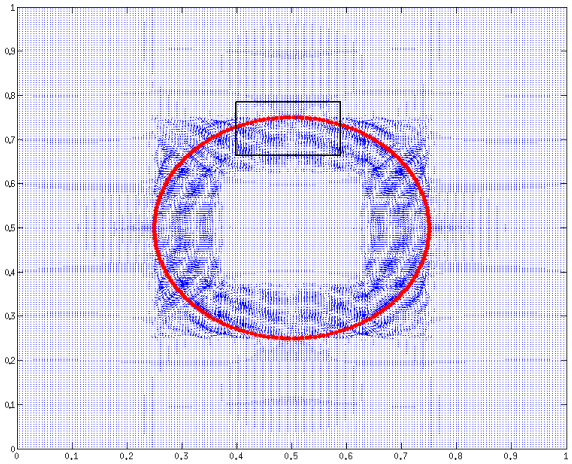
Figure 5.11: Illustration of the velocity vector for $h_f = 1/2$ and increasing N.

the velocity field. In order to look at these patterns more closely we zoom in to the region contained by the black square in those three plots.



(a) Velocity vector for $N = 2$.

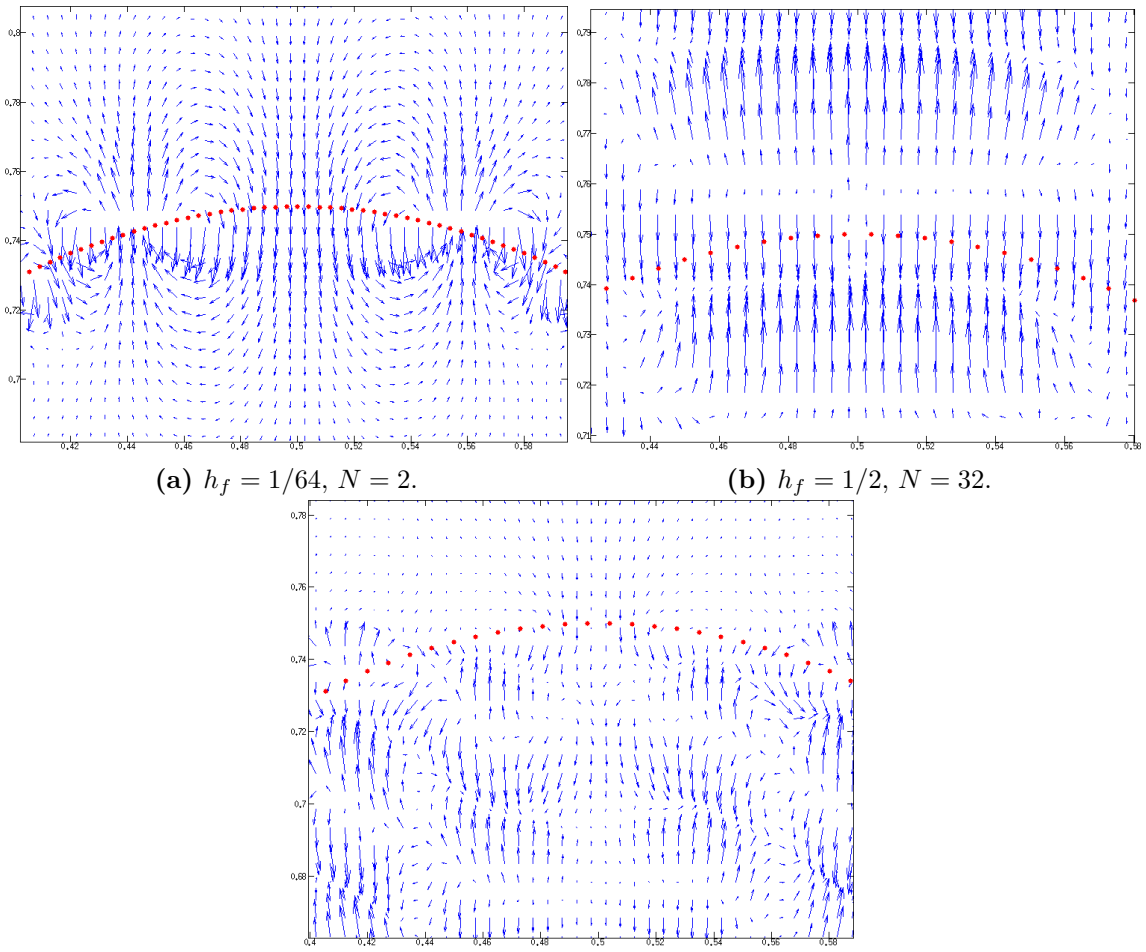
(b) Velocity vector for $N = 4$.



(c) Velocity vector for $N = 8$.

Figure 5.12: Illustration of the velocity vector for $h_f = 1/8$ and increasing N.

Fig. 5.13 illustrates the velocity field zoomed into the region contained inside the black squares illustrated in the plots 5.10f, 5.11e and 5.12c. The velocity vector is a lot more local for the smaller mesh width. In Fig. 5.13a the velocity can be seen to be pushing through the immersed boundary towards the centre of the circle. Whilst there is no *leakage* here (the velocity field acts to push the immersed boundary towards the centre of the circle) the majority of the motion in the direction of the axes is directed towards the centre. There are small recirculatory regions on the diagonal. Figs. 5.13b and 5.13c have larger velocity vectors inside the immersed boundary in the zoomed plots. However, this is actually a good thing in this context. We notice, that although there is still motion towards the centre of the circle, there are also velocity vectors pushing



(a) $h_f = 1/64$, $N = 2$. **(b)** $h_f = 1/2$, $N = 32$.

(c) $h_f = 1/8$, $N = 8$.

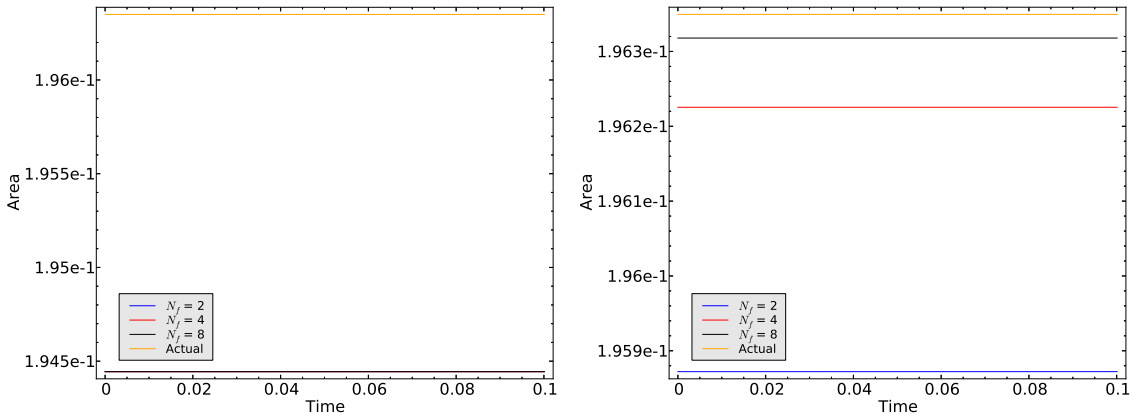
Figure 5.13: Zoomed plots of the velocity vector.

outwards from the centre of the circle. From the plots, it appears that these velocity vectors are of equal magnitude to the ones pushing inwards. This is a consequence of the incompressibility constraint being better satisfied when a high order polynomial is used, as illustrated in Fig. 5.9.

Physically, the immersed membrane wants to push inwards, however due to the fluid being incompressible, the immersed membrane cannot collapse towards the centre and thus a pressure is generated inside the membrane. Therefore, numerically we require something pushing outwards from the centre to prevent the collapse of the immersed membrane. This is precisely what happens when SE-IBM is used. Whilst it is physically realistic to have a local force (because the force distribution for the membrane should only be non-zero in the region where the immersed membrane resides), the

incompressibility constraint is not as accurately enforced. This is another argument to suggest that the use of the Spectral/hp method of Karniadakis and Sherwin [40] would be a good option. Peskin [59] suggested modifying the stencil used in the finite difference computations, as well as the discrete delta function, in order to improve the area (2D)/volume (3D) loss problems. Whilst this succeeded, the method increased the amount of computational time spent in the spreading and interpolation phases. All of the results given above were for when a Semi-Implicit Euler (SIME) method (4.2.1) was used to move the membrane. Here, we consider a Velocity Correction Scheme (VCS) so as to preserve the area inside the membrane up to the order of $O((\Delta t)^2)$. The derivation of the scheme is given in the appendix C.1.

Following the same procedure as above, Fig. 5.14 shows a comparison of area loss between $K_b = K_b(h_f)$ and $K_b = K_b(N)$ where we fix $h_f = 1/8$ so that we have an 8 by 8 grid of elements. In both of the plots in Fig. 5.14, the area remains almost constant

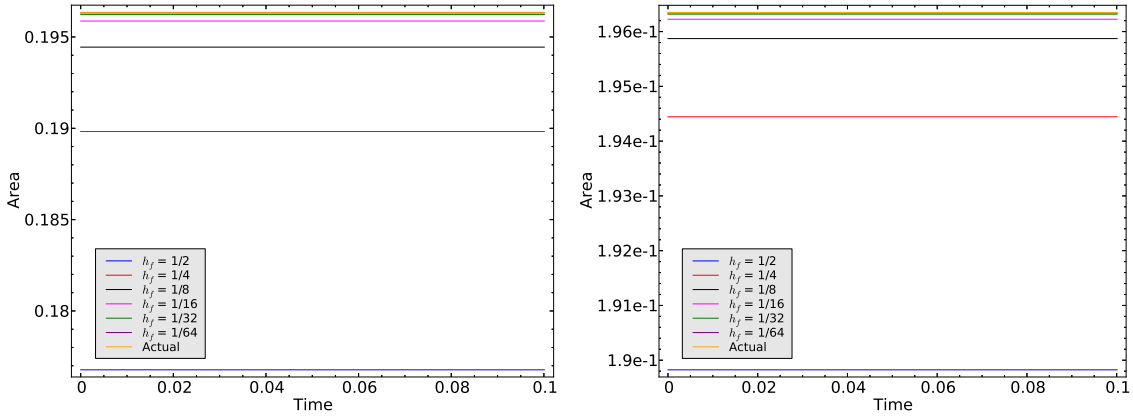


(a) $K_b = K_b(h_f)$ **(b)** $K_b = K_b(N)$

Figure 5.14: Comparison of the area conservation for different values of K_b, for increasing N when $h_f = 1/8$.

throughout the entire of the simulation and that when $K_b = K_b(h_f)$ is used, the initial value of the area contained inside the membrane is lower than when $K_b = K_b(N)$ due to the fewer number of points which make up the immersed membrane.

Fig. 5.15 shows a comparison between $K_b = K_b(h_f)$ and $K_b = K_b(N)$ where $N = 2$ and we consider h-type convergence. Clearly the area conservation is better when $K_b = K_b(N)$ than $K_b(h_f)$. It is hard to tell from the plots whether the order of con-



(a) $K_b = K_b(h_f)$ **(b)** $K_b = K_b(N)$

Figure 5.15: Comparison of the area conservation for different values of K_b when $N = 2$ for decreasing mesh width.

vergence has changed and whether SE-IBM gives better orders than one would expect from FE-IBM.

Table 5.6 compares the order of convergence for the area conservation for decreasing mesh width when $N = 2$ and $K_b = K_b(h_f)$ and $K_b = K_b(N)$. We can infer an order

$K_b = K_b(h_f)$		$K_b = K_b(N)$	
h_f	Order	h_f	Order
1/2	-	1/2	-
1/4	1.58471	1/4	1.77582
1/8	1.77587	1/8	1.99697
1/16	1.99683	1/16	1.94311
1/32	1.94318	1/32	1.97132
1/64	1.97137	1/64	1.99998

Table 5.6: Orders of convergence of the area with respect to h_f for increasing N, for both $K_b(h_f)$ and $K_b(N)$.

of 2 for both $K_b = K_b(h_f)$ and $K_b(N)$ which is a considerable improvement on the order given in Table 5.1. Table. 5.7 details the order of convergence for the area with respect to N when $h_f = 1/2$ and $K_b = K_b(h_f)$ and $K_b(N)$. The reason the zero order

$K_b = K_b(h_f)$		$K_b = K_b(N)$	
N	Order	N	Order
2	-	2	-
4	-1.33432×10^{-6}	4	1.77584
8	9.16950×10^{-7}	8	1.99685
16	2.16173×10^{-7}	16	1.94319
32	5.58729×10^{-7}	32	1.97137

Table 5.7: Orders of convergence for the area with respect to N for both $K_b = K_b(h_f)$ and $K_b(N)$ when $h_f = 1/2$.

of convergence when $K_b = K_b(h_f)$ is because the number of points on the immersed membrane does not change as N is increased, therefore the area is constant between each value of N. We can infer an order of 2 when $K_b = K_b(N)$ just as before in Table 5.2. Table 5.8 details the order of convergence for the area with respect to N when $h_f = 1/8$ and $K_b = K_b(h_f)$ and $K_b(N)$ We have the same issue when $K_b = K_b(h_f)$ as

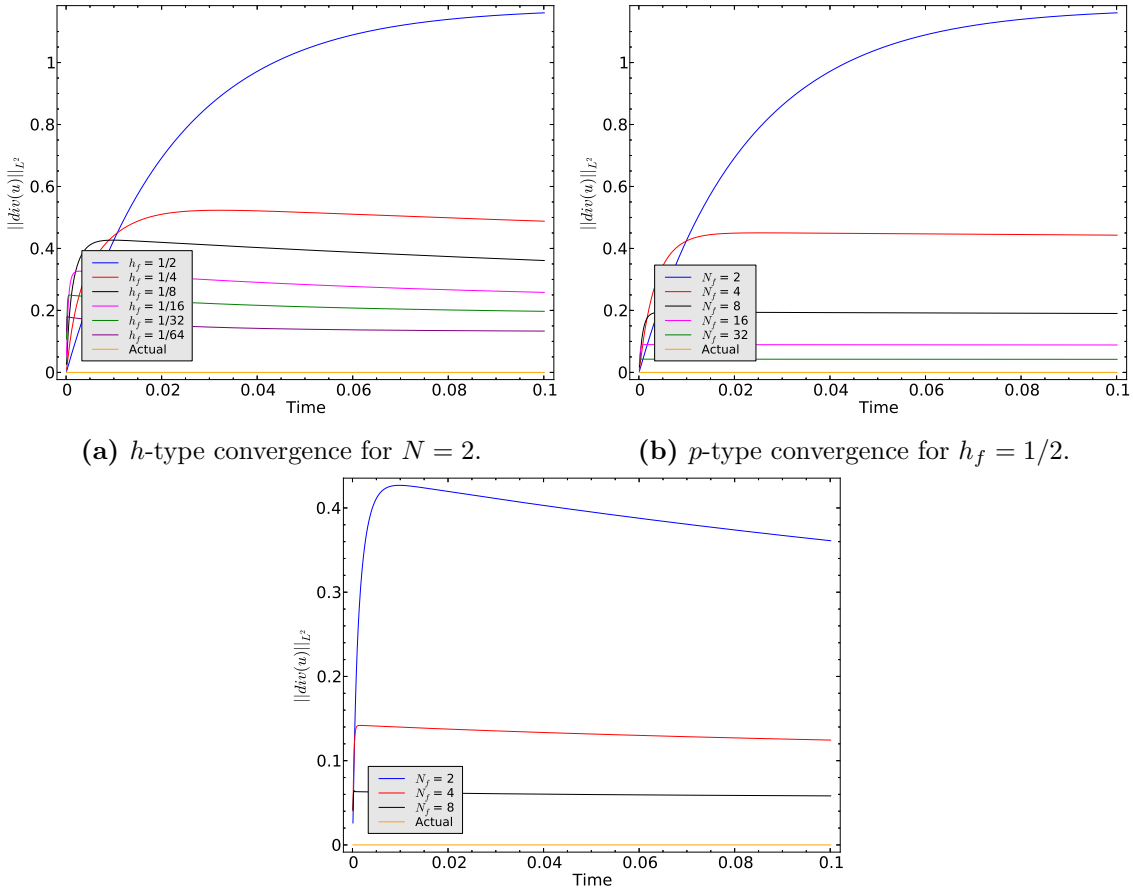
$K_b = K_b(h_f)$		$K_b = K_b(N)$	
N	Order	N	Order
2	-	2	-
4	-1.55824×10^{-5}	4	1.94310
8	5.81535×10^{-7}	8	1.97136

Table 5.8: Orders of convergence of the area with respect to N for both $K_b = K_b(h_f)$ and $K_b(N)$ when $h_f = 1/8$.

we saw in Table 5.7 and we can again infer the same order of convergence of 2 that we inferred in Tables 5.7 and 5.6.

As we have guaranteed that the area is conserved up to the order $O((\Delta t)^2)$, we are curious to see how that affects the divergence-free constraint. Fig. 5.16 illustrates the L^2 norm of the divergence of the velocity in time for h-type convergence and p-type

convergence for both $h_f = 1/2$ and $h_f = 1/8$. The first thing we notice is that there is



(a) h-type convergence for $N = 2$.

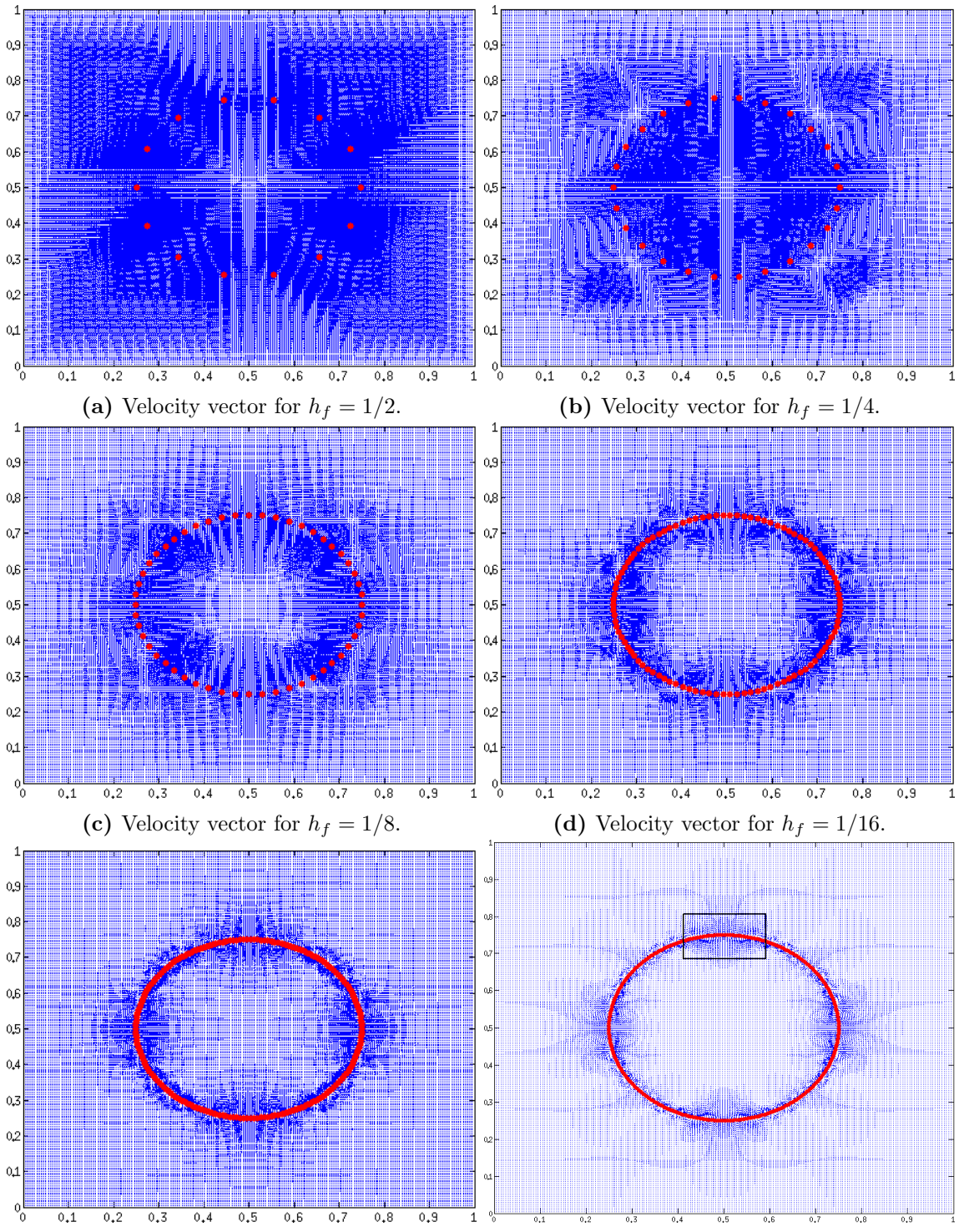
(b) p-type convergence for $h_f = 1/2$.

(c) p-type convergence for $h_f = 1/8$.

Figure 5.16: Comparison of the $\|\nabla \cdot \boldsymbol{u}_N\|_{L^2(\Omega)}$.

very little difference between the plots given in Fig. 5.16 and the ones given in Fig. 5.9 - in fact they are almost identical. This shows that although the velocity is corrected to prevent the membrane from collapsing, the actually computed velocity solution remains unchanged. Thus, we do not consider the order of convergence for the divergence of the velocity in the L^2 norm because we expect it to be the same as in Table 5.5.

The velocity correction scheme works - as the name suggests - by correcting the velocity so as to guarantee the area is conserved up to the order $O((\Delta t)^2)$. Therefore, we wish to see how it affects the velocity field. Fig. 5.17 illustrates the velocity field when $K_b = K_b(N)$ and $N = 2$ for $h_f = 1/2, \ldots, 1/64$. We note that there is absolutely no difference between the velocity vectors given in Fig. 5.17 when compared to the plots in Fig. 5.10. This is because the correction in the velocity is only used for the movement



(a) Velocity vector for $h_f = 1/2$.

(b) Velocity vector for $h_f = 1/4$.

(c) Velocity vector for $h_f = 1/8$.

(d) Velocity vector for $h_f = 1/16$.

(e) Velocity vector for $h_f = 1/32$.

(f) Velocity vector for $h_f = 1/64$.

Figure 5.17: Illustration of the velocity vector for decreasing mesh width when $N = 2$ and $K_b = K_b(N)$.

of the immersed boundary. Perhaps it would be better, once a timestep has finished,

to set the velocity field to be the corrected velocity field used in the IB movement. As the velocity vector fields are the same we do not include the plots for increasing N or the zoomed plots. We do not make use of the VCS in this thesis however it would be interesting to see how it performs when applied to an Oldroyd-B fluid and we hope to do this in the future.

In this section, we investigated the area conservation properties of the spectral element immersed boundary method (SE-IBM). The main source of error in the area conservation of the IBM is in the spreading and interpolation phases. Therefore, we hoped that using a higher-order polynomial present in a spectral discretisation would improve the accuracy of the spreading and interpolation phases and hence improve the area conservation properties of the method. In this section, we illustrated that the SE-IBM could achieve better area conservation than the FE-IBM, at a reduced computational time. Indeed, a mesh width $h_f = 1/8$ and a polynomial degree $N = 8$ yielded better area conservation than a mesh width $h_f = 1/64$ and a polynomial degree $N = 2$ and does so at a fraction of the computational time. However, we also illustrated that a coarse mesh width $h_f = 1/2$ and a high polynomial degree $N = 32$ yields very good area conservation. However, the time taken is considerably larger than the other cases considered. Therefore, we can conclude that a medium mesh width, e.g. $h_f = 1/8$, with a medium polynomial degree, e.g. $N = 8$, would be the best option as it gives a good balance between good area conservation (and therefore accurate spreading and interpolation phases) and at a small computational time.

Additionally, in this section we looked at the effect the higher-order polynomial had on the velocity field. The velocity field at low order polynomials and fine mesh widths is local to the membrane (or boundary) and is primarily directed inwards towards the centre of the immersed circular membrane. However, when a high-order polynomial is used, the velocity field not only directs inwards but also outwards from the centre of the immersed circular membrane. We believe this is the reason for the better area conservation that can be obtained when using the higher-order methods. The fluid contained inside the membrane is incompressible, therefore the membrane should not be able to collapse inwards. As the IBM is an unfitted method, the satisfaction of boundary conditions on the interface between the fluid and the structure can not be

enforced strongly. Therefore, we require something inside the membrane that pushes outwards in order to prevent the collapse (and hence the area loss). This is precisely what we see in the velocity field. Therefore, we can conclude that the higher-order polynomial is an important asset in immersed boundary computations.

Chapter 6

Numerical Investigation of XSEM

The purpose of this chapter is to numerically investigate the eXtended Spectral Element Method (XSEM) discussed in §5.3. First we consider the approximation of a discontinuous function which we use as validation for our error estimates given in §5.3.2. Then we discuss the quadrature used in this thesis for the extended part of the enriched approximation.

6.1 Approximation of a discontinuous function

Before we consider the XSEM approximation of a discontinuous function, we consider the approximation of a continuous function. Due to the accuracy of spectral methods, we choose to approximate a non-polynomial function.

Let $\Omega = [-1, 1]$ contain two subdomains $\Omega_1 = [-1, -1/3)$ and $\Omega_2 = [-1/3, 1]$ such that

$$\Gamma = \partial\Omega_1 \cap \partial\Omega_2 = -\frac{1}{3}$$

denotes the interface between the two regions. Later, Γ will denote the discontinuity in the function. However, we begin by considering a continuous function $f\colon \Omega \to \mathbb{R}$

defined by:

$$f(x) = \sin(x) \tag{6.1.1}$$

For smooth functions, $f \in C^\infty$, spectral methods can obtain exponential order of convergence, see e.g. [22, 23]. Therefore, as $\sin \in C^\infty$, we expect exponential order of convergence for the spectral element and extended spectral element approximations of f. For simplicity, we only consider a single element. Therefore the spectral element and extended spectral element approximations, denoted f_N and f_N^Γ respectively, on the domain Ω are given by:

$$f_N(x) = \sum_{i=0}^{N} f_i h_i(x) \tag{6.1.2}$$

$$f_N^\Gamma(x) = \sum_{i=0}^{N} f_i h_i(x) + \sum_{i=0}^{N} \alpha_i h_i(x) \phi_i(x) \tag{6.1.3}$$

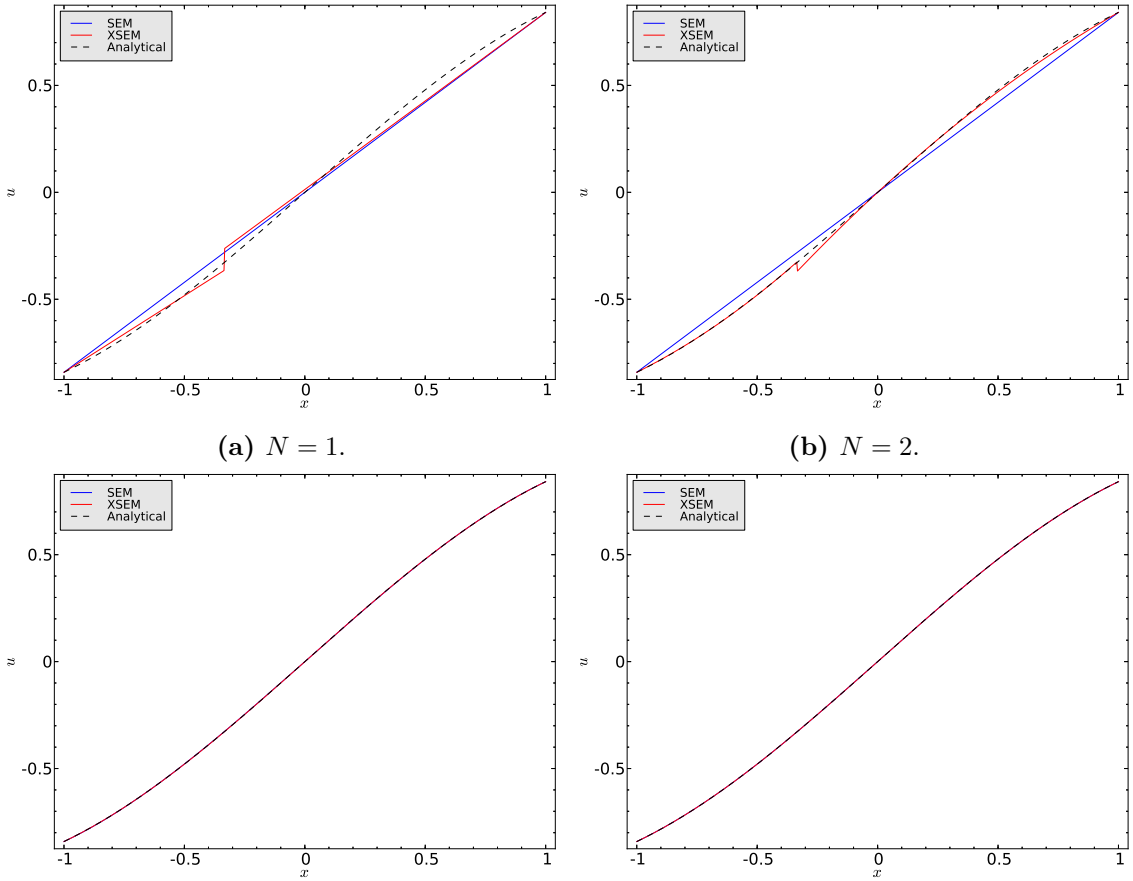
where $f_i = f(x_i)$, $i = 0, \ldots, N$, h_i are the Lagrange interpolants defined in (5.2.15), α_i are the additional degrees of freedom as a result of the enrichment and ϕ_i is the enrichment function defined by:

$$\phi_i(x) = H(x) - H(x_i) \tag{6.1.4}$$

where $H(x)$ is the Heaviside step-function defined by:

$$H(x) = \begin{cases} 0 & \text{in } \Omega_1 \\ 1 & \text{in } \Omega_2 \end{cases} \tag{6.1.5}$$

Fig. 6.1 illustrates the approximation to the solution for both SEM and XSEM as N is increased from 2 to 8. Unfortunately, we are unable to go to $N = 16$ for the reasons discussed in §5.3.1. Notice that the XSEM approximation includes a discontinuity in ϕ_i. However, it can still obtain the analytical solution exactly even though the solution is continuous. Table 6.1 illustrates the order of convergence for both SEM and XSEM with respect to the L^2 norm. To calculate the L^2 norm, as we are using high order polynomials, we look to Gauss-Lobatto-Legendre (GLL) quadrature discussed in Appendix B. We interpolate the function f using both SEM and XSEM to a finer GLL grid of the order $N_q = 1000$. This GLL grid is then used in the quadrature for the



(a) $N = 1$. **(b)** $N = 2$.

(c) $N = 4$. **(d)** $N = 8$.

Figure 6.1: Comparison of the SEM and XSEM approximation of continuous f against the analytical solution for varying values of N. The y axis label, u, denotes the function values f, f_N and f_N^Γ.

L^2 norm. It is clear from table 6.1 that for the continuous function, both SEM and XSEM obtain similar orders of convergence and that the order is high when $N = 8$. This is due to the smoothness of the function we are approximating. The reason why the SEM error for $N = 1$ and $N = 2$ is the same, is that when $N = 1$, the line cuts the y axis at $x = 0$ which is a GLL point and it just so happens that the value when $N = 2$ at $x = 0$ is the same as the value cut by the linear polynomial.

Now we add a discontinuity to our function f at the point $x = -1/3$. Thus the function

N	$\|f - f_N\|_{L^2(\Omega)}$	Order	$\left\|f - f_N^\Gamma\right\|_{L^2(\Omega)}$	Order
1	0.06084	-	0.04709	-
2	0.06084	0.00000	0.01407	1.74328
4	6.9496×10^{-4}	6.45193	6.0328×10^{-5}	7.86509
8	1.3837×10^{-8}	15.61611	2.6788×10^{-10}	17.78089

Table 6.1: Order of convergence with respect to N for both SEM and XSEM.

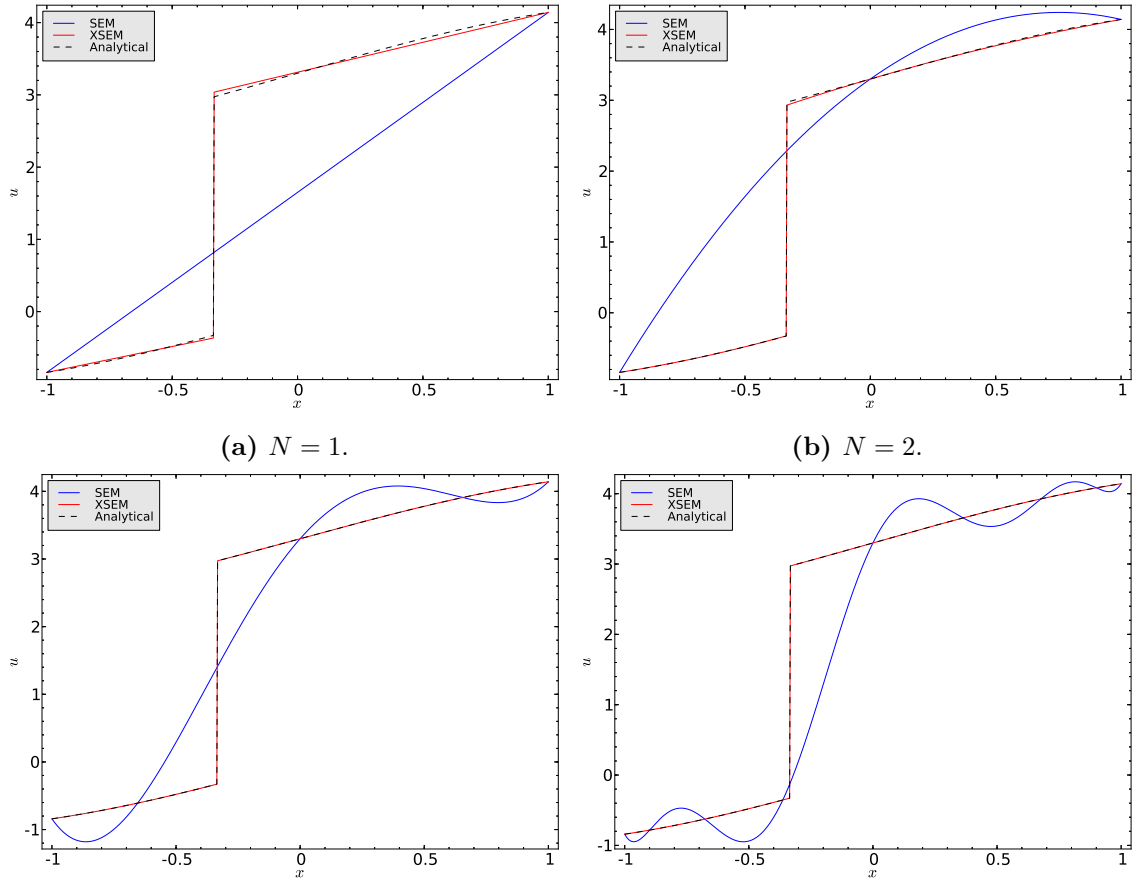
f is now defined as:

$$f(x) = \begin{cases} \sin(x) & \text{in } \Omega_1 \\ \sin(x) + 3.3 & \text{in } \Omega_2 \end{cases} \tag{6.1.6}$$

We follow the same procedure as above and approximate the function using both SEM and XSEM. Fig. 6.2 illustrates both the SEM and XSEM approximations of the discontinuous function. It is clear that SEM struggles to obtain the analytical solution due to the presence of the discontinuity. We can see clearly the oscillations present around the discontinuity (when $N = 8$) as we did in §5.2.3. Table 6.2 shows the order of convergence for both SEM and XSEM approximation of discontinuous f. Firstly we note that the SEM approximation struggles and in fact the error increases when $N = 8, 64$. The error for SEM no longer converges monotonically to the asymptotic value, instead it seems to oscillate towards the asymptotic value - which we can infer to be ≈ 0.5. These oscillations are most likely due to Gibbs phenomenon. It is clear from table 6.2, that the order of convergence for the XSEM approximation of a discontinuous function is same as when XSEM approximates a continuous function, Table 6.1. This illustrates the power of an enriched method. For a function of lower regularity, we can maintain the desired order of convergence. Therefore, we can infer that we have validated the approximation estimates given in §5.3.2.

6.2 Quadrature

In this thesis, we use XSEM to approximate the pressure in the governing equations for fluid flow. In the weak formulation of Stokes flow (5.2.19), the bilinear form involving



(a) $N = 1$. **(b)** $N = 2$.

(c) $N = 4$. **(d)** $N = 8$.

Figure 6.2: Comparison of the SEM and XSEM approximation of discontinuous f against the analytical solution for varying values of N. The y axis label, u, denotes the function values f, f_N and f_N^Γ.

the pressure term (when XSEM is used) is given by:

$$\begin{aligned} b(\boldsymbol{u}_N, q_N^\Gamma) &= \int_\Omega q_N^\Gamma \nabla \cdot \boldsymbol{u}_N \, \mathrm{d}\Omega \\ &= \int_\Omega q_N \nabla \cdot \boldsymbol{u}_N \, \mathrm{d}\Omega + \int_\Omega q_N^X \nabla \cdot \boldsymbol{u}_N \, \mathrm{d}\Omega \end{aligned} \tag{6.2.1}$$

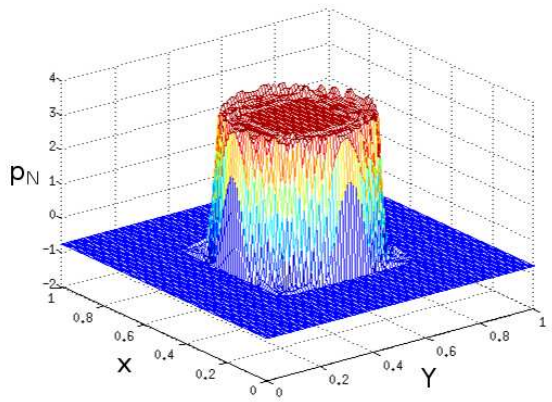
where q_N^Γ is the enriched approximation composed of the standard approximation, q_N, and an extended part q_N^X such that $q_N^\Gamma = q_N + q_N^X$. In this thesis, the function q_N^X is discontinuous as we are enriching a strong discontinuity. Therefore, using the standard GLL quadrature on the second integral above would not be accurate due to the integrand being discontinuous. In the XFEM literature, see e.g. [38], the standard procedure is to *refine* the mesh in an element which contains a discontinuity. This is not considered re-meshing as no additional degrees of freedom are added - the refinement

N	$\|f - f_N\|_{L^2(\Omega)}$	Order	$\|f - f_N^\Gamma\|_{L^2(\Omega)}$	Order
1	1.57870	-	0.04709	-
2	1.37584	0.19843	0.01407	1.74328
4	0.80355	0.77586	6.0328×10^{-5}	7.86509
8	1.06387	-0.40487	2.6811×10^{-10}	17.77967
16	0.62365	0.77051	-	-
32	0.32542	0.93845	-	-
64	0.43223	-0.40950	-	-
128	0.28190	0.61661	-	-
256	0.16528	0.77031	-	-

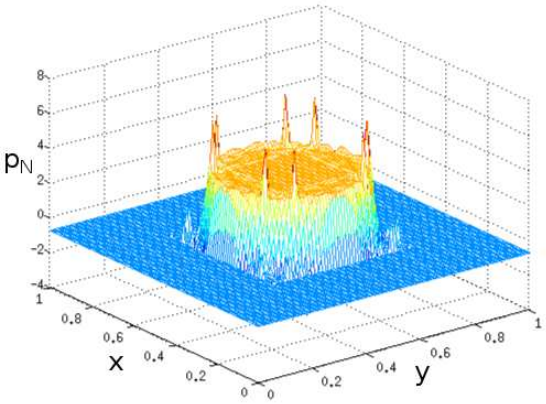
Table 6.2: Order of convergence in the L^2 norm with respect to N for both SEM and XSEM.

is *only* used for the quadrature. This ensures that the second integral above is exact. However, in this thesis, we do not employ this procedure. Instead we over-integrate. Below we present some results to illustrate that over-integration is valid and that if we do not choose the number of quadrature points N_q high enough then a lack of enrichment is seen.

