RHEOLOGY AND MAGNETIZATION OF DILUTE FERROFLUID EMULSIONS

VICTOR GUINANCIO E ABICALIL

DISSERTAÇÃO DE MESTRADO EM CIÊNCIAS MECÂNICAS
DEPARTAMENTO DE ENGENHARIA MECÂNICA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA MECÂNICA

RHEOLOGY AND MAGNETIZATION OF DILUTE FERROFLUID EMULSIONS

REOLOGIA E MAGNETIZAÇÃO DE EMULSÕES DILUÍDAS DE FERROFLUIDO

VICTOR GUINANCIO E ABICALIL

ORIENTADOR: TAYGOARA FELAMINGO DE OLIVEIRA

DISSERTAÇÃO DE MESTRADO EM CIÊNCIAS MECÂNICAS

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA MECÂNICA

RHEOLOGY AND MAGNETIZATION OF DILUTE FERROFLUID EMULSIONS

VICTOR GUINANCIO E ABICALIL

DISSERTAÇÃO DE MESTRADO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA MECÂNICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE.

APROVADA POR:

_____

Prof. Dr. Taygoara Felamingo de Oliveira – ENM/Universidade de Brasília
Orientador

_____

Prof. Dr. Francisco Ricardo da Cunha – ENM/Universidade de Brasília
Membro Interno

_____

Prof. Dr. Márcio da Silveira Carvalho – DEM/Pontífica Universidade Católica do Rio de Janeiro
Membro Externo

BRASÍLIA, 2 DE AGOSTO DE 2020.

ii

_____

Victor Guinancio e Abicalil

Departamento de Engenharia Mecânica (ENM) - FT
Universidade de Brasília (UnB)
Campus Darcy Ribeiro
CEP 70919-970 - Brasília - DF - Brasil

ACKNOWLEDGMENTS

RESUMO

Título: Reologia e Magnetização de Emulsões Diluídas de Ferrofluido
Autor: Victor Guinancio e Abicalil
Orientador: Taygoara Felamingo de Oliveira
Programa de Pós-Graduação em Ciências Mecânicas
Brasília, 2 de agosto de 2020

Este trabalho apresenta um estudo acerca da reologia e magnetização de emulsões diluídas de ferrofluido, quando submetidas a escoamentos cisalhantes simples e oscilatórios e à ação de campos magnéticos externos. Para tal, são realizadas simulações numéricas de uma gota de ferrofluido imersa em um fluido de fase contínua, assumindo que ambos os fluidos são Newtonianos, imiscíveis e de mesma viscosidade, que o ferrofluido é superparamagnético, e que o problema magnético pode ser tratado como magnetostático. As simulações foram realizadas com a utilização do código computacional FENRir, desenvolvido para este trabalho, em que o escoamento dos fluidos é calculado com base em um discretização em diferenças finitas de segunda ordem das equações de Navier-Stokes incompressíveis, com um método de projeção de segunda ordem para a discretização temporal e o método de Level Set para captura da posição da interface entre os dois fluidos. De modo a permitir a solução eficiente dos sistemas lineares associados à resolução do problema, são utilizadas rotinas de solução direta por análise de Fourier e o método dos Gradientes Conjugados com um precondicionador Multigrid. São apresentados resultados tanto para gotas não-magnéticas quanto gotas de ferrofluido, para os casos de escoamentos cisalhantes simples e oscilatórios, e sujeitas a campos magnéticos externos de diferentes direções e intensidades. Para o caso de escoamentos cisalhantes simples, são apresentados dados de morfologia da emulsão, representados pela deformação e ângulo de inclinação da gota, dados de reologia, representados pela viscosidade reduzida e primeira e segunda diferenças normais de tensão, e dados de magnetização, representados por sua magnitude e direção. Apesar de a gota ser considerada superparamagnética, as deformações induzidas pelo escoamento cisalhante fazem com que sua magnetização não seja perfeitamente alinhada ao campo magnético externo, induzindo um torque magnético no sistema, o qual é neutralizado por um torque hidrodinâmico oposto de igual magnitude, garantindo a conservação de momento angular em regimes de escoamento de Stokes. Para o caso de escoamentos cisalhantes oscilatórios, são apresentados valores para os módulos de perda e armazenamento, correspondentes aos comportamentos viscosos e elásticos da emulsão, respectivamente.

Palavras-chave: Reologia, Magnetização, Emulsão de Ferrofluido, Análise Numérica.

ABSTRACT

Title: Rheology and Magnetization of Dilute Ferrofluid Emulsions
Author: Victor Guinancio e Abicalil
Supervisor: Taygoara Felamingo de Oliveira
Graduate Program in Mechanical Sciences
Brasília, August 2nd, 2020

This works presents an investigation regarding the rheology and magnetization of dilute ferrofluid emulsions, when subjected to simple and oscillatory shear flows and to external magnetic fields. To this end, numerical simulations of a ferrofluid droplet immersed in a continuous fluid phase are performed, with both fluids assumed to be Newtonian, immiscible and of the same viscosity, the ferrofluid assumed to be superparamagnetic and the magnetic problem assumed to be magnetostatic. Simulations were performed using the computational code FENRir, developed for this work, in which the fluid flow problem is solved by using a second-order, finite difference discretization of the incompressible Navier-Stokes equations, alongside a second-order projection method for temporal discretization and the Level Set method to capture the position of the interface between the two fluids. To allow for an efficient solution of the linear systems of equations associated to the solutions of the governing equations, the computational code uses direct solution via Fourier analysis and the Conjugate Gradient method with Multigrid preconditioning. Results are presented both for non-magnetic and ferrofluid droplets, subjected to simple shear and oscillatory shear flows and to external magnetic fields of varying intensities and directions. In the case of simple shear flows, results are presented for the emulsion's morphology, represented by the droplet's deformation and inclination angle, for the emulsion's rheology, represented by the reduced viscosity and the first and second normal stress differences, and for it's magnetization, represented by it's magnitude and direction. Despite the superparamagnetic assumption for the ferrofluid phase, shear-induced droplet deformations cause the emulsion's magnetization to not perfectly align to the external magnetic field, giving rise to a magnetic torque in the system, which is then counteracted by an equal and opposite hydrodynamic torque, ensuring the conservation of angular momentum in the Stokes flow regime. For oscillatory shear flows, results are presented for the storage and loss moduli, corresponding to the elastic and viscous behaviours of the emulsion, respectively.

Keywords: Rheology, Magnetization, Ferrofluid Emulsion, Numerical Analysis.

SUMMARY

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 CONTEXTUALIZATION

Emulsions are suspensions of one or more immiscible liquids, consisting of a matrix and a dispersed phase, with microscopic droplets of the dispersed phase immersed in the continuous matrix phase. Given the microscopic size of the droplets, buoyancy effects are minimal, so that these suspensions take a very long time to settle, or do not settle at all. Additionally, since the length scales associated to the flow around the droplets are much smaller than the length scales associated with flows in practical applications, emulsions can usually be analyzed as a single fluid in a macroscopic scale, although one with complex rheological behaviour, originating from the microscopic multi-phase flows and the effects of surface tension.

Emulsions are present in a wide variety of applications, such as in the medical, food, and cosmetic industries, and can be dilute, in which interactions between droplets are negligible, or concentrated, in which interactions between droplets become significant. They can also be characterized as monodisperse, in which all dispersed droplets have the same size, or polydisperse, in which droplets have several different sizes.

Due to the effects of surface tension and droplet deformation, emulsions display a complex, non-Newtonian behaviour. When subjected to simple shear flows, the droplets tend to deform and align themselves with the flow, an effect that becomes more pronounced with higher shear rates, and which causes a reduction in the viscosity of the emulsion. Thus, dilute emulsions usually display a shear-thinning behaviour. When subjected to unsteady shear flows, emulsions display a viscoelastic behaviour. In this case, as surface tension forces attempt to return a deformed droplet to a spherical shape, they introduce a stress component which is dependent on strain, and thus an additional elastic behaviour. The relative magnitude of the viscous and elastic effects depends on the frequency associated with the flow and the relaxation time of the droplets, and, as such, emulsions can display widely different behaviours depending on the characteristics of the flows they are subjected to.

Ferrofluids, meanwhile, are colloidal suspensions of ferromagnetic particles. Due to the presence of such magnetic particles, the characteristics of a ferrofluid can be altered by manipulating the magnetic fields which permeate them. This allows for an active control of the fluid's behaviour, and has given rise to several different possible applications, such as in microfluidics, in the aerospace industry, where it could be used for spacecraft propulsion, for example, and in the medical industry, where it could be used for targeted drug delivery or for cancer treatments via hyperthermia.

Recently, several works have focused on studying the behaviour of ferrofluid droplets immersed in another liquid and subjected to a shear flow, in order to better understand the behaviour of such droplets and how they affect the overall fluid flow. In the case of microscopic droplets, this characterizes a dilute, monodisperse ferrofluid emulsion, i.e. an emulsion in which the dispersed phase is a ferrofluid. Such an emulsion, then, will have a very complex behaviour, by combining the non-Newtonian characteristics of general emulsions with the magnetic response and active control associated with ferrofluids.

In this context, the goal for this work is to study the rheological and magnetic behaviour of dilute, monodisperse ferrofluid emulsions when subjected to an external magnetic field, both under simple shear and oscillatory shear flows. This is accomplished by performing numerical simulations of such emulsions and analyzing their response to the shear flows and magnetic fields, in order to better understand the morphology, rheology and magnetization of such emulsions and, ultimately, to determine the feasibility of controlling the rheology and overall behaviour of such emulsions with the use of external magnetic fields.

1.2   BIBLIOGRAPHICAL REVIEW

The rheology of colloids has been a subject of active research for over a century, dating back to the seminal work of Einstein [6] in 1906, which presented a theoretical prediction for the viscosity of a suspension of solid spheres. This work was later expanded upon by Taylor [2], in 1932, in which he presented a prediction for the viscosity of a suspension of liquid spheres. The dynamics of a suspended droplet were further analyzed in [1], in which Taylor presented experimental results on the deformation of droplets under simple shear and extensional flows, alongside a theoretical model for droplet deformation in the limit of small deformations. Since these seminal works of Taylor, a wide range of works have since been published regarding the dynamics of suspended droplets, including theoretical, experimental, and numerical results.

Some other early experimental works, investigating the behaviour of dilute emulsions under simple shear flows, include those of Rumscheidt and Mason [7], Torza, Cox and Mason [8], and Bentley and Leal [9], among others. More recent works include those of Sibillo et al. [10] and Vananroye et al. [11, 12, 13], which investigated the influence of confinement on the dynamics of droplets undergoing simple shear flows (a review regarding the effects of confinement is presented in [14]), as well as those of Wannaborworn, Mackley and Renardy [15], Cavallo, Guido and Simeone [16], and Janpaen et al. [17], which investigated droplet deformation and breakup under oscillatory shear flows. Theoretical works, focusing on the development of small deformation theories, include those of Schowalter, Chaffey and Brenner [18], Barthes-Biesel and Acrivos [19], Grmela, Bousmina and Palierne [20], and

Yu et al. [21], for simple shear flows, and those of Frankel and Acrivos [22], Palierne [23] and Yu et al. [24] for oscillatory shear flows. A summary of small deformation theories is presented by Rallison [25], and recent reviews regarding the rheology of emulsions are presented by Derkach [26] and Pal [27].

Regarding the numerical simulation of dilute emulsions, several different methods have been employed. One of the most common is the Boundary Integral method, since it allows for a three-dimensional problem to be represented on a two-dimensional surface, and was used, among others, by Rallison [28] and Kennedy, Pozrikidis and Skalak [29]. It is, however, unable to solve the complete Navier-Stokes equations, being limited to the case of creeping flow, and requires complex mesh recreation algorithms to be able to deal with problems of droplet breakup. Extensions to the Boundary Integral method were also presented by Janssen and Anderson [30], to allow for simulations of confined droplets, and by Cunha and Couto [31], to allow for simulations of magnetic droplets. Other works have also been based on the Volume of Fluid method, such as that of Li, Renardy and Renardy [32], the Front-Tracking method, such as that of Li and Sarkar [33], and the Lattice-Boltzmann method, such as those of Xi and Duncan [34] and Ioannou, Liu and Zhang [35]. Gounley et al. [36] used a numerical method coupling the Boundary Element method with the Finite Element method to investigate the influence of interfacial viscosity on droplet dynamics.

In the case of ferrofluids, a general review regarding possible applications currently under research is presented by Torrez-Diaz and Rinaldi [37]. Since the pioneering works of Sherwood [38] and Sero-Guillaume et al. [39], several studies have investigated the behaviour of ferrofluid droplets in quiescent flows when subjected to external magnetic fields, with noteworthy works including those of Afkhami et al. [40], which presented a theoretical prediction for droplet deformation alongside matching numerical simulations, and of Rowghanian, Meinhart and Campas [41], which analyzed numerical results for droplet shape when compared to an ellipsoid.

The dynamics of a ferrofluid droplet when subjected to both an external magnetic field and a shear flow, however, has only recently been investigated in the literature. Jesus, Roma and Ceniceros [4] presented a small deformation theory alongside three-dimensional, highly accurate numerical results based on the Immersed Boundary method, for external magnetic fields parallel to the main velocity gradient direction. However, the numerical results focused on cases of small droplet deformation, with only few cases of moderate deformation presented. Capobianchi, Lappa and Oliveira [42] presented three-dimensional results based on a hybrid Level Set - Volume of Fluid method, for external magnetic fields parallel to the main velocity gradient direction, with results focusing on the transient relaxation behaviour of the droplets, in addition to it's steady-state deformation and inclination angle.

Meanwhile, Hassan, Zhang and Wang [43] presented two-dimensional simulations of a ferrofluid droplet, based on the Level Set method with a finite element discretization, and

external magnetic fields in arbitrary directions. Although limited to two dimensions, the results included cases of large droplet deformations, and spanned a wide range of magnetic field orientations. The aforementioned results indicated that the droplets tend to align and deform alongside the external magnetic field direction, with the amount of influence the shear flow has on droplet shape being dependent on both the flow and magnetic field conditions. The droplet deformation is largest for external magnetic fields in a $45°$ angle between the main flow direction and the main velocity gradient direction, with magnetic fields parallel to the main velocity gradient direction leading to larger deformations than magnetic fields parallel to the main flow direction.

The aforementioned works, however, focused solely on the droplet's shape. The work of Cunha et al. [44], meanwhile, investigated the influence of an external magnetic field on the emulsion's rheology and on droplet breakup, using two-dimensional numerical simulations based on the Level Set method. It found that external magnetic fields have a major influence on the emulsion's reduced viscosity, with fields parallel to the main flow direction leading to significant decreases in viscosity, and fields parallel to the main velocity gradient direction leading to significant increases in viscosity. Moreover, it also found that external magnetic fields can be used to control the droplet's breakup process, either promoting or preventing it, depending on the magnetic field direction and intensity, as well as influencing the time to breakup and the size and number of satellite droplets. This analysis of droplet breakup was later extended by Hassan and Wang [45] to include a wider range of viscosity ratios and magnetic field directions and intensities, although still in the two-dimensional case.

In a later work, Cunha et al. [46] presented a new formulation for the droplet's particle stress tensor, which accurately captures the magnetic forces acting on the droplet interface. Based on this new formulation, the authors were able to further investigate the rheology of the ferrofluid emulsion, presenting results for the first normal stress difference in addition to the reduced viscosity. The authors also investigated the emulsion's magnetization, and found that a hydrodynamic torque is present in the system, counteracting a magnetic torque that arises from the emulsion's magnetization not perfectly aligning to the external magnetic field.

This investigation regarding a ferrofluid emulsion's rheology was then extended to the three-dimensional case by Ishida and Matsunaga [3], which used numerical simulations based on the Lattice-Boltzmann method to present results for the emulsion's reduced viscosity as well as it's first and second normal stress differences, with external magnetic fields in either of the three main orthogonal directions and for varying shear flow and magnetic field intensities. Abicalil et al. [5] later extended the investigation regarding the emulsion's magnetization to the three-dimensional case, as well as to wider range of external magnetic field intensities, with results based on numerical simulations using the Level Set method. The authors observed qualitatively similar results to those of the two-dimensional case [46] for

external magnetic fields parallel to the main flow and main velocity gradient directions, while for external magnetic fields parallel to the main vorticity direction the emulsion's magnetization perfectly aligns to the external magnetic field, so that no magnetic or hydrodynamic torques arise in the system.

To the best of the author's knowledge, there hasn't yet been any work published regarding the rheology of a ferrofluid emulsion subjected to an oscillatory shear flow. For ferrofluid emulsions subjected to simple shear flows, the author is currently not aware of any works presenting experimental results, or works investigating droplet breakup from a three-dimensional perspective.

1.3 PROBLEM FORMULATION

1.3.1 Studied Flows

In this work, the rheological behaviour of ferrofluid emulsions is studied from a microscopic point of view, through numerical simulations of shear flows, both steady and oscillatory, with one droplet of the dispersed phase, assumed to be superparamagnetic, immersed in the matrix fluid, assumed to be non-magnetic. Figure 1.1 presents a schematic of the computational domain used for these simulations, with dimensions $L_x \times L_y \times L_z$ and one initially spherical droplet of radius a placed in its center. This system can then be subjected to an externally applied, uniform magnetic field $\mathbf{H_0}$, parallel to either one of the Cartesian axes. As the flow develops, the droplet is deformed and, thus, can be characterized based on its length L, breadth B, and inclination angle θ[1]. The droplet deformation can then be measured using Taylor's deformation parameter $D = (L - B)/(L + B)$.

The boundaries normal to the y axis are walls moving with velocities $U/2$ and $-U/2$ for the upper and lower ones, respectively. These walls have Dirichlet boundary conditions for velocity, corresponding to the impenetrability and no-slip constraints, while all other boundary conditions are periodic.

For oscillatory shear flows, the wall velocities follow a sinusoidal function given by $U = U_0 \sin(\omega t)$, where U_0 is the peak relative velocity between the walls, ω is the angular velocity of oscillation and t is the elapsed time. For steady shear flows the wall velocities are constant ($U = U_0$). The peak average shear rate is then defined as $\dot{\gamma}_0 = U_0/L_Y$. The viscosities of the fluids are η_m and η_d, for the matrix and dispersed phases, respectively, and both phases have an equal density ρ. The magnetic permeabilities of the fluids are μ_0 for the matrix phase, assumed to be equal to that of free space, and μ_d for the dispersed phase. The interface between the two fluids is assumed to be clean, so that it has a constant surface

---

[1]L, B and θ are measured in the plane normal to the z axis that crosses the center of the droplet.
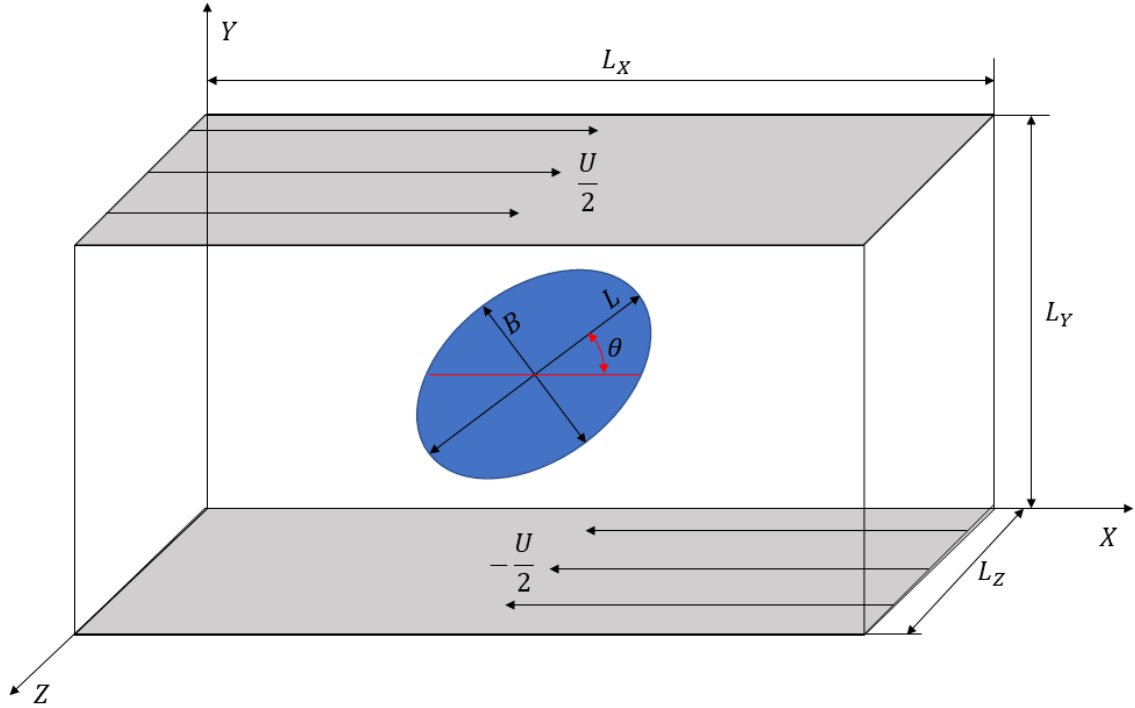
Figure 1.1 – Schematic of a droplet immersed in a simple shear flow.

tension coefficient of σ.