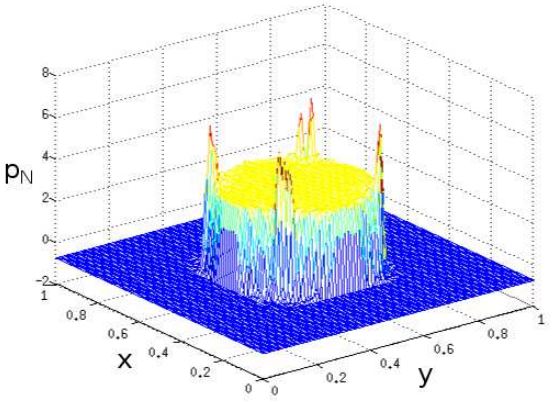
To illustrate that over-integration is valid and some potential problems seen with it, we use the same test problem that we use throughout this thesis; that is the example of a static closed membrane immersed in a Newtonian fluid, as depicted by Fig. 4.3. We do not go into details of the analytical solution here as they are given in Chapter 7. Fig. 6.3 illustrates the enriched pressure solution for varying values of N_q. It is important to note, that the figures in the left column of 6.3 are *not* the standard SEM approximation. Instead, they are the original part p_N of the enriched approximation $p_N^\Gamma = p_N + p_N^X$. First we notice that as N_q is increased from 2 to 18, most of the oscillations around the discontinuity have been removed. However, there are four big peaks which can clearly be seen. Following the discussion in §5.3.3, it is possible that these peaks are related to *regions of small support* and therefore may be due to a loss of inf-sup stability. As we are over-integrating, it is possible that we have a pseudo-spurious mode caused by a lack of enrichment. Indeed, this seems to be suggested by the plot in Fig. 6.3e. Increasing N_q from 18 to 20 removes those big peaks seen in Fig. 6.3c, thus implying that we have an error in our approximation due to a lack of enrichment rather than an inf-sup instability. However, it is evident in Fig. 6.3e that there are still small oscilla-



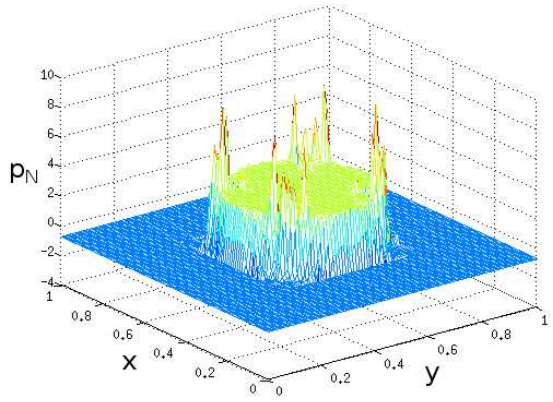
(a) p_N for $N_q = 2$.



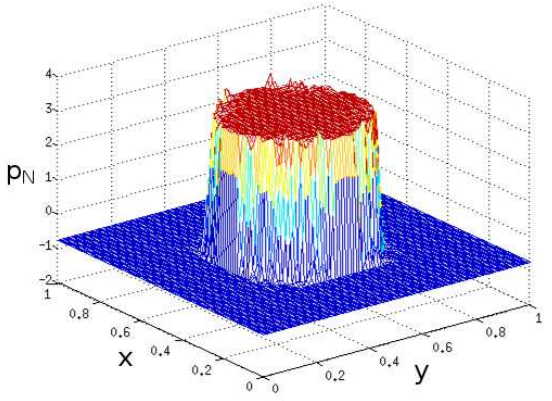
(b) p_N^Γ for $N_q = 2$.



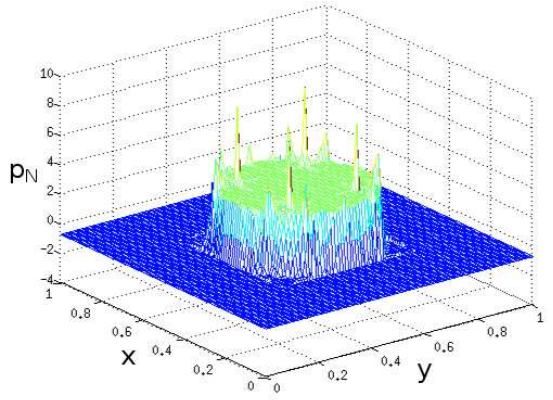
(c) p_N for $N_q = 18$.



(d) p_N^Γ for $N_q = 18$.



(e) p_N for $N_q = 20$.



(f) p_N^Γ for $N_q = 20$.

Figure 6.3: Plots of the pressure p_N and $p_N^\Gamma = p_N + p_N^X$ for $N_q = 2, 18, 20$.

tions local to the discontinuity. Again we believe that these are simply due to a lack of enrichment and that increasing the number of quadrature points N_q would resolve this issue or that refining the mesh local to the discontinuity would yield better results.

The right column in Fig. 6.3 is dedicated to the full enriched approximation p_N^Γ. It is clear that the oscillations present in Fig. 6.3a are also present in Fig. 6.3b. However, there are also additional peaks which are large in comparison to the other oscillations. Increasing N_q from 2 to 18 does *not* remove these peaks. Therefore, this implies that these peaks are the result of regions of small support and hence show inf-sup instability. This agrees with the work of [38]. Additionally, in Fig. 6.3f, we can see more peaks just inside the membrane/interface. These are present due to problems with blending. Blending is well known within the XFEM literature, e.g. [28]. Blending problems occur when a node is shared between a enriched element and a standard element. Legay *et al.* [42] showed that when N is increased, blending elements are not needed. However, in that article, the extended part of their enriched approximation was of a degree less than the standard approximation and they were not approximating the pressure. Here, not only are we using the same degree polynomials for the standard approximation and the extended part of the enriched approximation, but those approximations are based on the interior GLL nodes. Therefore, blending problems are seen to occur at shared nodes. The removal of the blending problems is a subject of future research.

One would expect that the standard part of the enriched pressure solution would remain the same as the general idea of the additional degrees of freedom is to correct any issues that occur in the original approximation. Therefore, we were quite surprised that there is a change in the standard part of the pressure in the left hand column of Fig. 6.3. As we mentioned before, in §5.3.3, when using the XSEM approximation of the pressure, we split the incompressibility constraint into two parts so that:

$$b(\boldsymbol{u}_N, q_N) = 0 \qquad\qquad b(\boldsymbol{u}_N, q_N^X) = 0 \tag{6.2.2}$$

and that this implies a doubly-constrained minimisation problem. Clearly, $\mathcal{Q}_N^X \subset \mathcal{Q}_N$ and therefore, one could argue that the second constraint involving q_N^X is automatically satisfied if the first constraint is satisfied. Therefore, we could conclude that separating the incompressibility constraint is what alters the standard part of the enriched approximation.

Chapter 7

Validation

The purpose of this chapter is to validate the fluid solver and immersed boundary method. The chapter is organised as follows. First we validate the Newtonian and Oldroyd-B fluid solvers against an example problem - transient Poiseuille flow. Then we validate the immersed boundary method using four well known model problems found in the literature; a static and oscillating membrane and a static and oscillating shell.

7.1 Transient Poiseuille Flow

7.1.1 Newtonian Fluid

We validate our Newtonian fluid solver using the time-dependent analytical solution of transient Poiseuille flow through a channel of length L and height H. Poiseuille flow is achieved by applying a constant pressure gradient along the channel. For the purposes of this validation, we use the non-dimensionalised formulation given in (2.2.14). The

analytical solution for transient Poiseuille flow is given by [41]:

$$u_x(\boldsymbol{x},t) = A(y) - 32\sum_{n=1}^{\infty}\frac{\sin\left(\nu(n)y/H\right)}{\nu(n)^3}\exp\left(-\frac{\nu(n)^2 t}{Re}\right) \tag{7.1.1a}$$

$$u_y(\boldsymbol{x},t) = 0 \tag{7.1.1b}$$

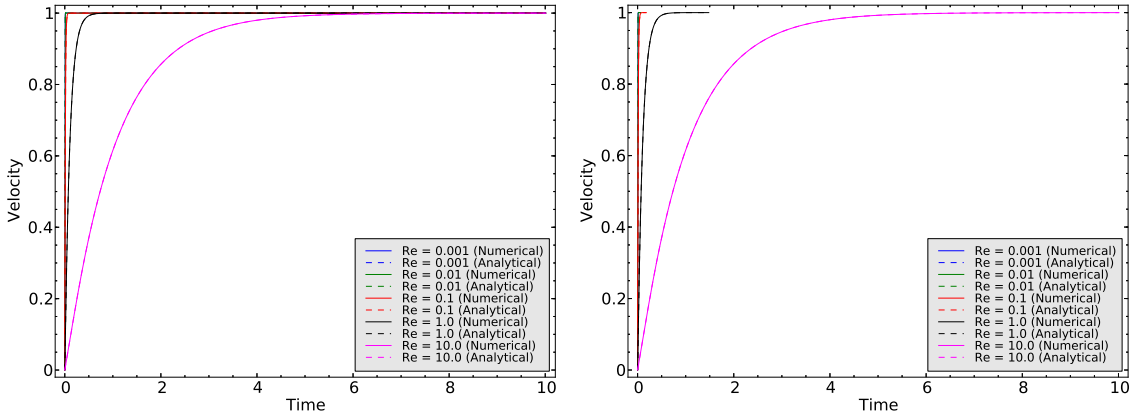
$$p(\boldsymbol{x},t) = \frac{-8}{H}x \tag{7.1.1c}$$

where $\boldsymbol{u} = (u_x, u_y)$ is the fluid velocity, p is the pressure, Re is the Reynolds number defined in (2.2.15), $A(y) = -4y(y-1)$ and $\nu(n) = (2n-1)\pi$.

In this example, we take the channel to be of length $L = 8$ and height $H = 1$, $\Omega = [0,8]\times[0,1]$, with zero initial velocity, $\boldsymbol{u}(\boldsymbol{x},0) = \boldsymbol{0}$. We use the analytical solution to impose Dirichlet boundary conditions at the inflow and outflow boundaries of the channel (i.e. when $x = 0, 8$) and impose no-slip boundary conditions are the top and bottom boundaries of the channel (i.e. when $y = 0, 1$). In all simulations, we terminate at time $T = 10$. We choose to terminate the infinite sum given in (7.1.1a) after 20 steps because for $N = 4$ (the lowest value of N considered here) and $t \geq 10^{-4}$ (the lowest timestep considered here) we observed that the 20th term in the summation had a value in the region of 10^{-5} and lower. Kynch [41], Van Os [68] and Van Os and Phillips [69] also used 20 terms in the summation. The parameters are as follows: $N = 4, 6, 8, 12$, $Re = 0.001, 0.01, 0.1, 1, 10$ and $\Delta t = 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}$.

We consider two meshes to validate the Newtonian solver. We decompose our domain in uniform quadrilateral spectral elements with mesh widths $h_f = 1, 1/4$. Fig. 7.1 illustrates the velocity solution at the penultimate GLL node just inside the outflow boundary at $y = 0.5$ for both meshes and all values of the Re. It is clear that we have good agreement with the analytical solution and our solution matches the results given in the thesis of R. Kynch [41] and R. Van Os [68]. In Fig. 7.1b, for $Re \leq 1$ the code identifies that we have reached a steady state and terminates early.

Table 7.1 details the H^1 norm of the velocity error at $T = 0.8$ for all meshes, all values of N and all values of Δt when $Re = 1$. For a mesh width $h_f = 1$, it is clear from



(a) $h_f = 1$. **(b)** $h_f = 1/4$.

Figure 7.1: Computed and analytical u_x solution for $Re = 0.001, 0.01, 0.1, 1, 10$ with $N = 6$ and $\Delta t = 10^{-2}$.

Table 7.1 that spatial errors are dominating the H^1 norm of the velocity error when $N = 4, 6$. However, at the higher polynomial degrees ($N = 8, 12$) it is clear that spatial errors are no longer the dominating factors and we obtain second-order accuracy with respect to Δt. For a mesh width of $h_f = 1/4$, it is clear from Table 7.1 that when $N = 4$ we have almost obtained second-order accuracy with respect to Δt and that it is achieved for $N \geq 6$.

Concerning spatial refinement, we see from Table 7.1, that for a fixed value of $N = 4$ and $\Delta t = 10^{-3}$, that the h-type refinement improves the H^1 error of the velocity. However at larger values of $N = 8, 12$, there is very little improvement in the error as the mesh is refined. Correspondingly, for a fixed mesh width $h_f = 1$ and $\Delta t = 0.001$, the p-type refinement improves the H^1 error of the velocity. However, when $h_f = 1/4$, we see that there is very little improvement in the error. This agrees with the results of R. Kynch [41].

7.1.2 Oldroyd-B

We validate our Oldroyd-B fluid solver using the same example as the previous subsection; the time-dependent analytical solution of transient Poiseuille flow through a channel of length L and height H. Poiseuille flow is achieved by applying a constant

h_f	Δt	$N = 4$	$N = 6$	$N = 8$	$N = 12$
1	10^{-1}	1.7884×10^{-4}	2.0278×10^{-4}	2.0429×10^{-4}	2.0424×10^{-4}
	10^{-2}	8.8883×10^{-6}	1.1078×10^{-6}	1.1207×10^{-6}	1.1168×10^{-6}
	10^{-3}	1.0092×10^{-5}	9.3914×10^{-8}	1.0497×10^{-8}	1.0446×10^{-8}
	10^{-4}	1.0103×10^{-5}	9.4699×10^{-8}	6.1731×10^{-10}	1.0357×10^{-10}
$\frac{1}{4}$	10^{-1}	2.0522×10^{-4}	2.0432×10^{-4}	2.0424×10^{-4}	2.0432×10^{-4}
	10^{-2}	1.1225×10^{-6}	1.1173×10^{-6}	1.1159×10^{-6}	1.1159×10^{-6}
	10^{-3}	1.2485×10^{-8}	1.0450×10^{-8}	1.0438×10^{-8}	1.0438×10^{-8}
	10^{-4}	6.7263×10^{-9}	1.0254×10^{-10}	1.0219×10^{-10}	9.6361×10^{-11}

Table 7.1: H^1 norm error of the velocity at $T = 0.8$ when $Re = 1.0$.

pressure gradient across the channel. Again, for the purposes of this validation, we use the non-dimensionalised formulation given in (2.2.26). The analytical solution for transient Poiseuille flow of an Oldroyd-B fluid was derived by Waters and King [72]:

$$u_x(\boldsymbol{x}, t) = A(y) - 32 \sum_{n=1}^{\infty} \frac{\sin(\nu(n)y)}{\nu(n)^3} \exp\left(-\frac{\alpha_\nu t}{2We}\right) G_\nu(t) \tag{7.1.2a}$$

$$u_y(\boldsymbol{x}, t) = 0 \tag{7.1.2b}$$

where $\boldsymbol{u} = (u_x, u_y)$ is the fluid velocity, $A(y) = -4y(y-1)$, $\nu(n) = (2n-1)\pi$ and $G_\nu(t)$ is given by:

$$G_\nu(t) = \cosh\left(\frac{\beta_\nu t}{2We}\right) + \left[\frac{s_\nu}{\beta_\nu}\right] \sinh\left(\frac{\beta_\nu t}{2We}\right) \tag{7.1.3}$$

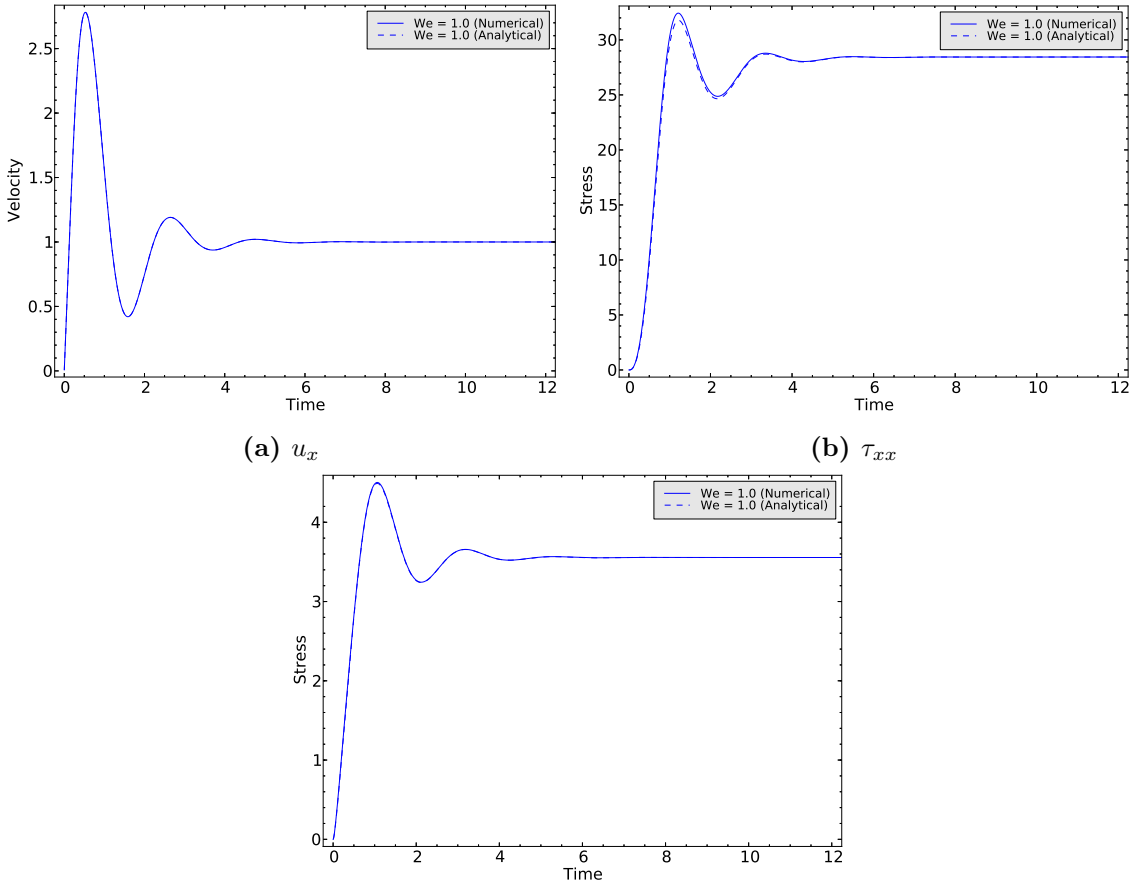
where We is the Weissenberg number defined in (2.2.28) and

$$El = \frac{We}{Re}, \quad \alpha_\nu = 1 + \beta\nu(n)^2 El$$
$$\beta_\nu^2 = \alpha_\nu^2 - 4\nu(n)^2 El, \quad s_\nu = 1 + (\beta - 2)\nu(n)^2 El$$

The analytical expressions for the polymeric stress components may be found in Carew *et al.* [24] or the thesis of R. van Os [68]. In this section, we compare our results against the work of R. Kynch [41], R. van Os [68] and Van Os and Phillips [69].

Following Van Os and Phillips [69], we take the channel to be of length 64 and height 1, $\Omega = [0, 64] \times [0, 1]$, with zero initial velocity, $\boldsymbol{u}(\boldsymbol{x}, 0) = \mathbf{0}$ and zero initial stress, $\boldsymbol{\tau}(\boldsymbol{x}, 0) = \mathbf{0}$. We use the analytical solution of the velocity to impose Dirichlet boundary conditions at the inflow and outflow boundaries of the channel (i.e. when $x = 0, 64$) and impose no-slip boundary conditions are the top and bottom boundaries of the channel (i.e. when $y = 0, 1$). The analytical solution for the stress is used to impose Dirichlet boundary conditions at the inflow boundary only. In all simulations, we terminate at time $T = 40$. We terminate the infinite sum given in (7.1.2a) after 20 steps as was done for the Newtonian validation in §7.1.1. Van Os and Phillips [69], and Kynch [41], also used 20 steps in the calculation of the infinite sum. In order to compare our results to the work of Van Os and Phillips [69], Van Os [68] and Kynch [41], we use the same parameters as they did. We let $Re = 1$, the viscosity ratio $\beta = 1/9$ and consider $We = 1, 10, 100$. The other parameters are as follows: $N = 2, 4, 8, 12$ and $\Delta t = 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}$.

Again, we consider two meshes to validate the Oldroyd-B solver as we did when we validated the Newtonian solver. We decompose our domain in uniform quadrilateral spectral elements with mesh widths $h_f = 1, 1/4$. Fig. 7.2 illustrates the velocity solution at the penultimate GLL node just inside the outflow boundary at $y = 0.5$, the τ_{xx} component and the τ_{xy} component at the penultimate GLL node just inside the outflow boundary and just above $y = 0$ when $N = 8$, $\Delta t = 10^{-3}$ and $We = 1$. Clearly, the computed solution for the velocity component and the τ_{xy} component are barely discernible from their analytical solutions and all the components experience an overshoot followed by an undershoot before finally settling down to a steady state. However, the τ_{xx} component differs slightly from the analytical solution between times of approximately $T = 1$ and $T = 3$. This difference was also found by Kynch [41] and Van Os [68]. The analytical solution for the τ_{xx} component contains a double infinite sum whereas the velocity and shear stress, τ_{xy} component, only contain a single infinite sum. It is possible that this is where the error in the τ_{xx} component can be found. In fact, Kynch [41] found that the difference between the analytical τ_{xx} solution when 20 terms were used from that computed when 100 terms were used, was quite large during $T = 0$ to $T = 5$.

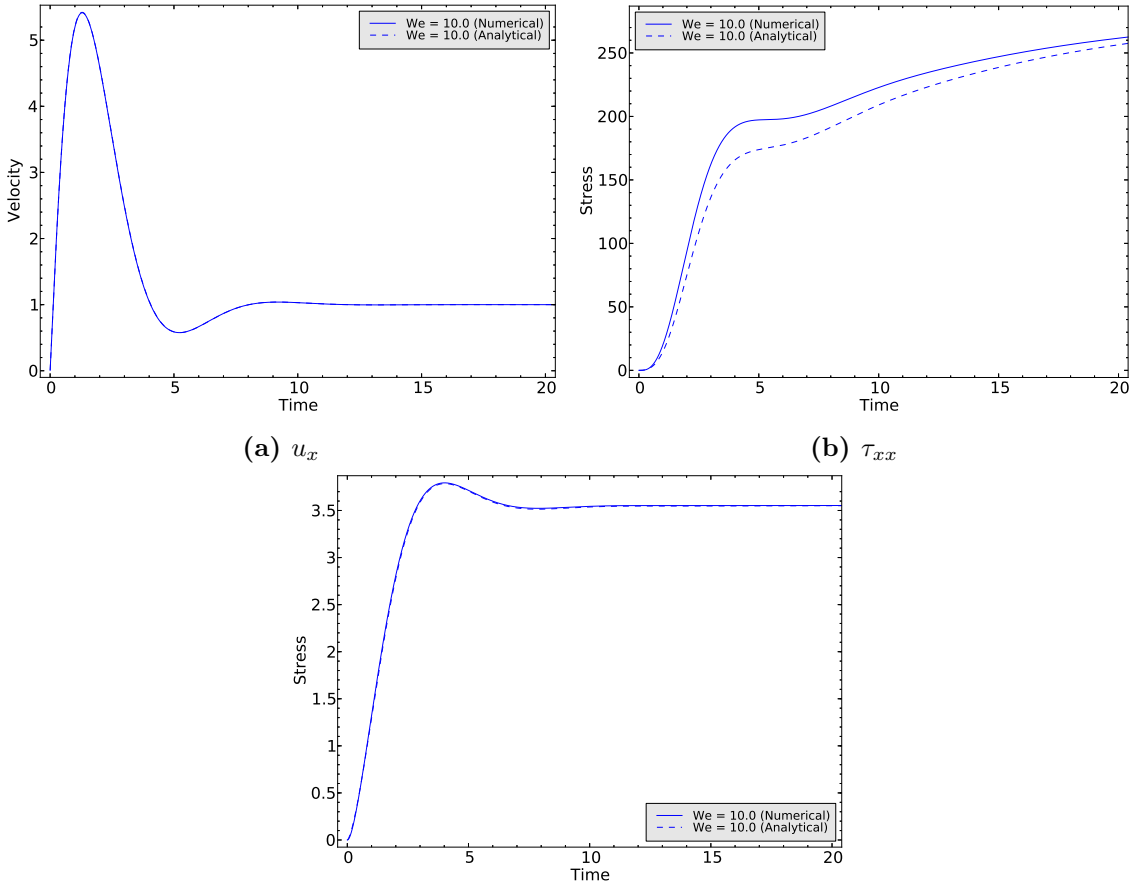


(a) u_x **(b)** τ_{xx}

(c) τ_{xy}

Figure 7.2: Computed and analytical u_x, τ_{xx} and τ_{xy} solutions for $We = 1$ with $N = 8$ and $\Delta t = 10^{-3}$.

Fig. 7.3 illustrates the u_x, the τ_{xx} and the τ_{xy} solutions plotted at the same test points that were used in Fig. 7.2 when $N = 4$, $\Delta t = 10^{-3}$ and $We = 10$. Clearly, the computed solution for the velocity component and the shear stress component are again barely discernible from their analytical solutions. However, there is quite a big discreprency in the τ_{xx} component. This discrepency was also found by Kynch [41] and Van Os and Phillips [69]. In this thesis, we employ a high order explicit scheme to approximate the non-linear deformation terms which appear in the upper-convected derivative present in the Oldroyd-B constitutive equation (2.2.26c). We mentioned in §4.1.2 that at the higher Weissenberg numbers one would require a more implicit treatment of the deformation terms. Therefore, due to the higher Weissenberg number considered in Fig. 7.3 it is possible that a more implicit representation of these non-



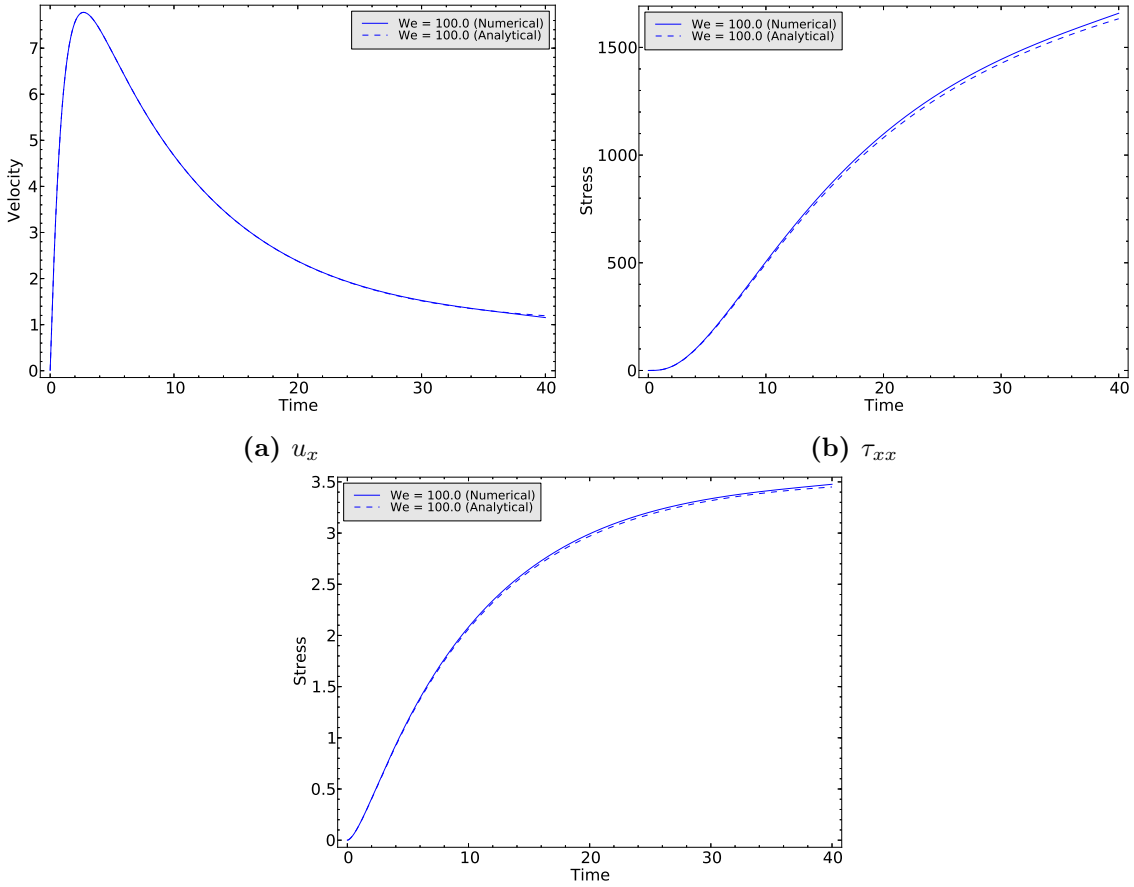
(a) u_x **(b)** τ_{xx}

(c) τ_{xy}

Figure 7.3: Computed and analytical u_x, τ_{xx} and τ_{xy} solutions for $We = 10$ with $N = 4$ and $\Delta t = 10^{-3}$.

linear terms would yield better results.

Fig. 7.4 illustrates the u_x, the τ_{xx} and the τ_{xy} solutions plotted at the same test points used in Fig. 7.2 when $N = 4$, $\Delta t = 10^{-3}$ and $We = 100$. Clearly, the computed solution for the velocity and shear stress still have very good agreement with the analytical solution, although not perfect. It appears as if the computed τ_{xx} solution is closer to its analytical solution and therefore one may think that we have a better approximation at a higher Weissenberg number. However, it is clear that neither the velocity, shear stress or normal stress have reached a steady state and we expect that if the computation was run for longer, it would eventually break down. These results agree with the findings of Kynch [41].



(a) u_x **(b)** τ_{xx}

(c) τ_{xy}

Figure 7.4: Computed and analytical u_x, τ_{xx} and τ_{xy} solutions for $We = 100$ with $N = 4$ and $\Delta t = 10^{-3}$.

Following the Newtonian validation, we consider the H^1 norm of the velocity error at a specific point in time. Also we consider the L^2 norm of the stress error at the same time. Table 7.2 details the H^1 norm of the velocity error at $T = 5$ for all meshes, all values of N and all values of Δt when $We = 1$. For a mesh width $h_f = 1$, it is clear from Table 7.2 that when $N = 4$ we have quite a large error in the H^1 norm of the velocity which does not improve as we decrease the size of Δt. Initially, one would believe that we have similar behaviour to that of the Newtonian fluid given in Table 7.1 and that spatial errors are the main cause. However, it is clear from Table 7.2 that when $N = 12$ (so that any contribution from spatial errors are eliminated) we still do not obtain second-order accuracy with respect to Δt. Therefore, the loss in the accuracy must be related to the temporal discretisation. Refining the mesh to $h_f = 1/4$ does not reduce the error as much as we would like. It is clear that we cannot get below 10^{-5}. Kynch [41] found the same loss in accuracy and also could not achieve

h_f	Δt	$N = 4$	$N = 6$	$N = 8$	$N = 12$
1	10^{-1}	2.1039×10^{-2}	2.2548×10^{-2}	2.1059×10^{-2}	2.1971×10^{-2}
	10^{-2}	9.6642×10^{-4}	5.9747×10^{-4}	5.5509×10^{-4}	4.4495×10^{-4}
	10^{-3}	1.1613×10^{-3}	4.7598×10^{-5}	5.6603×10^{-5}	5.0847×10^{-5}
	10^{-4}	1.1964×10^{-3}	1.4944×10^{-5}	1.1750×10^{-5}	2.4609×10^{-5}
$\frac{1}{4}$	10^{-1}	4.2796×10^{-2}	4.7722×10^{-2}	1.2415×10^{-1}	-
	10^{-2}	7.3051×10^{-4}	4.5503×10^{-4}	4.6935×10^{-4}	-
	10^{-3}	8.1685×10^{-5}	6.4789×10^{-5}	1.1528×10^{-4}	-
	10^{-4}	2.5108×10^{-5}	4.1359×10^{-5}	9.8918×10^{-5}	-

Table 7.2: H^1 norm of the velocity error at $T = 5$ when $We = 1.0$.

an error lower than 10^{-5} in the H^1 norm. On the plus side, in this thesis we have been able to obtain results when $N = 8$ which were omitted in the thesis of Kynch [41]. We were unable to deal with $N = 12$ because the computational time required was greater than the maximum wall time available. As we are using a second-order OIFS scheme for the material derivative and a second-order Adams-Bashforth scheme for the non-linear deformation terms, we expected second-order accuracy and therefore it is unclear where the origins of this loss in accuracy are.

Concerning spatial refinement, we notice that if we fix $\Delta t = 0.001$ and $N = 4$, then the error does improve as we refine the mesh. If we fix $\Delta t = 0.001$ and fix the mesh width $h_f = 1$, we notice that the error decreases from $N = 2$ to 4 but then increases from $N = 4$ to 8. For a mesh width of $h_f = 1/4$, we see an improvement of the error as N increases from 4 to 6 but then the error decreases when $N = 8$. We believe the reason for this is the inherent instability of the solution.

Table 7.3 details the L^2 norm of the polymeric stress error at $T = 5$ for all meshes, all values of N and all values of Δt when $We = 1$. For a mesh width $h_f = 1$, it is clear from Table 7.3 that we have not obtained second-order accuracy with respect to Δt for any value of N. In fact for any value of $\Delta t < 10^{-2}$ the L^2 norm of the stress error

h_f	Δt	$N = 4$	$N = 6$	$N = 8$	$N = 12$
1	10^{-1}	1.3466×10^{-1}	1.6935×10^{-1}	1.7094×10^{-1}	1.7693×10^{-1}
	10^{-2}	3.8357×10^{-2}	3.2532×10^{-2}	2.5197×10^{-2}	2.2420×10^{-2}
	10^{-3}	4.5619×10^{-2}	3.4140×10^{-2}	2.5823×10^{-2}	2.0334×10^{-2}
	10^{-4}	4.6356×10^{-2}	3.4445×10^{-2}	2.6095×10^{-2}	2.0403×10^{-2}
$\frac{1}{4}$	10^{-1}	1.6993×10^{-1}	1.8634×10^{-1}	1.9678×10^{-1}	-
	10^{-2}	2.4573×10^{-2}	2.3328×10^{-2}	1.9376×10^{-2}	-
	10^{-3}	2.5978×10^{-2}	2.0465×10^{-2}	1.7259×10^{-2}	-
	10^{-4}	2.6319×10^{-2}	2.0452×10^{-2}	1.7382×10^{-2}	-

Table 7.3: L^2 norm of stress error at $T = 5$ when $We = 1.0$.

seems to get worse. Refining the mesh does not improve things either, we still cannot achieve second-order accuracy in time and we are unable to go below an error of 10^{-2} which is disappointing.

Out of curiosity, we consider the same error but for the highest Weissenberg number $We = 100$. Table 7.4 details the L^2 norm of the stress error at $T = 5$ for all meshes, all values of N and all values of Δt when $We = 100$. It is evident in Table 7.4 that we see the same behaviour that was seen in Table 7.3 when $We = 1$. We notice that for $\Delta t = 0.1$, as we increase N the L^2 norm of the stress error increases. Mesh refinement does improve the error on the lower values of N but makes the error worse for larger values of N.

As we have been able to obtain good agreement with the results of Kynch [41], Van Os [68] and Van Os and Phillips [69], for both Newtonian and Oldroyd-B fluids, we can conclude that we have validated our fluid solvers. In the next section, we validate our immersed boundary solver.

h_f	Δt	$N = 4$	$N = 6$	$N = 8$	$N = 12$
1	10^{-1}	7.2050	8.4864	9.8649	11.3769
	10^{-2}	7.0079	4.6787	3.7107	3.1338
	10^{-3}	7.6719	5.2102	3.9752	2.9386
	10^{-4}	7.7415	5.2699	4.0133	2.9364
$\frac{1}{4}$	10^{-1}	9.1282	11.6734	11.3798	-
	10^{-2}	3.4969	3.4196	44.6983	-
	10^{-3}	3.9296	3.1063	70.7928	-
	10^{-4}	3.9829	3.0896	74.0297	-

Table 7.4: L^2 norm of the stress error at $T = 5$ when $We = 100.0$.

7.2 Immersed Boundary Method

We validate the immersed boundary method using four well known sample problems. The first two involve a one-dimensional structure, such as a fibre, immersed in a two-dimensional Newtonian fluid, while the second two involve a two-dimensional structure, such as a shell, immersed in a two-dimensional Newtonian fluid. Where possible, we shall compare the computed solution against the analytical solution but in all cases we shall refer to the results for the FE-IBM of Boffi *et al.* [17] who considered the same sample problems.

Let $\Omega(t) \subset \mathbb{R}^2$ be a region containing the current configuration of a viscoelastic continuum. Let $\Omega^f(t) \subset \Omega$ be the subdomain containing the current configuration of a Newtonian fluid and $\Omega^s(t) \subset \Omega$ be the subdomain containing the current configuration of the immersed viscoelastic structure such that $\bar{\Omega}^f(t) \cup \bar{\Omega}^s(t) = \bar{\Omega}(t)$ and $\Omega^f(t) \cap \Omega^s(t) = \emptyset$, $\forall t \in [0, T]$. Note that $\Omega^f(t) \subset \mathbb{R}^2$ and $\Omega^s(t) \subset \mathbb{R}^d$, $d = 1, 2$. Also, we write $\Omega = \Omega(t)$ to emphasize that the domain containing both the fluid and the immersed structure does not change in time. Additionally, we define $\Gamma(t) = \bar{\Omega}^f(t) \cap \bar{\Omega}^s(t) = \partial\Omega^f(t) \cap \partial\Omega^s(t)$ to be the interface between the fluid and the immersed structure. For the immersed viscoelastic structure, it makes sense for us to define the reference configuration of the

material. Let Ω_r^f denote the reference configuration of the fluid domain, Ω_r^s denote the reference configuration of the immersed viscoelastic structure and Γ_r denote the reference configuration of the interface between fluid and structure.