The main nondimensional parameters that govern this flow are:

- The Reynolds number, Re, corresponding to the ratio between inertial and viscous effects in the flow;

- The capillary number, Ca, corresponding to the ratio between viscous and surface tension forces in the flow;

- The magnetic capillary number, Ca_{mag}, corresponding to the ratio between magnetic and surface tension forces in the flow;

- The viscosity ratio between the two fluids, λ; and

- The magnetic permeability ratio between the two fluids, ζ.

The Reynolds number can be defined as either $Re_m = \rho\dot{\gamma}_0 a^2/\eta_m$, based on the matrix phase viscosity, or $Re_d = \rho\dot{\gamma}_0 a^2/\eta_d$, based on the dispersed phase viscosity. The capillary number can be defined as either $Ca_m = \eta_m a\dot{\gamma}_0/\sigma$, based on the matrix phase viscosity, or $Ca_d = \eta_d a\dot{\gamma}_0/\sigma$, based on the dispersed phase viscosity. The magnetic capillary number, viscosity ratio, and permeability ratio are defined as $Ca_{mag} = \mu_0 a||\mathbf{H_0}||^2/\sigma$, $\lambda = \eta_d/\eta_m$ and $\zeta = \mu_d/\mu_0$, respectively.

It is worth noting that such a flow has three main characteristic times: the droplet's relaxation time, related to surface tension forces ($\tau_\sigma = \eta a/\sigma$), the bulk flow characteristic time, related to the shear rate ($\tau_u = 1/\dot{\gamma}_0$), and the magnetic characteristic time, related to magnetic forces ($\tau_{mag} = \eta/\mu_0 ||\mathbf{H_0}||^2$). In this way, the capillary number measures the ratio between the droplet's relaxation time and the bulk flow characteristic time, with $Ca = \tau_\sigma/\tau_u$, while the magnetic capillary number measures the ratio between the droplet's relaxation time and the magnetic characteristic time, with $Ca_{mag} = \tau_\sigma/\tau_{mag}$. The capillary number is also the ratio between stresses nondimensionalized using τ_σ and τ_u, respectively, with $Ca = \sigma_\sigma/\sigma_u$.

1.3.2 Magnetic problem

The magnetic problem is assumed to be magnetostatic, such that relaxation times for the magnetic field are assumed to be negligible when compared to the time scales associated with fluid flow and droplet motion. In this case, and considering the absence of electric fields and currents, Maxwell's equations reduce to $\nabla \cdot \mathbf{B} = 0$ and $\nabla \times \mathbf{H} = 0$, where $\mathbf{B}$ is the magnetic induction and $\mathbf{H}$ is the magnetic field. Assuming that the ferrofluid droplet is superparamagnetic, with constant magnetic susceptibility χ, it's magnetization $\mathbf{M}$ follows the linear relation $\mathbf{M} = \chi\mathbf{H}$ and, since the magnetic induction is composed of contributions from the magnetic field and the local magnetization, it can be defined as $\mathbf{B} = \mu_0(\mathbf{M} + \mathbf{H})$.

To generalize this definition to the entire domain, the permeability ratio is extended to the entire domain as $\zeta(\mathbf{x})$, where $\mathbf{x}$ is the position vector of a given point, defined as $\zeta(\mathbf{x}) = 1$ in the non-magnetic matrix fluid, and $\zeta(\mathbf{x}) = \zeta = 1 + \chi$ in the ferrofluid droplet. Using this definition, the magnetic induction is defined as $\mathbf{B} = \mu_0\zeta(\mathbf{x})\mathbf{H}$ and, since the magnetic field is irrotational, it can be defined as the gradient of a potential field ψ, such that $\mathbf{H} = \nabla\psi$. Given that the magnetic induction $\mathbf{B}$ is solenoidal, the magnetic potential can be obtained from the relation [44]

$$\nabla \cdot (\mu_0\zeta(\mathbf{x})\nabla\psi) = 0\,, \tag{1.1}$$

with boundary conditions representing the external magnetic field. These correspond to imposing Neumann boundary conditions on all domain boundaries, defined as $\nabla\psi \cdot \hat{\mathbf{n}} = \mathbf{H}_0 \cdot \hat{\mathbf{n}}$, with $\hat{\mathbf{n}}$ the unit vector normal to the boundary.

1.3.3 Hydrodynamic problem

The flows studied in this work are all assumed to be incompressible, and so are governed by the incompressible Navier-Stokes equations [44]

$$\rho\left(\frac{\partial\mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla\mathbf{u}\right) = -\nabla P + \nabla \cdot \left[\eta(\mathbf{x})(\nabla\mathbf{u} + \nabla\mathbf{u}^T)\right] + \mathbf{F_s} + \mathbf{F_{mag}}\,, \tag{1.2}$$

$$\nabla \cdot \mathbf{u} = 0 \,, \tag{1.3}$$

with the addition of the body force terms $\mathbf{F_s}$ and $\mathbf{F_{mag}}$, which account for the surface tension in the droplet surface and for the magnetic forces, respectively. In the above equations, $\mathbf{u}$ is the velocity field, P is the pressure field, and $\eta(\mathbf{x})$ is the viscosity, and, since both fluids are assumed to have equal densities, gravitational effects are not considered. The surface tension force term, derived from the Young-Laplace equation, is $\mathbf{F}_s = -\sigma\kappa\hat{\delta}(\mathbf{x} - \mathbf{x}_\Gamma)\hat{\mathbf{n}}$, where κ is the local mean curvature of the interface, $\hat{\mathbf{n}}$ is the unit vector normal to the interface, $\hat{\delta}$ is the Dirac delta function, and $\mathbf{x}_\Gamma$ is the position of the interface, and the magnetic force term is $\mathbf{F}_m = \mu_0(\zeta(\mathbf{x}) - 1)\mathbf{H} \cdot \nabla\mathbf{H}$.

Defining the nondimensional variables $\tilde{\mathbf{u}} = \mathbf{u}/\dot{\gamma}a$, $\tilde{t} = t\dot{\gamma}$, $\tilde{\mathbf{x}} = \mathbf{x}/a$, $\tilde{P} = P/\rho a^2\dot{\gamma}^2$, $\tilde{\eta}(\mathbf{x}) = \eta(\mathbf{x})/\eta_m$, and $\tilde{\mathbf{H}} = \mathbf{H}/|\mathbf{H}_0|$, and dropping the tilde in nondimensional variables to alleviate the notation, the Navier-Stokes equations can be rewritten in nondimensional form as

$$\begin{aligned}
\frac{\partial\mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla\mathbf{u} = &-\nabla P + \frac{1}{Re_m}\nabla \cdot \left[\eta(\mathbf{x})(\nabla\mathbf{u} + \nabla\mathbf{u}^T)\right] \\
&-\frac{1}{Re_mCa_m}\kappa\hat{\delta}(\mathbf{x} - \mathbf{x}_\Gamma)\hat{\mathbf{n}} + \frac{Ca_{mag}}{Re_mCa_m}(\zeta(\mathbf{x}) - 1)\mathbf{H} \cdot \nabla\mathbf{H} \,,
\end{aligned} \tag{1.4}$$

$$\nabla \cdot \mathbf{u} = 0 \,. \tag{1.5}$$

When using the droplet viscosity η_d instead of the matrix fluid viscosity η_m as a characteristic viscosity, the only difference in Eq. 1.4 is that Re_m and Ca_m are replaced by Re_d and Ca_d, respectively. This alternative formulation is extremely useful when performing numerical simulations of emulsions with a high viscosity ratio, since it avoids large jumps in nondimensional viscosity across the droplet interface that could otherwise lead to numerical instabilities.

2 DISCRETIZATION OF THE GOVERNING EQUATIONS

2.1 PROJECTION METHODS

2.1.1 General Description

When solving the incompressible Navier-Stokes equations, one of the main difficulties arises from the accurate calculation of the pressure term, since there isn't a thermodynamic equation of state for it such as in the case of compressible flows, as well as from the requirement of imposing the incompressibility constraint. One alternative would be to solve the incompressible Navier-Stokes equations as a coupled system [47], but this would incur in significant computational costs. Another alternative, although limited to some two-dimensional flows, is to use the vorticity-streamfunction formulation [48].

One of the most popular alternatives, however, is the use of so-called projection methods, which are used in this work. These methods, initially proposed by Chorin [49], consist, essentially, in splitting the solution of the Navier-Stokes equations, first by using an absent or approximate pressure term to calculate a tentative velocity field u^*, which is not divergence-free, then by calculating a pressure field which would render this tentative velocity field incompressible, and, finally, updating the tentative velocity field to obtain the final velocity field. As a basic example, consider the Navier-Stokes equations for a single Newtonian fluid and, for the sake of simplicity, approximate the time derivative by an explicit Euler scheme:

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = -\mathbf{u}^n \cdot \nabla \mathbf{u}^n - \nabla P^n + \frac{1}{Re} \nabla^2 \mathbf{u}^n. \tag{2.1}$$

Splitting the solution, and ignoring the pressure term when calculating the tentative velocity field, the scheme becomes

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -\mathbf{u}^n \cdot \nabla \mathbf{u}^n + \frac{1}{Re} \nabla^2 \mathbf{u}^n, \tag{2.2}$$

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^*}{\Delta t} = -\nabla P^n, \tag{2.3}$$

where it is clear that the sum of Eqs. 2.2 and 2.3 is equal to Eq. 2.1. Finally, by taking the divergence of Eq. 2.3 and imposing the incompressibility constraint, an equation for

Pressure can then be obtained as

$$\nabla^2 P^n = \frac{1}{\Delta t} \nabla \cdot \mathbf{u}^* \,, \tag{2.4}$$

aptly named the Pressure Poisson Equation. Note that this formulation imposes the incompressibility constraint on every time step, ensuring that the flow doesn't violate this constraint due to an accumulation of numeric errors across several steps. It is also worth noting that an approximate pressure term $\tilde{P}$, usually extrapolated from previous time steps, can be included in Eq. 2.2. With such a term, the resulting projection method would be defined as

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -\mathbf{u}^n \cdot \nabla \mathbf{u}^n - \nabla \tilde{P} \frac{1}{Re} \nabla^2 \mathbf{u}^n \,, \tag{2.5}$$

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^*}{\Delta t} = -\nabla P^n + \nabla \tilde{P} \,, \tag{2.6}$$

$$\nabla^2 P^n = \frac{1}{\Delta t} \nabla \cdot \mathbf{u}^* + \nabla^2 \tilde{P} \,. \tag{2.7}$$

Due to the presence of the approximate pressure term, such methods are referred to as incremental pressure projection methods. Note that, in this case, $\mathbf{u}^*$ is a close approximation of $\mathbf{u}^{n+1}$, whereas the $\mathbf{u}^*$ defined in Eq. 2.1 is not. This difference in $\mathbf{u}^*$ means that care must be taken when imposing boundary conditions, in order to ensure that such conditions are sufficiently accurate for the tentative velocity field, but it does not decrease the overall accuracy of the method. In the case of incremental pressure projection methods, $\mathbf{u}^*$ is such an accurate approximation of the true velocity field, in fact, that the errors associated with it are similar to those associated with $\mathbf{u}^{n+1}$ [50], with $\mathbf{u}^*$ not being divergence-free and $\mathbf{u}^{n+1}$ having inexact boundary conditions.

Finally, it is essential to note that the formulations previously described are not complete, since they lack the required boundary conditions for Eq. 2.4. Additionally, such a boundary condition is not present in the physical description of the problem, since it only involves the no-slip velocity boundary conditions, and an inexact boundary condition can lead to the creation of numeric boundary layers which reduce the accuracy of the solution method. Then, what boundary conditions can one use to solve the Pressure Poisson Equation? This is one of the major issues of projection methods, and has been extensively discussed in the literature and led to several different methods. Excellent reviews of different projection methods, as well as their advantages and disadvantages, are presented in [51] and [50].

2.1.2 The Pojection Method of Kim and Moin

In this work, the Navier-Stokes equations are solved using the projection method proposed by Kim and Moin in [52]. This method, expected to be second order accurate in time, is based on a Crank-Nicolson scheme for the time discretization, with implicit treatment of the viscous term, and uses an auxiliary, pressure-like scalar field $\mathfrak{P}$, instead of the true pressure P. Note that this method can also be used with other time-stepping schemes. This scheme, assuming constant viscosity, can be defined as

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -(\mathbf{u} \cdot \nabla \mathbf{u})^{n+\frac{1}{2}} + \frac{1}{2Re}(\nabla^2 \mathbf{u}^* + \nabla^2 \mathbf{u}^n) + \mathbf{F}^{n+\frac{1}{2}} \,, \tag{2.8}$$

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^*}{\Delta t} = -\nabla \mathfrak{P}^{n+1} \,, \tag{2.9}$$

$$\nabla^2 \mathfrak{P}^{n+1} = \frac{1}{\Delta t} \nabla \cdot \mathbf{u}^* \,, \tag{2.10}$$

where $\mathbf{F}$ is a generic forcing term, extrapolated from previous iterations as $\mathbf{F}^{n+\frac{1}{2}} = \frac{3}{2}\mathbf{F}^n - \frac{1}{2}\mathbf{F}^{n-1}$, with the nonlinear advective term extrapolated in a similar way. As such, the current method requires the solution of four elliptic partial differential equations on each time step, one Helmholtz equation for each component of $\mathbf{u}$ and one Poisson equation for $\mathfrak{P}$. Since $\mathfrak{P}$ is an auxiliary variable, the exact instant in which it is defined is arbitrary, and does not alter the resulting computational code. Thus, although defining it at the instant $n + \frac{1}{2}$ would be more consistent with the Crank-Nicolson scheme, in this work it is defined at the instant $n + 1$ solely for ease of notation. The actual pressure P, if required, can be calculated by the relation

$$P^{n+\frac{1}{2}} = \mathfrak{P}^{n+1} + \frac{\Delta t}{2Re} \nabla^2 \mathfrak{P}^{n+1} \,, \tag{2.11}$$

obtained by taking the sum of Eqs. 2.8 and 2.9, substituting $\mathbf{u}^* = \Delta t \nabla \mathfrak{P}^{n+1} + \mathbf{u}^{n+1}$, commuting the gradient and laplacian operators, and comparing the result to a similar Crank-Nicolson discretization of the Navier-Stokes equations, without a projection method

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = -\nabla P^{n+\frac{1}{2}} - (\mathbf{u} \cdot \nabla \mathbf{u})^{n+\frac{1}{2}} + \frac{1}{2Re}(\nabla^2 \mathbf{u}^{n+1} + \nabla^2 \mathbf{u}^n) + \mathbf{F}^{n+\frac{1}{2}} \,. \tag{2.12}$$

Note that $P^{n+\frac{1}{2}}$, as defined in Eq. 2.11, is a first order in time approximation of P^{n+1}. In order to solve Eqs. 2.8 and 2.10, boundary conditions for $\mathbf{u}^*$ and $\mathfrak{P}$ are required. Accurate boundary conditions for $\mathbf{u}^*$ can be obtained by using Eq. 2.9 and the corresponding boundary conditions for $\mathbf{u}$, using the extrapolation $\mathfrak{P}^{n+1} = 2\mathfrak{P}^n - \mathfrak{P}^{n-1}$. For the boundary conditions for $\mathfrak{P}$, it is evident from Eq. 2.11 that they do not enforce a boundary condition for P, and it can be seen in the boundary conditions for $\mathbf{u}^*$ that the boundary conditions for $\mathfrak{P}$ also do not influence the boundary conditions of $\mathbf{u}$. As such, boundary conditions for $\mathfrak{P}$

are arbitrary, and thus can be set as the homogeneous Neumann condition of $(\nabla \mathfrak{P}) \cdot \hat{n} = 0$, where $\hat{n}$ is a unit vector normal to the boundary, for convenience and to simplify the boundary conditions for $\mathbf{u}^*$. It is worth noting that, although the above method should obtain second order accuracy for both $\mathbf{u}$ and P, numerical experiments presented in [51] for a very similar method displayed a deterioration to first order accuracy for P near the boundaries. However, since the current work doesn't require accurate values of P, especially near the boundaries, and this numerical boundary layer doesn't cause a deterioration of accuracy for $\mathbf{u}$, such a deterioration of accuracy, if present, would not be an issue.

In order to extend this method to the case of variable viscosity, as is the case when $\lambda \neq 1$, one can simply substitute the viscous term $\frac{1}{Re} \nabla^2 \mathbf{u}$ with $\frac{1}{Re} \nabla \cdot (\eta(\mathbf{x})(\nabla \mathbf{u} + \nabla \mathbf{u}^T))$, where $\mathbf{u}$ can represent $\mathbf{u}^n$, $\mathbf{u}^{n+1}$ or $\mathbf{u}^*$. This, however, would result in variable coefficients when the corresponding Helmholtz equations are discretized, which would require recalculating the coefficients on every time step, as well as making the equations harder to solve. One solution to this problem is to use the semi-implicit treatment of the viscous term proposed in [53], which consists in extrapolating the true viscous term while adding the term $\frac{1}{2Re}(\nabla^2 \mathbf{u}^* - \nabla^2 \tilde{\mathbf{u}}^{n+1})$ to Eq. 2.8, with $\tilde{\mathbf{u}}^{n+1}$ extrapolated from previous values as $\tilde{\mathbf{u}}^{n+1} = 2\mathbf{u}^n - \mathbf{u}^{n-1}$. The resulting method thus consists in substituting Eq. 2.8 for

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -(\mathbf{u} \cdot \nabla \mathbf{u})^{n+\frac{1}{2}} + \frac{1}{Re} \nabla \cdot (\eta(\mathbf{x})(\nabla \mathbf{u} + \nabla \mathbf{u}^T))^{n+\frac{1}{2}} + \frac{1}{2Re}(\nabla^2 \mathbf{u}^* - \nabla^2 \tilde{\mathbf{u}}^{n+1}) + \mathbf{F}^{n+\frac{1}{2}},$$
(2.13)

while the rest of the method remains unchanged. The resulting method will still have some stability requirements associated with the viscous forces, since the viscous term is not treated in a fully implicitly fashion, but these requirements are significantly less strict than those associated with an explicit treatment of the viscous term. In order to better understand this, consider the following split of the viscous term:

$$\nabla \cdot (\eta(\mathbf{x})(\nabla \mathbf{u} + \nabla \mathbf{u}^T)) = \eta(\mathbf{x}) \nabla \cdot (\nabla \mathbf{u} + \nabla \mathbf{u}^T) + (\nabla \eta(\mathbf{x})) \cdot (\nabla \mathbf{u} + \nabla \mathbf{u}^T).$$ (2.14)

Since $\nabla \cdot \mathbf{u} = 0$, we have that $\nabla \cdot (\nabla \mathbf{u}^T) = 0$, such that the viscous term can be rewritten as

$$\nabla \cdot (\eta(\mathbf{x})(\nabla \mathbf{u} + \nabla \mathbf{u}^T)) = \eta(\mathbf{x}) \nabla^2 \mathbf{u} + (\nabla \eta(\mathbf{x})) \cdot (\nabla \mathbf{u} + \nabla \mathbf{u}^T),$$ (2.15)

where it can be seen that the aforementioned semi-implicit scheme treats the first component of the viscous term implicitly, which is the one associated with the strictest stability requirements. Thus, the resulting method will not be subject to the strict stability requirements associated with the first component, but will still be subjected to the less strict requirements associated with the second component. It is worth noting that, for flows of variable density, a similar semi-implicit scheme can also be applied [54, 55].