In all the validation examples given here, we consider a viscoelastic structure immersed in two-dimensional fluid flow. The domain $\Omega = [0,1] \times [0,1]$. The continuous weak formulation of Stokes flow with an immersed structure is: find $(\boldsymbol{u}, p, \boldsymbol{X}) \in \mathcal{V} \times \mathcal{Q} \times \mathcal{X}$ such that

$$\eta_s a(\boldsymbol{u}, \boldsymbol{v}) + b(\boldsymbol{v}, p) = \langle \boldsymbol{f}, \boldsymbol{v} \rangle \qquad \forall \boldsymbol{v} \in \mathcal{V} \tag{7.2.1a}$$

$$b(\boldsymbol{u}, q) = 0 \qquad \forall q \in \mathcal{Q} \tag{7.2.1b}$$

$$\langle \boldsymbol{f}, \boldsymbol{v} \rangle = -\int_{\Omega_r^s} \mathbb{P}^s \colon \nabla_s \boldsymbol{v}(\boldsymbol{X}(\boldsymbol{s}, t))\, \mathrm{d}\Omega \qquad \forall \boldsymbol{v} \in \mathcal{V} \tag{7.2.1c}$$

$$\frac{\partial \boldsymbol{X}}{\partial t} = \boldsymbol{u}(\boldsymbol{X}(\boldsymbol{s}, t), t) \qquad \forall \boldsymbol{s} \in \Omega_r^s \tag{7.2.1d}$$

with suitable initial, $\boldsymbol{u}(\boldsymbol{x}, 0) = \boldsymbol{u}^0$ and $\boldsymbol{X}(\boldsymbol{s}, 0) = \boldsymbol{X}^0$, and boundary conditions $\boldsymbol{u} = \boldsymbol{u}_D$ on $\partial\Omega$. The bilinear forms $a(\cdot, \cdot)$ and $b(\cdot, \cdot)$ are defined as in (5.2.3). The temporal discretisation of the problem is performed as in Chapter 4. We also use a semi-implicit Euler method for the movement of the immersed boundary; i.e.

$$\frac{\boldsymbol{X}^{n+1} - \boldsymbol{X}^n}{\Delta t} = \boldsymbol{u}^{n+1}(\boldsymbol{X}^n) \tag{7.2.2}$$

Spectral elements will be used for the spatial discretisation of the fluid as given in Chapter 5. For the immersed one-dimensional structure, piecewise linear finite elements will be used where as linear spectral elements will be used for the immersed two-dimensional structure as described in §5.4.1. For the immersed membrane, we give results for both Stokes and Navier-Stokes equations. The reason is that the analytical solutions for the model problems are solutions of Stokes equations and it is possible that the loss in area will induce a contribution from the material derivative that shouldn't exist.

7.2.1 Static and Oscillating Membrane

Static

This example, as mentioned by Boffi *et al.* [17], is the simplest example that can be considered and therefore has been widely used as a model problem in immersed boundary computations [59, 12, 16, 52]. Indeed we used this same example for our study of the volume/area loss given in §5.4.2. The structure is a one-dimensional fibre whose reference configuration is initially stressed and takes the position of a circle with radius $R = 0.25$. The fibre Ω_r^s is parameterised by arc-length $s \in [0, 2\pi R]$ so that the initial configuration of the fibre is given by:

$$\boldsymbol{X}(s,0) = \begin{pmatrix} R\cos\left(\dfrac{s}{R}\right) + 0.5 \\ R\sin\left(\dfrac{s}{R}\right) + 0.5 \end{pmatrix} \tag{7.2.3}$$

$\forall s \in [0, 2\pi R]$. The velocity is chosen to satisify homogeneous Dirichlet boundary conditions, $\boldsymbol{u}_D = 0$ on $\partial\Omega$, with zero initial condition, $\boldsymbol{u}(\boldsymbol{x}, 0) = \boldsymbol{0}$. Fortunately, due to the simplicity of the example, an analytical solution exists and is given in Boffi *et al.* [17] by:

$$\boldsymbol{u}(\boldsymbol{x},t) = 0 \qquad \forall \boldsymbol{x} \in \Omega,\ \forall t \in [0,T] \tag{7.2.4a}$$

$$p(\boldsymbol{x},t) = \begin{cases} \kappa\left(\dfrac{1}{R} - \pi R\right) & |\boldsymbol{x} - \boldsymbol{c}| \leq R \\ -\kappa\pi R & |\boldsymbol{x} - \boldsymbol{c}| > R \end{cases} \qquad \forall t \in [0,T] \tag{7.2.4b}$$

where $\boldsymbol{c} = (0.5, 0.5)$ is the centre of the circle. Boffi *et al.* [17] considered a hyperelastic fibre satsifying a neo-Hookean model. Therefore, the expression for the Piola-Kirchhoff stress tensor, $\mathbb{P}^s$, was found from the strain energy density function for the fibre so that:

$$\mathbb{P}^s = \kappa \frac{\partial \boldsymbol{X}}{\partial s} = \kappa \mathbb{F} \tag{7.2.5}$$

where $\mathbb{F}$ is the deformation gradient tensor. Therefore, the duality pairing $\langle \boldsymbol{f}, \boldsymbol{v} \rangle$ is given by

$$\langle \boldsymbol{f}, \boldsymbol{v} \rangle = -\kappa \int_0^{2\pi R} \frac{\partial \boldsymbol{X}}{\partial s} \cdot \frac{\partial \boldsymbol{v}(\boldsymbol{X}(\boldsymbol{s},t))}{\partial s}\, \mathrm{d}s \tag{7.2.6}$$

Incidentally, the same expression can be obtained by considering the variational formulation of the original immersed boundary method; that is

$$\boldsymbol{F}(s,t) = \kappa \frac{\partial^2 \boldsymbol{X}}{\partial s^2} \tag{7.2.7}$$

then use integration by parts in the weak formulation. Interestingly, the representation of the Lagrangian force density is the same whether we consider a Hookean spring law (as is done with the original IBM, §3.1.1) or a neo-Hookean hyperelastic material (as is done in [17]). Boffi *et al.* [17] mentioned that the force given to the fluid by the immersed fibre is directed inwards to the circle's centre along its radii. The parameters used in this example are as follows: $\eta_s = 1$, $R = 0.25$, $\kappa = 1$, $N = 2$, $\Delta t = 0.005$ and the simulation was run for 600 timesteps. Note that for the purposes of this section, we let $K_b = K_b(h_f)$ as defined in (5.4.8). We consider both Stokes equations and Navier-Stokes equations as we believe it is possible that for the coarser mesh widths, the loss in area, and therefore the collapsing of the membrane, may induce a convection which shouldn't exist.

In the interest of verification, we consider the convergence with respect to mesh width. For a Stokesian fluid, Table 7.5 details the error after a single timestep, as was done by Boffi *et al.* [17], and the orders of convergence for the velocity and pressure. We can infer an order of 1.5 in the L^2 norm of the velocity and an order of 0.5 in the H^1 norm of the velocity and the L^2 norm of the pressure. Similarly, for the Navier-Stokes

h_f	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{H^1(\Omega)}$	Order	$\|p - p_N\|_{L^2(\Omega)}$	Order
1/2	0.18504	-	1.25404	-	1.40229	-
1/4	0.04274	2.11426	0.58168	1.10828	0.99751	0.49137
1/8	0.01720	1.31293	0.45462	0.35557	0.70264	0.50556
1/16	0.00660	1.38249	0.34352	0.40425	0.49579	0.50306
1/32	0.00254	1.37929	0.25911	0.40684	0.35377	0.48691
1/64	0.00092	1.46203	0.18508	0.48541	0.25265	0.48570

Table 7.5: Stokes: Order of convergence, after a single timestep, with respect to h_f when $N = 2$ and $K_b = K_b(h_f)$ for an immersed membrane.

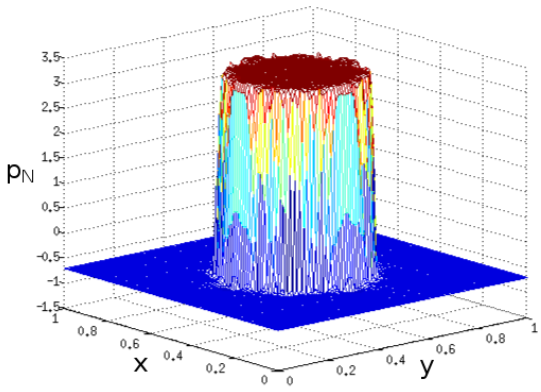
equations Table 7.6 details the error after a single timestep and the orders of convergence for the velocity and pressure. We can again infer an order of convergence of 1.5 in the L^2 norm of the velocity and an order of 0.5 in the H^1 norm of the velocity and the L^2 norm of the pressure. Similar orders of convergence, and similar errors, were

h_f	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{H^1(\Omega)}$	Order	$\|p - p_N\|_{L^2(\Omega)}$	Order
1/2	0.02453	-	0.18283	-	1.40229	-
1/4	0.01578	0.63642	0.26018	-0.50895	1.01171	0.47099
1/8	0.01162	0.44177	0.32625	-0.32649	0.72907	0.47267
1/16	0.00587	0.98511	0.30993	0.07404	0.50369	0.53351
1/32	0.00246	1.25524	0.25190	0.29909	0.35515	0.50412
1/64	0.00091	1.42860	0.18372	0.45536	0.25279	0.49047

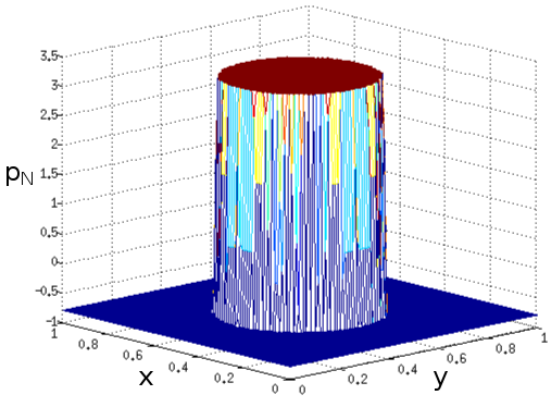
Table 7.6: Navier-Stokes: Order of convergence with respect to h_f when $N = 2$ and $K_b = K_b(h_f)$ for an immersed membrane.

inferred by Boffi *et al.* [17] (Table 1 in their article). However, we note some erroneous orders for the H^1 norm of the velocity on the coarser meshes. We believe this is due to a combination of error in the first timestep of the OIFS method discussed in §4.1.1 and an inaccurate spreading of the Lagrangian force density.

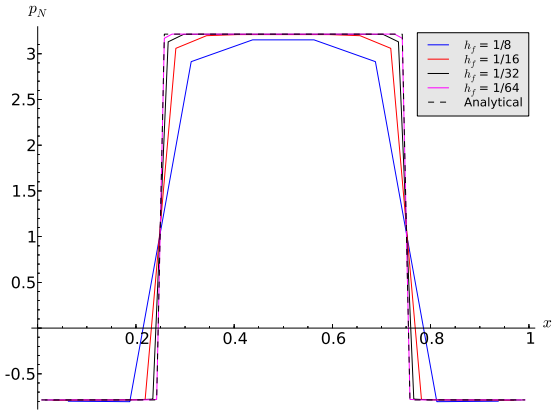
Figs. 7.5 and 7.6 illustrate the computed and exact pressure solutions and along a cut at $y = 0.5$ at the end of the simulation for both Stokes and Navier-Stokes, respectively. It is clear that we have obtained good agreement with the exact solution in both cases. Additionally, Figs. 7.7 and 7.8 show a comparison of the movement of the immersed membrane when $N = 2$ for $h_f = 1/8, 1/64$ for both Stokes and Navier-Stokes, respectively. Clearly, there is very little difference between the movement of the immersed boundary for Stokes and Navier-Stokes and that a mesh width of $h_f = 1/64$ gives much better area conservation than $h_f = 1/8$.



(a) Computed pressure solution for $h_f = 1/64$.

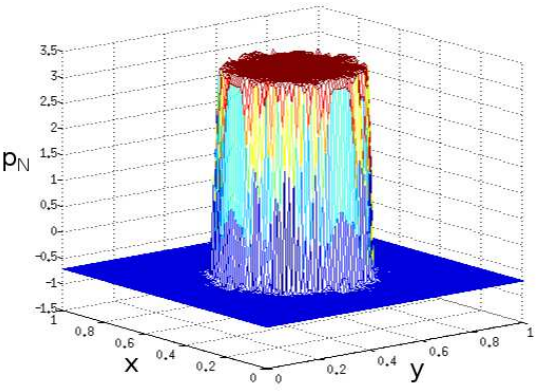


(b) Analytical pressure solution.

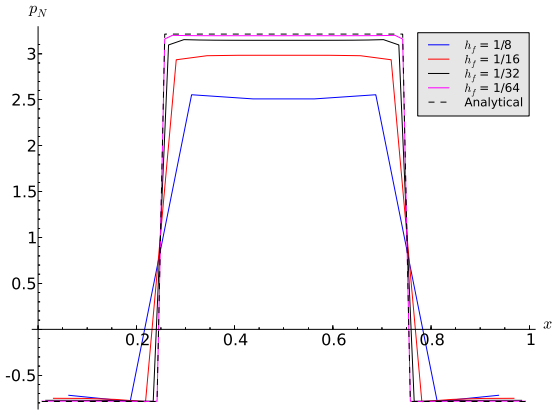


(c) Pressure solution along the cut $y = 0.5$ for $h_f = 1/8, 1/16, 1/32, 1/64$.

Figure 7.5: Stokes: Computed and analytical pressure plots, at the end of the simulation, for $N = 2$ and $K_b = K_b(h_f)$.

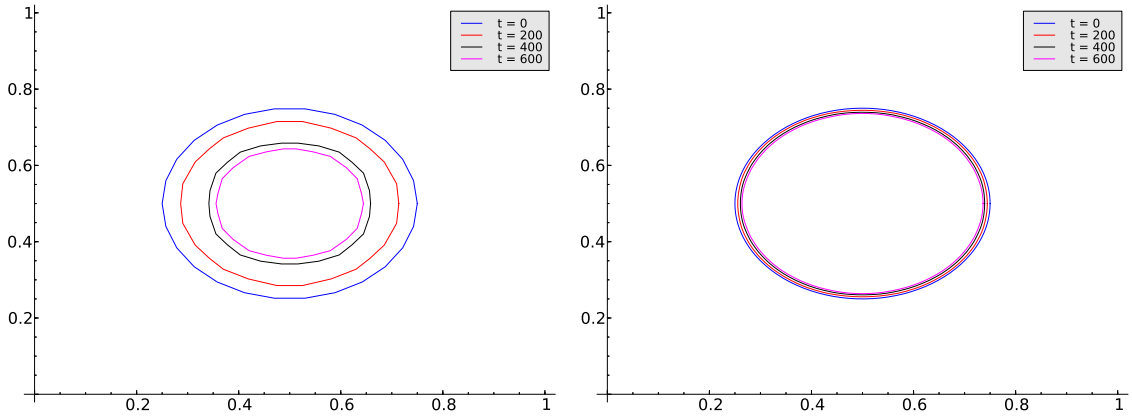


(a) Computed pressure solution for $h_f = 1/64$.



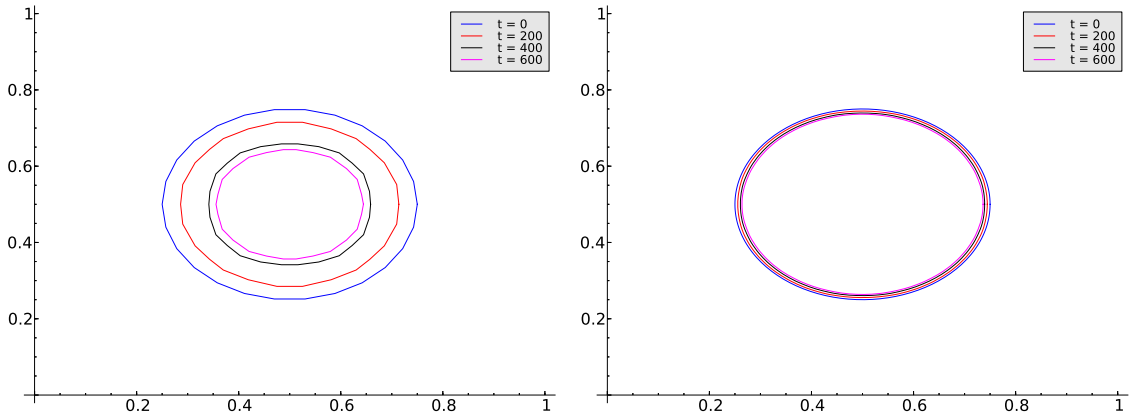
(b) Computed pressure solution along the cut $y = 0.5$ for $h_f = 1/8, 1/16, 1/32, 1/64$.

Figure 7.6: Navier-Stokes: Computed pressure plots, at the end of the simulation, for $N = 2$ and $K_b = K_b(h_f)$.



(a) $h_f = 1/8$. **(b)** $h_f = 1/64$.

Figure 7.7: Stokes: IB evolution, every 200 timesteps, when $K_b = K_b(h_f)$ and $N = 2$.



(a) $h_f = 1/8$. **(b)** $h_f = 1/64$.

Figure 7.8: Navier-Stokes: IB evolution, every 200 timesteps, when $K_b = K_b(h_f)$ and $N = 2$.

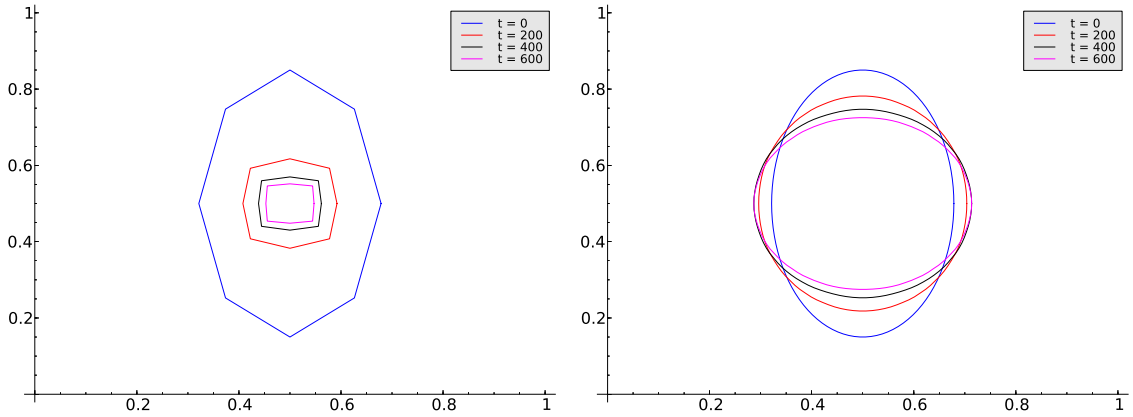
Oscillating

Another example considered by Boffi *et al.* [17], which we consider, is that of an oscillating membrane. It is a simple extension of the static case, however no analytical solution exists for such a problem, as far as we are aware. Similarly to Boffi *et al.*, we perturb the y-component of the IB position $\boldsymbol{X}$ by a small amount so that the initial configuration is an ellipse rather than a circle:

$$\boldsymbol{X}(s,0) = \begin{pmatrix} (R-\alpha)\cos\left(\dfrac{s}{R}\right) + 0.5 \\ (R+0.1)\sin\left(\dfrac{s}{R}\right) + 0.5 \end{pmatrix} \tag{7.2.8}$$

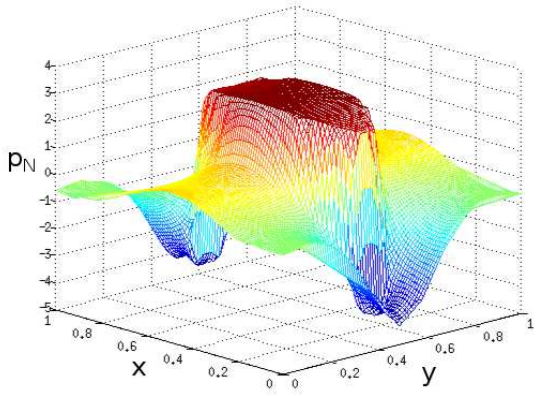
where the parameter α is chosen so that the area inside the ellipse is the same as the area in the circle for the static case; here $\alpha = 1/14$. Note that this isn't exactly the same initial condition as that considered by Boffi *et al.* as they considered $\alpha = 0$. What we expect to see here, is that the IB relaxes from an ellipse into a circle. Again, we consider both Stokes and Navier-Stokes equations here to see if the inclusion of the material derivative has any effect on the relaxing membrane.

Figs. 7.9a and 7.9b illustrate the movement of the IB at different time intervals for $h_f = 1/2$ and $h_f = 1/32$, respectively. Immediately one can see how much area is lost



(a) IB evolution for $h_f = 1/2$.

(b) IB evolution for $h_f = 1/32$.



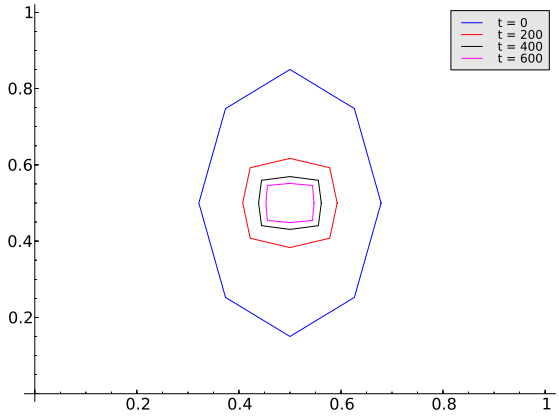
(c) Pressure solution for $h_f = 1/32$ after a single timestep.

Figure 7.9: Stokes: Immersed boundary evolution for $h_f = 1/2, 1/32$ and $N = 2$ and the pressure solution, after a single timestep, for $h_f = 1/32$.

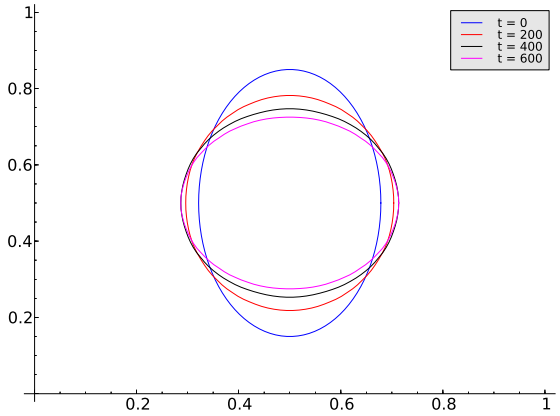
for the coarsest example. Fig. 7.9c is the computed pressure solution after a single timestep when $h_f = 1/32$. Immediately one can notice how smooth the solution is - in fact there don't appear to be any oscillations, yet the pressure solution should still be

discontinuous across the membrane. Comparing our pressure solution here to that of Boffi *et al.* [17] (their Fig. 3(b)) gives good agreement, even though we have considered a slightly different initial configuration.

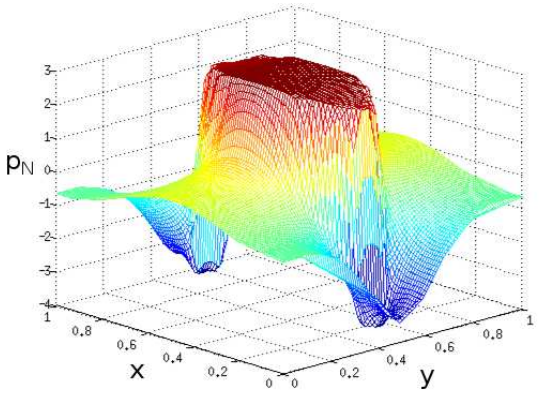
Now we consider the Navier-Stokes equations. We do not expect any difference in the results. Figs. 7.10a and 7.10b illustrate the movement of the immersed membrane. Once again, we have a large amount of area loss on the coarsest mesh. Comparing



(a) IB evolution for $h_f = 1/2$.



(b) IB evolution for $h_f = 1/32$.



(c) Pressure solution for $h_f = 1/32$ after a single timestep.

Figure 7.10: Navier-Stokes: Immersed boundary evolution for $h_f = 1/2, 1/32$ and $N = 2$ and the pressure solution, after a single timestep, for $h_f = 1/32$.

Figs. 7.9 and 7.10 shows that there is very little difference between the plots.

For the rest of the examples in this thesis, we do not use the Navier-Stokes equations but the Stokes equations as there is very little difference between the results. The only

exceptions to that are the results involving the Oldroyd-B fluid, the study of p-type convergence for the static immersed membrane and any time XSEM is used.

7.2.2 Static and Oscillating Shell

Static

The FE-IBM, and therefore the SE-IBM, has the ability to deal with *thick* immersed structures; for example, a two-dimensional structure immersed in a two-dimensional fluid. The simplest immersed thick structures we can consider, are generalisations of the two examples considered in §7.2.1 where the membrane is *thickened* to a width w. This example was considered by Boffi *et al.* [17]. Following their example, we define the reference configuration of the immersed structure as $\Omega_r^s = [0, 2\pi R] \times [0, w]$. We denote a material point in the reference configuration as $\boldsymbol{s} = (s, r)$ where $s \in [0, 2\pi R]$ is the arc-length parameter and $r \in [0, w]$ is the radial parameter. We follow Boffi *et al.*, and define the Piola-Kirchhoff stress tensor by:

$$\mathbb{P}^s = \frac{\kappa}{w} \begin{pmatrix} \dfrac{\partial X}{\partial s} & 0 \\ \dfrac{\partial Y}{\partial s} & 0 \end{pmatrix} \tag{7.2.9}$$

where $\boldsymbol{X} = (X, Y)$. It is evident from the above expression, that this example should be equivalent to the original IB with multiple fibres placed at certain $r \in [0, w]$. This means that we are assuming that the deformation of the structure occurs only in the tangential direction; in other words the deformation is fibre-like. The initial configuration of the IB position is given by:

$$\boldsymbol{X}(\boldsymbol{s}, 0) = \begin{pmatrix} (R + r)\cos\left(\dfrac{s}{R}\right) + 0.5 \\ (R + r)\sin\left(\dfrac{s}{R}\right) + 0.5 \end{pmatrix} \tag{7.2.10}$$

Note that is *not* the same initial configuration that was given by Boffi *et al.* [17]. Their initial condition implies a radial vector $\boldsymbol{r}$ of length $R(1 + w)$, where as the analytical solution they give implied a radial vector of length $R + w$. So we have edited the initial condition so that it matches the analytical solution. The analytical solution stated in

[17] as:

$$\boldsymbol{u}(\boldsymbol{x},t) = 0 \tag{7.2.11a}$$

$$p(\boldsymbol{x},t) = \begin{cases} p_0 + \dfrac{\kappa}{R} & |\boldsymbol{x}-\boldsymbol{c}| \leq R \\ p_0 + \dfrac{\kappa}{w}\dfrac{1}{R}\left(R+w-r\right) & R < |\boldsymbol{x}-\boldsymbol{c}| \leq R+w \\ p_0 & R+w < |\boldsymbol{x}-\boldsymbol{c}| \end{cases} \tag{7.2.11b}$$

where $r = |\boldsymbol{x}-\boldsymbol{c}|$, $\boldsymbol{c} = (0.5, 0.5)$ is the centre of the shell and the constant p_0 is chosen so that the zero integral pressure condition is satisifed. Therefore, p_0 is given by:

$$p_0 = \frac{\kappa\pi}{3w}\left[R^2 - \frac{(R+w)^3}{R}\right] \tag{7.2.12}$$

Note that it was shown by Boffi *et al.* [17] that the force density is again directed inwards along the radial direction. It is clear that the analytical solution is continuous everywhere and linear in the region where the immersed structure resides. The parameters chosen for this example are the same as given in [17] and are as follows: $\eta_s = 1$, $R = 0.25$, $\kappa = 1$, $w = 0.0625$, $\Delta t = 0.005$ and the simulation was run for 600 timesteps. Just as was done previously for the case of an immersed membrane, we must choose the number of structure elements so that $h^s \leq h_f/2$. In this section, we assume that the total number of elements is given by $K_b = K_b^s \times K_b^r$, where $K_b^s = K_b^s(h_f)$ is calculated using the relation in (5.4.8) and K_b^r is calculated using:

$$K_b^r \geq \frac{2w}{h_f} \tag{7.2.13}$$

These values of K_b^s and K_b^r ensure that the decomposition of Ω_r^s is composed of approximately uniform quadrilateral elements. Again, the analytical solution given in (7.2.11) is the analytical solution of Stokes flow. Therefore, here we consider Stokes flow rather than Navier-Stokes.

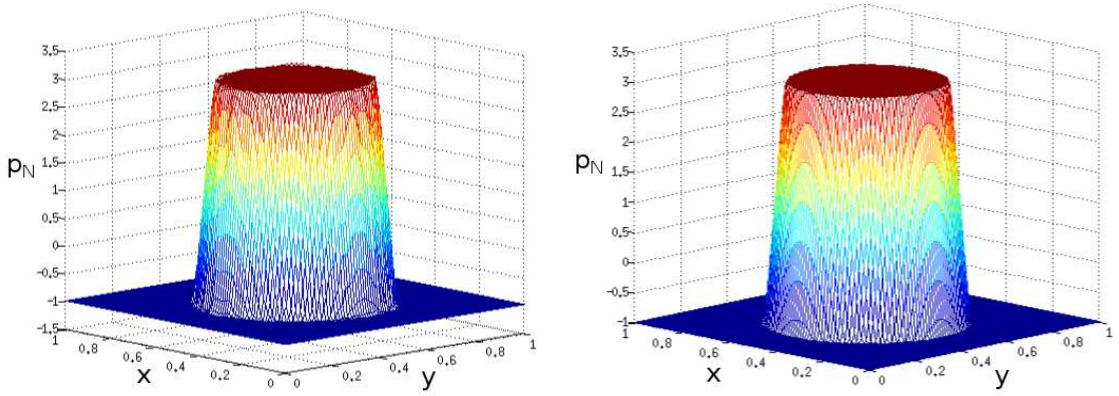
Table 7.7 details the error after a single timestep, as was done by Boffi *et al.* [17], and orders of convergence for the velocity and pressure solutions. We can infer that the order of convergence for the velocity with respect to the L^2 and H^1 norms is approximately 2 and 1, respectively. Our orders of convergence are lower than that found by

h_f	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{H^1(\Omega)}$	Order	$\|p - p_N\|_{L^2(\Omega)}$	Order
1/2	0.20474	-	1.37396	-	1.73205	-
1/4	0.03691	2.47169	0.49431	1.47484	0.81880	1.08089
1/8	0.01904	0.95512	0.48566	0.02548	0.61515	0.41257
1/16	0.00724	1.39482	0.34108	0.50982	0.34656	0.82784
1/32	0.00196	1.88576	0.18088	0.91506	0.18187	0.93017
1/64	0.00051	1.93991	0.09342	0.95324	0.09351	0.95972

Table 7.7: Order of convergence, after a single timestep, with respect to h_f when $N = 2$ for a shell immersed in a Stokesian fluid.

Boffi *et al.* [17] who inferred an order of 2.5 in the L^2 norm and 1.5 in the H^1 norm for the velocity. The pressure solution can be seen to have an order of approximately 1 which again is lower than the order inferred by Boffi *et al.* [17], who inferred an order of 1.5. We believe the reduced order of convergence we see here is related to the discretisation of the immersed shell. We believe that if we increased the number of marker particles that make up the shell, we would see better order of convergence.

Figs. 7.11a and 7.11b show the computed and analytical pressure solutions when $h_f = 1/64$. Due to the analytical solution being continuous, we should *not* see any oscillations local to the interface. Fig. 7.11a appears to show that there are in fact very



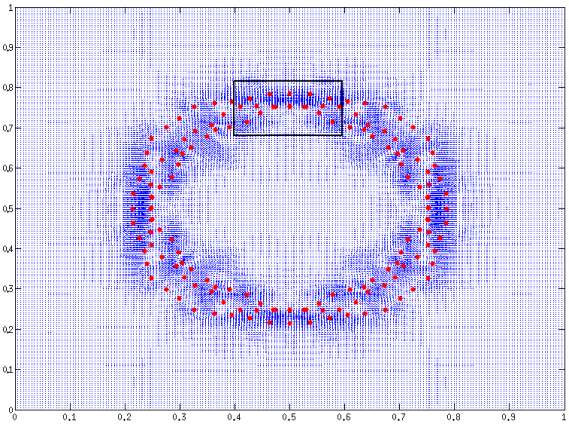
(a) Computed pressure solution. **(b)** Analytical pressure solution.

Figure 7.11: Pressure plots, at the end of the simulation, for $h_f = 1/64$ and $N = 2$.

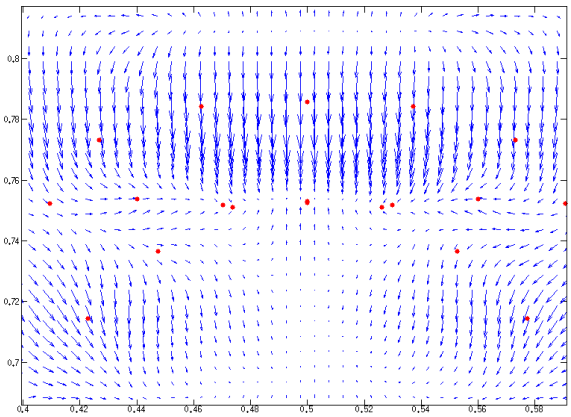
small oscillations local to the interface. We believe these oscillations are present due to the linear interpolation to a uniform grid computed in MATLAB in order to produce the plots. It is clear that the computed solution is a very good match of the analytical solution. Incidentally, the computed pressure solution given by Boffi *et al.* [17] contains fairly large oscillations local to the interface (see their Fig. 4f). It is possible that these are present due to the initial condition not matching their analytical solution, or it could be that they used higher-order splines to obtain the three-dimensional plots of the pressure solution.

Fig. 7.12 shows the velocity vector at the end of the simulation for $h_f = 1/16, 1/32$ and $1/64$. First we notice that the pattern of the velocity field does not really change, it just becomes more local. This is the same behaviour that we saw in the case of an immersed membrane, Fig. 5.10. Clearly, on the coarser mesh $h_f = 1/16$, Fig. 7.12a, there is separation in the immersed structure marker particles. This is clearer in the zoomed plot in Fig. 7.12b. This is due to the large local velocity field around each marker particle. It is evident from Figs. 7.12c and 7.12e that the separation on the diagonal is improved as the mesh is refined. However, the separation along the axes still takes place and it is clear from Figs. 7.12d and 7.12f that the local velocity field is pushing the marker particles apart. We believe that this is the reason for the sub-optimal order of convergence seen in Table 7.7. Even the finest mesh $h_f = 1/64$ has this behaviour although it is considerably less than the coarser meshes. In our opinion, this also illustrates a weakness in the immersed boundary method modelling thick structures. The movement of the immersed boundary method is achieved on a node by node basis and therefore, each marker particle will move independently from one another with the local velocity field and this is the reason why the structure can separate in the manner seen in Fig. 7.12.

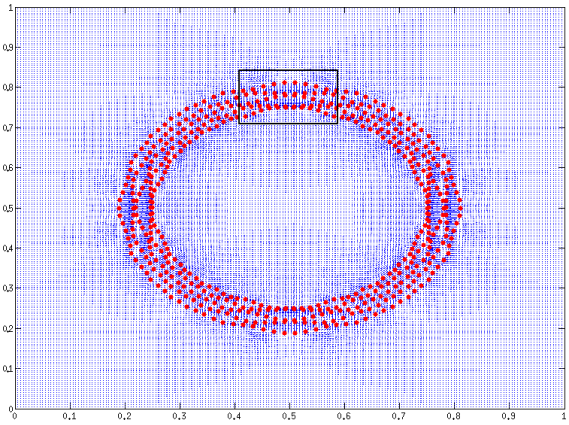
Finally, Fig. 7.13 shows the initial and final configuration of the immersed boundary for $h_f = 1/16, 1/32$ and $1/64$ when $N = 2$. Notice the separation of the immersed structure along the axes for Figs. 7.13b and 7.13c. Later we will see what effect the higher-order polynomial will have on this movement and the velocity field.



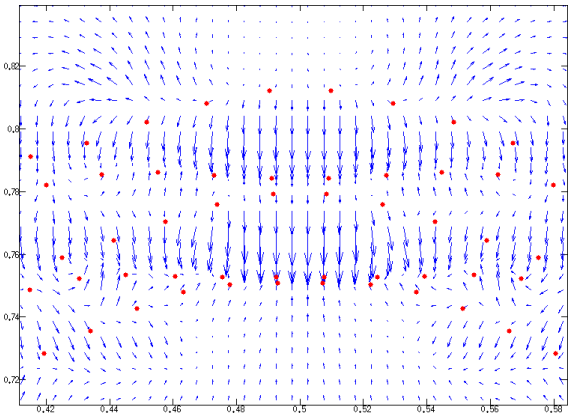
(a) Velocity vector plot for $h_f = 1/16$.



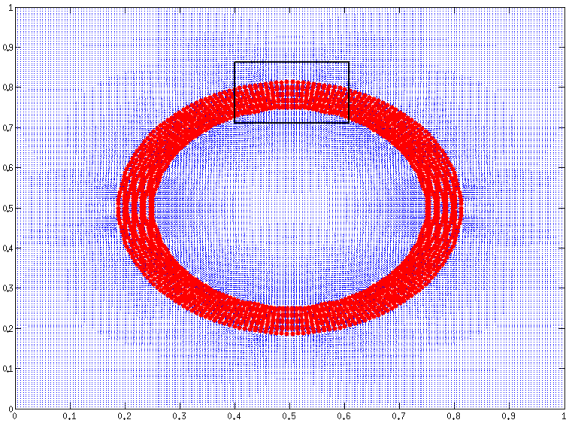
(b) Zoomed velocity vector plot for $h_f = 1/16$.



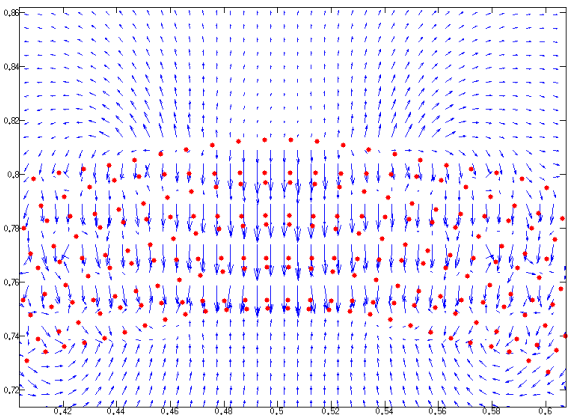
(c) Velocity vector plot for $h_f = 1/32$.



(d) Zoomed velocity vector plot for $h_f = 1/32$.

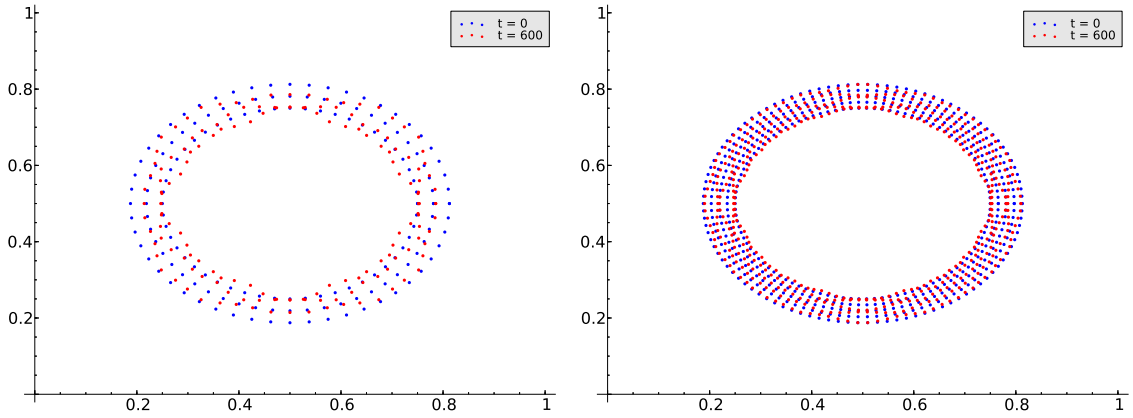


(e) Velocity vector plot for $h_f = 1/64$.



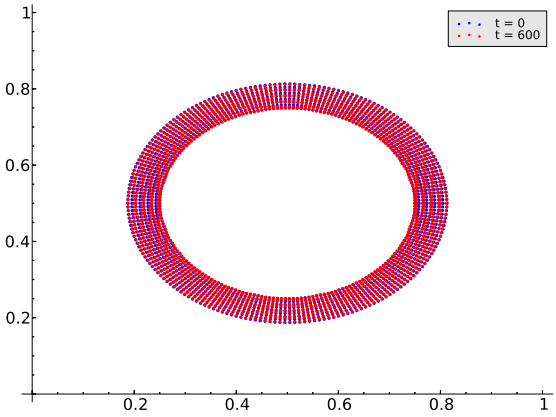
(f) Zoomed velocity vector plot for $h_f = 1/64$.

Figure 7.12: Velocity vectors when $N = 2$ for decreasing mesh width. The red dots on each plot are the final position of the IB marker particles. The black box is the region which is shown in the zoomed plots.



(a) IB evolution for $h_f = 1/16$.

(b) IB evolution for $h_f = 1/32$.



(c) IB evolution for $h_f = 1/64$.

Figure 7.13: Immersed shell evolution for $N = 2$ and $h_f = 1/16, 1/32, 1/64$.

Oscillating

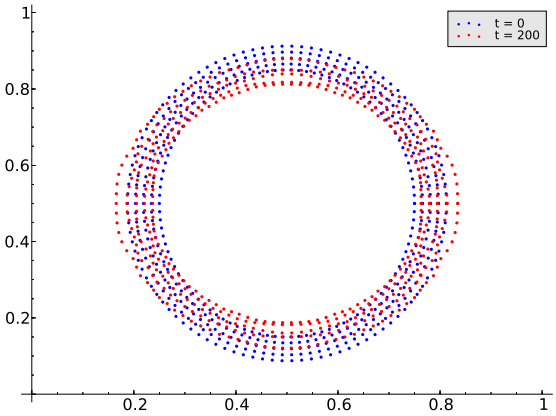
Again we consider a simple extension of the static shell to an oscillating shell. In §7.2.1, we considered a perturbation of the original static example so that the area contained inside the ellipse was the same as that for the static circle. However, here we consider a similar perturbation to that considered by Boffi *et al.* [17]:

$$\boldsymbol{X}(\boldsymbol{s}, 0) = \begin{pmatrix} (R + r)\cos\left(\dfrac{s}{R}\right) + 0.5 \\ (R + 0.1 + r)\sin\left(\dfrac{s}{R}\right) + 0.5 \end{pmatrix} \tag{7.2.14}$$

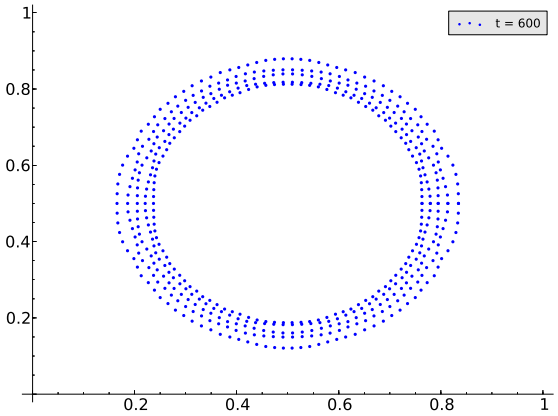
Again, we point out that our initial condition is slightly different to that of Boffi *et al.*. Again the parameters chosen are $R = 0.25$, $\kappa = 1$, $w = 0.0625$, $\Delta t = 0.005$ and the simulation was run for 600 timesteps. Boffi *et al.* [17] let $\eta_s = 0.005$ which we

were unable to consider as the computation broke down. As we discussed in §4.2, it is known within the IB literature that the method is unstable for small viscosity or large forcing parameters. Therefore, Boffi *et al.* [17] must have decreased their timestep but, unfortunately, there is no mention in their article about whether or not the timestep is changed. Therefore, in this section we keep $\eta_s = 1$.

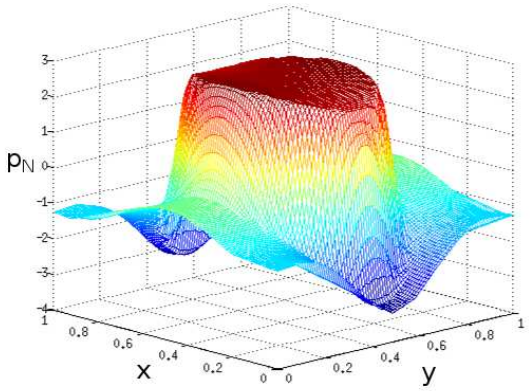
Figures 7.14a and 7.14b illustrate the evolution of the immersed boundary configuration initially and at the end of the simulation (when $T = 3$). Fig. 7.14a illustrates



(a) IB evolution for $h_f = 1/32$ at $T = 0$ and $T = 3$.



(b) IB evolution for $h_f = 1/32$ when $T = 3$.



(c) Pressure solution for $h_f = 1/32$ after a single timestep.

Figure 7.14: Immersed shell evolution for $h_f = 1/32$ and $N = 2$ and the pressure solution, after a single timestep, when $h_f = 1/32$.

that the shell is beginning to relax towards a circular shape. We can see from Fig. 7.14b that around $x = 0.2$ and 0.8, the structure at the end of the simulation has some separation between the marker particles. We saw similar behaviour for the static shell.

It is possible that increasing the number of marker particles on the shell will eliminate this behaviour.

Fig. 7.14c illustrates the pressure solution after a single timestep. For the case of the static closed shell, the pressure solution was continuous and therefore in the oscillating example, we still expect the pressure to be continuous. It is clear from Fig. 7.14c that the computed solution is smooth and hence no oscillations can be seen.

In this section, we have validated our immersed boundary solver against the work of Boffi *et al.* [17]. We found good qualitative and quantitative agreement with Boffi *et al.* as well as the analytical solutions. In the next section, we apply the SE-IBM to the same sample problems but we use a higher-order polynomial.

Chapter 8

Numerical Calculations: Newtonian Fluids

In this chapter we consider the p-type convergence of the same example problems discussed in Chapter 7. Also, in the case of an immersed membrane, we apply the XSEM approximation of the pressure.

8.1 Static Closed Membrane

8.1.1 SEM

As our interest is with the spectral element method, we wish to consider the orders of convergence with respect to the polynomial degree N. We let the number of immersed boundary marker particles K_b be dependent on N, as in (5.4.9) because, as discussed in §5.4.2, better area conservation is seen. We wish to draw a comparison with the results given in §7.2 and therefore, we consider both the Stokes and Navier-Stokes equations.

For a Stokesian fluid, Table 8.1 details the error after a single timestep and orders of convergence for the velocity and pressure solutions. We can infer that for a mesh

N	$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{H^1(\Omega)}$	Order	$\|p-p_N\|_{L^2(\Omega)}$	Order
2	0.01720	-	0.45144	-	0.70314	-
4	0.00206	3.06120	0.15411	1.55056	0.33923	1.05155
8	0.00042	2.30616	0.07032	1.13189	0.22521	0.59099
16	0.00010	2.12344	0.03505	1.00452	0.14737	0.61186

Table 8.1: Stokes: Orders of convergence, after a single timestep, with respect to N when $h_f = 1/8$ for an immersed membrane.

width $h_f = 1/8$, the order of convergence for the velocity with respect to the L^2 and H^1 norms is approximately 2 and 1, respectively. The orders of convergence for the velocity are higher than those found previously (Table 7.5) and higher than those by Boffi *et al.* [17]. However, due to the irregularity of the pressure solution, we do not expect SE-IBM to perform any better than the FE-IBM. Indeed, we find that the order of convergence for the pressure is approximately 0.5 which is the same as inferred previously (Table 7.5) and by Boffi *et al.* [17].

Similarly, for the Navier-Stokes equations, Table 8.2 details the error after a single timestep and orders of convergence for velocity and pressure. We can infer that for a

N	$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{H^1(\Omega)}$	Order	$\|p-p_N\|_{L^2(\Omega)}$	Order
2	0.01159	-	0.32233	-	0.72812	-
4	0.00195	2.57209	0.14631	1.13953	0.33871	1.10415
8	0.00041	2.24182	0.06959	1.07203	0.22521	0.58875
16	0.00010	2.11085	0.03497	0.99264	0.14737	0.61187

Table 8.2: Navier-Stokes: Orders of convergence, after a single timestep, with respect to N when $h_f = 1/8$ for an immersed membrane.

mesh width $h_f = 1/8$, the order of convergence for the velocity with respect to the L^2 and H^1 norms is approximately 2 and 1, respectively. For the L^2 norm of the pressure we can again infer an order of 0.5. These agree with the results of Stokes flow given in Table 8.1. We note that we do not see erroneous orders of convergence for the H^1

norm in this case, most likely due to the mesh being sufficiently fine for there not to be a problem.

The orders of convergence for the velocity found in Tables 8.1 and 8.2 are optimal and it appears that the regularity of the pressure solution is not affecting the velocity order of convergence as was seen by Boffi *et al.* [17]. To check this, we ran the same example on a coarser mesh but with a higher value for the polynomial degree N. For Stokes flow, Table 8.3 shows the order of convergence of the velocity and pressure solutions when $h_f = 1/2$ and $N = 2, 4, 8, 16, 32$. It is clear that we can infer an order of 2 and

N	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{H^1(\Omega)}$	Order	$\|p - p_N\|_{L^2(\Omega)}$	Order
2	0.18909	-	1.25811	-	1.40229	-
4	0.02781	2.76525	0.47229	1.41351	0.83724	0.74407
8	0.00544	2.35471	0.20530	1.20193	0.56314	0.57214
16	0.00122	2.16191	0.09508	1.11048	0.36653	0.61957
32	0.00029	2.09135	0.04565	1.05863	0.25108	0.54577

Table 8.3: Stokes: Orders of convergence, after a single timestep, with respect to N when $h_f = 1/2$ for an immersed membrane.

1 for the L^2 and H^1 norms of the velocity, respectively. We can also infer an order of approximately 0.5 for the L^2 norm of the pressure. Clearly, this shows that the irregularity of the pressure solution is *not* affecting the convergence rate when a high order approximation is used.

Similarly, for the Navier-Stokes equations, Table 8.4 shows the order of convergence for the velocity and pressure solutions. We can infer the same orders of convergence for the velocity and pressure as was from inferred from Table 8.3. However, we note that we have an erroneous order of convergence for the H^1 norm of the velocity. Again, we believe this is due to the error in the spreading of the Lagrangian force density to the surrounding fluid grid.

N	$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{H^1(\Omega)}$	Order	$\|p-p_N\|_{L^2(\Omega)}$	Order
2	0.02405	-	0.16884	-	1.40229	-
4	0.01354	0.82932	0.23442	-0.47349	0.80319	0.80396
8	0.00446	1.60321	0.17006	0.46311	0.56050	0.51903
16	0.00115	1.94793	0.09066	0.90743	0.36631	0.61363
32	0.00028	2.03608	0.04512	1.00671	0.25106	0.54505

Table 8.4: Navier-Stokes: Orders of convergence, after a single timestep, with respect to N when $h_f = 1/2$ for an immersed membrane.

According to Maday, Patera and Rønquist [49] (their Theorem 3.8) and Bernardi and Maday [7, p. 152] (their Theorem 7.7), for a velocity $\boldsymbol{u} \in H^m(\Omega)^2$ and a pressure $p \in H^{m-1}(\Omega)$, the approximation error can be given by:

$$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{H^1(\Omega)^2} + N^{-1/2}\|p-p_N\|_{L^2(\Omega)} \leq cN^{1-m}\left(\|\boldsymbol{u}\|_{H^m(\Omega)^2} + \|p\|_{H^{m-1}(\Omega)}\right) \quad (8.1.1)$$

from which it can be inferred:

$$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{L^2(\Omega)^2} \leq cN^{-m}\left(\|\boldsymbol{u}\|_{H^m(\Omega)^2} + \|p\|_{H^{m-1}(\Omega)}\right) \quad (8.1.2a)$$

$$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{H^1(\Omega)^2} \leq cN^{1-m}\left(\|\boldsymbol{u}\|_{H^m(\Omega)^2} + \|p\|_{H^{m-1}(\Omega)}\right) \quad (8.1.2b)$$

$$\|p-p_N\|_{L^2(\Omega)} \leq cN^{\frac{3}{2}-m}\left(\|\boldsymbol{u}\|_{H^m(\Omega)^2} + \|p\|_{H^{m-1}(\Omega)}\right) \quad (8.1.2c)$$

From these results we can see that the order of convergence for the velocity if $m = 2$ is 2 and 1 for the L^2 and H^1 norms, respectively, and 0.5 for the L^2 norm of the pressure. This is exactly what we have inferred in the tables above. However, our exact pressure solution is *not* a H^1 function. Therefore, according to the estimates given above, we would expect the order of the velocity and pressure convergence to be impaired due to the irregularity of the pressure solution.

The question then is: Why do we see second-order and first-order convergence rates for the L^2 and H^1 norms of the velocity when the pressure solution is *not* in H^1? Formally, we believe the reason we see optimal convergence rates here is because of the improved

approximation of the incompressibility constraint. By increasing N, we are increasing the number of points inside the support of the Dirac delta function and that this, as was already discussed in §5.4.2, yields better area conservation - or equivalently, a more accurate divergence-free constraint. Let $\mathcal{V}^{div}$ denote the space of functions from $\mathcal{V} = H_0^1(\Omega)^2$ which are divergence-free, i.e.:

$$\mathcal{V}^{div} = \{\boldsymbol{v} \in \mathcal{V} \colon \nabla \cdot \boldsymbol{v} = 0\} \tag{8.1.3}$$

and assume that our exact solution $\boldsymbol{u} \in \mathcal{V}^{div}$. Therefore, the discrete velocity $\boldsymbol{u}_N$ belongs to:

$$\mathcal{V}_N^{div} = \{\boldsymbol{v}_N \in \mathcal{V}_N \colon (\nabla \cdot \boldsymbol{v}_N, q_N)_N = 0,\ \forall q_N \in \mathcal{Q}_N\} \tag{8.1.4}$$

According to Bernardi and Maday [8], $\boldsymbol{u}_N$ is therefore a solution of:

$$(\nabla \boldsymbol{u}_N, \nabla \boldsymbol{v}_N)_N = (\boldsymbol{f}, \boldsymbol{v}_N)_N \quad \forall \boldsymbol{v}_N \in \mathcal{V}_N \tag{8.1.5}$$

so that the error estimate involves the regularity of $\boldsymbol{u}$ and can be given by, [8]:

$$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{L^2(\Omega)^2} \leq cN^{-m} \|\boldsymbol{u}\|_{H^m(\Omega)^2} \tag{8.1.6a}$$

$$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{H^1(\Omega)^2} \leq cN^{1-m} \|\boldsymbol{u}\|_{H^m(\Omega)^2} \tag{8.1.6b}$$

$$\|p - p_N\|_{L^2(\Omega)} \leq cN^{\frac{3}{2}-m} \left(\|\boldsymbol{u}\|_{H^m(\Omega)^2} + \|p\|_{H^{m-1}(\Omega)} \right) \tag{8.1.6c}$$

In such a case, the error estimates on the velocity are mainly related to the exact velocity solution rather than the pressure. Boffi *et al.* [17] concluded that their impaired order of convergence for the velocity was because of the pressures irregularity. However, from a numerical perspective, the divergence-free constraint is solved implicitly. Therefore, the discrete velocity solution should always be a member of $\mathcal{V}_N^{div}$. In the immersed boundary method, the main reason for a loss in area/volume (or error in the divergence-free constraint) is because the interpolated velocity $\boldsymbol{u}_N(\boldsymbol{X}_h) \notin \mathcal{V}_N^{div}$ (or equivalently errors in the spreading and interpolation phases). Therefore, the results given here suggest that the higher-order polynomial reduces the error in the divergence of the interpolated velocity sufficiently so that the numerics can obtain velocity orders which are independent of the pressures regularity. We believe this is the reason why the regularity of the exact pressure does not affect the velocity convergence rate. However,

what we cannot explain, is the reason why the order of convergence for the pressure is *not* impaired by the lack of regularity on the exact solution. As far as we are aware, error estimates for the spectral approximation of the pressure from a broken Sobolev space (as is the case here) do not currently exist.

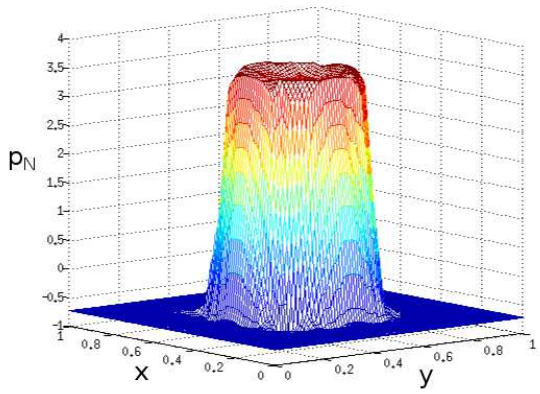
Fig. 8.1 compares the computed pressure solution for $h_f = 1/8$ and $N = 4, 8, 16$ against the analytical solution. It is clear that we have obtained good agreement with the analytical solution. As N is increased, the Gibbs phenomenon becomes more apparent.

Fig. 8.2 illustrates that we still obtain the solution when $N = 32$ and $h_f = 1/2$. Again, good agreement with the analytical solution in Fig. 8.1d can be seen. However, there are quite a few oscillations away from the membrane which we believe are caused by the Gibbs phenomenon as the mesh is very coarse. These oscillations can be seen further in Fig. 8.2b (the pink line).

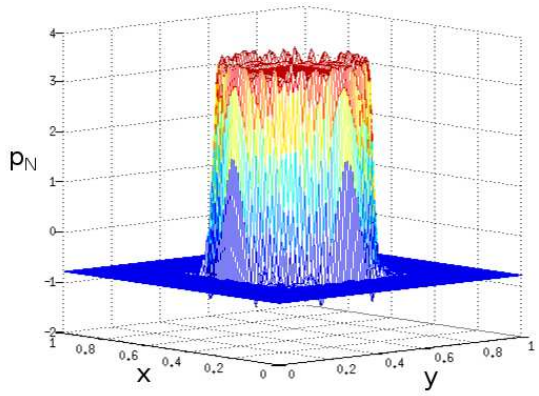
Additionally, Fig. 8.3 illustrates the good area conservation that is obtained when using high order polynomials, even on very coarse meshes. Comparing Figs. 8.1 and 8.2 with those in the work of Boffi *et al.* gives good agreement.

8.1.2 XSEM

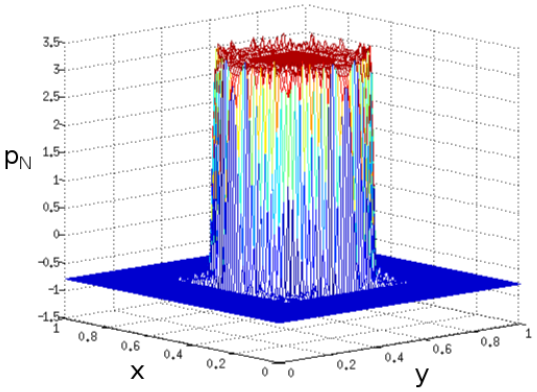
In this section we look at the influence of the XSEM approximation of the pressure. Throughout this section, we have used the OIFS method that we discussed in §4.1.1 to approximate the material derivative that appears in the momentum equation and we take a mesh width of $h_f = 1/8$ and $K_b = K_b(N)$. Also, in this section we employ the non-dimensional formulation (2.2.14) rather than the dimensional form. We choose the Reynolds number $Re = 1$. Additionally, the number of quadrature points, used for the numerical integration of the extended part of the enriched approximation p_N^X, is given by: $N_q = 2N + 2$. Including XSEM, introduces two additional matrices to the linear system. These matrices correspond to the additional degrees of freedom (α) and the



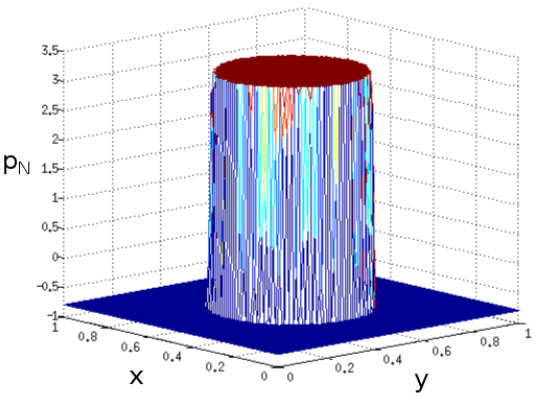
(a) Computed pressure solution for $N = 4$.



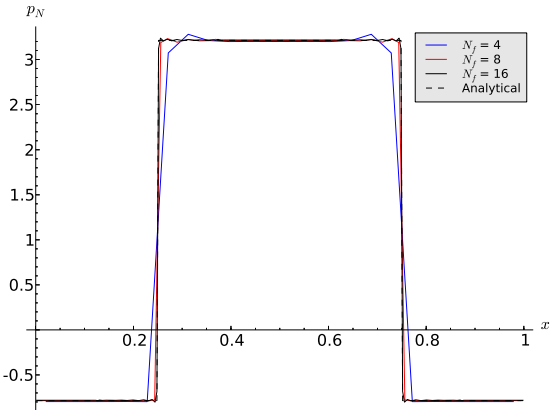
(b) Computed pressure solution for $N = 8$.



(c) Computed pressure solution for $N = 16$.



(d) Analytical pressure solution.

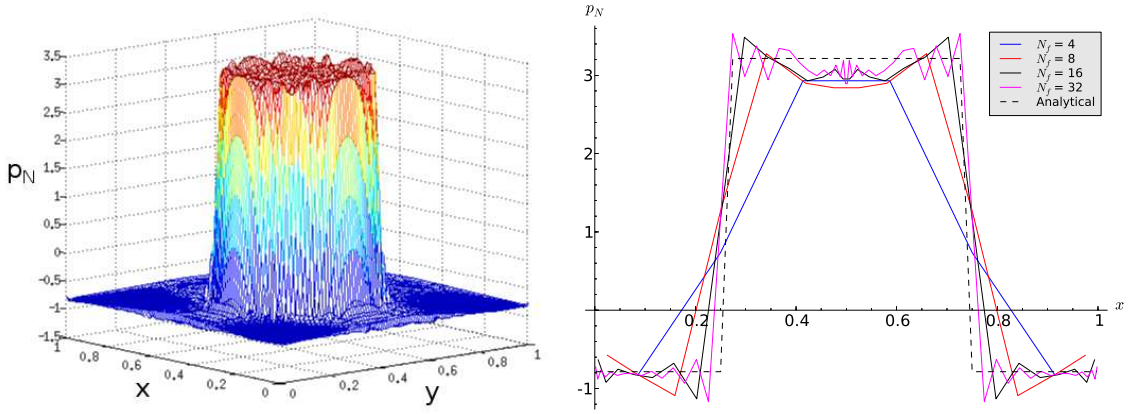


(e) Computed pressure solution along the cut $y = 0.5$ for $N = 8, 16, 32$.

Figure 8.1: Stokes: Computed and analytical pressure plots, at the end of the simulation, for increasing N when $h_f = 1/8$.

linear system becomes:

$$\begin{pmatrix} A & B^T & C^T \\ B & 0 & 0 \\ C & 0 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{u} \\ p \\ \alpha \end{pmatrix} = \begin{pmatrix} \boldsymbol{f} \\ g \\ \widehat{g} \end{pmatrix} \tag{8.1.7}$$



(a) Computed pressure solution.

(b) Computed pressure solution along the cut $y = 0.5$ for $N = 4, 8, 16, 32$.

Figure 8.2: Stokes: Computed and analytical pressure plots, at the end of the simulation, for $h_f = 1/2$.

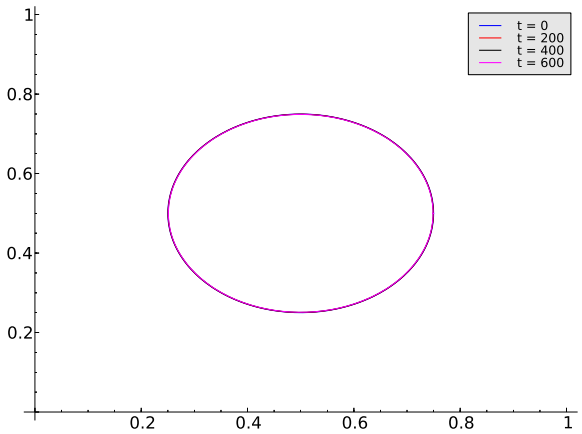
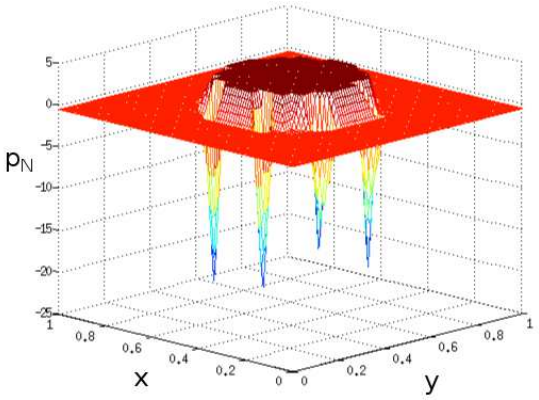


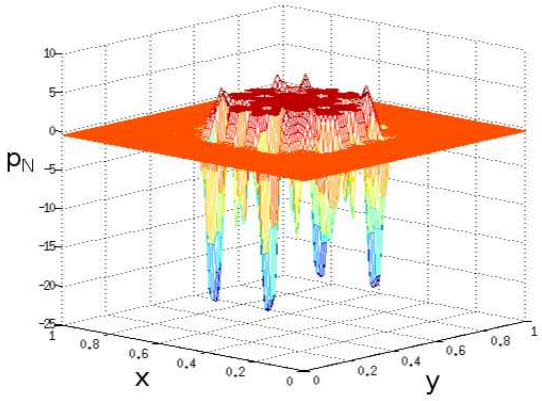
Figure 8.3: IB evolution for $N = 32$ and $h_f = 1/2$.

These additional matrices are very sparse and increase the condition number quite significantly. Additionally, we were unable to use a preconditioner as using the diagonal preconditioner used throughout this thesis gave spurious results when applied to XSEM. We wish to investigate this further in the future.

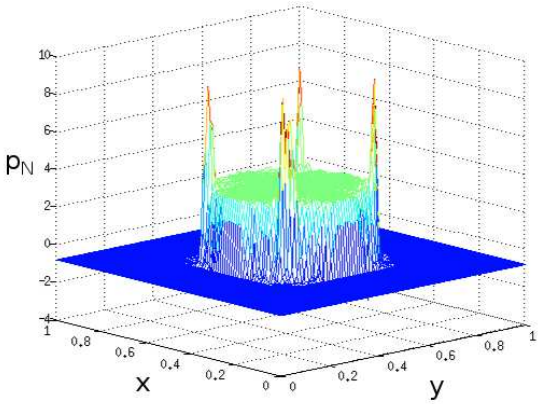
To check that we have obtained the analytical solution, we plot the computed pressure solution for $N = 2, 4$ and 8 in Fig 8.4. The enriched approximation is given by: $p_N^\Gamma = p_N + p_N^X$; the left hand column in Fig. 8.4 illustrates the standard part of the enriched approximation (p_N) and the right hand column illustrates the full enriched approximation (p_N^Γ). Clearly when $N = 4$ there are some spurious peaks in both the



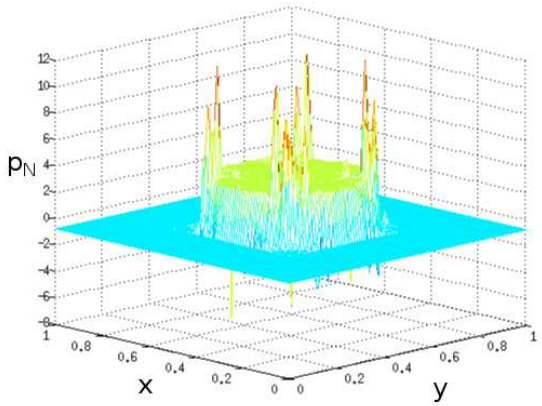
(a) p_N for $N = 4$.



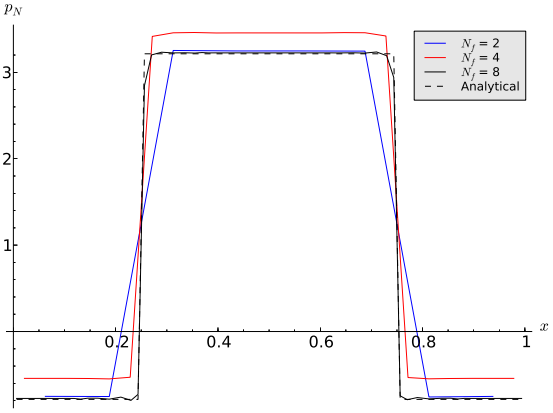
(b) p_N^Γ for $N = 4$.



(c) p_N for $N = 8$.



(d) p_N^Γ for $N = 8$.



(e) p_N along the cut $y = 0.5$ for $N = 2, 4, 8$.

Figure 8.4: Computed pressure plots for $h_f = 1/8$.

standard part and the enriched solution. As we discussed in §6.2, we believe these spurious peaks are an error due to a lack of enrichment but could also be related to the regions of small support. Obviously, these peaks are going to interfere with the pressure error. There are also peaks present when $N = 8$ and it is known from §6.2

that these are in fact due to a lack of enrichment caused by not enough quadrature points. Note also from Fig. 8.4d that blending is a problem when $N = 8$. Eliminating the blending problems is a subject of future work. The plot of the pressure p_N along a cut $y = 0.5$ given in Fig. 8.4e shows that very few oscillations are present local to the membrane. In fact, if we ignore all of the spurious peaks in the pressure plots, it is clear that most of the oscillations due to Gibbs phenomenon have been removed.

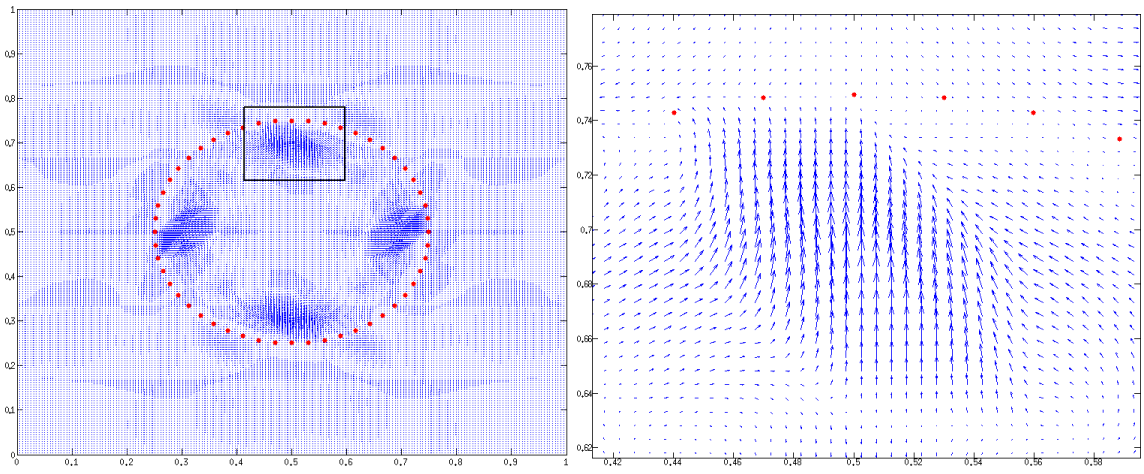
Table 8.5 details the error after a single timestep and orders of convergence for the velocity solution and the pressure solution p_N . For the SEM approximation of the pressure, the orders of convergence for the velocity and pressure in the L^2 norm were 2 and 0.5, respectively. First of all we notice how much better the velocity error is

N	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{H^1(\Omega)}$	Order	$\|p - p_N\|_{L^2(\Omega)}$	Order
2	0.00067	-	0.02890	-	0.76630	-
4	0.00011	2.60440	0.01116	1.37353	3.24687	-2.08306
8	0.00003	1.72818	0.00914	0.28793	0.32272	3.33069

Table 8.5: Orders of convergence, when XSEM is used to approximate the pressure, with respect to N when $h_f = 1/8$ for an immersed membrane.

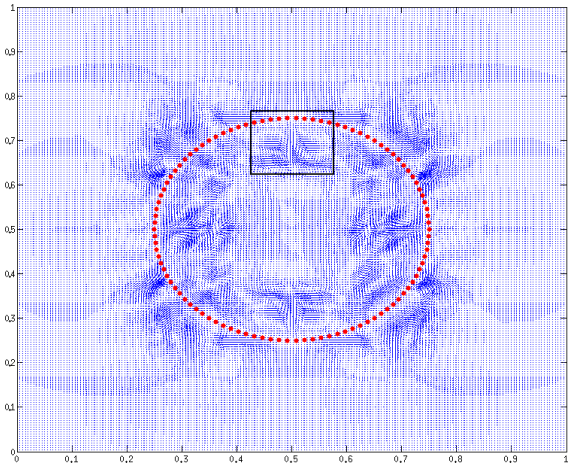
in the L^2 and H^1 norms in comparison to the standard SEM approximation in Table 8.2. This was to be expected, as the use of an enriched method will greatly reduce the spurious velocities that are seen local to the membrane. However, the pressure error is behaving strangely. It is not surprising, given the figures above, that the error increases when $N = 4$. However, as we increase $N = 4$ to 8 we see very good order of convergence. Thus it is possible that considering a higher value for N_q and a higher value for N would yield better order of convergence for p_N.

We are interested to see the effect that using XSEM to approximate the pressure has on the velocity field. Fig. 8.5 illustrates the velocity vectors for $h_f = 1/8$ and $N = 2, 4, 8$ when $Re = 1$. It is clear how different the velocity field is in Fig. 8.5a when compared to Fig. 5.10c. The magnitude of the velocity field for all plots in Fig. 8.5 is clearly a lot less when XSEM is used to approximate the pressure. This is as expected as

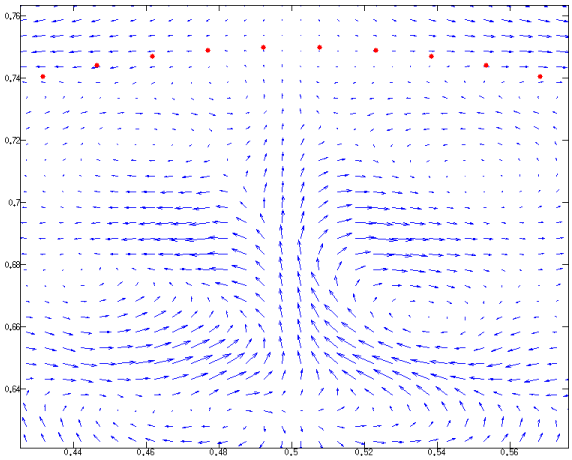


(a) Velocity vector for $N = 2$.

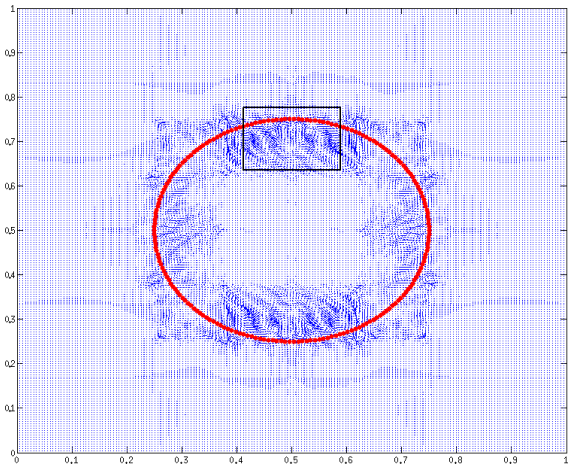
(b) Velocity vector for $N = 2$.



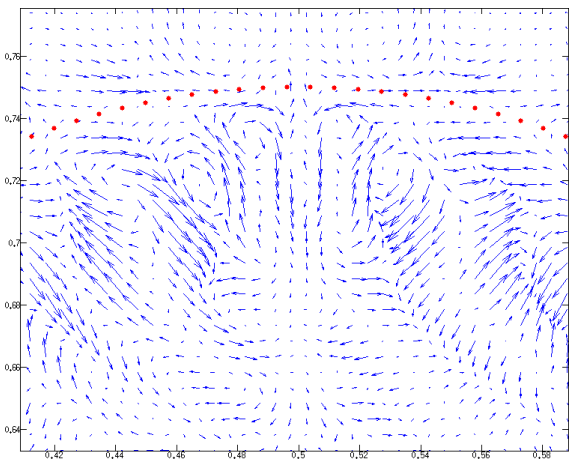
(c) Velocity vector for $N = 4$.



(d) Velocity vector for $N = 4$.



(e) Velocity vector for $N = 8$.



(f) Velocity vector for $N = 8$.

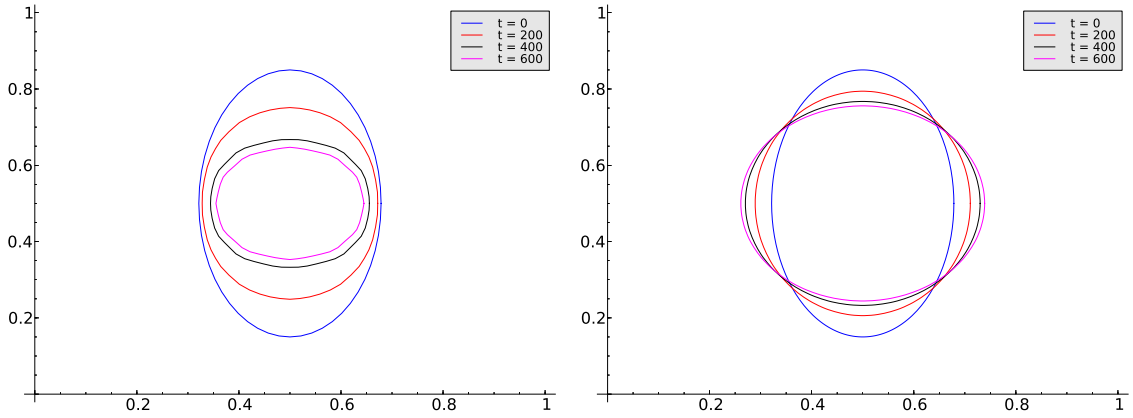
Figure 8.5: Comparison of the velocity vector for $h_f = 1/8$, $N = 2, 4, 8$ when XSEM is used to approximate the pressure.

the discontinuity in the pressure causes spurious velocities. Therefore, if the pressure solution is enriched we expect to see a reduction in spurious velocities.