2.2 LEVEL-SET METHOD

2.2.1 General Description

In addition to solving the Navier-Stokes equations, the modelling of multiphase flows also requires the use of a method to accurately determine the position and motion of the interface between the two fluids, as well as the interface's geometric properties that determine the surface tension forces. Such methods can be split into two main categories: interface tracking methods, in which the position of the interface is determined explicitly based on a Lagrangian approach, using surface-fitted meshes, and interface capturing methods, in which the position of the interface is inferred based on some auxiliary variable, which is defined on the same Eulerian mesh used for the Navier-Stokes equations. Examples of interface tracking methods include the Boundary Integral method [56] and Immersed Boundary methods [57], while examples of interface capturing methods include Volume of Fluid methods [58], Phase-Field methods [59], and Level Set methods [60], each having distinct advantages and disadvantages. For example, it is quite complicated to deal with drastic geometry changes, such as droplet breakup and coalescence, when using interface tracking methods, since it essentially requires the creation of a new mesh, while such phenomena are dealt with automatically by interface capturing methods. Determining the interface's geometric properties is not trivial when using Volume of Fluid methods, but is so when using Level Set methods. Level Set methods, meanwhile, have the downside of requiring constant reinitialization, among other considerations. It is worth mentioning that it is also possible to combine the aforementioned methods, such as in the coupled Level Set and Volume of Fluid method presented in [61] and the hybrid Level Set and Immersed Boundary method presented in [62].

In this work, the Level Set method is used to capture the interface's position and motion. Initially proposed in [63], this method consists in creating a scalar function, referred to as the Level Set function and defined in the entire computational domain, and defining a surface as the set of points in which this function has a specific value. In this work, one of the most common alternatives is used, in which the Level Set function ϕ is initially defined as a signed distance function to the interface, with negative values inside the droplet and positive values outside it. Thus, the interface is then defined as the set of points in which the function has a value of zero. Furthermore, geometric properties of the interface can also be determined based on the Level Set function. Given that $\phi = 0$ throughout the interface, it is clear that the unit vector normal to it can be defined as $\hat{\mathbf{n}} = \nabla\phi/|\nabla\phi|$, since the components of $\nabla\phi$ tangential to the interface are all zero. The interface's curvature can then be defined as $\kappa = \nabla \cdot \hat{\mathbf{n}}$ [64] or, assuming a constant $|\nabla\phi|$, as $\kappa = \nabla^2\phi/|\nabla\phi|$. Note that, in this way, $\hat{\mathbf{n}}$ and κ can be defined across the entire domain, and not only strictly at the interface, which is essential for the Level Set implementation used in this work, as will be discussed in section 2.2.3.

With this definition of the Level Set function, the Heaviside function $H(\phi)$ can then be defined as

$$H(\phi) = \begin{cases} 0, & \text{if } \phi < 0, \\ \frac{1}{2}, & \text{if } \phi = 0, \\ 1, & \text{if } \phi > 0, \end{cases} \tag{2.16}$$

with a corresponding one-dimensional Dirac Delta function defined as $\delta(\phi) = \frac{dH(\phi)}{d\phi}$. The viscosity and permeability for a given point can then be determined by using the following relations [65]:

$$\eta(\phi) = \lambda + (1 - \lambda)H(\phi), \tag{2.17}$$

$$\zeta(\phi) = \zeta + (1 - \zeta)H(\phi). \tag{2.18}$$

In order to accurately determine the interface's motion, note that, locally, the interface can only move in the normal direction, and with the same velocity as the underlying fluid flow, given that no particles can cross the interface. The interface's motion is then governed by the Level Set equation [66]:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \hat{\mathbf{n}}|\nabla \phi| = 0, \tag{2.19}$$

or, alternatively [67]:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = 0. \tag{2.20}$$

Notice that Eq. 2.20 is the conservation equation $\frac{D\phi}{Dt} = 0$, ensuring that values of ϕ remain constant for a given fluid particle as it is transported by the flow. One consequence of this is that, unless the velocity fields are uniform or strictly tangential to the interface, the Level Set function ϕ will diverge from a signed distance function, as the trajectories of fluid particles outside the interface move them closer or further from it. In the continuous case this is negligible, but, for the discrete, practical implementations of the method, this becomes a major issue, for reasons that will be discussed in section 2.2.3.

2.2.2 Surface Tension Forcing Term

With the curvature of the interface κ and normal unit vector $\hat{\mathbf{n}}$ already defined in the previous section, the only term of the surface tension body force not yet fully defined in this Level Set implementation is the Dirac delta function, i.e. how to represent $\hat{\delta}(\mathbf{d})$, with $\mathbf{d} = \mathbf{x} - \mathbf{x}_\Gamma$, using the Level Set function ϕ. First, note that the delta function $\hat{\delta}$ is the

directional derivative of the Heaviside function H in the normal direction [64]:

$$\hat{\delta}(\mathbf{d}) = \nabla H(\phi(\mathbf{d})) \cdot \hat{\mathbf{n}} . \tag{2.21}$$

Then, by applying the chain rule, and recalling the definition of $\hat{\mathbf{n}}$ in the Level Set formulation:

$$\nabla H(\phi(\mathbf{d})) \cdot \hat{\mathbf{n}} = \frac{dH(\phi)}{d\phi}\nabla\phi \cdot \frac{\nabla\phi}{|\nabla\phi|} = \delta(\phi)|\nabla\phi| , \tag{2.22}$$

Thus, the surface tension term in Eq. 1.4 can be rewritten, using the Level Set function ϕ, as

$$\mathbf{F_s} = -\frac{1}{ReCa}\kappa\delta(\phi)|\nabla\phi|\hat{\mathbf{n}} , \tag{2.23}$$

or, recalling the definition of $\hat{\mathbf{n}}$, as [60]:

$$\mathbf{F_s} = -\frac{1}{ReCa}\kappa\delta(\phi)\nabla\phi \tag{2.24}$$

2.2.3 Interface Thickness

When considering the implementation of the methodology described previously, it is clear that it would not work adequately in a discrete setting. The sudden jumps in fluid properties would lead to numerical instabilities and, more importantly, the delta function would fail to correctly impose the stress jump due to surface tension where the interface does not exactly coincide with a mesh point. One solution to this issue, first presented in [60] and later updated in [65], is to smooth the interface, so that it has a finite thickness, larger than the spacing between mesh points. This is accomplished by using smoothed versions of the Heaviside and delta functions, with the smoothed Heaviside function H_ϵ defined as

$$H_\epsilon(\phi) = \begin{cases} 0, & \text{if } \phi < -\epsilon, \\ \frac{1}{2}[1 + \frac{\phi}{\epsilon} - \frac{1}{\pi}\sin(\pi\phi/\epsilon)], & \text{if } |\phi| \leq \epsilon, \\ 1, & \text{if } \phi > \epsilon, \end{cases} \tag{2.25}$$

with ϵ an arbitrary value, set as $\epsilon = 1.5\Delta x$ in this work. The corresponding smoothed delta function is then defined as $\delta_\epsilon = \frac{dH_\epsilon(\phi)}{d\phi}$. In this formulation, it is clear that the interface's thickness depends on the gradient of ϕ near the interface. In fact, the thickness is equal to $2\epsilon/|\nabla\phi|$, assuming that $|\nabla\phi|$ is constant in the normal direction. Thus, as ϕ deviates significantly from a signed distance function, i.e. $|\nabla\phi|$ deviates significantly from 1, the interface's thickness also diverges significantly from its expected value of $3\Delta x$. Such variations in interface thickness can cause a loss of stability and accuracy of the method, and so must be avoided. In order to do so, the Level Set equation must be kept close to a signed distance function, at least near the interface. This can be achieved by a process called reinitialization,

which is discussed in detail in section 2.2.4.

As a remark, it is worth mentioning that it is possible to have a sharp interface using Level Set methods, such as with the Ghost-Fluid method [68]. Such an approach is not straightforward, however, and is not used in this work.

2.2.4 Level Set Reinitialization

As previously mentioned, it is essential to the stability and accuracy of the numerical scheme that the Level Set function ϕ remains a good approximation of a signed distance function. The main solution to this issue is to periodically reinitialize the Level Set function, such that it becomes a close approximation to a signed distance function while preserving the interface's position, although it is possible to implement Level Set formulations that don't require reinitialization [69]. Reinitialization can be performed in several different ways, such as by solving a reinitialization equation [60], by using Fast Marching Methods [70] or Fast Sweeping Methods [71], or using a novel approach based on the Hopf-Lax formula [72].

In this work, reinitialization is performed by solving the reinitialization equation

$$\frac{\partial \phi}{\partial \tau} + S(\phi)(|\nabla \phi| - 1) = 0\,, \tag{2.26}$$

where $S(\phi)$ is a signum function and τ is a virtual time. It clear that such an equation preserves the position of the interface, since $\frac{\partial \phi}{\partial \tau} = 0$ if $\phi = 0$, and that it's steady-state solution, with $\frac{\partial \phi}{\partial \tau} = 0$ everywhere, results in ϕ being a signed distance function. Additionally, note that the reinitialization equation can be rewritten as the advection equation

$$\frac{\partial \phi}{\partial \tau} + \mathbf{w} \cdot \nabla \phi - S(\phi) = 0\,, \tag{2.27}$$

with

$$\mathbf{w} = S(\phi)\frac{\nabla \phi}{|\nabla \phi|}\,. \tag{2.28}$$

In this advective formulation, it is clear that the reinitialization equation is, effectively, propagating information away from the interface, with a velocity of 1 and in the direction normal to the interface. As a result, it quickly reaches steady-state in regions close to the interface. This is a major advantage, since ϕ only needs to be a signed distance function in regions close to the interface, which is quickly accomplished by the reinitialization equation, without the need to actually reach steady-state across the entire domain.

Similarly to what is used for the Heaviside and delta functions, it is also essential to define a smoothed signum function $S_\epsilon(\phi)$, in order to ensure that it is suitable for a discrete

16

implementation. In this work, the smoothed signum function is defined as [73]

$$S_\epsilon(\phi) = \frac{\phi}{\sqrt{\phi^2 + |\nabla\phi|^2\epsilon^2}}\,, \tag{2.29}$$

with $\epsilon = \Delta x$. Note that, in this definition, $S_\epsilon(\phi)$ is not simply a smoothed version of the traditional sharp signum function, with it's values varying with $|\nabla\phi|$ in such a way that it speeds up convergence of the reinitialization equation when the Level Set function is flat ($|\nabla\phi| < 1$), and dampens it when the Level Set function is steep ($|\nabla\phi| > 1$). The result is a more consistent reinitialization procedure, avoiding the slow convergence that would otherwise be associated with a flat Level Set function, as well as the possible instabilities and numerical errors that would otherwise be associated with a steep Level Set function.

When performing reinitialization on a discrete setting, however, numerical errors can lead to changes in the interface's position and, in turn, to unphysical variations in droplet volume. In order to counteract this issue, Sussman and Fatemi [74] presented an updated version of the reinitialization equation, which imposes a local conservation of droplet volume on each reinitialization step. This conservation is imposed by ensuring that

$$\frac{\partial}{\partial\tau}\int_\Omega H_\epsilon(\phi)dV = 0\,, \tag{2.30}$$

where Ω is an arbitrary, fixed domain. In order to effectively enforce this conservation, the reinitialization equation is modified to

$$\frac{\partial\phi}{\partial\tau} + \mathfrak{L}(\phi) - \lambda f(\phi) = 0\,, \tag{2.31}$$

$$\mathfrak{L}(\phi) = S_\epsilon(\phi)(|\nabla\phi| - 1)\,, \tag{2.32}$$

with the parameter λ defined as being only a function of time inside the domain Ω, and $f(\phi)$ an arbitrary function.

From Eq. 2.30, we have that

$$\frac{\partial}{\partial\tau}\int_\Omega H_\epsilon(\phi)dV = \int_\Omega H'_\epsilon(\phi)\frac{\partial\phi}{\partial\tau}dV = \int_\Omega H'_\epsilon(\phi)(-\mathfrak{L}(\phi) + \lambda f(\phi))dV = 0\,, \tag{2.33}$$

and, since λ is only a function of time inside Ω, we have that

$$\lambda = \frac{\int_\Omega H'_\epsilon(\phi)\mathfrak{L}(\phi)dV}{\int_\Omega H'_\epsilon(\phi)f(\phi)dV}\,. \tag{2.34}$$

The arbitrary function $f(\phi)$ can then be defined as [74]

$$f(\phi) = H'_\epsilon(\phi)|\nabla\phi|\,, \tag{2.35}$$

ensuring that this volume-preserving correction is only applied at the interface. Note that $H'_\epsilon(\phi) = \delta_\epsilon(\phi)$. When implementing a discrete version of this correction scheme, the domain Ω is set as an individual grid cell, so that any interface displacement is corrected locally.

2.2.5 Local Level Set

As evidenced by the definitions of the smoothed Heaviside and delta functions, the actual shape of the Level Set function is only important on a narrow band around it's zero level set, corresponding to the thickness of the smoothed interface, and so it's shape outside of this region is unimportant. Thus, accurately solving the Level Set's transport and reinitialization equations across the entire domain would be a significant waste of computational time, given that they only need to be accurately solved in a narrow region around the interface. In order to take advantage of this, and drastically reduce the cost of using Level Set methods, Peng et al. [73] proposed a fast local Level Set method.

This local Level Set scheme consists in, first, defining two band thickness parameters ϵ_2 and ϵ_3, in addition to the interface thickness ϵ, with $\epsilon < \epsilon_2 < \epsilon_3$. These parameters are then used to define two bands in the domain, an inner one defined by $\phi < \epsilon_2$, in which the Level Set function is accurately determined, and an outer one, defined by $\epsilon_2 < \phi < \epsilon_3$, which acts as a transition layer between the inner band, where the Level Set function is accurately calculated, and the outer region in which the Level Set function is not calculated at all. Note that the interface is entirely contained inside the inner band. Based on these parameters, the method then introduces a cutoff function $C(\phi)$, defined as

$$C(\phi) = \begin{cases} 1, & \text{if } |\phi| \le \epsilon_2, \\ (|\phi| - \epsilon_3)^2(2|\phi| + \epsilon_3 - 3\epsilon_2)/(\epsilon_3 - \epsilon_2)^3, & \text{if } \epsilon_2 < |\phi| \le \epsilon_3, \\ 0, & \text{if } |\phi| > \epsilon_3. \end{cases} \tag{2.36}$$

In [73], the authors suggest different different values of ϵ_2 and ϵ_3 depending on the size of the stencil used to calculate the derivatives of ϕ. For a fifth-order WENO scheme, as is the case in this work, the band thicknesses are defined as $\epsilon_2 = 3\Delta x$ and $\epsilon_3 = 6\Delta x$. Using this cutoff function, the Level Set advection equation is then rewritten as:

$$\frac{\partial \phi}{\partial t} + C(\phi)\mathbf{u} \cdot \nabla\phi = 0\,, \tag{2.37}$$

ensuring that no numerical errors arise due to sharp transitions between the accurately up-

dated band and the outer region where the Level Set function is not updated. For the reinitialization equation, a cutoff function is not necessary, since information propagates strictly away from the interface, and values of the Level Set function outside of the inner band do not affect the reinitialization of the values inside it. The Level Set function is then defined outside of the outer band as

$$\phi = \begin{cases} -\epsilon_3, & \text{if } \phi < -\epsilon_3, \\ \phi, & \text{if } |\phi| \leq \epsilon_3, \\ \epsilon_3, & \text{if } \phi > \epsilon_3. \end{cases} \tag{2.38}$$

2.2.6 Surface and Volume Integrals

In applications containing interfaces, it is often necessary to evaluate volume integrals over the region bounded by the interface, or surface integrals over the interface. When using Level Set methods, both can be evaluated in a fairly straightforward way. To evaluate an arbitrary function $\mathfrak{F}$ over the region Ω_i bounded by the interface, one can simply use the identity [64]

$$\int_{\Omega_i} \mathfrak{F} dV = \int_V \mathfrak{F} H_\epsilon(-\phi) dV \,, \tag{2.39}$$

where V is the entire domain. Alternatively, for an integral over the region Ω_o outside of the interface:

$$\int_{\Omega_o} \mathfrak{F} dV = \int_V \mathfrak{F} H_\epsilon(\phi) dV \,. \tag{2.40}$$

For surface integrals over the interface Γ, these can be evaluated as

$$\int_\Gamma \mathfrak{F} dS = \int_V \mathfrak{F} \delta_\epsilon(\phi) |\nabla \phi| dV \,. \tag{2.41}$$

2.3 FINITE DIFFERENCE DISCRETIZATION

In this work, the governing equations are discretized using a finite difference scheme, with derivatives approximated by differences between the values of a variable on given grid points. As an example, consider a one dimensional arbitrary variable ξ, defined in a grid with N points ($x_1, x_2, ..., x_N$), with uniform spacing Δx. For a given point x_i, the first derivative of this function can be approximated with either a forward difference

$$\frac{d\xi(x_i)}{dx} = \frac{\xi(x_{i+1}) - \xi(x_i)}{\Delta x} + O(\Delta x) \,, \tag{2.42}$$

19

or a backward difference

$$\frac{d\xi(x_i)}{dx} = \frac{\xi(x_i) - \xi(x_{i-1})}{\Delta x} + O(\Delta x), \tag{2.43}$$

both with first order accuracy in respect to Δx, or with a centered discrete derivative

$$\frac{d\xi(x_i)}{dx} = \frac{\xi(x_{i+1}) - \xi(x_{i-1})}{2\Delta x} + O(\Delta x^2), \tag{2.44}$$

with second order accuracy in respect to Δx, albeit with a larger stencil. This derivative can also be approximated with second order accuracy and a smaller stencil, defined at the midpoint between two grid points as

$$\frac{d\xi(x_{i+\frac{1}{2}})}{dx} = \frac{\xi(x_{i+1}) - \xi(x_i)}{\Delta x} + O(\Delta x^2), \tag{2.45}$$

which also has a smaller error when compared to the approximation with a larger stencil. To obtain an approximation for the second derivative, one can take the finite difference between the first derivatives on the two adjacent midpoints $x_{i-\frac{1}{2}}$ and $x_{i+\frac{1}{2}}$, defined as

$$\frac{d^2\xi(x_i)}{dx^2} = \frac{\xi'(x_{i+\frac{1}{2}}) - \xi'(x_{i-\frac{1}{2}})}{\Delta x} + O(\Delta x^2), \tag{2.46}$$

$$\frac{d^2\xi(x_i)}{dx^2} = \frac{\xi(x_{i+1}) - 2\xi(x_i) + \xi(x_{i-1})}{\Delta x^2} + O(\Delta x^2). \tag{2.47}$$

Partial derivatives of multidimensional functions can be approximated analogously, and higher order approximations are possible, albeit with larger stencils.