8.2 Oscillating Closed Membrane

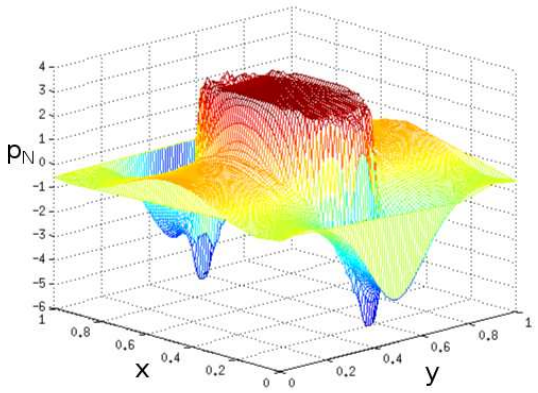
8.2.1 SEM

In this section, we apply the SE-IBM to the oscillating immersed membrane that we considered in §7.2.1. Fig. 8.6 illustrates the movement of the immersed boundary for $h_f = 1/8$ when $N = 2$ and 8 and also the pressure solution after a single timestep. Clearly we see the loss in area when $N = 2$ that we expect to see for such a coarse



(a) IB evolution for $N = 2$.

(b) IB evolution for $N = 8$.



(c) Pressure solution for $N = 8$ after a single timestep.

Figure 8.6: Immersed boundary evolution for $h_f = 1/8$ and $N = 2, 8$ and the pressure solution for $h_f = 1/8$ when $N = 8$.

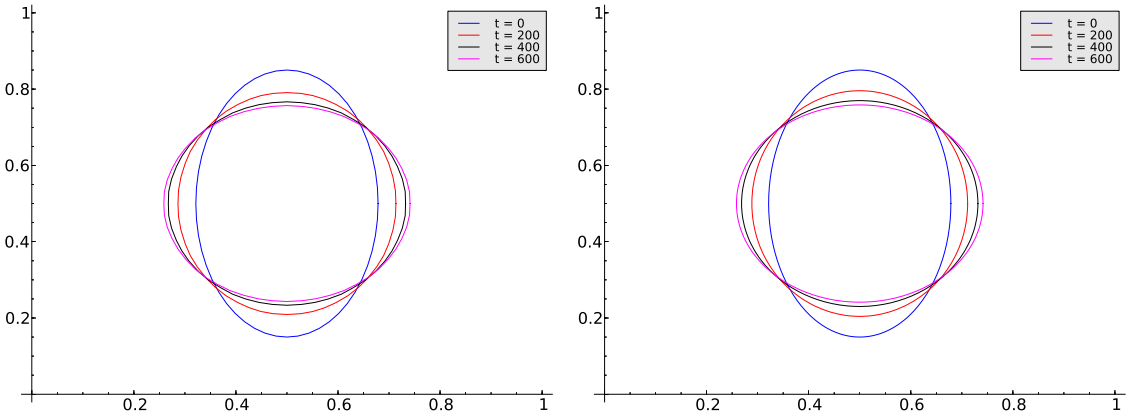
approximation. Comparing Fig. 8.6c to 7.9c shows that the solutions are very similar except that when a high order polynomial is used, small oscillations can be seen local to the interface. Although the analytical solution is not known, we expect the pressure solution is to be discontinuous across the membrane. These small oscillations that are present are therefore most likely due to Gibbs phenomenon.

8.2.2 XSEM

Once again we consider the example of an oscillating closed membrane. However, in this section we apply the XSEM to the approximation of the pressure solution. Fig. 8.7 illustrates the movement of the immersed boundary and also the pressure solution after a single timestep. Once again our parameters are: $h_f = 1/8$ and $N = 2, 8$. Note that for the purposes of this section we have used the non-dimensionalised Navier-Stokes equations (2.2.14) and we let $Re = 1$. First off, we notice that when $N = 2$, we have much better area conservation than previously in Fig. 8.6a. This suggests that the enriched approximation of the pressure is working as there must be reduced spurious velocities local to the membrane in order to prevent the area loss on such a coarse approximation. However, Figs. 8.7c and 8.7d have considerably more oscillations than the standard approximation given in Fig. 8.6c. This is most likely due to either a lack of enrichment or regions of small support.

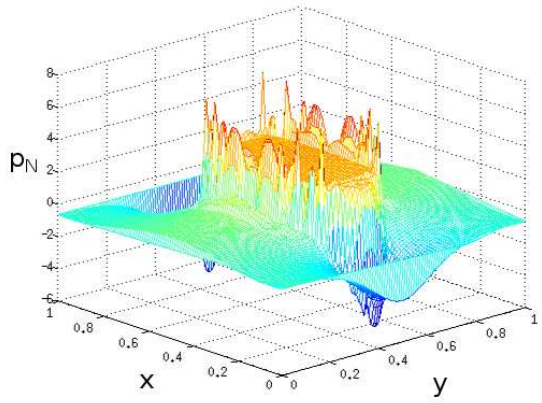
8.3 Static Closed Shell

In this section we consider the SE-IBM applied to a thick immersed structure and consider the p-type convergence of the method. We set up the example exactly as was done in §7.2.2. Again, we point out that the analytical solution (7.2.11) is continuous everywhere and linear in the region where the immersed structure resides, but it is not continuously differentiable. Again, the parameters chosen for this example are the same as given in [17] and are as follows: $\eta_s = 1$, $R = 0.25$, $\kappa = 1$, $w = 0.0625$, $\Delta t = 0.005$ and the simulation was run for 600 timesteps. As we are looking at p-type convergence here, we consider two values of the mesh width $h_f = 1/2$ and $h_f = 1/8$. Just as was done previously for the case of an immersed membrane in §8.1.1, we must

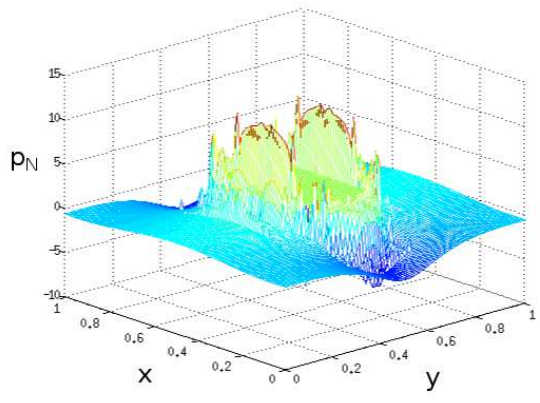


(a) IB evolution for $N = 2$.

(b) IB evolution for $N = 8$.



(c) Standard part of the enriched pressure solution for $N = 8$ after a single timestep.



(d) Full enriched pressure solution for $N = 8$ after a single timestep.

Figure 8.7: Immersed boundary evolution for $h_f = 1/8$ and $N = 2, 8$ and the pressure solution for $h_f = 1/8$ when $N = 8$ and when XSEM is used to approximate the pressure.

choose the number of structure elements so that $h^s \leq h_f/2$. Just as we did in §7.2.2, we assume that the total number of elements is given by $K_b = K_b^s \times K_b^r$, where K_b^s is calculated using the relation in (5.4.9) and K_b^r is calculated using:

$$K_b^r \geq \frac{2w}{h_f} \tag{8.3.1}$$

where this time K_b^r and K_b^s are dependent on N. These values of K_b^s and K_b^r ensure that the decomposition of Ω_r^s is composed of approximately uniform quadrilateral elements. Again, the analytical solution given in (7.2.11) is the analytical solution of Stokes flow. Therefore, here we consider Stokes flow rather than Navier-Stokes.

Table 8.6 details the error after a single timestep for a mesh width of $h_f = 1/8$ and orders of convergence for the velocity and pressure solutions. We can infer that the

N_f	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{H^1(\Omega)}$	Order	$\|p - p_N\|_{L^2(\Omega)}$	Order
2	0.01889	-	0.47424	-	0.61513	-
4	0.00092	4.35444	0.06499	2.86740	0.09170	2.74584
8	0.00008	3.58092	0.01273	2.35195	0.02406	1.93034

Table 8.6: Order of convergence with respect to N_f when $h_f = 1/8$ for a shell immersed in a Stokesian fluid.

order of convergence for the velocity with respect to the L^2 and H^1 norms is approximately 3 and 2, respectively. These orders of convergence are higher than those found by Boffi *et al.* [17] who inferred an order of 2.5 in the L^2 norm and 1.5 in the H^1 norm for the velocity. The pressure solution can be seen to have an order of approximately 2 which again is higher than the order of 1.5 inferred by Boffi *et al.* [17]. The order found here is also higher than what we found earlier in §7.2.2. Unfortunately, we were unable to run $N_f = 16$ due to computational time and memory issues - something we hope to rectify in the future.

To ensure that we are in fact obtaining higher-orders of convergence than was seen by Boffi *et al.* [17] we consider the order of convergence for a mesh width of $h_f = 1/2$. Table 8.7 details the error after a single timestep and orders of convergence for the velocity and pressure solutions. We can infer from Table 8.7 that we have an order of

N_f	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{H^1(\Omega)}$	Order	$\|p - p_N\|_{L^2(\Omega)}$	Order
2	0.20952	-	1.38309	-	1.73205	-
4	0.03071	2.77024	0.50879	1.44275	0.57180	1.59890
8	0.00469	2.71096	0.17514	1.53853	0.25584	1.16026
16	0.00035	3.76494	0.03091	2.50226	0.06881	1.89459

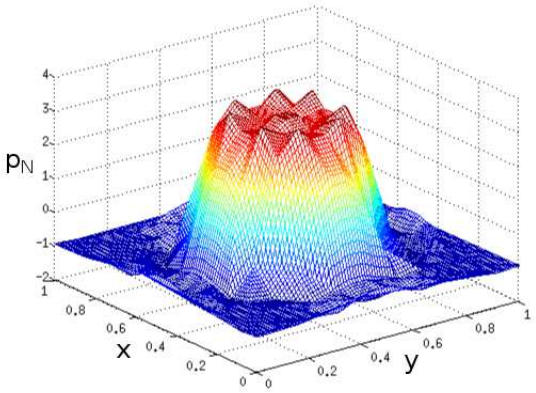
Table 8.7: Order of convergence with respect to N_f when $h_f = 1/2$ for a shell immersed in a Stokesian fluid.

3 and 2 in the L^2 and H^1 norm of the velocity error and an order of 1.5 in the L^2 norm of the pressure error. The velocity order is higher than the order found by Boffi *et al.* [17] but the order of pressure given here is the same as that inferred by Boffi *et al.*. We attribute these higher-orders to the divergence of the interpolated velocity (to the immersed structure) being closer to zero, just as we did in §8.1.1. Again, unfortunately we were unable to run $N = 32$ due to computational time.

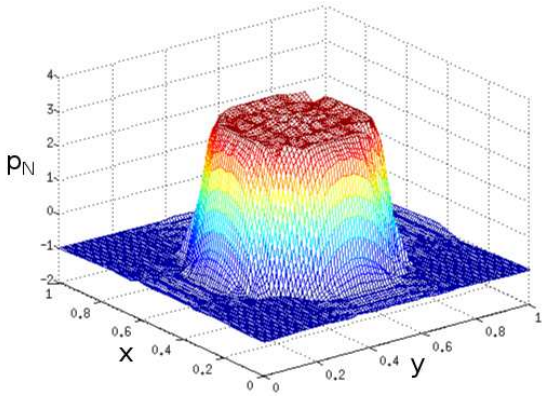
We believe that the reason we obtain optimal orders of convergence in the tables above, is due to the more accurate representation of the incompressibility constraint as we discussed in §8.1.1. According to the error estimates given in (8.1.6), we expect the order of convergence for the pressure approximation to be impaired due to the lack of regularity of the exact pressure solution. However, just as with the membrane, we do not see that impairment here. The analytical pressure solution is a H^1 function but not a H^2 function, therefore we can assume that it exists in an interpolation space, somewhere between H^1 and H^2. Boffi *et al.* stated that the pressure solution belongs to $H^{3/2-\varepsilon}$, $\varepsilon > 0$ and they attributed their sub-optimal orders of convergence to the pressure being from this interpolation space. Unfortunately, as far as we are aware, error estimates for the spectral approximation of a function from an interpolation space such as $H^{3/2-\varepsilon}$ do not exist.

Figs. 8.8a and 8.8b show the computed pressure solutions when $N = 8$ and 16 for $h_f = 1/2$. Due to the analytical solution being continuous, we should *not* see any oscillations local to the interface. Comparing those figures against the analytical solution, Fig. 8.8c, yields good agreement. Unfortunately, oscillations can be seen local to the shell. We believe these oscillations are caused by the coarseness of the mesh and hence the coarse representation of the shell. Refining the mesh to $h_f = 1/8$ and taking $N = 8$ yields better results as shown in Fig. 8.9.

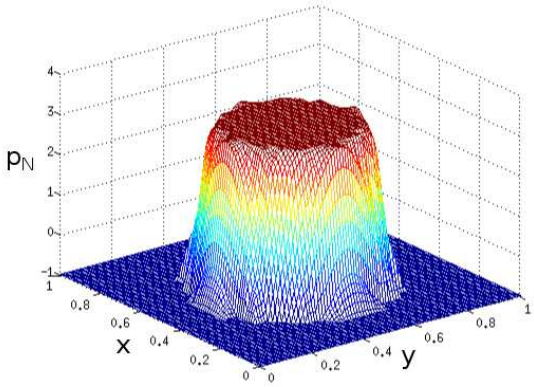
Figure 8.10 shows the velocity vector at the end of the simulation when $N = 4, 8, 16$ for $h_f = 1/2$. First we notice that the pattern of the velocity field around the shell is becoming more pronounced and by the time $N = 16$ it has a clear shape. Clearly, when $N_f = 4$ and 8, Figs. 8.10a and 8.10c, there is separation in the immersed structure marker particles. This is clearer in the zoomed plots in Figs. 8.10b and 8.10d. This



(a) Computed pressure solution $N = 8$.



(b) Computed pressure solution $N = 16$.



(c) Analytical pressure solution.

Figure 8.8: Pressure plots, at the end of the simulation, for an immersed shell when $h_f = 1/2$ and $N = 4, 8$.

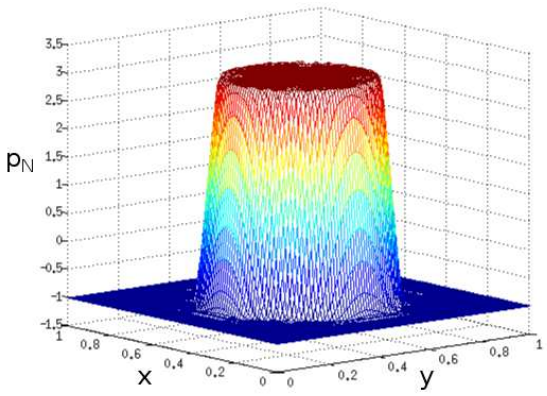
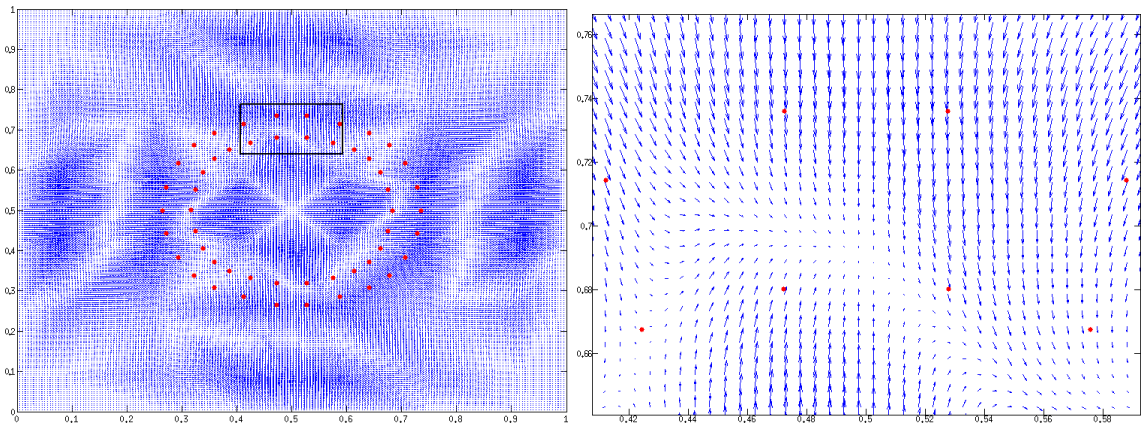


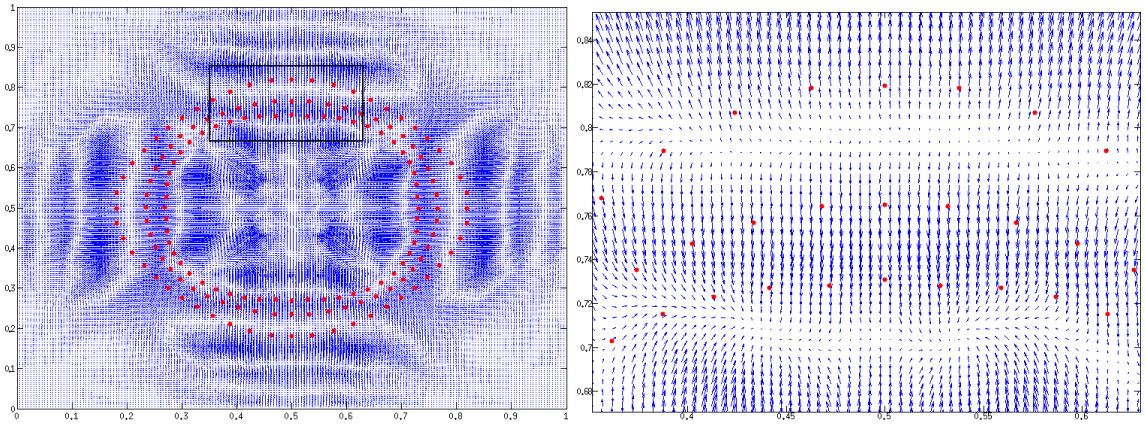
Figure 8.9: Pressure plot, at the end of the simulation, for an immersed shell when $h_f = 1/8$ and $N = 8$.

is, once again, due to the local velocity field around each marker particle. It is evident from Figs. 8.10a and 8.10c that the separation is around $x \approx 0.2$ to 0.3 and $x \approx 0.7$ to 0.8 when $y \approx 0.5$. This is similar to what we found in Fig. 7.12. However, clearly



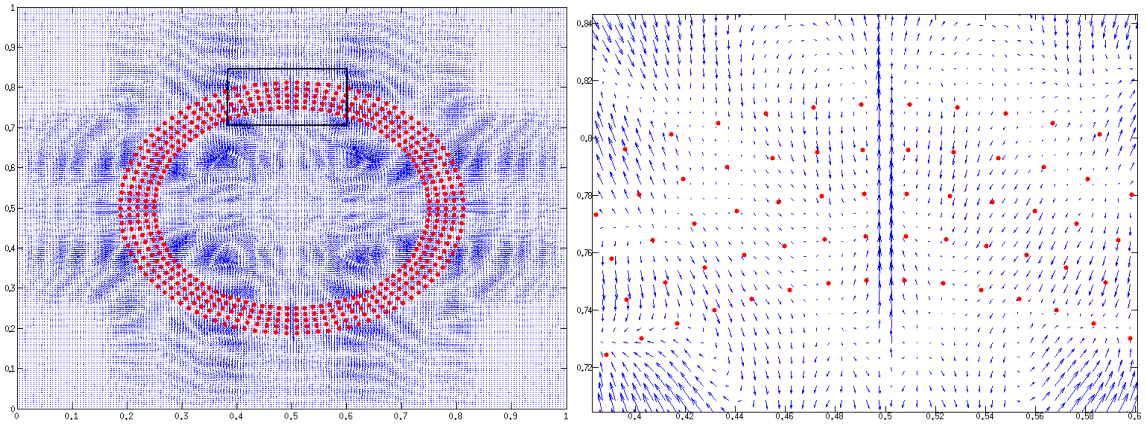
(a) Velocity vector plot for $N_f = 4$.

(b) Zoomed velocity vector plot for $N_f = 4$.



(c) Velocity vector plot for $N_f = 8$.

(d) Zoomed velocity vector plot for $N_f = 8$.



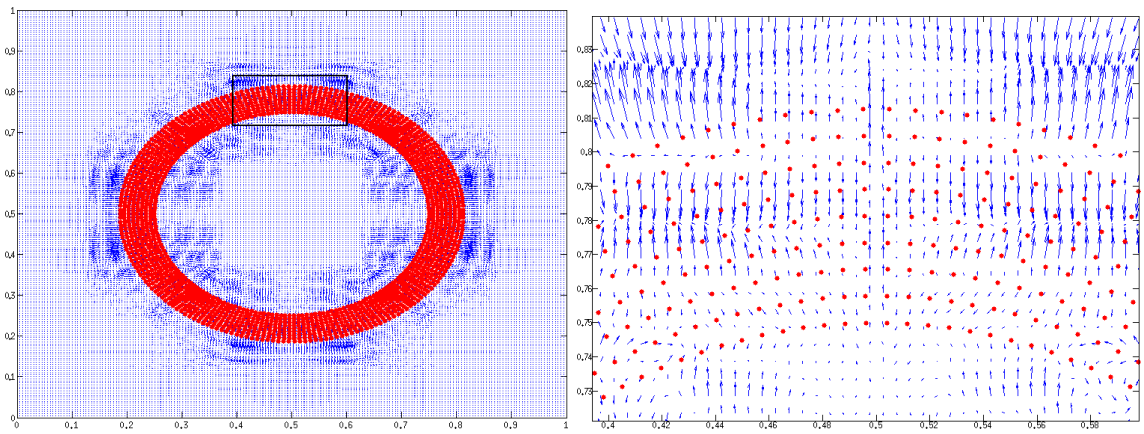
(e) Velocity vector plot for $N_f = 16$.

(f) Zoomed velocity vector plot for $N_f = 16$.

Figure 8.10: Velocity vector for $h_f = 1/2$ with increasing N. The red dots on each plot are the final position of the IB marker particles. The black box is the region which is shown in the zoomed plots.

by the time $N = 16$, Fig. 8.10e, there is very little separation and the position of the immersed boundary marker particles are more or less uniformly arranged. This is an improvement on the results given in §7.2.2 when one takes into account that the fluid mesh is very coarse.

Fig. 8.11 illustrates the velocity field when $N = 8$ and $h_f = 1/8$. The plot of the



(a) Velocity vector plot for $N_f = 8$. **(b)** Zoomed velocity vector plot for $N_f = 4$.

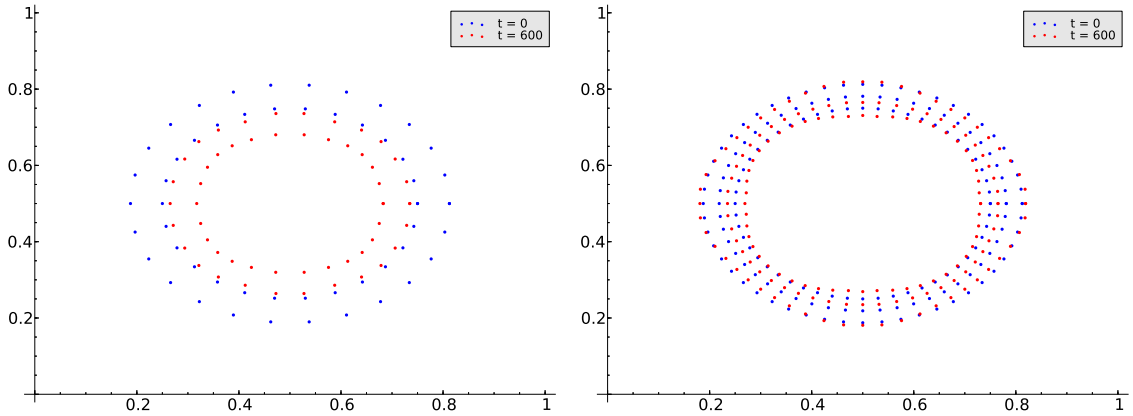
Figure 8.11: Velocity vectors for $h_f = 1/8$ and $N = 8$. The red dots on each plot are the final position of the IB marker particles. The black box is the region which is shown in the zoomed plots.

immersed boundary in Fig. 8.11 is after 200 timesteps rather than 600 because the solution had reached a steady state in the velocity field and so the program terminated early. Just as with Fig. 8.10e we notice that the immersed boundary has remained intact and there is no separation of the *layers* of the immersed structure.

Finally, Fig. 8.12 shows the initial and final configuration of the immersed shell when $N = 4, 8, 16$ for $h_f = 1/2$ and $N = 8$ when $h_f = 1/8$. It is evident that there is separation between the marker particles when $N = 8$ and $h_f = 1/2$. However, when $N = 16$ and $h_f = 1/2$ the marker particles barely move due to the their number being greater and also because of the improved accuracy of the solution.

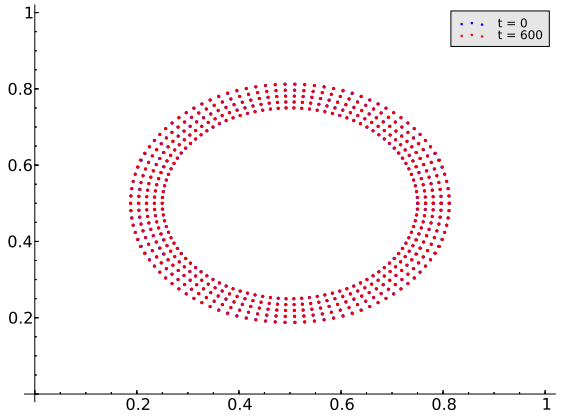
8.4 Oscillating Closed Shell

In this section we apply the SE-IBM to the oscillating closed shell example considered in §7.2.2. Fig. 8.13 illustrates the movement of the immersed shell and also the pressure solution after a single timestep. Once again our parameters are: $h_f = 1/8$ and $N = 8$. The simulation was terminated at $T = 3$. It is clear from the motion of the

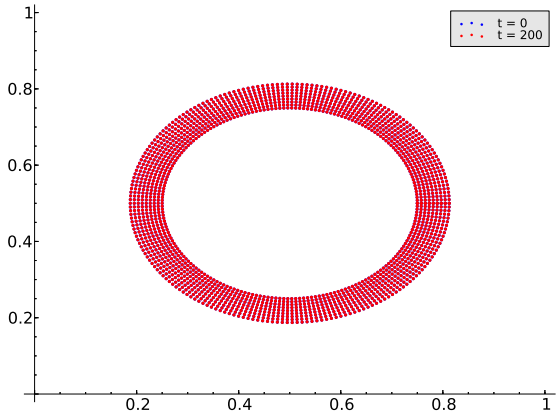


(a) IB evolution for $N_f = 4$, $h_f = 1/2$.

(b) IB evolution for $N_f = 8$, $h_f = 1/2$.



(c) IB evolution for $N_f = 16$, $h_f = 1/2$.

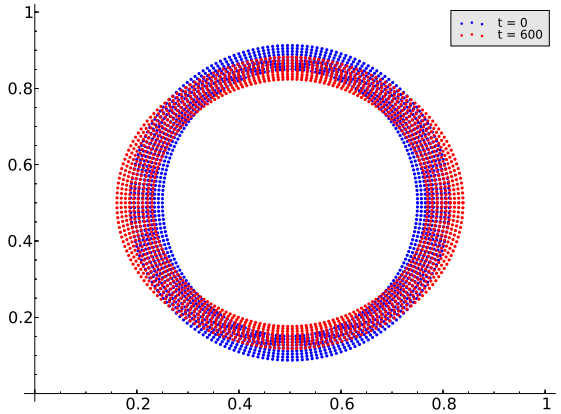


(d) IB evolution for $N_f = 8$, $h_f = 1/8$.

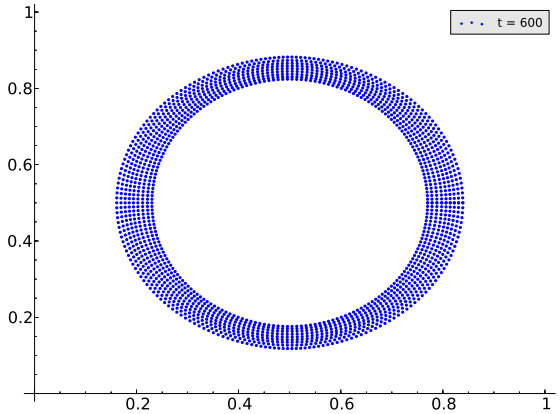
Figure 8.12: Immersed shell evolution.

shell that there is no separation between the *layers* of marker particles. This could be due to the number of marker particles that make up the shell or the more accurate velocity field as a result of the higher-order approximation. However, it seems from Fig. 8.13b that the shell is thinner at the top and bottom than it is at the sides. This again is most likely caused by the local fluid velocity and it may be that increasing the order beyond $N = 8$ will produce a more uniform shell. Unfortunately we were unable to take N beyond 8 due to the limits in computational time.

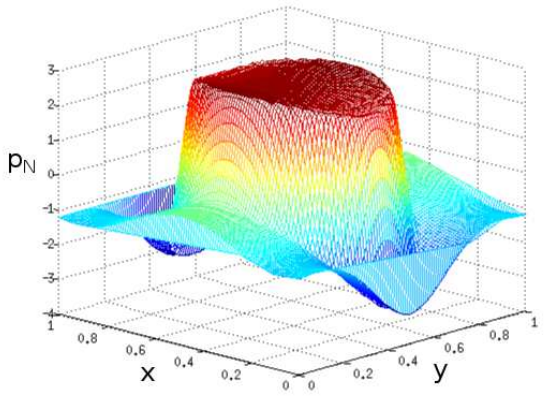
Finally, we see have good agreement between our pressure solution in Fig. 8.13c and the solution in Fig. 7.14c. Notice that there are very small oscillations local to the shell but we believe these are most likely caused by the linear interpolation in MATLAB which was used to produce the plot.



(a) IB evolution for $N = 8$ at $T = 0$ and $T = 3$.



(b) IB evolution for $N = 8$ at $T = 3$.



(c) Computed pressure solution for $N = 8$ after a single timestep.

Figure 8.13: Immersed shell evolution for $h_f = 1/8$ and $N = 2, 8$ and the pressure solution, after a single timestep, for $h_f = 1/8$ when $N = 8$.

Chapter 9

Numerical Calculations: Non-Newtonian Fluids

In this chapter, we consider an elastic membrane and shell immersed in an Oldroyd-B fluid. For the immersed shell examples we only consider using SE-IBM, however for the immersed membrane examples we use both SE-IBM and XSEM. Where possible we investigate the order of convergence. Throughout this chapter, we have used the Operator-Integration-Factor-Splitting scheme which we discussed in §4.1.1 to approximate the material derivative that appears in the momentum equation and the constitutive equation. Also, we consider the non-dimensionalised formulation of the governing equations.

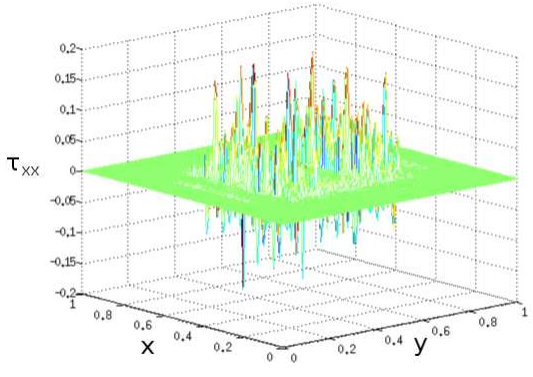
9.1 Static Closed Membrane

9.1.1 SEM

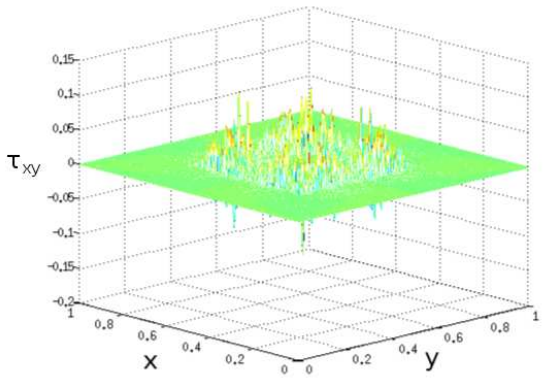
In this subsection, we look at the spectral element approximation of an Oldroyd-B fluid with a static, closed immersed membrane. For simplicity, we limit ourselves to the same domain that was considered in §7.2.1 and §8.1.1; that is $\Omega = [0, 1]^2$ as illustrated in Fig. 4.3. The domain is decomposed into uniform quadrilateral spectral elements of

width $h_f = 1/8$. The polynomial degree N can take the values $2, 4, 8$. The reason why we have not considered $N = 16$ is computational time. The other parameters are as follows: $Re = 1$, $We = 1, 10, 100$, $\beta = 1/9$, $\Delta t = 10^{-4}$, $R = 0.25$ and $\kappa = 1$. Also throughout this subsection, we let $K_b = K_b(N)$ as defined in (5.4.9).

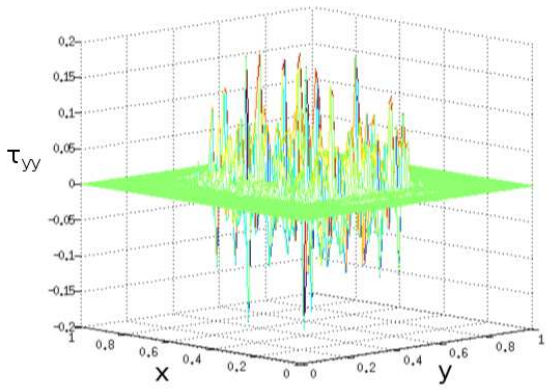
Fig. 9.1 illustrates the components of the polymeric stress tensor for $h_f = 1/8$, $We = 1.0$ and $N = 8$ at the end of the simulation (600 timesteps). The analyti-



(a) Computed τ_{xx} solution.



(b) Computed τ_{xy} solution.



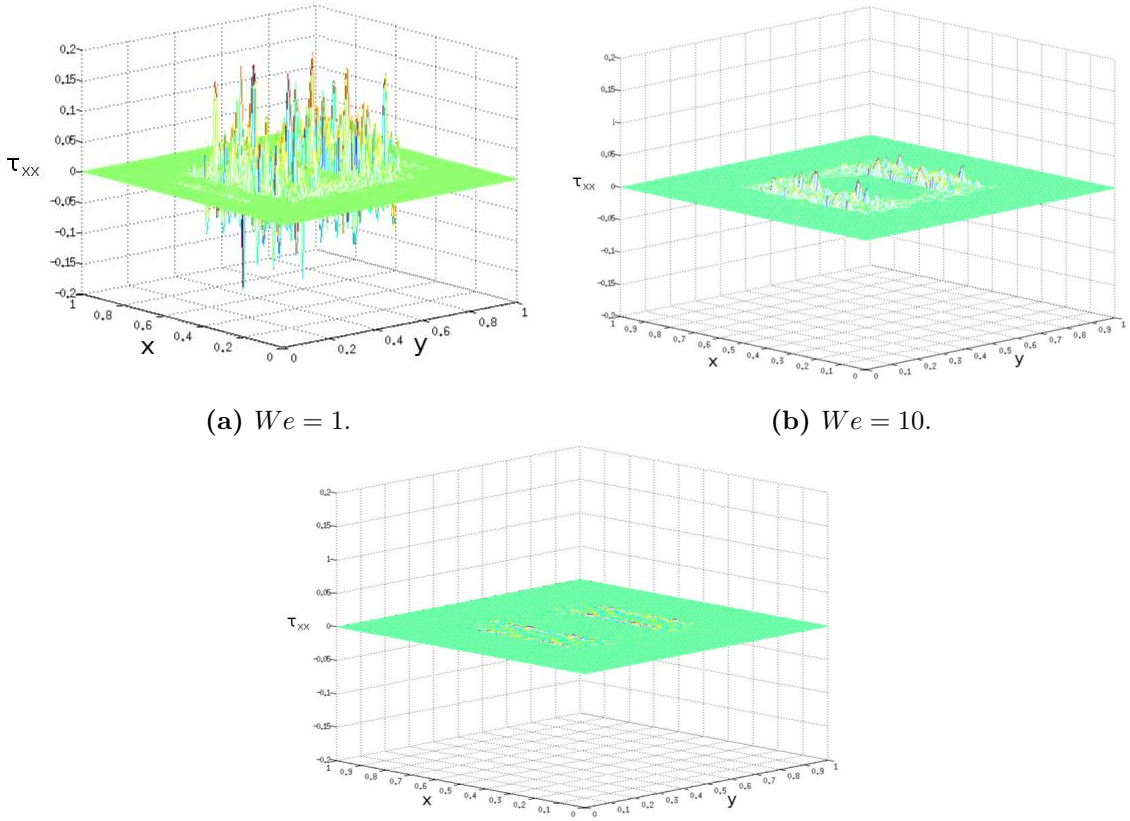
(c) Computed τ_{yy} solution.

Figure 9.1: Plots of the components of the computed polymeric stress for $We = 1$, $h_f = 1/8$ and $N = 8$.

cal solution for the polymeric stress is $\boldsymbol{\tau} = 0$, therefore we expect all components to be zero. However, the discontinuity in the pressure causes spurious velocities local to the membrane. These spurious velocities cause oscillations in the polymeric stress components. Although this example is very simple, these oscillations are worrying. The reason is that the constitutive equation (2.2.26c) is hyperbolic and it is possible that these oscillations may cause the computation to break down. Therefore, we look

at the effect increasing the Weissenberg number has on each of the components.

Fig. 9.2 illustrates the τ_{xx} component for increasing Weissenberg number for $h_f = 1/8$ and $N = 8$. All the figures have the same scale for the axes so that the effect of Weis-



(a) $We = 1$.