Although a discretization with all variables co-located on the same grid points may seem more straightforward, such an arrangement proves to be troublesome when used to solve the Navier-Stokes equations [75, 76]. The reason for this is that, when calculating the pressure gradient for the momentum equations or the divergence of the velocity for the continuity equation on co-located grids, derivatives with an enlarged ($2\Delta x$) stencil must be used. Since these derivatives only use the two adjacent points in its calculation (x_{i-1} and x_{i+1}), they can remain oblivious to massive oscillations. For example, if $P = 100$ on even points and $P = 300$ on odd points, such a derivative would indicate that $\nabla P = 0$ across the entire domain. This, in turn, can lead to the accumulation of numerical errors and result in unstable and/or inaccurate solutions.

One solution to this issue, which is commonly employed when discretizing simple geometries and is used in this work, is to use a staggered, rather than co-located arrangement of variables. This staggered arrangement is commonly referred to in the literature as Marker and Cell (MAC), in reference to the scheme presented in [77] which first introduced it, and

consists in dividing the domain in cells, with scalar variables located on cell centers and components of vector variables located on the center of cell faces normal to their directions. In this arrangement, Eq. 2.45 can be used to calculate the pressure gradient for the momentum equations, since the velocities are defined in the midpoints between pressure grid points, and to calculate the divergence of the velocity field for the continuity equation, since pressure is defined in the midpoints between velocity grid points. Although this arrangement works well for simulations involving simple geometries, the increased discretization complexity is not well-suited for simulations involving complex geometries, and so different solution must be employed when dealing with such cases.

In this work, both the Navier-Stokes equations and the magnetic potential equation are discretized with the second order, centered derivatives previously described, with the exception of the velocity derivatives for the advective term $\mathbf{u} \cdot \nabla \mathbf{u}$, which use a second order Essentially Non-Oscillatory (ENO) scheme with upwinding, presented in Section 2.4. This, however, is not a requirement for flows with low Reynolds numbers, since the large viscous diffusion stabilizes the solution. When required, values located between grid points are approximated using linear interpolation.

Special attention must be given to the discretization of the curvature term in Eq. 1.4. It can be calculated as $\kappa = \nabla \cdot (\nabla\phi/|\nabla\phi|)$, without any further assumptions. However, this requires approximating a first derivative twice with Eq. 2.44, which results in an enlarged stencil. An alternative is to calculate the curvature as $\kappa = \nabla^2\phi/|\nabla\phi|$, assuming that $|\nabla\phi|$ is locally constant. This results in a smaller stencil, approximating the second derivatives with Eq. 2.47, at the cost of reduced stability and generality. In this work the first approach is used, as it offers better stability and generality, although the second is used on some particular cases, which are specified accordingly.

When discretizing the terms of the type $\nabla \cdot (\zeta(\mathbf{x})\nabla\psi)$, special care must be taken to ensure that it does not use an extended stencil. To this end, the term $\zeta(\mathbf{x})\nabla\psi$ must be evaluated in the midpoints between grid points of ψ, so that the stencil only encompasses the points adjacent to the central one. In the one dimensional case, this results in

$$
\begin{aligned}
\frac{d}{dx}\left(\zeta(x_i)\frac{d\psi(x_i)}{dx}\right) &\approx \frac{\zeta(x_{i+\frac{1}{2}})\psi'(x_{i+\frac{1}{2}}) - \zeta(x_{i-\frac{1}{2}})\psi'(x_{i-\frac{1}{2}})}{\Delta x} \\
&\approx \frac{1}{\Delta x}\left[\zeta(x_{i+\frac{1}{2}})\left(\frac{\psi(x_{i+1}) - \psi(x_i)}{\Delta x}\right) - \zeta(x_{i-\frac{1}{2}})\left(\frac{\psi(x_i) - \psi(x_{i-1})}{\Delta x}\right)\right],
\end{aligned}
\tag{2.48}
$$

with a straightforward extension to the multidimensional case. Note, however, that ζ might not be defined in $x_{i+\frac{1}{2}}$, requiring some interpolation procedure. A simple linear interpolation may be adequate for some particular cases, such as when simulating flows with bubbles [78], although it is not, in general, well suited for this purpose.

An improved alternative is the one presented in [76], which is based on ensuring a con-

21

stant flux $\Phi = \zeta \nabla \psi$ between grid cells. In the one dimensional case, and assuming a sharp transition between $\zeta(x_i)$ and $\zeta(x_{i+1})$ in the midpoint between the cells, such a conservation of flux can be represented as

$$\Phi = \zeta(x_i) \left(\frac{\psi(x_{i+\frac{1}{2}}) - \psi(x_i)}{0.5\Delta x} \right) = \zeta(x_{i+1}) \left(\frac{\psi(x_{i+1}) - \psi(x_{i+\frac{1}{2}})}{0.5\Delta x} \right) , \qquad (2.49)$$

for both half-cells individually, or, combined, as

$$\Phi = \zeta(x_{i+\frac{1}{2}}) \left(\frac{\psi(x_{i+1}) - \psi(x_i)}{\Delta x} \right) . \qquad (2.50)$$

Equations 2.49 and 2.50 can be solved as a system of equations, resulting in

$$\frac{2}{\zeta(x_{i+\frac{1}{2}})} = \frac{1}{\zeta(x_i)} + \frac{1}{\zeta(x_{i+1})} , \qquad (2.51)$$

which corresponds to the harmonic mean between the two adjacent grid points. Thus, harmonic means are used when estimating values of ζ (and η, analogously) between grid points, rather than arithmetic means, which would correspond to a linear interpolation. It is worth mentioning that this scheme was first introduced for the solution of the heat equation, although it is well suited for other applications. A more complex scheme, made specifically for the case of viscous stresses, is presented in [79], although such an approach was not pursued in this work.

2.4   DISCRETIZATION OF THE LEVEL SET EQUATIONS

Due to the advective nature of the Level Set transport and reinitialization equations, a simple central differencing discretization scheme would result in an unphysical downwind transport of information and in an unstable numerical solution. Furthermore, since numerical errors in the solution of these equations can lead to a loss of droplet volume, it is desirable that their solution is highly accurate. Thus, the Level Set equations are discretized with high-order schemes developed for hyperbolic conservation laws, commonly found in the solution of compressible flows [80, 81]. This section presents a general description of the methods used to solve the Level Set equations in this work, although an in-depth and comprehensive review of such methods is outside the scope of this work.

The time-stepping of the transport and reinitialization equations is performed with a third-order, Strong Stability Preserving (SSP)[1] explicit Runge-Kutta scheme [82]. Explicit

---

[1]SSP Runge-Kutta schemes were previously referred to as TVD (Total Variation Diminishing) Runge-Kutta schemes

methods are ideal in this case, since neither of the equations are stiff, as they are computationally cheaper than implicit methods, which would require the solution of linear systems of equations on each time step, and allow for a straightforward implementation of upwinding schemes. Strong Stability Preserving schemes are a special class of Runge-Kutta schemes, developed for the solution of hyperbolic conservation laws, aimed at ensuring the stability of the solution even if it has discontinuities.

Consider a generic hyperbolic conservation law

$$\frac{\partial \phi}{\partial t} = L(\phi, t) \,, \tag{2.52}$$

with L a term containing the PDE's spacial derivatives. For example, $L(\phi, t) = -\mathbf{u}(t) \cdot \nabla \phi$ for the Level Set transport equation, and $L(\phi, t) = S_\epsilon(\phi)(1 - |\nabla \phi|)$ for the reinitialization equation. Using the method of lines, this conservation law can be represented as a semi-discrete system of ordinary differential equations (ODEs)

$$\frac{\partial \phi}{\partial t} = \mathfrak{L}(\phi, \mathfrak{t}) \,, \tag{2.53}$$

with $\mathfrak{L}$ a discrete approximation of L. With a first-order explicit Euler scheme for time-stepping, this results in

$$\phi^{n+1} = \phi^n + \Delta t \mathfrak{L}(\phi^{\mathfrak{n}}, \mathfrak{t}^{\mathfrak{n}}) \,, \tag{2.54}$$

for an arbitrary instant n.

Although such a scheme can be stable even in the case of a discontinuous solution if a proper discretization is used for the derivatives, it is only first order accurate in time. To obtain higher order schemes in time, while still preserving stability, Shu and Osher [83] devised SSP Runge-Kutta schemes consisting of convex linear combinations of explicit Euler steps. The third-order scheme, used in this work, can be written as

$$
\begin{aligned}
\phi_1 &= \phi^n + \Delta t L(\phi^n, t^n) \,, \\
\phi_2 &= \frac{3}{4}\phi^n + \frac{1}{4}\phi_1 + \frac{1}{4}\Delta t L(\phi_1, t_1) \,, \\
\phi^{n+1} &= \frac{1}{3}\phi^n + \frac{2}{3}\phi_2 + \frac{2}{3}\Delta t L(\phi_2, t_2) \,,
\end{aligned}
\tag{2.55}
$$

where ϕ_1 and ϕ_2 are auxiliary intermediate variables, or, alternatively, as

$$
\begin{aligned}
\tilde{\phi}^{n+1} &= \phi^n + \Delta t L(\phi^n, t^n)\,, \\
\tilde{\phi}^{n+2} &= \tilde{\phi}^{n+1} + \Delta t L(\tilde{\phi}^{n+1}, t^n + \Delta t)\,, \\
\tilde{\phi}^{n+\frac{1}{2}} &= \frac{3}{4}\phi^n + \frac{1}{4}\tilde{\phi}^{n+2}\,, \\
\tilde{\phi}^{n+\frac{3}{2}} &= \tilde{\phi}^{n+\frac{1}{2}} + \Delta t L(\tilde{\phi}^{n+\frac{1}{2}}, t^n + \frac{1}{2}\Delta t)\,, \\
\phi^{n+1} &= \frac{1}{3}\phi^n + \frac{2}{3}\tilde{\phi}^{n+\frac{3}{2}}\,,
\end{aligned}
\tag{2.56}
$$

where the tilde is used to indicate auxiliary intermediate variables.