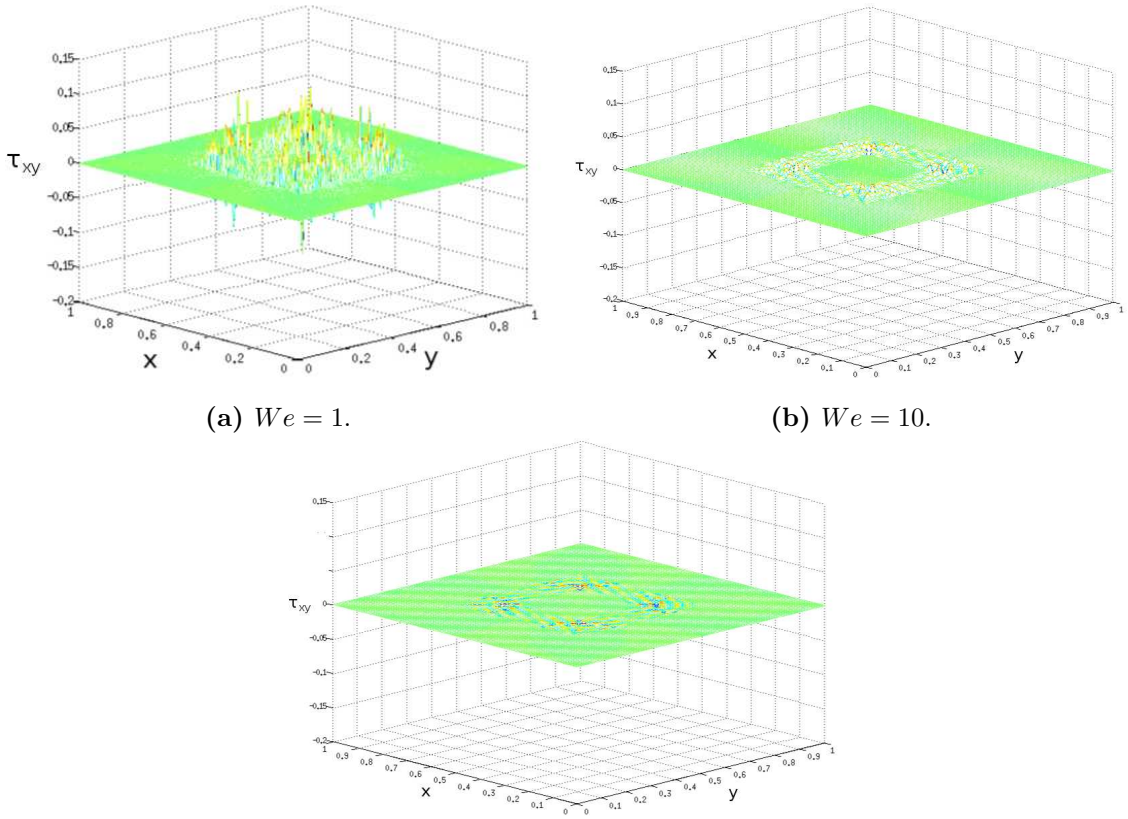
(b) $We = 10$.

(c) $We = 100$.

Figure 9.2: Plots of the τ_{xx} component for $h_f = 1/8$ and $N = 8$.

senberg number is emphasized. In all cases, we notice that as the Weissenberg number is increased, the oscillations actually decrease. This example is simple and therefore it may be possible that this example does not exhibit the high Weissenberg number problem. On the contrary, it may be that if the simulation was run for longer then the oscillations could increase.

We verify that the same behaviour is seen for all components of the polymeric stress. Fig. 9.3 illustrates the τ_{xy} component for increasing Weissenberg number for $h_f = 1/8$ and $N = 8$. Again the figures have the same scale for the axes so that the effect of



(a) $We = 1$.

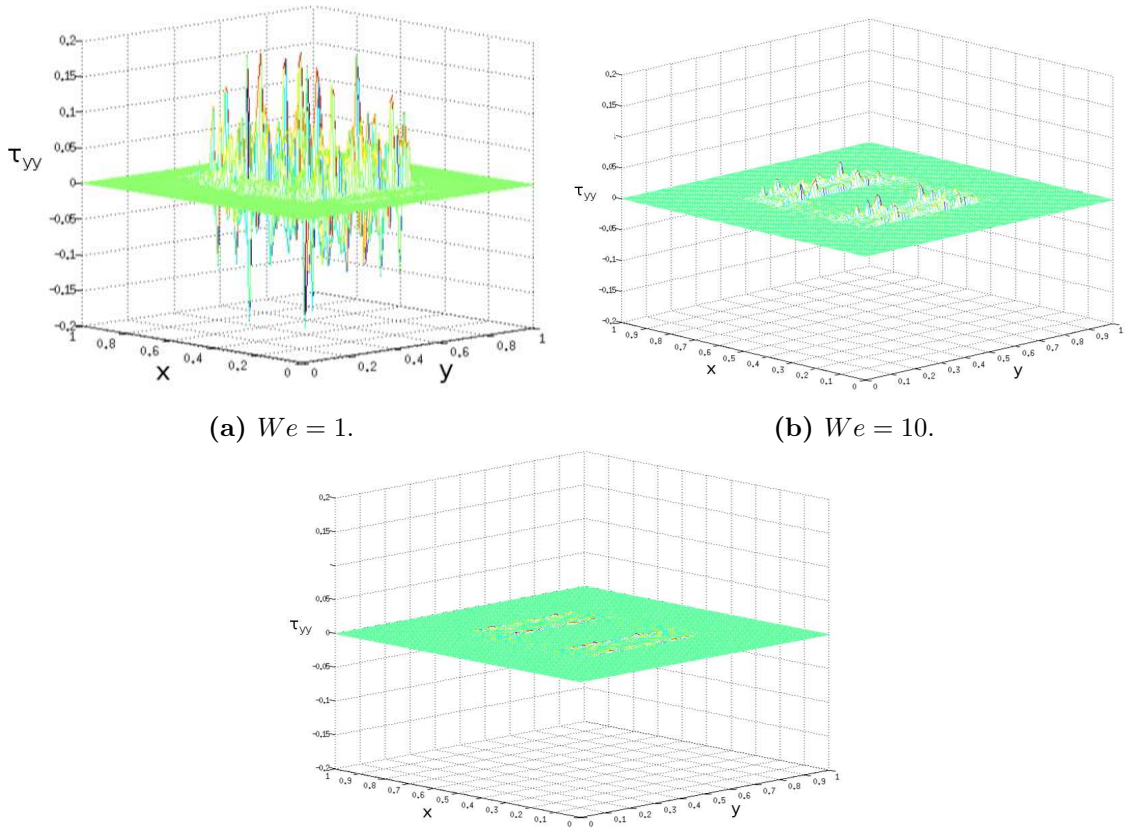
(b) $We = 10$.

(c) $We = 100$.

Figure 9.3: Plots of the τ_{xy} component for $h_f = 1/8$ and $N = 8$.

the Weissenberg number can be seen. Again, we see that the oscillations are decreased as the Weissenberg number is increased. Fig. 9.4 illustrates the τ_{yy} component for increasing Weissenberg number for $h_f = 1/8$ and $N = 8$. Once again, we see that the oscillations decrease as the Weissenberg number is increased. This is not what we expected. We expected that increasing the Weissenberg number would increase the oscillations and cause the numerics to break down. However, we still believe this would happen if the simulation was run for a lot longer.

Table 9.1 details the error and the order of convergence for the velocity, pressure and stress when $h_f = 1/8$ and $We = 1$, at the end of the simulation. In Table 9.1, we can infer an order of 2 and 1 for the L^2 and H^1 norm of the velocity, 0.5 for the L^2 norm of the pressure and 1 for the L^2 norm of the polymeric stress. According to Gerritsma and Phillips [34], if $\boldsymbol{u} \in (H^2(\Omega) \cap H_0^1(\Omega))^2$, $p \in H^1(\Omega)$ and $\boldsymbol{\tau} \in (H^1(\Omega))_s^4$, then we expect for the velocity second-order convergence in the L^2 norm and first-order conver-



(a) $We = 1$.

(b) $We = 10$.

(c) $We = 100$.

Figure 9.4: Plots of the τ_{yy} component for $h_f = 1/8$ and $N = 8$.

gence in the H^1 norm, an order of 0.5 for the pressure in the L^2 norm and first-order convergence for the polymeric stress in the L^2 norm. This is precisely the scenario we see above, even though our pressure is not a H^1 function.

Table 9.2 details the error and the order of convergence for the velocity, pressure and stress when $h_f = 1/8$ and $We = 10$, at the end of the simulation. Clearly, we obtain the same order of convergence for the velocity and pressure in the L^2 norm and the same order for the velocity in the H^1 norm. However, we note that the order of convergence for the polymeric stress has decayed slightly. This may indicate a dependence on the Weissenberg number.

To verify this dependence, Table 9.2 details the error and the order of convergence for the velocity, pressure and stress when $h_f = 1/8$ and $We = 100$, at the end of

N	$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{H^1(\Omega)}$	Order
2	0.10506	-	2.64735	-
4	0.01024	3.35859	0.74030	1.83837
8	0.00197	2.37525	0.31411	1.23683

N	$\|p-p_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{\tau}-\boldsymbol{\tau}_N\|_{L^2(\Omega)}$	Order
2	0.72174	-	0.16140	-
4	0.32806	1.13753	0.07381	1.12876
8	0.22712	0.53048	0.03697	0.99746

Table 9.1: Order of convergence with respect to N when $h_f = 1/8$ for a membrane immersed in an Oldroyd-B fluid with $We = 1$.

the simulation. Once again, we see no change in the order of convergence for the velocity and pressure in the L^2 norm and the H^1 norm of the velocity. However, once again we see that the order of convergence for the stress has once again slightly deteriorated. Although not conclusive, this could show that this example does indeed have a dependence on the Weissenberg number.

9.1.2 XSEM

In this section we apply XSEM to the approximation of an Oldroyd-B fluid with an immersed static membrane. Note that XSEM is only applied to the pressure solution and that standard SEM is used for both the approximation of the velocity and polymeric stress. Due to computational time, we have only considered a mesh width of $h_f = 1/8$ for $N = 2, 4, 8$. Unfortunately we could not consider $N = 16$. As we mentioned earlier, including XSEM introduces two additional sparse matrices to the linear system which increase the condition number quite significantly. Just as before, we let $N_q = 2N + 2$ and let $Re = 1$, $We = 1, 10, 100$.

In the previous section, Fig. 9.1 illustrated the stress components on a 8x8 mesh with

N	$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{H^1(\Omega)}$	Order
2	0.12468	-	3.13066	-
4	0.01608	2.95529	1.14193	1.45500
8	0.00342	2.23306	0.54679	1.06241

N	$\|p-p_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{\tau}-\boldsymbol{\tau}_N\|_{L^2(\Omega)}$	Order
2	0.73056	-	0.01934	-
4	0.32517	1.16781	0.00986	0.97222
8	0.22659	0.52109	0.00510	0.95162

Table 9.2: Order of convergence with respect to N when $h_f = 1/8$ for a membrane immersed in an Oldroyd-B fluid with $We = 10$.

$N = 8$ for $We = 1$. In Fig. 9.5 below, we compare those results (left hand column) against the results when XSEM is used to approximate the pressure (right hand column). We have used the same scale for the axes in both plots so as to emphasize the effect that XSEM has on the stress solution. Clearly, using an enriched approximation for the pressure greatly reduces the oscillations seen in the polymeric stress. This is a big advantage to using enriched approximations within viscoelastic fluids. In many cases, the oscillations that exist within the stress are propagated throughout the computation due to the hyperbolicity of the constitutive equations. Those oscillations can then cause the computation to break down in certain cases. Therefore, it is theoretically possible to use an enriched method to approximate any discontinuous functions to reduce the oscillations seen in the polymeric stress and therefore, potentially, preventing the computation from breaking down. We note that we have not enriched the stress here, only the pressure.

Fig. 9.6 illustrates the same plots as Fig. 9.5 but when the Weissenberg number is 10. Once again, we have used the same scale for the axes as the original plots so as to emphasize the effect of XSEM. Even though we are at a higher Weissenberg number, we see that using XSEM to approximate the pressure, greatly reduces the oscillations found in the polymeric stress.

N	$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u}-\boldsymbol{u}_N\|_{H^1(\Omega)}$	Order
2	0.12708	-	3.18976	-
4	0.01689	2.91133	1.19817	1.41261
8	0.00363	2.21907	0.58046	1.04556

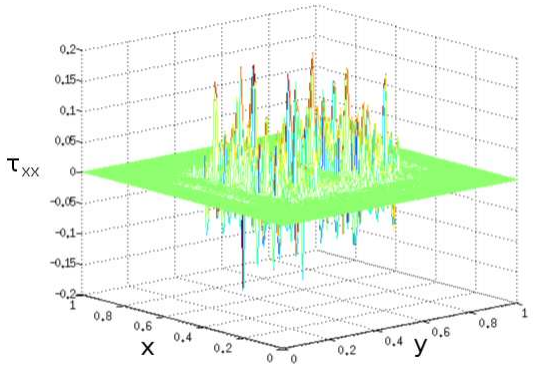
N	$\|p-p_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{\tau}-\boldsymbol{\tau}_N\|_{L^2(\Omega)}$	Order
2	0.73171	-	0.00197	-
4	0.32496	1.17099	0.00102	0.95378
8	0.22658	0.52028	0.00053	0.94720

Table 9.3: Order of convergence with respect to N when $h_f = 1/8$ for a membrane immersed in an Oldroyd-B fluid with $We = 100$.

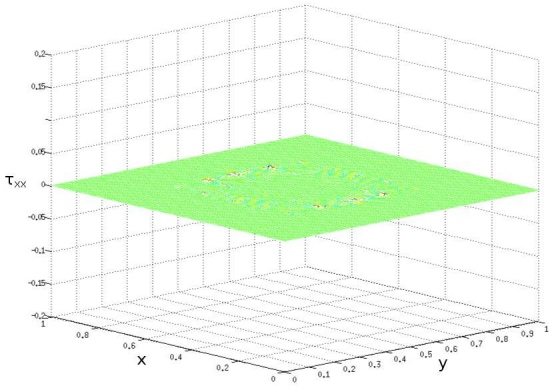
Fig. 9.7 illustrates the same plots as Fig. 9.6 but when the Weissenberg number is 100. Note that the oscillations actually reduce for increasing Weissenberg number, but we still expect that XSEM will reduce these oscillations further. Once again we use the same scales to emphasize the effect of the XSEM approximation. Clearly, once again we see that the oscillations have been reduced quite dramatically.

Figs. 9.5, 9.6 and 9.7 show that using XSEM to approximate the pressure can greatly reduce the oscillations present in the polymeric stress components, irrespective of Weissenberg number. However, this example is extremely simple and so may not exhibit problems at high Weissenberg number, or not at least until extremely high values.

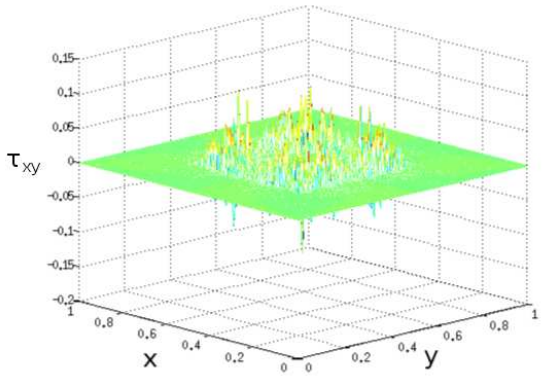
It would be very interesting to apply XSEM to a model which does exhibit problems at high Weissenberg numbers to see if XSEM has a stabilising effect. It also would be interesting to approximate the polymeric stress using the XSEM. Unfortunately, there are potential problems with such an approximation. For example, if the polymeric stress exhibited a strong discontinuity, the enriched approximation would take the



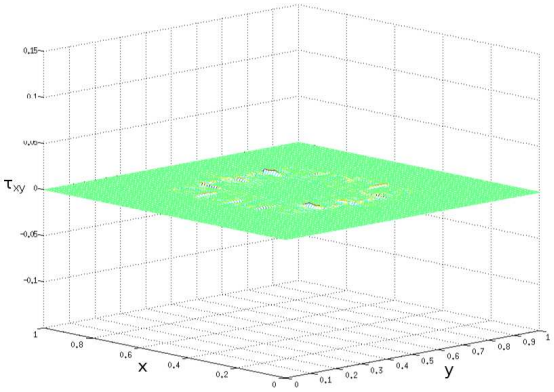
(a) Computed τ_{xx} solution for p_N.



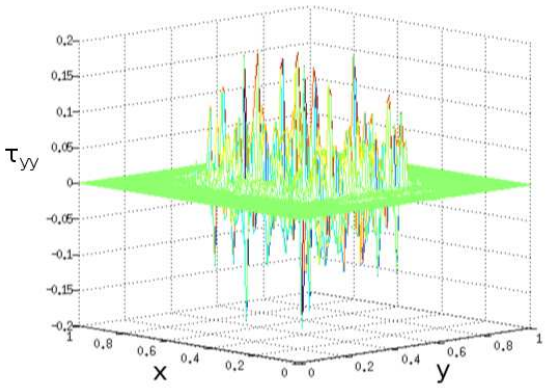
(b) Computed τ_{xx} solution for p_N^Γ.



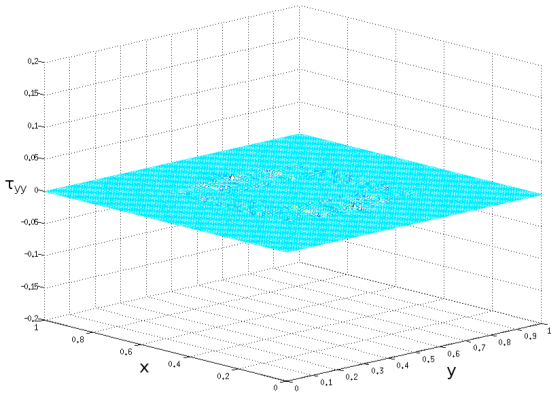
(c) Computed τ_{xy} solution for p_N.



(d) Computed τ_{xy} solution for p_N^Γ.



(e) Computed τ_{yy} solution for p_N.

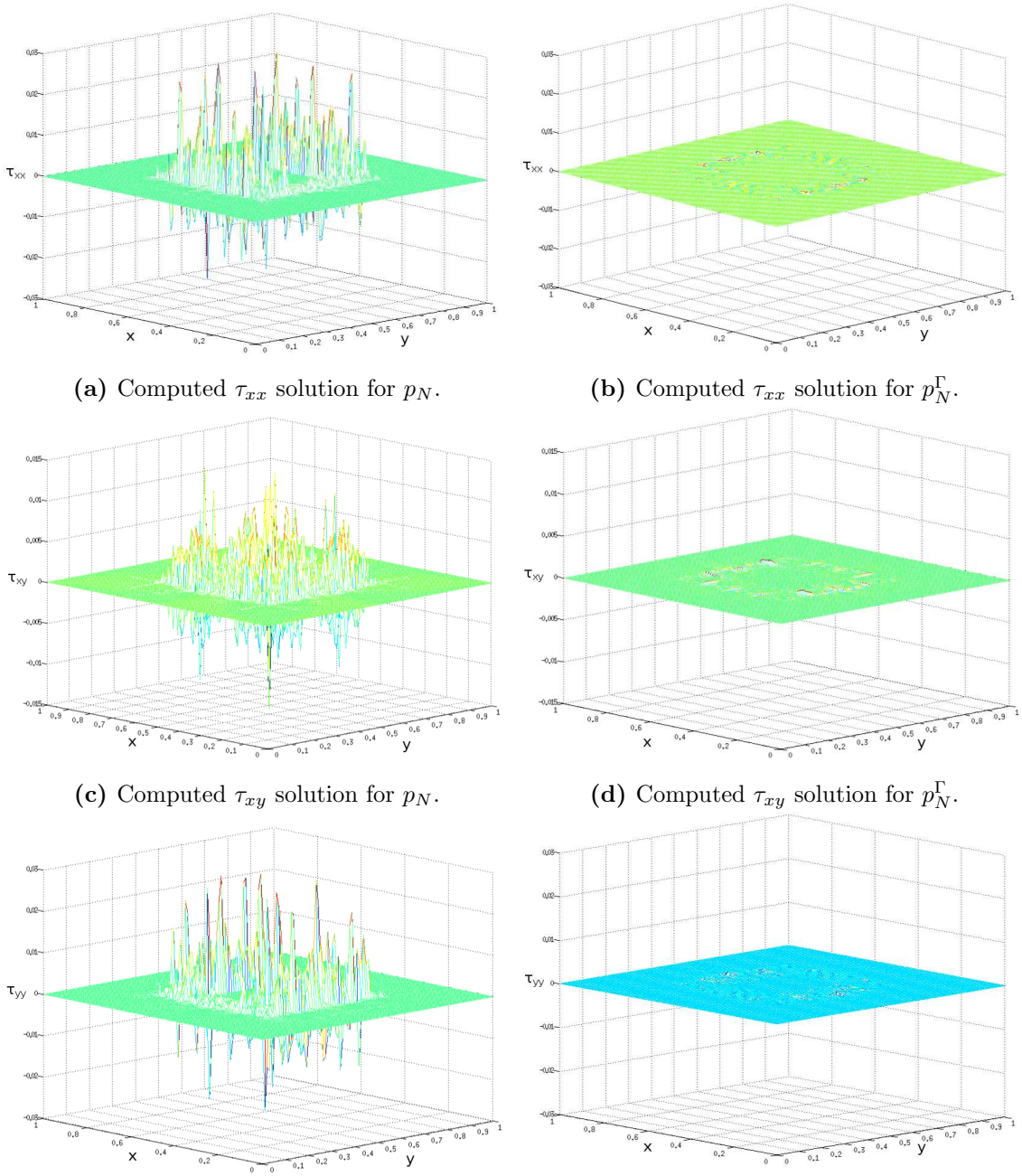


(f) Computed τ_{yy} solution for p_N^Γ.

Figure 9.5: Plots of the components of the computed polymeric stress for $We = 1$, $h_f = 1/8$ and $N = 8$. Left hand column are the original SEM approximation and the right hand column are the XSEM approximation.

form:

$$\begin{aligned}\boldsymbol{\tau}_N^\Gamma &= \boldsymbol{\tau}_N + \boldsymbol{\tau}_N^X \\ &= \sum_{i,j=0}^{N} \boldsymbol{\tau}_{ij} h_i(\xi) h_j(\eta) + \sum_{i,j=0}^{N} \boldsymbol{\alpha}_{ij} h_i(\xi) h_j(\eta) \boldsymbol{\phi}_{ij}(\xi,\eta) \end{aligned} \tag{9.1.1}$$



(a) Computed τ_{xx} solution for p_N.

(b) Computed τ_{xx} solution for p_N^Γ.

(c) Computed τ_{xy} solution for p_N.

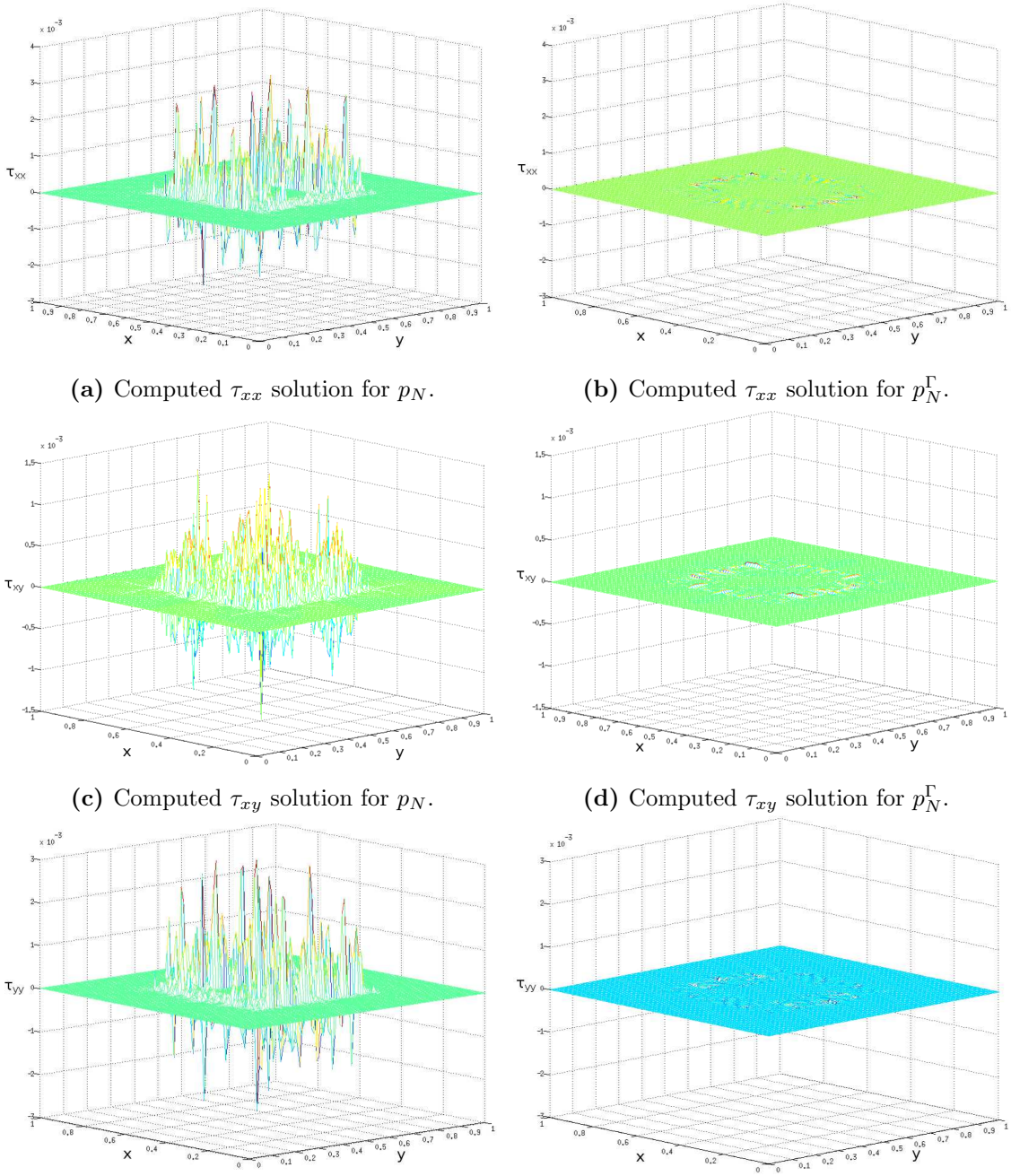
(d) Computed τ_{xy} solution for p_N^Γ.

(e) Computed τ_{yy} solution for p_N.

(f) Computed τ_{yy} solution for p_N^Γ.

Figure 9.6: Plots of the components of the computed polymeric stress for $We = 10$, $h_f = 1/8$ and $N = 8$. Left hand column are the original SEM approximation and the right hand column are the XSEM approximation.

Unfortunately, $\boldsymbol{\tau}_N^X \equiv 0$ for all values of (ξ, η) on the GLL grid. For example, for values of (ξ, η) in which $\boldsymbol{\phi}_{ij} \neq 0$, the Lagrange interpolants $h_i(\xi) = 0$ or $h_j(\eta) = 0$ and for values of (ξ, η) in which $h_i(\xi)$ and $h_j(\eta)$ are non zero, the enrichment function $\boldsymbol{\phi}_{ij}$ is zero. The use of over-integration as discussed in §6.2 may alleviate this issue but may



(a) Computed τ_{xx} solution for p_N.

(b) Computed τ_{xx} solution for p_N^Γ.

(c) Computed τ_{xy} solution for p_N.

(d) Computed τ_{xy} solution for p_N^Γ.

(e) Computed τ_{yy} solution for p_N.

(f) Computed τ_{yy} solution for p_N^Γ.

Figure 9.7: Plots of the components of the computed polymeric stress for $We = 100$, $h_f = 1/8$ and $N = 8$. Left hand column are the original SEM approximation and the right hand column are the XSEM approximation.

not be accurate enough. Legay, Wang and Belytschko [42] used a different polynomial degree for the extended part ($\boldsymbol{\tau}_N^X$). However, in our opinion, this will impair the order of convergence similar to what is found for the pressure for the standard SEM approximation. Additionally, it would also be interesting to apply the XSEM approximation

to the Upper-Convected Maxwell model. The UCM model is very difficult to deal with numerically, due to the equation being purely hyperbolic. It would interesting to see if the use of XSEM has a stabilising effect.

Finally, Table 9.4 illustrates the orders of convergence for the velocity, the standard part of the enriched approximation of the pressure and the stress when $We = 1$ at the end of the simulation. Clearly, the use of the XSEM approximation for the pressure

N	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{H^1(\Omega)}$	Order
2	0.00267	-	0.11392	-
4	0.00019	3.77634	0.01630	2.80510
8	0.00004	2.34288	0.00879	0.89056

N	$\|p - p_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{\tau} - \boldsymbol{\tau}_N\|_{L^2(\Omega)}$	Order
2	0.76675	-	0.00728	-
4	3.06285	-1.99805	0.00145	2.32383
8	1.18313	1.37226	0.000726	1.00328

Table 9.4: Order of convergence with respect to N when $h_f = 1/8$ for a membrane immersed in an Oldroyd-B fluid with $We = 1$.

has given similar spurious orders of convergence that we saw in §8.1.2. However, the order of convergence for the polymeric stress has been improved slightly.

9.2 Oscillating Closed Membrane

9.2.1 SEM

In this section we immerse an oscillating membrane in an Oldroyd-B fluid. Just as we did for the static case §9.1.1, we consider a mesh width of $h_f = 1/8$ and let $N = 8$. We also let $Re = 1$ and $We = 1, 10, 100$. Again we use the OIFS method that we discussed

in §4.1.1. The other parameters are again as follows: $\beta = 1/9$, $\Delta t = 10^{-4}$, $R = 0.25$ and $\kappa = 1$.

To illustrate that the membrane is behaving as expected, Fig. 9.8 shows the movement of the immersed boundary for $h_f = 1/8$, $N = 8$ and $We = 1$. The simulation is run

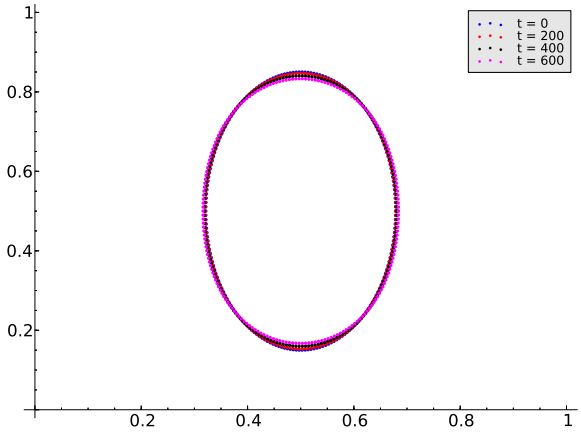
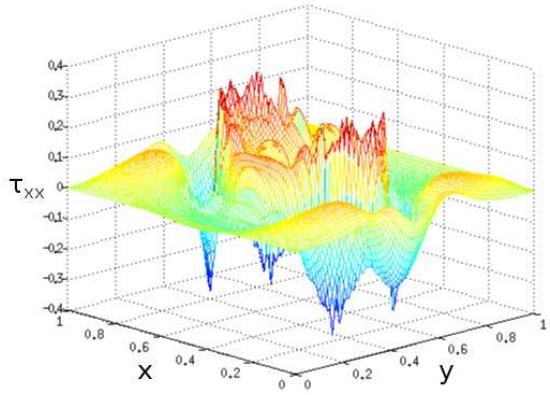


Figure 9.8: Immersed boundary evolution for $N = 8$ and $h_f = 1/8$ when $We = 1$.

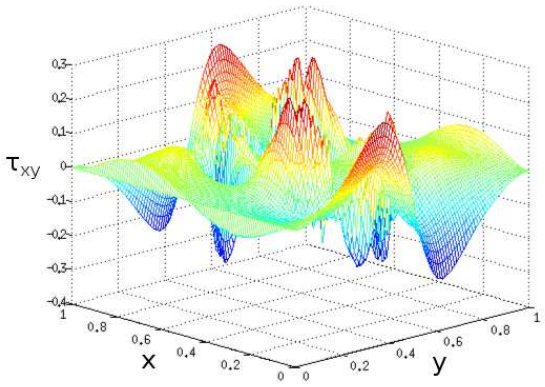
for 600 timesteps. However, at $Re = 1$ there is not much movement of the immersed boundary. However, it is clear from Fig. 9.8 that the membrane has started to relax towards the static closed membrane.

Following the static example given earlier, we plot the components of the stress. Fig. 9.9 illustrates the components of the polymeric stress for $We = 1$ at the end of the simulation. Clearly there are oscillations local to the membrane due to the pressure discontinuity. However, the whole plot of each stress component is *oscillatory*. This is due to the motion of the membrane as it relaxes. In the next section we apply the XSEM to approximate the pressure and it will be interesting to see whether the motion is stabilised. Fig. 9.9d shows that the highest stress occurs local to the membrane (the red dots).

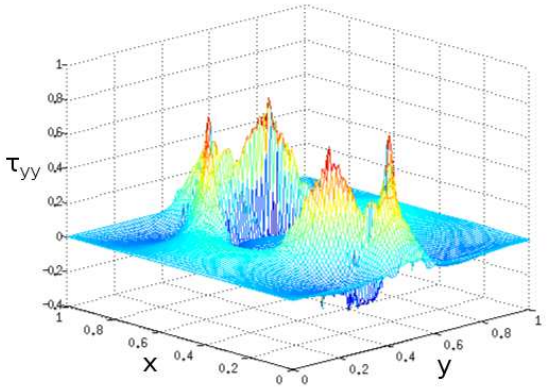
Fig. 9.10 illustrates the τ_{xx} component as we increase the Weissenberg number from $We = 1$ to 100. Following the results of the static immersed membrane, we expect that the oscillations will decrease. Just as we did in the static case, we have set the scales



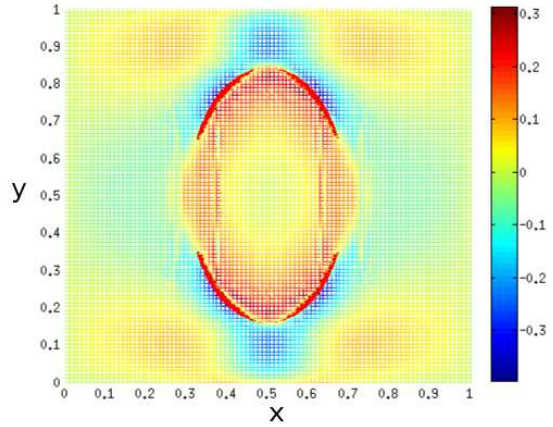
(a) Computed τ_{xx} solution.



(b) Computed τ_{xy} solution.



(c) Computed τ_{yy} solution.

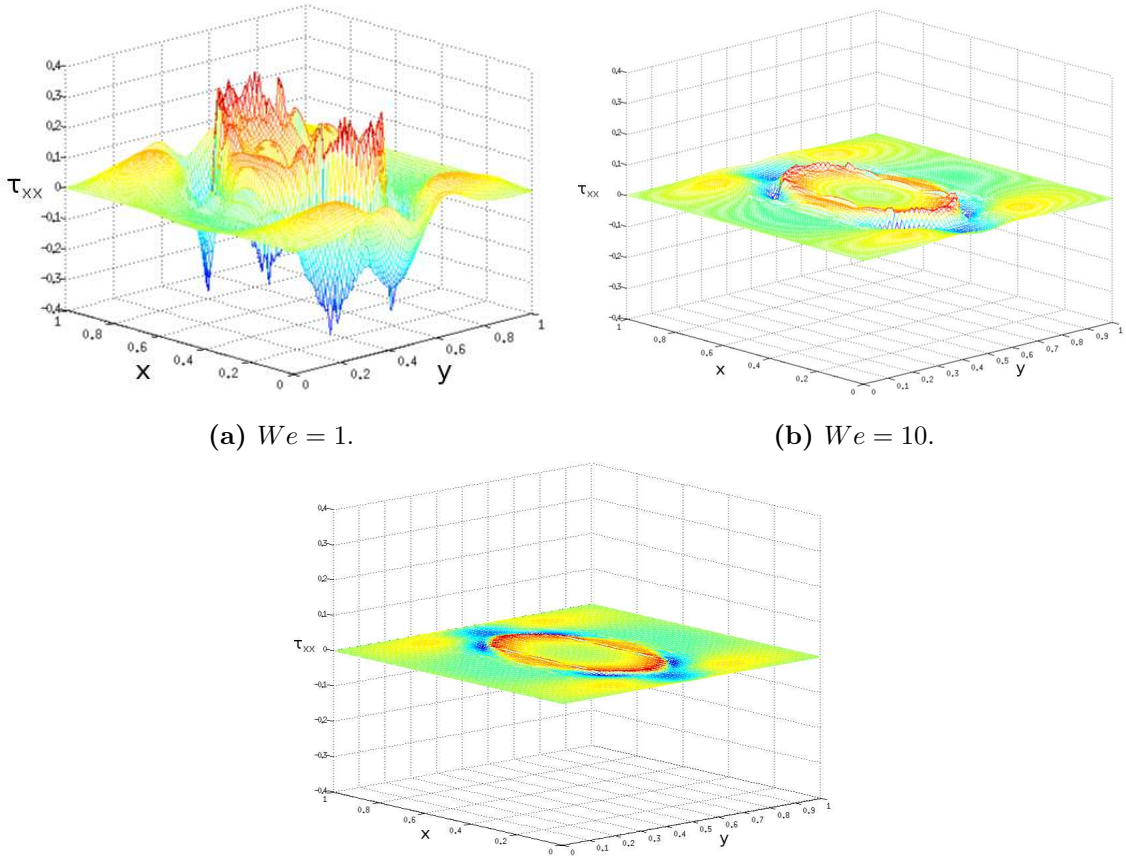


(d) Computed τ_{xx} solution looking down from above.

Figure 9.9: Plots of the components of the computed polymeric stress for $We = 1$, $h_f = 1/8$ and $N = 8$.

of the axes to be the same to emphasize the effect of the Weissenberg number. Just as before, we see that as the Weissenberg number is increased, the oscillations local to the membrane decrease as we expected.

Just as before, we check that the other components exhibit the same behaviour. Fig. 9.11 illustrates the τ_{xy} component as we increase the Weissenberg number from $We = 1$ to 100. Judging from our previous results, we expect that the oscillations will decrease and this is precisely what we see. It is once again, quite a drastic reduction in oscillations. Finally, Fig. 9.12 illustrates the τ_{yy} component as we increase the Weissenberg number from $We = 1$ to 100. Again, we see a reduction in oscillations as the Weissenberg number is increased. We believe that if we run the simulation for much longer,



(a) $We = 1$.

(b) $We = 10$.

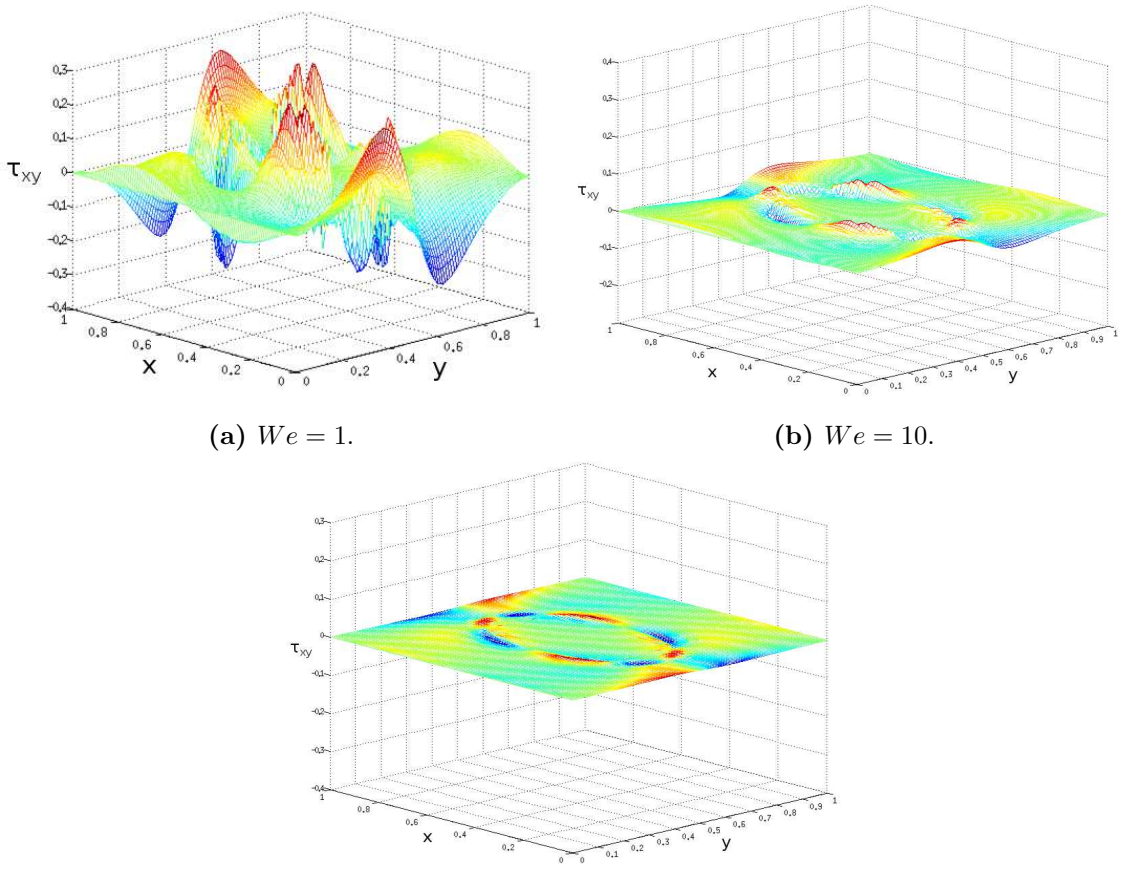
(c) $We = 100$.

Figure 9.10: Plots of the τ_{xx} component of the computed polymeric stress for $We = 1, 10, 100$, $h_f = 1/8$ and $N = 8$.

we would eventually see greater oscillations. Additionally, it would be interesting to run the same example for a *very* high Weissenberg number, e.g. in the region of $1,000$ to $10,000$ to see if the simulation breaks down.

9.2.2 XSEM

In this section we apply the XSEM to the pressure and see how it affects the polymeric stress when the immersed membrane is initially perturbed. Just as with the previous section, we let $Re = 1$, $We = 1, 10, 100$, $\beta = 1/9$, $\Delta t = 10^{-4}$, $R = 0.25$ and $\kappa = 1$. Also we consider a mesh width of $h_f = 1/8$ and $N = 8$. As we are using XSEM, we let $N_q = 2N + 2$ just as before.



(a) $We = 1$.

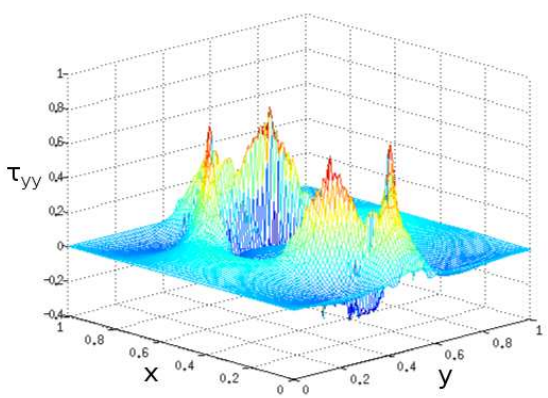
(b) $We = 10$.

(c) $We = 100$.

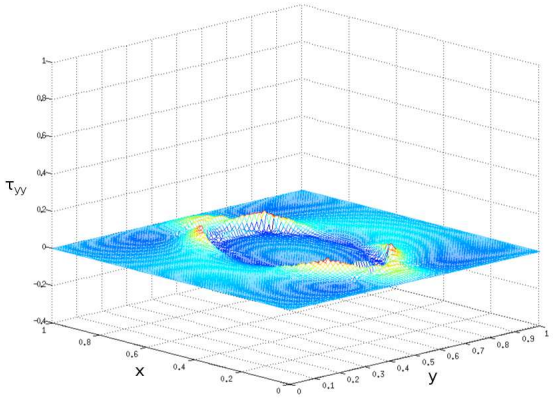
Figure 9.11: Plots of the τ_{xy} component of the computed polymeric stress for $We = 1, 10, 100$, $h_f = 1/8$ and $N = 8$.

Previously in §9.1.2, we compared the standard SEM approximation and the XSEM approximation and we do the same comparison here. In the previous section, Fig. 9.9 illustrated the stress components on a 8x8 mesh with $N = 8$ for $We = 1$. In Fig. 9.13, we compare those results (left hand column) against the results when XSEM is used to approximate the pressure (right hand column). Clearly, using the XSEM approximation for the pressure does very little to the polymeric stress when the immersed membrane is initially perturbed. This is most puzzling, and really the only explanation is a lack of enrichment or indeed no enrichment.

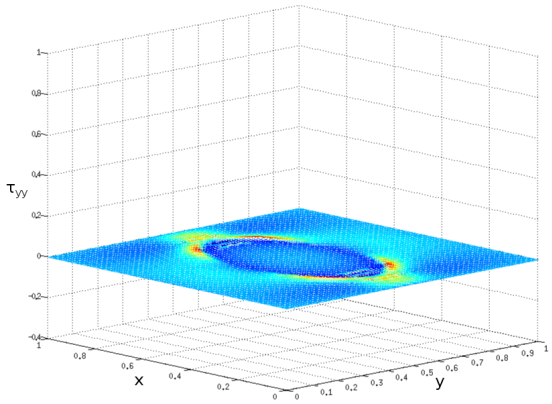
To verify that we have a lack of enrichment, we take a look at the pressure solution at the end of the simulation. Fig. 9.14 illustrates a comparison between the standard SEM approximation of the pressure against the XSEM approximation of the pressure.



(a) $We = 1$.



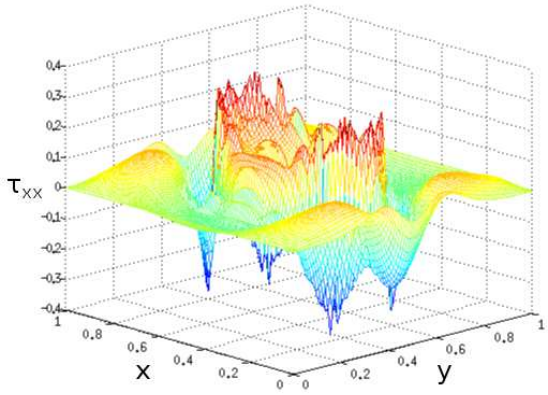
(b) $We = 10$.



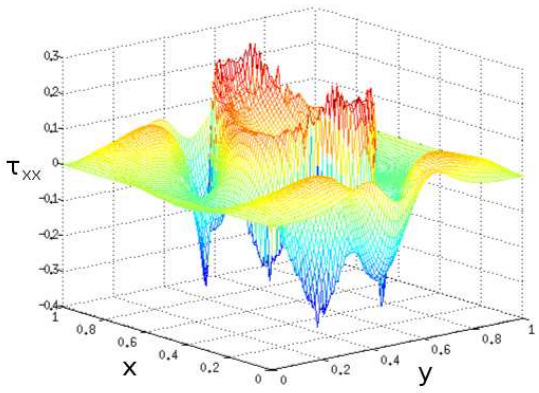
(c) $We = 100$.

Figure 9.12: Plots of the τ_{yy} component of the computed polymeric stress for $We = 1, 10, 100$, $h_f = 1/8$ and $N = 8$.

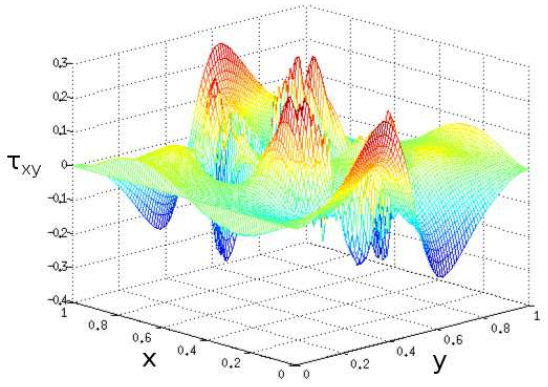
It is clear from Fig. 9.14 that the XSEM approximation is not improving the solution. This is quite disappointing, however - to a certain degree - not too surprising. The oscillations that are present in Fig. 9.14b are most likely caused by either blending problems, region of small support or a lack of enrichment as we found in §6.2. Taking that into account, it is not too surprising that the stress remains, mostly, unaffected. More research is required to determine the reason why XSEM does not improve the pressure solution in this case. We do not consider the plots for higher Weissenberg number in this case as they exhibit the same behaviour as is seen here.



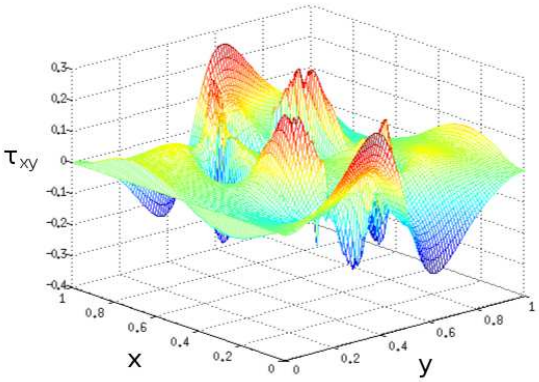
(a) Computed τ_{xx} solution for p_N.



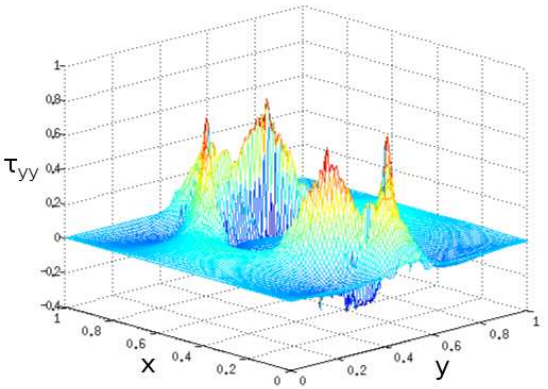
(b) Computed τ_{xx} solution for p_N^Γ.



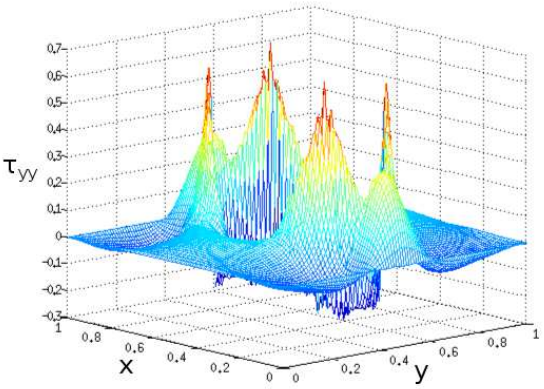
(c) Computed τ_{xy} solution for p_N.



(d) Computed τ_{xy} solution for p_N^Γ.

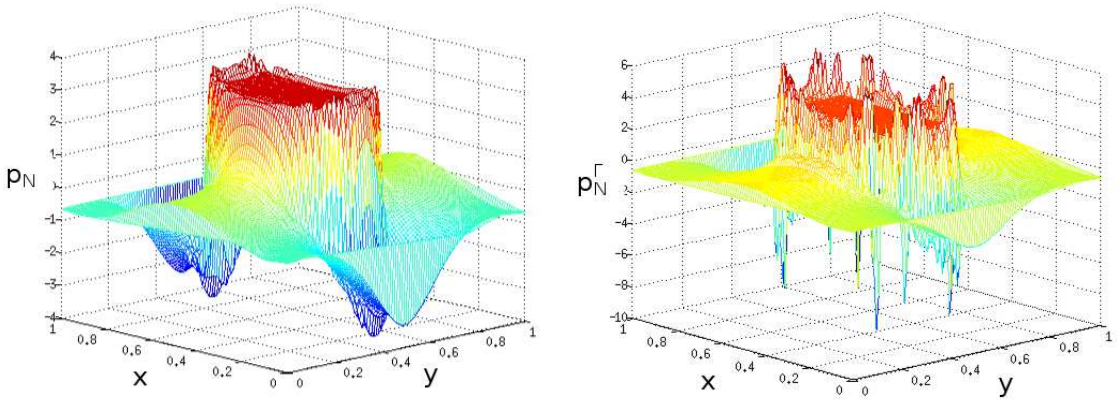


(e) Computed τ_{yy} solution for p_N.



(f) Computed τ_{yy} solution for p_N^Γ.

Figure 9.13: Plots of the components of the computed polymeric stress for $We = 1$, $h_f = 1/8$ and $N = 8$. Left hand column are the original SEM approximation and the right hand column are the XSEM approximation.



(a) p_N. **(b)** p_N^Γ.

Figure 9.14: Comparison of the standard SEM approximation against the enriched pressure solution for $We = 1$, $h_f = 1/8$ and $N = 8$.

9.3 Static Closed Shell

In this section we consider the SE-IBM applied to a thick structure immersed in an Oldroyd-B fluid and consider the p-type convergence of the method. We set up the example exactly as was done in §7.2.2. Again, we point out that the analytical solution (7.2.11) is continuous everywhere and linear in the region where the immersed structure resides, but it is not continuously differentiable. The analytical solution for the polymeric stress is $\boldsymbol{\tau} = 0$.

The parameters chosen for this example are similar to those given previously and are as follows: $Re = 1$, $We = 1, 10, 100$, $R = 0.25$, $\kappa = 1$, $w = 0.0625$, $\Delta t = 10^{-4}$ and the simulation was run for 300 timesteps. We chose to run the simulation for 300 timesteps because the maximum walltime available meant that we could not run 600 timesteps. We decompose the fluid domain into uniform quadrilateral spectral elements with mesh width $h_f = 1/8$ and let $N = 2, 4, 8$. Just as was done previously for the case of an immersed membrane in §8.1.1, we must choose the number of structure elements so that $h^s \leq h_f/2$. Just as we did in §7.2.2, we assume that the total number of elements is given by $K_b = K_b^s \times K_b^r$, where K_b^s is calculated using the relation in (5.4.9) and K_b^r is calculated using:

$$K_b^r \geq \frac{2w}{h_f} \tag{9.3.1}$$

where this time K_b^r and K_b^s are dependent on N. These values of K_b^s and K_b^r ensure that the decomposition of Ω_r^s is composed of approximately uniform quadrilateral elements. Again, the analytical solution given in (7.2.11) is the analytical solution for Stokes flow. Therefore, here we consider Stokes flow rather than Navier-Stokes.

Table 9.5 details the error and the orders of convergence for the velocity, pressure and polymeric stress at the end of the simulation for $We = 1$. According to Gerritsma

N	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{u} - \boldsymbol{u}_N\|_{H^1(\Omega)}$	Order
2	0.12269	-	3.13549	-
4	0.00570	4.42735	0.38900	3.01085
8	0.00041	3.80922	0.06476	2.58657

N	$\|p - p_N\|_{L^2(\Omega)}$	Order	$\|\boldsymbol{\tau} - \boldsymbol{\tau}_N\|_{L^2(\Omega)}$	Order
2	0.61734	-	0.08677	-
4	0.09180	2.74944	0.01632	2.41067
8	0.02366	1.95598	0.00369	2.14607

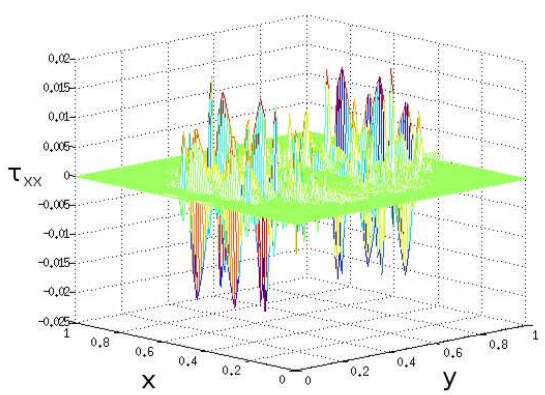
Table 9.5: Order of convergence with respect to N when $h_f = 1/8$ for a shell immersed in an Oldroyd-B fluid with $We = 1$.

and Phillips [34], for a velocity $\boldsymbol{u} \in (H^3(\Omega) \cap H_0^1(\Omega))^2$, $p \in H^2(\Omega)$ and $\boldsymbol{\tau} \in (H^2(\Omega))_s^4$, then we expect third-order convergence in the L^2 norm and second-order convergence in the H^1 norm for the velocity, an order of 1.5 for the pressure in the L^2 norm and second-order convergence for the polymeric stress in the L^2 norm. We can infer these orders from the Table 9.5 above. Once again we see that we have obtained the optimal order of convergence.

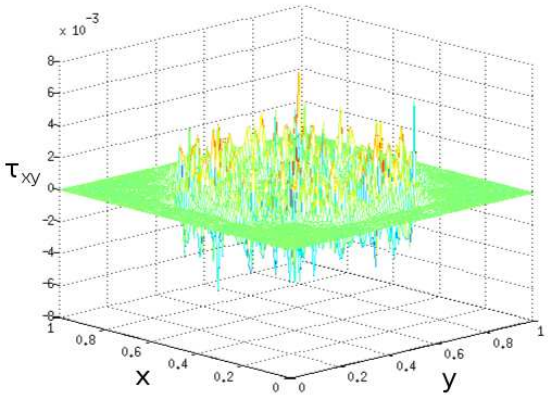
According to the theory of Gerritsma and Phillips [34], the regularity of the pressure does *not* affect the order for either the velocity or stress. However, the pressure's regularity should affect the order of convergence for the pressure solution. As we stated earlier, the pressure solution is not a H^2 function therefore we would expect an im-

paired order of convergence for the pressure. However we do not see an impaired order of convergence either in this section or in §8.3. As we mentioned previously, we are unaware of approximation errors for a spectral method over a broken Sobolev space or an interpolation space.

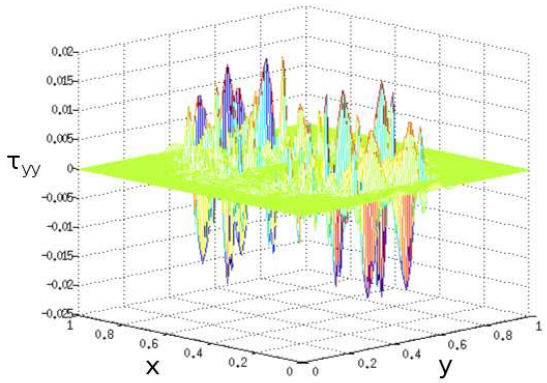
Fig. 9.15 illustrates the components of the computed polymeric stress solution for $We = 1$ and $N = 8$. Clearly all of the components have oscillations local to the shell.



(a) Computed τ_{xx} solution.



(b) Computed τ_{xy} solution.



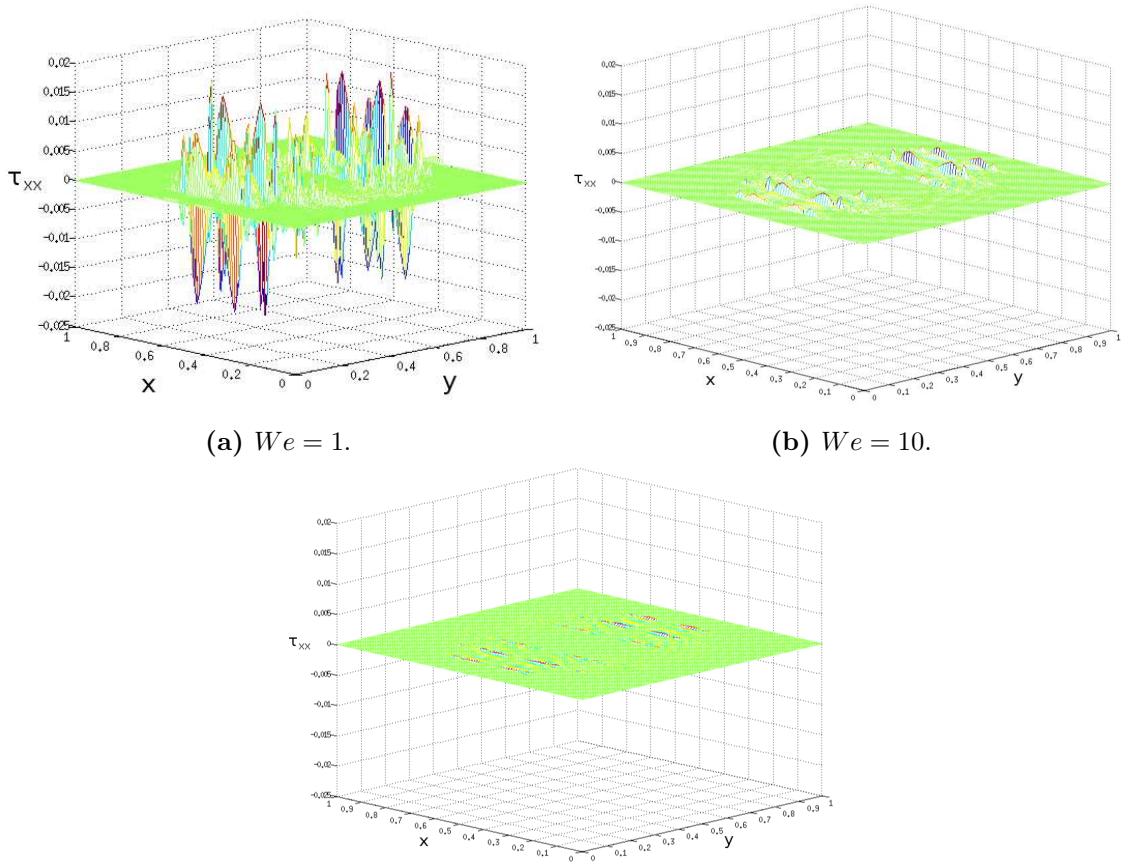
(c) Computed τ_{yy} solution.

Figure 9.15: Components of the computed polymeric stress for $h_f = 1/8$, $N = 8$ and $We = 1$.

We expected that these oscillations would be relatively smooth. However, they appear to be pretty sharp which is quite surprising given that the pressure solution is continuous. However, the spurious velocities that exist local to the structure could have fairly steep gradients and it is possible that these steep gradients cause the sharp oscillations seen in the polymeric stress. Additionally, the shell is entirely unconstrained and can move freely within the computational domain. Therefore, spurious velocities

will always be created local to the shell. Unfortunately, due to computational time we were unable to take the values of N any higher than 8. This is something we wish to address in the future.

Just as was done previously, we look at the effect of increasing Weissenberg number. Fig. 9.16 illustrates the τ_{xx} component for increasing Weissenberg number. We expect that the oscillations will decrease with increasing Weissenberg number just as we saw for the case of the immersed membrane. Again, we have used the same scale for the



(a) $We = 1$.

(b) $We = 10$.

(c) $We = 100$.

Figure 9.16: Plots of the computed τ_{xx} component for $h_f = 1/8$, $N = 8$ and $We = 1, 10, 100$.

axes so that the effect of Weissenberg number is emphasized. Once again, we see that as the Weissenberg number is increased, the oscillations local to the shell are greatly reduced. It is once again our belief that running the simulation for longer, would in fact cause these oscillations to grow and eventually break down the computation. We do not include the plots for the other components of the polymeric stress here as similar

behaviour can be seen.

9.4 Oscillating Closed Shell

The purpose of this section is apply the SE-IBM to an oscillating shell immersed in an Oldroyd-B fluid. Once again we consider decomposing our domain into uniform quadrilateral spectral elements of mesh width $h_f = 1/8$ and let $N = 8$. Once again our parameters are: $Re = 1$, $We = 1, 10, 100$, $R = 0.25$, $\Delta t = 10^{-4}$, $w = 0.0625$ and $\kappa = 1$. The simulation was once again run for 300 timesteps.

Fig. 9.17 illustrates the initial and final configuration of the immersed shell. As we

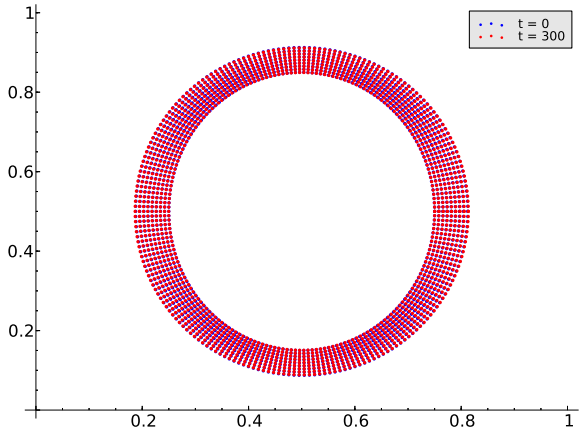
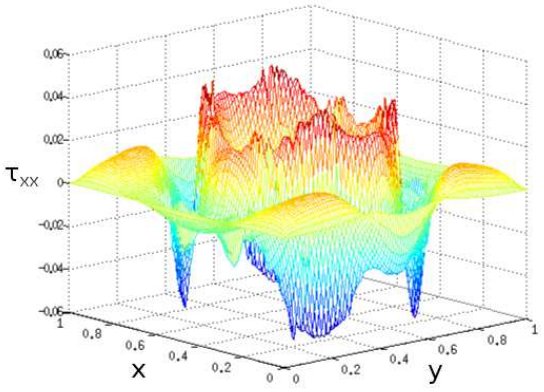


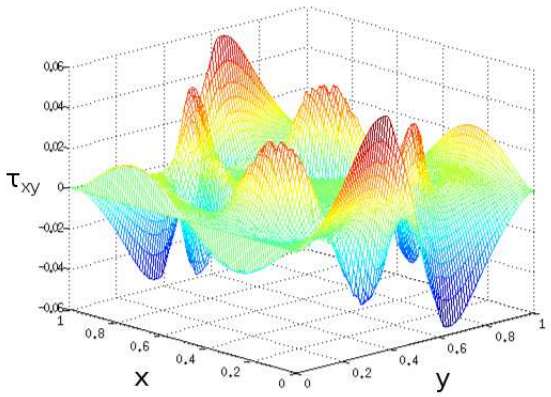
Figure 9.17: Immersed shell evolution for $N = 8$ and $h_f = 1/8$ when $We = 1$.

have considered $Re = 1$, the relaxation of the immersed shell from an ellipse to a circle is slow and, as we have only considered 300 timesteps, the amount of movement is very little and can barely be seen in Fig. 9.17. The blue points correspond to the initial configuration while the red correspond to the final configuration.

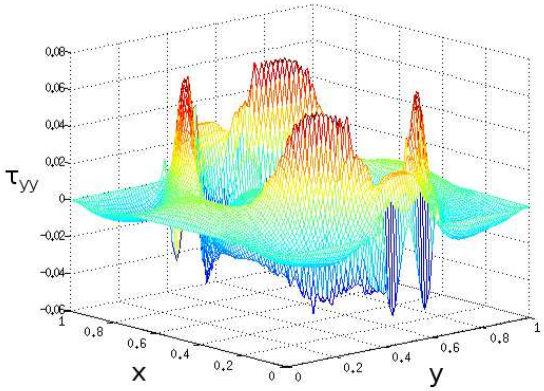
Fig. 9.18 illustrates the components of the computed polymeric stress when $We = 1$. We expect to see lots of oscillation local to the shell. We see similar behaviour to what we saw in §9.2.1 for the case of an immersed perturbed membrane. These oscillations are caused by spurious velocities that exist in the region where the structure resides.



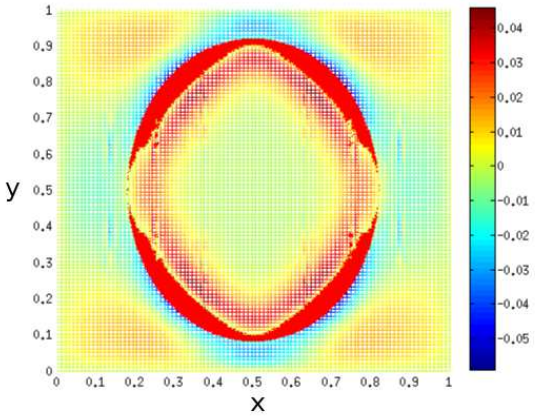
(a) Computed τ_{xx} solution.



(b) Computed τ_{xy} solution.



(c) Computed τ_{yy} solution.

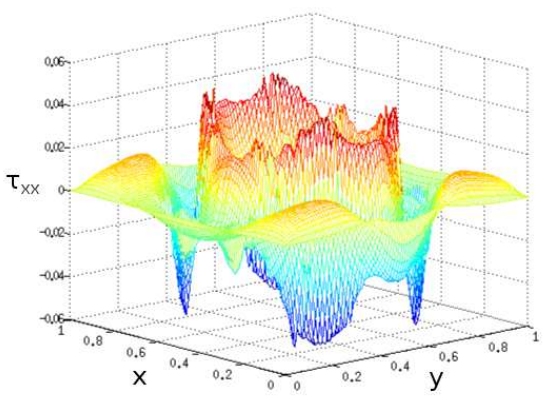


(d) Computed τ_{xx} solution looking down from above.

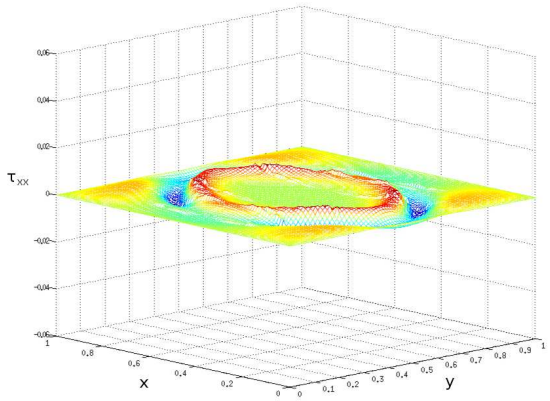
Figure 9.18: Plots of the computed polymeric stress components for $h_f = 1/8$, $N = 8$ and $We = 1$.

Fig. 9.18d illustrates that the largest stress can be seen local to the region where the shell resides (the red dots).

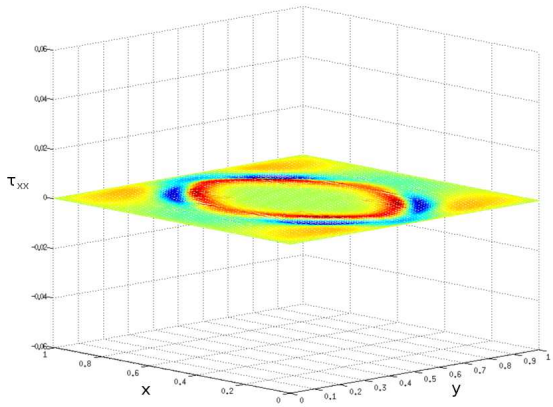
Finally we look at the effect of increasing the Weissenberg number. Fig. 9.19 illustrates the τ_{xx} component for increasing Weissenberg number. We expect that the oscillations will again decrease as the Weissenberg number is increased. Once again we have used the same scales for the axes so that the effect of the Weissenberg number can be seen clearly. As expected, and as we have seen throughout this chapter, as the Weissenberg number increases the oscillations decrease. We do not plot the other components because they behave similarly.



(a) $We = 1$.



(b) $We = 10$.



(c) $We = 100$.

Figure 9.19: Plots of the computed τ_{xx} component for $h_f = 1/8$, $N = 8$ and $We = 1, 10, 100$.

Chapter 10

Conclusions and Future Work

In this thesis, we have successfully applied a spectral element approximation to the finite element immersed boundary method (FE-IBM). FE-IBM, and immersed boundary methods in general, suffer from the limitation of assuming that the entire computational domain has a constant viscosity. While this may be a reasonable assumption in the case of an immersed fibre, membrane or boundary, it is not necessarily a good assumption in the case of an immersed shell or thick structure. In Chapter 3 we derived an extension to allow the thick immersed structure to have a different viscosity to that of the surrounding fluid. Unfortunately, we were unable to computationally validate our extension in this thesis but in the future we would like to apply the extension to the benchmark problem of flow past a sphere or cylinder where the sphere or cylinder will be ascribed very high viscosity so as to simulate a solid structure.

It is well known in the immersed boundary literature that the immersed boundary method can be stiff and therefore small timesteps may be required in order for the simulation to remain stable. The most common method used in the literature is the so-called FE/BE method where the Laplacian and gradient terms are dealt with implicitly and the force due to the immersed boundary is dealt with explicitly. The motion of the immersed boundary is usually governed by the semi-implicit Euler method. It is very well known that the immersed boundary method will struggle to converge at small viscosities and/or large forcing parameters. It is agreed that the best way to remove

the restriction on the timestep length is to consider a fully implicit coupled scheme. However, to do so would be computationally impractical and even the approximate implicit method introduced by Charles Peskin is not that useful because, although they allow larger timesteps, the computational time is increased essentially destroying any advantage that would be gained from using a larger timestep. In Chapter 4, we considered keeping the FE/BE representation but changing the evolution equation for the immersed boundary. We numerically investigated how the semi-implicit Euler method performed on a range of viscosities, stiffness parameters and timestep lengths. We confirmed that the semi-implicit Euler method struggled for small viscosities and/or large forcing parameters. We explored whether using an implicit method, to move the immersed boundary, would allow larger timestep lengths. This was indeed the case and using an implicit Euler method did allow slightly larger timesteps in certain cases. Additionally, we considered a third-order backward differentiation method and found that the method performed similarly to the implicit Euler but on certain cases allowed larger timesteps. We also considered a velocity correction scheme which is guaranteed to preserve the area contained inside an immersed closed membrane up to the order $O((\Delta t)^2)$. We found that the velocity correction scheme performed very badly for small viscosities. We also noticed that it was the ratio between the viscosity and the stiffness that was important. We found that the method was stabilised at large values of the stiffness if we increased the viscosity.

In Chapter 5 we introduced the spectral element method and illustrated the problems associated with approximating a function which is discontinuous, where the discontinuity is unfitted to the computational mesh. In such a scenario, oscillations can be seen local to the discontinuity; these oscillations are known as Gibbs phenomenon. We introduced the spectral element version of the extended finite element method which we called the extended spectral element method (XSEM). We illustrated that when approximating a function that is discontinuous, the XSEM can obtain the discontinuity exactly. Using the framework of A. Reusken we were able to obtain the spectral equivalents of his error estimates. We then discussed a possible inf-sup condition. The result of the inf-sup condition was slightly disappointing because the inf-sup parameter was still dependent on N. However, this is not that surprising as we assumed that the pressure approximation was of degree $N-2$. Another common problem within the immersed boundary method is the problem of so-called area loss. In the example of

a static closed membrane, the area contained inside the membrane is known to decay as the simulation progresses. In Chapter 5 we showed that using a high order polynomial can greatly improve the area conservation and therefore greatly improve the divergence-free constraint on the membrane.

In Chapter 6 we numerically investigated the XSEM. We showed that for a piecewise smooth function, one can still obtain exponential convergence. We then carried out a few numerical investigations to understand the effect of over-integrating the bilinear forms involving the enriched pressure term. We over-integrate in this thesis for simplicity and we showed that provided the number of quadrature points N_q is taken sufficiently high, good results can still be obtained. However, if N_q is too small the consequences can be disastrous.

After validating the Newtonian, Oldroyd-B and immersed boundary solvers in Chapter 7, we applied the SE-IBM to the same problems that we considered for the validation. In Chapter 8, when SE-IBM was applied to a static immersed membrane, we found optimal order of convergence for the velocity in both the L^2 and H^1 norms. This was quite surprising as we expected that the reduced regularity on the pressure would impair the order; this is what was seen in the FE-IBM. However, we believe that we obtain optimal order of convergence because the divergence-free constraint is better satisfied so that the pressure regularity does not affect the velocity error. Then we applied XSEM to the same example problem. We found that the majority of Gibbs phenomena was removed. However, there were some serious peaks which we believe are due to the value of N_q being too low. We also found that XSEM greatly reduced the spurious velocities which are normally present local to the membrane. Then we applied SE-IBM to an oscillating closed membrane and obtained good agreement with the validation. Applying XSEM to the oscillating closed membrane was unfortunately, disappointing. The pressure solution contained additional oscillations that were not seen when the standard SEM approximation was used. We are unsure of the reasons for the presence of these although we believe the oscillations are due to a combination of too small a value for N_q and inf-sup stability issues. Investigating this is a subject of future work.

The FE-IBM, and therefore SE-IBM, can deal with thick immersed structures. Therefore, in Chapters 7 and 8 we considered a thick immersed shell. Once again, SE-IBM was able to obtain optimal order of convergence for the velocity in the L^2 and H^1 norm, a result that once again was surprising. FE-IBM had sub-optimal order of convergence because, again, the lack of regularity on the pressure was impairing the order of convergence. Again, we attributed our optimal order of convergence to better area conservation and therefore a more divergence-free velocity.

Chapter 9 concerned non-Newtonian fluids, specifically an Oldroyd-B fluid. We applied the Oldroyd-B fluid to all of the immersed boundary benchmark problems that we had considered in Chapter 8. For the example of a closed membrane, both static and oscillating, the pressure discontinuity creates spurious velocities. These spurious velocities polluted the polymeric stress so that oscillations in the stress components were seen local to the membrane. Applying XSEM to that problem, reduced the majority of the oscillations seen in the polymeric stress components. This is a big advantage of using XSEM within viscoelastic fluids. In many cases, the oscillations that exist within the stress are propagated throughout the computation due to the hyperbolicity of the constitutive equation. However, for the case of an oscillating membrane, the enriched pressure approximation had little effect on the polymeric stress components. We also immersed a thick shell both static and oscillating in an Oldroyd-B fluid. These simulations produced some unexpected behaviour. As the pressure solution for an immersed shell is continuous, we expected to see a smoother solution for the polymeric stress than was seen for the immersed membrane. However, the polymeric stress components still contained rather sharp oscillations. We attributed these oscillations to the spurious velocities. An interesting phenomenon which we did not expect was that for all the Oldroyd-B examples considered in this thesis, the oscillations local the membrane or shell actually decrease with increasing Weissenberg. It is possible that these oscillations would grow if the simulation was run for longer. However, we do not know for sure.

There are a lot of topics which we would like to do in the future. As far as we are aware, a full comparison of the different methods, such as the original immersed boundary method, immersed finite element method and finite element immersed boundary method, and how they compare to the classical fluid-structure interaction formulation

(such as a partitioned approach) is absent from the immersed boundary literature and therefore we would like to carry out this comparison in the future. Additionally, we would like to look more closely at the possibility of an immersed boundary rheometer which was introduced by Dillon and Zhuo [31]. We would also like to numerically validate using variable and/or discontinuous viscosity within the immersed boundary method. The application of the SE-IBM to more realistic example problems, such as blood flow through a small blood vessel, is also something we would like to do in the future.

As far as we are aware, error estimates for the spectral approximation of a function from a broken Sobolev space or an interpolation space has not been considered in the literature and therefore we would like to look at this more closely in the future so that we may validate the p-type order of convergence we found for the pressure when a static membrane was immersed in a Newtonian fluid. Additionally, we would like to look at XSEM in more detail. We would like to consider higher-order (curved) interfaces and compare the over-integration scheme we used for the quadrature against the standard scheme used in the XFEM literature of subdividing the element which contains the discontinuity. We would also like to take a closer and more thorough look at the inf-sup condition for XSEM for both the velocity-pressure and velocity-pressure-stress formulations as well as removing regions of small support as suggested by Groß and Reusken [38]. We believe that XSEM could be very useful for viscoelastic flows and so we would like to apply XSEM to some viscoelastic benchmark problems such as contraction flows and die swell.

Throughout this thesis, there have been some parameter values which we have been unable to consider due to problems with computational time or memory issues and in the future we would like to be able to deal with those parameter values. In particular, we would like to run the high Weissenberg number examples considered in Chapter 9 for longer to see if the numerics break down. We would also like to consider the higher-order approximation of the immersed boundary position $\boldsymbol{X}$ as we have only considered linear approximations in this thesis. Finally, we like to look at the h-type (mesh width) convergence of the area when a medium polynomial degree is used, e.g. $N = 6, 8$.

Appendix A

Useful Definitions, Identities and Inequalities

A.1 Fréchet Derivative

The Fréchet derivative is a generalisation of the derivative of a function between Banach spaces. The definition given here is stated verbatim from [22]. Let A be a mapping between a Banach space X and a Banach space Y, i.e. $A\colon X \to Y$. We say A is Fréchet differentiable at a point $u_0 \in X$ if there exists a linear continuous operator L such that

$$\lim_{\substack{w \in X \\ \|w\|_X \to 0}} \frac{\|A(u_0 + w) - A(u_0) - Lw\|_Y}{\|w\|_X} = 0 \tag{A.1.1}$$

If this happens, the linear operator L is unique. It is termed the Fréchet derivative of A at the point u_0, and is denoted by $A'(u_0)$.

A.2 Minkowski's Inequality

We state Minkowski's inequality from [2]. Let $\Omega \subset \mathbb{R}^n$ and $1 \leq p < \infty$. If $u \in L^p(\Omega)$, $v \in L^p(\Omega)$ then $u + v \in L^p(\Omega)$ and

$$\|u + v\|_{L^p(\Omega)} \leq \|u\|_{L^p(\Omega)} + \|v\|_{L^p(\Omega)} \tag{A.2.1}$$

This inequality is also known as the triangle inequality.

Appendix B

Gauss-Lobatto Legendre Quadrature

For our numerical integration procedure we adopt Gauss-Lobatto Legendre quadrature. The fundamental concept of any numerical integration procedure is the approximation of the integral by a quadrature rule. Thus for a 1D function f defined on the interval $[-1, 1]$ we have:

$$\int_{-1}^{1} f(\xi)\, \mathrm{d}\xi \approx \sum_{p=0}^{N} w_p f(\xi_p) \tag{B.0.1}$$

where w_p are the weights and ξ_p represent specifically chosen abscissa of $N+1$ distinct points in $[-1, 1]$. Gaussian quadrature is particularly accurate for treating integrals where the integrand is smooth. For Gauss-Lobatto Legendre quadrature, the abscissa ξ_p, $p = 0, N$, are chosen to be the zeros of $(1 - \xi^2)L'_N(\xi)$ where L_N is the Legendre polynomial of degree N. The weights w_p are chosen so that the quadrature rule is exact for polynomials of degree less than or equal to $2N - 1$ [54], and are given by:

$$w_p = \int_{-1}^{1} h_p(\xi)\, \mathrm{d}\xi = \frac{2}{N(N+1)[L_N(\xi_p)]^2} \tag{B.0.2}$$

where h_p are the Lagrange interpolants defined in (5.2.15). The Gauss-Lobatto Legendre quadrature rule is easily extended to 2D:

$$\int_{-1}^{1}\int_{-1}^{1} f(\xi,\eta)\,\mathrm{d}\xi\,\mathrm{d}\eta \approx \sum_{p=0}^{N}\sum_{q=0}^{N} w_p w_q f(\xi_p,\eta_q) \tag{B.0.3}$$

where the weights are defined as above.

Appendix C

Immersed Boundary Appendix

C.1 Velocity Correction Scheme

The idea behind the Velocity Correction Scheme (VCS) is to ensure that $\nabla \cdot \boldsymbol{u} = 0$ in a discrete setting: i.e., $\nabla \cdot \boldsymbol{u}_N = 0$. The problem is to ensure that the area inside an enclosed membrane is conserved, i.e.

$$\int_{\Omega_i} \nabla \cdot \boldsymbol{u} \, \mathrm{d}\Omega = 0 \qquad \text{or} \qquad \int_{\partial\Omega_i} \boldsymbol{u} \cdot \boldsymbol{n} \, \mathrm{d}s = 0 \tag{C.1.1}$$

Discretising the boundary integral gives:

$$\int_{\partial\Omega_i} \boldsymbol{u} \cdot \boldsymbol{n} \, \mathrm{d}s \approx \sum_{j=1}^{K_b} \boldsymbol{u}_j \cdot \boldsymbol{n}_j \Delta s_j = 0 \tag{C.1.2}$$

where K_b is the number of segments. Suppose that we have a uniform discretisation of the interface $\partial\Omega_i$ so that the points are equally spaced. The tangent vector at the point $\boldsymbol{X}_i = (X_i, Y_i)$ is

$$\boldsymbol{t}_i = (X_{i+1} - X_{i-1}, Y_{i+1} - Y_{i-1}) \tag{C.1.3}$$

Thus the normal vector at the point $\boldsymbol{X}_i$ is given by:

$$\boldsymbol{n}_i = \pm\frac{(Y_{i+1} - Y_{i-1}, X_{i-1} - X_{i+1})}{|\boldsymbol{t}_i|} = \pm\frac{(Y_{i+1} - Y_{i-1}, X_{i-1} - X_{i+1})}{2\Delta s_i} \tag{C.1.4}$$

where $\Delta s_i = |\boldsymbol{t}_i|/2$. To enforce (C.1.2), determine the mean value of $\boldsymbol{u}_i \cdot \boldsymbol{n}_i$:

$$\mu = \frac{\sum_{i=1}^{K_b}(\boldsymbol{u}_i \cdot \boldsymbol{n}_i)\Delta s_i}{\sum_{i=1}^{K_b}\Delta s_i} \tag{C.1.5}$$

Then define the corrected velocity as: $\hat{\boldsymbol{u}}_i = \boldsymbol{u}_i - \mu\boldsymbol{n}_i$. Let $\boldsymbol{X}_i = (X_i, Y_i)$, as before, and $\boldsymbol{u}_i = (u_i, v_i)$ then the area contained inside a polygon is defined as:

$$\begin{aligned} A &= \frac{1}{2}\sum_{j=1}^{K_b}(X_{j-1}Y_j - X_jY_{j-1}) \\ &= \frac{1}{4}\sum_{j=1}^{K_b}X_j(Y_{j+1} - Y_{j-1}) + \frac{1}{4}\sum_{j=1}^{K_b}Y_j(X_{j-1} - X_{j+1}) \end{aligned} \tag{C.1.6}$$

Let $X_j^{n+1} = X_j^n + \Delta t\, u_j^{n+1}$ and $Y_j^{n+1} = Y_j^n + \Delta t\, v_j^{n+1}$ for $j = 1, \ldots, K_b$. Then the area at time t^{n+1} is given by:

$$
\begin{aligned}
A^{n+1} &= \frac{1}{4}\sum_{j=1}^{K_b} X_j^{n+1}(Y_{j+1}^{n+1} - Y_{j-1}^{n+1}) + \frac{1}{4}\sum_{j=1}^{K_b} Y_j^{n+1}(X_{j-1}^{n+1} - X_{j+1}^{n+1}) \\
&= \frac{1}{4}\sum_{j=1}^{K_b} \left(X_j^n + \Delta t\, u_j^{n+1}\right)\left(Y_{j+1}^n + \Delta t\, v_{j+1}^{n+1} - Y_{j-1}^n - \Delta t\, v_{j-1}^{n+1}\right) + \\
&\quad \frac{1}{4}\sum_{j=1}^{K_b} \left(Y_j^n + \Delta t\, v_j^{n+1}\right)\left(X_{j-1}^n + \Delta t\, u_{j-1}^{n+1} - X_{j+1}^n - \Delta t\, u_{j+1}^{n+1}\right) \\
&= \frac{1}{4}\sum_{j=1}^{K_b} X_j^n(Y_{j+1}^n - Y_{j-1}^n) + \frac{1}{4}\sum_{j=1}^{K_b} Y_j^n(X_{j-1}^n - X_{j+1}^n) + \\
&\quad \frac{\Delta t}{4}\sum_{j=1}^{K_b} X_j^n(v_{j+1}^{n+1} - v_{j-1}^{n+1}) + \frac{\Delta t}{4}\sum_{j=1}^{K_b} Y_j^n(u_{j-1}^{n+1} - u_{j+1}^{n+1}) + \\
&\quad \frac{\Delta t}{4}\sum_{j=1}^{K_b} u_j^{n+1}(Y_{j+1}^n - Y_{j-1}^n) + \frac{\Delta t}{4}\sum_{j=1}^{K_b} v_j^{n+1}(X_{j-1}^n - X_{j+1}^n) + \\
&\quad \frac{\Delta t^2}{4}\sum_{j=1}^{K_b} u_j^{n+1}(v_{j+1}^{n+1} - v_{j-1}^{n+1}) + \frac{\Delta t^2}{4}\sum_{j=1}^{K_b} v_j^{n+1}(u_{j-1}^{n+1} - u_{j+1}^{n+1}) \\
&= A^n + \frac{\Delta t}{2}\sum_{j=1}^{K_b} u_j^{n+1}(Y_{j+1}^n - Y_{j-1}^n) + \frac{\Delta t}{2}\sum_{j=1}^{K_b} v_j^{n+1}(X_{j-1}^n - X_{j+1}^n) + \\
&\quad \frac{\Delta t^2}{4}\sum_{j=1}^{K_b} u_j^{n+1}(v_{j+1}^{n+1} - v_{j-1}^{n+1}) + \frac{\Delta t^2}{4}\sum_{j=1}^{K_b} v_j^{n+1}(u_{j-1}^{n+1} - u_{j+1}^{n+1}) \qquad \text{(C.1.7)}
\end{aligned}
$$

Assuming that we have corrected our velocity so that (C.1.2) holds, then we have:

$$
\begin{aligned}
0 &= \sum_{j=1}^{K_b} \boldsymbol{u}_j \cdot \boldsymbol{n}_j \Delta s_j \\
&= \sum_{j=1}^{K_b} \boldsymbol{u}_j \cdot \left(\frac{(Y_{j+1} - Y_{j-1}, X_{j-1} - X_{j+1})}{2\Delta s_j}\right)\Delta s_j \\
&= \frac{1}{2}\sum_{j=1}^{K_b} u_j(Y_{j+1} - Y_{j-1}) + \frac{1}{2}\sum_{j=1}^{K_b} v_j(X_{j-1} - X_{j+1}) \qquad \text{(C.1.8)}
\end{aligned}
$$

Therefore, clearly the terms of $\mathcal{O}(\Delta t)$ in A^{n+1} vanish and the method preserves the area up to order $(\Delta t)^2$.

C.2 Semi-Implicit Euler Energy Estimate

In this section we derive the energy estimate for the semi-implicit Euler method. The derivation of the estimate given here, follows very closely to the derivation of the energy estimates provided by Newren *et al.* [52], and also resembles the discrete energy estimates given by Boffi *et al.* [15] which were then used to obtain a CFL condition [16]. First, we assume that the non-linear convection term which appears in the material derivative (2.2.3) is zero so that the momentum equation (2.2.11a) becomes:

$$\rho\frac{\partial \boldsymbol{u}}{\partial t} + \nabla p - \eta\nabla^2\boldsymbol{u} = \boldsymbol{f} \tag{C.2.1}$$

where, for simplicity, we have let $\eta = \eta_0$ be the viscosity of the fluid. Also, we define $\rho = 1$. Temporally discretising the momentum equation above gives:

$$\frac{\boldsymbol{u}^{n+1} - \boldsymbol{u}^n}{\Delta t} + \nabla p^{n+1} - \eta\nabla^2\boldsymbol{u}^{n+1} = \boldsymbol{f}^n \tag{C.2.2}$$

Take the inner product of the momentum equation above with $\boldsymbol{u}^{n+1}$ and the inner product of the incompressibility constraint (2.2.11b) with p^{n+1} to give:

$$\left(\frac{\boldsymbol{u}^{n+1} - \boldsymbol{u}^n}{\Delta t}, \boldsymbol{u}^{n+1}\right)_\Omega - \eta\left(\nabla^2\boldsymbol{u}^{n+1}, \boldsymbol{u}^{n+1}\right)_\Omega - \left(p^{n+1}, \nabla\cdot\boldsymbol{u}^{n+1}\right)_\Omega = \left(\boldsymbol{f}^n, \boldsymbol{u}^{n+1}\right)_\Omega \tag{C.2.3a}$$

$$\left(p^{n+1}, \nabla\cdot\boldsymbol{u}^{n+1}\right)_\Omega = 0 \tag{C.2.3b}$$

where the inner product $(\cdot,\cdot)_\Omega \equiv (\cdot,\cdot)_{L^2(\Omega)}$. Thus we have:

$$\frac{1}{\Delta t}\left(\boldsymbol{u}^{n+1} - \boldsymbol{u}^n, \boldsymbol{u}^{n+1}\right)_\Omega - \eta\left(\nabla^2\boldsymbol{u}^{n+1}, \boldsymbol{u}^{n+1}\right)_\Omega = \left\langle \boldsymbol{f}^n, \boldsymbol{u}^{n+1}\right\rangle \tag{C.2.4}$$

where $\langle\cdot,\cdot\rangle$ denotes a duality pairing. Then using the identity

$$\left(\boldsymbol{u}^{n+1} - \boldsymbol{u}^n, \boldsymbol{u}^{n+1}\right)_\Omega = \frac{1}{2}\left[\left(\boldsymbol{u}^{n+1}, \boldsymbol{u}^{n+1}\right)_\Omega - \left(\boldsymbol{u}^n, \boldsymbol{u}^n\right)_\Omega + \left(\boldsymbol{u}^{n+1} - \boldsymbol{u}^n, \boldsymbol{u}^{n+1} - \boldsymbol{u}^n\right)_\Omega\right] \tag{C.2.5}$$

and the definition $\|\boldsymbol{u}\|_\Omega^2 = (\boldsymbol{u}, \boldsymbol{u})_\Omega$, we arrive at:

$$\frac{1}{2\Delta t}\left[\left\|\boldsymbol{u}^{n+1}\right\|_\Omega^2 - \|\boldsymbol{u}^n\|_\Omega^2\right] - \eta\left(\nabla^2\boldsymbol{u}^{n+1}, \boldsymbol{u}^{n+1}\right)_\Omega = \frac{-1}{2\Delta t}\left\|\boldsymbol{u}^{n+1} - \boldsymbol{u}^n\right\|_\Omega^2 + \left\langle \boldsymbol{f}^n, \boldsymbol{u}^{n+1}\right\rangle \tag{C.2.6}$$

where, using the definition of the source term (3.1.16b) and the sifting property of the Dirac delta function, we can write:

$$\left\langle \boldsymbol{f}^n, \boldsymbol{u}^{n+1}\right\rangle = \int_\Gamma \kappa\frac{\partial^2\boldsymbol{X}^n}{\partial s^2}\cdot\boldsymbol{u}^{n+1}(\boldsymbol{X}^n(s))\,\mathrm{d}s = \left(\kappa\frac{\partial^2\boldsymbol{X}^n}{\partial s^2}, \boldsymbol{u}^{n+1}(\boldsymbol{X}^n(s))\right)_\Gamma \tag{C.2.7}$$

where s denotes the arclength parameter which describes the immersed boundary. Following Newren *et al.* [52] we take the inner product of the immersed boundary evolution equation (4.2.1) with $-\kappa\frac{\partial^2\boldsymbol{X}^{n+1}}{\partial s^2}$ and add it to the momentum equation to give:

$$\begin{aligned}\frac{1}{2\Delta t}\left[\left\|\boldsymbol{u}^{n+1}\right\|_\Omega^2 - \|\boldsymbol{u}^n\|_\Omega^2\right] - \eta\left(\nabla^2\boldsymbol{u}^{n+1}, \boldsymbol{u}^{n+1}\right)_\Omega + \frac{1}{\Delta t}\left(\boldsymbol{X}^{n+1} - \boldsymbol{X}^n, -\kappa\frac{\partial^2\boldsymbol{X}^{n+1}}{\partial s^2}\right)_\Gamma = \\ -\frac{1}{2\Delta t}\left\|\boldsymbol{u}^{n+1} - \boldsymbol{u}^n\right\|_\Omega^2 + \int_\Gamma \kappa\frac{\partial^2\boldsymbol{X}^n}{\partial s^2}\cdot\boldsymbol{u}^{n+1}(\boldsymbol{X}^n(s))\,\mathrm{d}s + \int_\Gamma -\kappa\frac{\partial^2\boldsymbol{X}^{n+1}}{\partial s^2}\cdot\boldsymbol{u}^{n+1}(\boldsymbol{X}^n(s))\,\mathrm{d}s\end{aligned} \tag{C.2.8}$$

As the Laplacian is negative definite and $\|\cdot\|_\Omega^2 \geq 0$, the above simplifies to:

$$\begin{aligned}&\frac{1}{2\Delta t}\left[\left\|\boldsymbol{u}^{n+1}\right\|_\Omega^2 - \|\boldsymbol{u}^n\|_\Omega^2\right] + \frac{1}{\Delta t}\left(\boldsymbol{X}^{n+1} - \boldsymbol{X}^n, -\kappa\frac{\partial^2\boldsymbol{X}^{n+1}}{\partial s^2}\right)_\Gamma \\ &\leq \kappa\int_\Gamma \frac{\partial}{\partial s}\left(\boldsymbol{X}^{n+1} - \boldsymbol{X}^n\right)\cdot\frac{\partial\boldsymbol{u}^{n+1}(\boldsymbol{X}^n(s))}{\partial s}\,\mathrm{d}s\end{aligned} \tag{C.2.9}$$

Newren *et al.* [52] stated that as the operator $-\kappa\partial^2/\partial s^2$ is both linear and self-adjoint, then the following identity must hold:

$$\begin{aligned}\left(\boldsymbol{X}^{n+1} - \boldsymbol{X}^n, -\kappa\frac{\partial^2\boldsymbol{X}^{n+1}}{\partial s^2}\right)_\Gamma = \frac{1}{2}\Bigg[&\left(\boldsymbol{X}^{n+1}, -\kappa\frac{\partial^2\boldsymbol{X}^{n+1}}{\partial s^2}\right)_\Gamma - \left(\boldsymbol{X}^n, -\kappa\frac{\partial^2\boldsymbol{X}^n}{\partial s^2}\right)_\Gamma + \\ &\left(\boldsymbol{X}^{n+1} - \boldsymbol{X}^n, -\kappa\frac{\partial^2}{\partial s^2}\left(\boldsymbol{X}^{n+1} - \boldsymbol{X}^n\right)\right)_\Gamma\Bigg]\end{aligned} \tag{C.2.10}$$

Substutiting the above identity into (C.2.9) and applying the negative definiteness of the operator $\partial^2/\partial s^2$ gives:

$$\frac{1}{2\Delta t}\left[\left\|\boldsymbol{u}^{n+1}\right\|_\Omega^2 - \left\|\boldsymbol{u}^n\right\|_\Omega^2 + \left(\boldsymbol{X}^{n+1}, -\kappa\frac{\partial^2 \boldsymbol{X}^{n+1}}{\partial s^2}\right)_\Gamma - \left(\boldsymbol{X}^n, -\kappa\frac{\partial^2 \boldsymbol{X}^n}{\partial s^2}\right)_\Gamma\right]$$
$$\leq \kappa\int_\Gamma \frac{\partial}{\partial s}\left(\boldsymbol{X}^{n+1} - \boldsymbol{X}^n\right)\cdot\frac{\partial \boldsymbol{u}^{n+1}(\boldsymbol{X}^n(s))}{\partial s}\,\mathrm{d}s \tag{C.2.11}$$

Now that we have obtained the continuous energy estimate (C.2.11) we can derive the discrete energy estimate. Let $\boldsymbol{X}^{n+1} \approx \boldsymbol{X}_h^{n+1}$ and $\boldsymbol{u}^{n+1} \approx \boldsymbol{u}_N^{n+1}$ be the finite element and spectral element approximations of the IB position $\boldsymbol{X}$ and velocity $\boldsymbol{u}$ at time level t^{n+1}, respectively. We begin by considering the right hand side of (C.2.11). We have

$$\kappa\int_\Gamma \frac{\partial}{\partial s}\left(\boldsymbol{X}^{n+1} - \boldsymbol{X}^n\right)\cdot\frac{\partial \boldsymbol{u}^{n+1}(\boldsymbol{X}^n(s))}{\partial s}\,\mathrm{d}s =$$
$$\kappa\sum_{i=1}^{K_b}\int_{s_{i-1}}^{s_i}\left(\frac{\boldsymbol{X}_{h,i}^{n+1} - \boldsymbol{X}_{h,i-1}^{n+1}}{\Delta s_i} - \frac{\boldsymbol{X}_{h,i}^{n} - \boldsymbol{X}_{h,i-1}^{n}}{\Delta s_i}\right)\cdot\frac{\partial \boldsymbol{u}_N^{n+1}(\boldsymbol{X}_h^n(s))}{\partial s}\,\mathrm{d}s \tag{C.2.12}$$

where K_b is the number of immersed boundary elements, $\boldsymbol{X}_{h,i} = \boldsymbol{X}_h(s_i)$ and $\Delta s_i = s_i - s_{i-1}$. Throughout this thesis, $\boldsymbol{X}_h$ is linear, therefore $\partial \boldsymbol{X}_h/\partial s$ are constants. Thus

$$\kappa\sum_{i=1}^{K_b}\left(\frac{\boldsymbol{X}_{h,i}^{n+1} - \boldsymbol{X}_{h,i-1}^{n+1}}{\Delta s_i} - \frac{\boldsymbol{X}_{h,i}^{n} - \boldsymbol{X}_{h,i-1}^{n}}{\Delta s_i}\right)\int_{s_{i-1}}^{s_i}\frac{\partial \boldsymbol{u}_N^{n+1}(\boldsymbol{X}_h^n(s))}{\partial s}\,\mathrm{d}s =$$
$$\kappa\sum_{i=1}^{K_b}\left(\frac{\boldsymbol{X}_{h,i}^{n+1} - \boldsymbol{X}_{h,i-1}^{n+1}}{\Delta s_i} - \frac{\boldsymbol{X}_{h,i}^{n} - \boldsymbol{X}_{h,i-1}^{n}}{\Delta s_i}\right)\left(\boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i}^n) - \boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i-1}^n)\right) \tag{C.2.13}$$

Finally, consider

$$\begin{aligned}\left(\boldsymbol{X}_h^k, -\kappa\frac{\partial^2 \boldsymbol{X}_h^k}{\partial s^2}\right)_\Gamma &= \kappa\left(\frac{\partial \boldsymbol{X}_h^k}{\partial s}, \frac{\partial \boldsymbol{X}_h^k}{\partial s}\right)_\Gamma \\ &= \kappa\int_\Gamma\left(\frac{\partial \boldsymbol{X}_h^k}{\partial s}\right)^2\,\mathrm{d}s \\ &= \kappa\sum_{i=1}^{K_b}\left(\frac{\boldsymbol{X}_{h,i}^k - \boldsymbol{X}_{h,i-1}^k}{\Delta s_i}\right)^2\int_{s_{i-1}}^{s_i}\,\mathrm{d}s \\ &= \kappa\sum_{i=1}^{K_b}\frac{\left(\boldsymbol{X}_{h,i}^k - \boldsymbol{X}_{h,i-1}^k\right)^2}{\Delta s_i}\end{aligned} \tag{C.2.14}$$

where $k = n, n+1$. Therefore, the discrete energy estimate is given by:

$$\left\|\boldsymbol{u}^{n+1}\right\|_{\Omega}^{2} + \kappa \sum_{i=1}^{K_b} \left[\frac{\left(\boldsymbol{X}_{h,i}^{n+1} - \boldsymbol{X}_{h,i-1}^{n+1}\right)^2}{\Delta s_i} \right] - \left\|\boldsymbol{u}^{n}\right\|_{\Omega}^{2} - \kappa \sum_{i=1}^{K_b} \left[\frac{\left(\boldsymbol{X}_{h,i}^{n} - \boldsymbol{X}_{h,i-1}^{n}\right)^2}{\Delta s_i} \right]$$
$$\leq 2\kappa \Delta t \sum_{i=1}^{K_b} \left[\frac{\boldsymbol{X}_{h,i}^{n+1} - \boldsymbol{X}_{h,i-1}^{n+1}}{\Delta s_i} - \frac{\boldsymbol{X}_{h,i}^{n} - \boldsymbol{X}_{h,i-1}^{n}}{\Delta s_i} \right] \left(\boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i}^{n}) - \boldsymbol{u}_N^{n+1}(\boldsymbol{X}_{h,i-1}^{n}) \right) \tag{C.2.15}$$

Bibliography

[1] S. Abbas, A. Alizada, and T.-P. Fries. The XFEM for High-Gradient Solutions in Convection-Dominated Problems. *Int. J. Numer. Meth. Engng.*, 82:1044–1072, 2010.

[2] R. A. Adams and J. J. F. Fournier. *Sobolev Spaces*, volume 140 of *Pure and Applied Mathematics*. Academic Press, 2005.

[3] K. M. Arthurs, L. C. Moore, C. S. Peskin, E. B. Pitman, and H. E. Layton. Modelling Arteriolar Flow and Mass Transport Using the Immersed Boundary Method. *J. Comp. Phys.*, 147:402–440, 1998.

[4] I. Babuška and J. M. Melenk. The Partition of Unity Method. *Int. J. Numer. Meth. Engng.*, 40:727–758, 1997.

[5] R. Becker, E. Burman, and P. Hansbo. A Nitsche eXtended Finite Element Method for Incompressible Elasticity with Discontinuous Modulus of Elasticity. *Comput. Methods Appl. Mech. Engrg.*, 198:3352–3360, 2009.

[6] T. Belytschko, N. Moës, S. Usui, and C. Parimi. Arbitrary Discontinuities in Finite Elements. *Int. J. Numer. Meth. Engng.*, 50:993–1013, 2001.

[7] C. Bernardi and Y. Maday. *Approximations Spectrales de Problèmes aux Limites Elliptiques*. Mathématiques and Applications. Springer-Verlag, 1992.

[8] C. Bernardi and Y. Maday. Uniform Inf-Sup Conditions for the Spectral Discretization of the Stokes Problem. *M3AS*, 9(3):395–414, 1999.

[9] R. P. Beyer and R. J. Leveque. Analysis of a One-Dimensional Model for the Immersed Boundary Method. *SIAM J. Numer. Anal.*, 29(2):332–364, 1992.

[10] R. P. Beyer Jr. A Computational Model of the Cochlea Using the Immersed Boundary Method. *J. Comp. Phys.*, 98:145–162, 1992.

[11] D. Boffi, N. Cavallini, F. Gardini, and L. Gastaldi. Immersed Boundary Method: Performance Analysis of Popular Finite Element Spaces. In M. Papadrakakis, E. Onate, and B. Schrefler, editors, *IV International Conference on Computational Methods for Coupled Problems in Science and Engineering*, 2011.

[12] D. Boffi, N. Cavallini, F. Gardini, and L. Gastaldi. Local Mass Conservation of Stokes Finite Elements. *J. Sci. Comput.*, 52:383–400, 2012.

[13] D. Boffi, N. Cavallini, and L. Gastaldi. Finite Element Approach to the Immersed Boundary Method with Different Fluid and Solid Densities. *M3AS*, 21(12):2523–2550, 2011.

[14] D. Boffi and L. Gastaldi. A Finite Element Approach for the Immersed Boundary Method. *Computers and Structures*, 81:491–501, 2003.

[15] D. Boffi, L. Gastaldi, and L. Heltai. Numerical Stability of the Finite Element Immersed Boundary Method. *M3AS*, 17(10):1479–1505, 2006.

[16] D. Boffi, L. Gastaldi, and L. Heltai. On the CFL Condition for the Finite Element Immersed Boundary Method. *Computers and Structures*, 85:775–783, 2007.

[17] D. Boffi, L. Gastaldi, L. Heltai, and C. S. Peskin. On the Hyper-Elastic Formulation of the Immersed Boundary Method. *Comput. Methods Appl. Mech. Engrg.*, 197:2210–2231, 2008.

[18] D. C. Bottino. Modeling Viscoelastic Networks and Cell Deformation in the Context of the Immersed Boundary Method. *J. Comp. Phys.*, 147:86–113, 1998.

[19] S. C. Brenner and L. R. Scott. *The Mathematical Theory of Finite Element Methods*. Texts in Applied Mathematics. Springer Science+Business Media, LLC, 3 edition, 2010.

[20] F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods*. Computational Mathematics. Springer-Verlag, 1991.

[21] A. Buffa and C. Ortner. Compact Embeddings of Broken Sobolev Spaces and Applications. *IMA J. Numer. Anal.*, 29:827–855, 2009.

[22] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, Jr. *Spectral Methods: Fundamentals in Single Domains*. Scientific Computation. Springer-Verlag, 2006.

[23] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, Jr. *Spectral Methods: Evolution to Complex Geometries and Applications to Fluid Dynamics*. Scientific Computation. Springer-Verlag, 2007.

[24] E. O. Carew, P. Townsend, and M. F. Webster. Taylor-Galerkin Algorithms for Viscoelastic Flow: Application to a Model Problem. *Numerical Methods for Partial Differential Equations*, 10:171–190, 1994.

[25] P. Chadwick. *Continuum Mechanics: Concise Theory and Problems*. Dover-Publications, Inc., 1999.

[26] K. W. Cheng and T.-P. Fries. Higher-Order XFEM for Curved Strong and Weak Discontinuities. *Int. J. Numer. Meth. Engng.*, 82:564–590, 2010.

[27] J. Chessa and T. Belytschko. Arbitrary Discontinuities in Space-Time Finite Elements by Level Sets and X-FEM. *Int. J. Numer. Meth. Engng.*, 61:2595–2614, 2004.

[28] J. Chessa, H. Wang, and T. Belytschko. On the Construction of Blending Elements for Local Partition of Unity Enriched Finite Elements. *Int. J. Numer. Meth. Engng.*, 57:1015–1038, 2003.

[29] Y. J. Choi, M. A. Hulsen, and H. E. H. Meijer. Simulation of the Flow of a Viscoelastic Fluid around a Stationary Cylinder using an eXtended Finite Element Method. *Computers and Fluids*, 57:183–194, 2012.

[30] S. Deparis, M. Discacciati, and A. Quarteroni. A Domain Decomposition Framework for Fluid-Structure Interaction Problems. In C. Groth and D. W. Zingg, editors, *Computational Fluid Dynamics 2004*, pages 41–58. Springer Berlin Heidelberg, 2004.

[31] R. H. Dillon and J. Zhuo. Using the Immersed Boundary Method to Model Complex Fluids-Structure Interaction in Sperm Motility. *Disc. Cont. Dyn. Sys. Series B*, 15(2):343–355, 2011.

[32] A. L. Fogelson and C. S. Peskin. A Fast Numerical Method for Solving the Three-Dimensional Stokes' Equations in the Presence of Suspended Particles. *J. Comp. Phys.*, 79:50–69, 1988.

[33] M. I. Gerritsma and T. N. Phillips. Discontinuous Spectral Element Approximations for the Velocity-Pressure-Stress Formulation of the Stokes Problem. *Int. J. Numer. Meth. Engng.*, 43:1401–1419, 1998.

[34] M. I. Gerritsma and T. N. Phillips. Compatible Spectral Approximations for the Velocity-Pressure-Stress Formulation of the Stokes Problem. *SIAM J. Sci. Comput.*, 20(4):1530–1550, 1999.

[35] A. Gerstenberger and W. A. Wall. An eXtended Finite Element Method/Lagrange Multiplier based Approach for Fluid-Structure Interaction. *Comput. Methods Appl. Mech. Engrg.*, 197:1699–1714, 2008.

[36] A. Gerstenberger and W. A. Wall. Enhancement of Fixed-Grid Methods Towards Complex Fluid-Structure Interaction Applications. *Int. J. Numer. Meth. Fluids*, 57:1227–1248, 2008.

[37] A. J. Gil, A. Arranz Carreño, J. Bonet, and O. Hassan. The Immersed Structural Potential Method for Haemodynamic Applications. *J. Comp. Phys.*, 229:8613–8641, 2010.

[38] S. Groß and A. Reusken. An eXtended Pressure Finite Element Space for Two-Phase Incompressible Flows with Surface Tension. *J. Comput. Phys.*, 224:40–58, 2007.

[39] T. J. R. Hughes, W. K. Liu, and T. K. Zimmerman. LagrangianEulerian Finite Element Formulation for Incompressible Viscous Flows. *Comput. Methods Appl. Mech. Engrg.*, 29:329, 1981.

[40] G. E. Karniadakis and S. Sherwin. *Spectral/hp Element Methods for Computational Fluid Dynamics*. Numerical Mathematics and Scientific Computation. Oxford University Press, 2005.

[41] R. Kynch. *Numerical Investigation of Sedimentation in Viscoelastic Fluids using Spectral Element Methods*. PhD thesis, Cardiff University, June 2013.

[42] A. Legay, H. W. Wang, and T. Belytschko. Strong and Weak Arbitrary Discontinuities in Spectral Finite Elements. *Int. J. Numer. Meth. Engng.*, 64:991–1008, 2005.

[43] A. J. Lew and G. C. Guscaglia. A Discontinuous-Galerkin based Immersed Boundary Method. *Int. J. Numer. Meth. Engng.*, 76:427–454, 2008.

[44] M. N. Linnick and H. F. Fasel. A High-Order Immersed Interface Method for Simulating Unsteady Incompressible Flows on Irregular Domains. *J. Comp. Phys.*, 204:157–192, 2005.

[45] W. K. Liu, D. W. Kim, and S. Tang. Mathematical Foundations of the Immersed Finite Element Method. *Comput. Mech.*, 39:211–222, 2007.

[46] W. K. Liu, Y. Liu, D. Farrell, L. Zhang, X. S. Wang, Y. Fukui, N. Patankar, Y. Zhang, C. Bajaj, J. Lee, J. Hong, X. Chen, and H. Hsu. Immersed Finite Element Method and its Applications to Biological Systems. *Comput. Methods Appl. Mech. Engrg.*, 195:1722–1749, 2006.

[47] Y. Maday and A. T. Patera. Spectral Element Methods for the Incompressible Navier-Stokes Equations. In A. K. Noor and J. T. Oden, editors, *State-of-the-Art Surveys on Computational Mechanics*, chapter 3, pages 71–143. ASME, 1989.

[48] Y. Maday, A. T. Patera, and E. M. Rønquist. An Operator-Integration-Factor Splitting Method for Time-Dependent Problems: Application to Incompressible Fluid Flow. *J. Sci. Comput.*, 5(4):263–292, 1990.

[49] Y. Maday, A. T. Patera, and E. M. Rønquist. The $P_N \times P_{N-2}$ Method for the Approximation of the Stokes Problem. Internal Report 92025, Laboratoire d'Analyse Numérique, Université Pierre et Marie Curie, Paris, 1992.

[50] N. Moës, J. Dolbow, and T. Belytschko. A Finite Element Method for Crack Growth Without Remeshing. *Int. J. Numer. Meth. Engng.*, 46:131–150, 1999.

[51] Y. Mori and C. S. Peskin. Implicit Second-Order Immersed Boundary Methods with Boundary Mass. *Comput. Methods Appl. Mech. Engrg.*, 197:2049–2067, 2008.

[52] E. P. Newren, A. L. Fogelson, R. D. Guy, and R. M. Kirby. Unconditionally Stable Discretizations of the Immersed Boundary Equations. *J. Comp. Phys.*, 222:702–719, 2007.

[53] R. K. Noutcheuwa and R. G. Owens. A New Incompressible Smoothed Particle Hydrodynamics-Immersed Boundary Method. *Int. J. Numer. Anal. Mod. B*, 3:126–167, 2012.

[54] R. G. Owens and T. N. Phillips. *Computational Rheology*. Imperial College Press, 2002.

[55] A. T. Patera. A Spectral Element Method for Fluid Dynamics: Laminar Flow in a Channel Expansion. *J. Comput. Phys.*, 54:468–488, 1984.

[56] C. S. Peskin. Flow Patterns Around Heart Valves: A Numerical Method. *J. Comp. Phys.*, 10:252–271, 1972.

[57] C. S. Peskin. Numerical Analysis of Blood Flow in the Heart. *J. Comp. Phys.*, 25:220–252, 1977.

[58] C. S. Peskin. The Immersed Boundary Method. *Acta Numerica*, pages 479–517, 2002.

[59] C. S. Peskin and B. F. Printz. Improved Volume Conservation in the Computation of Flows with Immersed Elastic Boundaries. *J. Comp. Phys.*, 105:33–46, 1993.

[60] A. Reusken. Analysis of an eXtended Pressure Finite Element Space for Two-Phase Incompressible Flows. *Comput. Visual Sci.*, 11:293–305, 2008.

[61] I. Robertson, S. Sherwin, and J. Graham. Comparison of Wall Boundary Conditions for Numerical Viscous Free Surface Flow Simulation. *J. Fluids Struct.*, 19(4):525–542, 2004.

[62] C. R. Schneidesch and M. O. Deville. Chebyshev Collocation Method and Multidomain Decomposition for Navier-Stokes Equations in Complex Curved Geometries. *J. Comput. Phys.*, 106:234–257, 1993.

[63] L. Shi, T.-W. Pan, and R. Glowinski. Numerical Simulation of Lateral Migration of Red Blood Cells in Poiseuille Flows. *Int. J. Numer. Meth. Fluids*, 68(11):1393–1408, 2012.

[64] J. M. Stockie and B. R. Wetton. Analysis of Stiffness in the Immersed Boundary Method and Implications of Time-Stepping Schemes. *J. Comp. Phys.*, 154:41–64, 1999.

[65] J. M. Stockie and B. T. R. Wetton. Stability Analysis for the Immersed Fiber Problem. *SIAM J. Appl. Math.*, 55(6):1577–1591, 1995.

[66] J. Teran, L. Fauci, and M. Shelley. Peristaltic Pumping and Irreversibility of a Stokesian Viscoelastic Fluid. *Phys. Fluids*, 20:073101, 2008.

[67] C. Tu and C. S. Peskin. Stability and Instability in the Computation of Flows with Moving Immersed Boundaries: A Comparison of Three Methods. *SIAM J. Sci. Stat. Comput.*, 13(6):1361–1376, 1992.

[68] R. van Os. *Spectral Element Methods for Predicting the Flow of Polymer Solutions and Melts*. PhD thesis, The University of Wales, Aberystwyth, September 2004.

[69] R. G. M. van Os and T. N. Phillips. The Prediction of Complex Flows of Polymer Melts using Spectral Elements. *J. Non-Newtonian Fluid Mech.*, 122:287–301, 2004.

[70] X. Wang and W. K. Liu. Extended Immersed Boundary Method using FEM and RKPM. *Comput. Methods Appl. Mech. Engrg.*, 193:1305–1321, 2004.

[71] X. S. Wang, L. T. Zhang, and W. K. Liu. On Computational Issues of Immersed Finite Element Methods. *J. Comp. Phys.*, 228:2535–2551, 2009.

[72] N. D. Waters and M. J. King. Unsteady Flow of an Elastico-Viscous Liquid. *Rheol. Acta*, 9:345–355, 1970.

[73] L. Zhang, A. Gerstenberger, X. Wang, and W. K. Liu. Immersed Finite Element Method. *Comput. Methods Appl. Mech. Engrg.*, 193:2051–2067, 2004.