For the time-stepping of the Level Set transport equation, Δt is the same one set for Navier-Stokes equations, and it will always be enough to ensure stability, since both equations are subject to similar CFL constraints, with the Navier-Stokes equations subject to stricter constraints arising from the magnetic and surface tension forcing terms. In the case of the reinitialization equation, however, the time steps are completely independent, since it is evolved in the virtual time τ. From Eq. 2.28, the characteristic propagation speed of the reinitialization equation is $|\mathbf{w}| = 1$, resulting in a Courant number

$$
C = \frac{|\mathbf{w}|\Delta\tau}{\Delta x} = \frac{\Delta\tau}{\Delta x}\,.
\tag{2.57}
$$

Since $C \leq 0.5$ ensures that the scheme is stable and monotone [73], the time step can then be set as $\Delta\tau = 0.5\Delta x$.

As previously mentioned, the SSP Runge-Kutta scheme will result in a stable solution, provided that an appropriate discretization is used for the spacial derivatives. Several different schemes satisfy this condition, generally referred to as shock capturing schemes [84, 85], due to their distinct ability to deal with discontinuities in the solution. One of the most prominent classes of such methods are ENO (Essentially Non-Oscillatory) and WENO (Weighted ENO) schemes, using polynomials that fit the data to approximate the derivatives.

There are, however, different possible polynomials for a given stencil. The ENO scheme, then, chooses the smoothest one, to avoid crossing a discontinuity while approximating a derivative. The WENO scheme, though, instead considers all possible polynomials, approximating the derivative with a dynamically weighted, convex combination of the polynomials [86]. If all polynomials are smooth, the weights are chosen such as to maximize the order of accuracy, which will be higher than that of an ENO scheme of equal stencil. If strong oscillations appear in one or more polynomials, however, their respective weights are reduced, resulting in a lower order of accuracy but ensuring the stability of the solution. In the worst case scenario, the WENO scheme reverts to the ENO scheme of equal stencil.

For the discretization of the advective term of the Navier-Stokes equations, a second

order ENO scheme is used, as presented in [64]. Although the implementation of higher-order schemes would be straightforward, the additional computational cost would not be justified, given that all other terms are discretized with second order schemes. Using an auxiliary variable m, defined as $m = i - 1$ when calculating the backward derivative ϕ_x^-, or as $m = i$ when calculating the forward derivative ϕ_x^+, the first and second divided differences can be calculated as

$$
\begin{aligned}
D^1_{m+\frac{1}{2}} &= \frac{\phi_{m+1} - \phi_m}{\Delta x} \,, \\
D^2_m &= \frac{\phi_{m+1} - 2\phi_m + \phi_{m-1}}{2\Delta x^2} \,, \\
D^2_{m+1} &= \frac{\phi_{m+2} - 2\phi_{m+1} + \phi_m}{2\Delta x^2} \,.
\end{aligned}
\tag{2.58}
$$

A second auxiliary variable c is then introduced, defined as $c = D^2_m$ if $|D^2_m| \leq |D^2_{m+1}|$, or as $c = D^2_{m+1}$ if $|D^2_{m+1}| < |D^2_m|$. Note that this choice of c, opting for the divided difference of least magnitude, ensures that the derivative is approximated using the smoothest of the available polynomials. The second order accurate derivative is then approximated as

$$
\phi_x = D^1_{m+\frac{1}{2}} + c(2(i - m) - 1)\Delta x \,.
\tag{2.59}
$$

For the discretization of the Level Set transport and reinitialization equations, a fifth order WENO scheme is used. Initially proposed in [87] for derivatives approximated at the midpoint between grid points, and later extended for derivatives approximated on grid points in [88], this method approximates the derivatives as a convex combination of the three possible approximations of the third order ENO scheme of equal stencil. Rewritten in the form presented in [64], the scheme is as follows. When calculating a backward derivative ϕ_x^-, first set the parameters

$$
\begin{aligned}
v_1 &= \frac{\phi_{i-2} - \phi_{i-3}}{\Delta x} \,, \\
v_2 &= \frac{\phi_{i-1} - \phi_{i-2}}{\Delta x} \,, \\
v_3 &= \frac{\phi_i - \phi_{i-1}}{\Delta x} \,, \\
v_4 &= \frac{\phi_{i+1} - \phi_i}{\Delta x} \,, \\
v_5 &= \frac{\phi_{i+2} - \phi_{i+1}}{\Delta x} \,.
\end{aligned}
\tag{2.60}
$$

25

When calculating a forward derivative ϕ_x^+, these parameters are otherwise defined as

$$
\begin{aligned}
v_1 &= \frac{\phi_{i+3} - \phi_{i+2}}{\Delta x} \,, \\
v_2 &= \frac{\phi_{i+2} - \phi_{i+1}}{\Delta x} \,, \\
v_3 &= \frac{\phi_{i+1} - \phi_i}{\Delta x} \,, \\
v_4 &= \frac{\phi_i - \phi_{i-1}}{\Delta x} \,, \\
v_5 &= \frac{\phi_{i-1} - \phi_{i-2}}{\Delta x} \,.
\end{aligned}
\tag{2.61}
$$

With these parameters set, the rest of the scheme becomes identical for both backward and forward derivatives. For an ENO scheme of equal stencil, the three possible derivatives are

$$
\begin{aligned}
\phi_x^1 &= \frac{v_1}{3} - \frac{7v_2}{6} + \frac{11v_3}{6} \,, \\
\phi_x^2 &= -\frac{v_2}{6} + \frac{5v_3}{6} + \frac{v_4}{3} \,, \\
\phi_x^3 &= \frac{v_3}{3} + \frac{5v_4}{6} - \frac{v_5}{6} \,,
\end{aligned}
\tag{2.62}
$$

and the resulting WENO derivative, a convex combination of these three ENO derivatives, is defined as

$$
\phi_x = \omega_1 \phi_x^1 + \omega_2 \phi_x^2 + \omega_3 \phi_x^2 \,,
\tag{2.63}
$$

with the optimal weights of $\omega_1 = 0.1$, $\omega_2 = 0.6$ and $\omega_3 = 0.3$ resulting in fifth order accuracy. Simply using these weights would lead to unstable approximation on non-smooth regions, however, so that the weights must be dynamically calculated in order to ensure a balance between stability and accuracy of the approximation. To this end, smoothness coefficients associated with each of the three derivatives are calculated as

$$
\begin{aligned}
S_1 &= \frac{13}{12}(v_1 - 2v_2 + v_3)^2 + \frac{1}{4}(v_1 - 4v_2 + 3v_3)^2 \,, \\
S_2 &= \frac{13}{12}(v_2 - 2v_3 + v_4)^2 + \frac{1}{4}(v_2 - v_4)^2 \,, \\
S_3 &= \frac{13}{12}(v_3 - 2v_4 + v_5)^2 + \frac{1}{4}(3v_3 - 4v_4 + v5)^2 \,,
\end{aligned}
\tag{2.64}
$$

which in turn are used to calculate the parameters

$$\alpha_1 = \frac{0.1}{(S_1 + \epsilon)^2},$$
$$\alpha_2 = \frac{0.6}{(S_2 + \epsilon)^2}, \tag{2.65}$$
$$\alpha_3 = \frac{0.3}{(S_3 + \epsilon)^2},$$

with $\epsilon = 10^{-6}$. The weights for Eq. 2.63 are then defined as the normalized parameters

$$\omega_1 = \frac{\alpha_1}{\alpha_1 + \alpha_2 + \alpha_3},$$
$$\omega_2 = \frac{\alpha_2}{\alpha_1 + \alpha_2 + \alpha_3}, \tag{2.66}$$
$$\omega_3 = \frac{\alpha_3}{\alpha_1 + \alpha_2 + \alpha_3},$$

with $\omega_1 + \omega_2 + \omega_3 = 1$. Note that, if all three smoothness coefficients have similar values, the weights for Eq. 2.63 assume the optimal values for a fifth order accurate derivative.

For the advective terms of the Navier-Stokes equations and the Level Set transport equation, the choice of which derivative to use is straightforward, using a simple upwinding scheme. Since the velocity field $\mathbf{u}$ is previously known and continuous, the derivatives can simply be chosen as ϕ_x^- if $u \geq 0$ or ϕ_x^+ if $u < 0$, and similarly for the other directions. For the Level Set reinitialization equation, however, the propagation velocity $S(\phi)\nabla\phi/|\nabla\phi|$ is discontinuous across the interface, where it changes sign. Thus, in order to obtain an accurate and stable solution, a more elaborate scheme must be used. There are two canonical monotone schemes which are adequate for this case [73], the Engquist-Osher scheme [63] and Godunov's scheme [89], the latter of which is used in this work. Since a lengthier discussion regarding such methods is outside the scope of this work, only a brief description of Godunov's scheme, as applied to the Level Set reinitialization equation, is presented here.

When discretizing an advective term of the type $a\nabla\phi \cdot \nabla\phi$, the characteristic propagation velocity is $a\nabla\phi$, with a an arbitrary scalar. With an upwinding scheme, $\nabla\phi$ can be approximated with either backward or forward derivatives, so that there are two possible propagation velocities in each direction, for example $a\phi_x^-$ and $a\phi_x^+$ in the x direction. If both velocities have the same sign, then use of upwinding is straightforward, since both alternatives indicate the use of the same derivative. An issue arises, however, if the two velocities have different signs, since it is not clear which derivative should be used. In this case, the core idea of Godunov's scheme is to choose the most meaningful solution.

If $a\phi_x^- < 0$ and $a\phi_x^+ > 0$, this indicates an expansion, with information flowing from a grid point in both directions. In this case, Godunov's scheme sets ϕ as being locally flat, with $\phi_x = 0$. On the other hand, if $a\phi_x^- > 0$ and $a\phi_x^+ < 0$, this indicates coalescing characteristics,

with information flowing towards a grid point from both directions. In this case, Godunov's scheme assumes that the grid point reacts to the information that first reaches it, and thus approximates ϕ_x with the largest in magnitude between ϕ_x^- and ϕ_x^+. This scheme can be elegantly represented as [90]

$$\phi_x^2 = \begin{cases} max\left(max(\phi_x^-,0)^2, min(\phi_x^+,0)^2\right), & \text{if } a > 0\,, \\ max\left(min(\phi_x^-,0)^2, max(\phi_x^+,0)^2\right), & \text{if } a < 0\,. \end{cases} \qquad (2.67)$$

In the case of the Level Set reinitialization equation, $a = S(\phi)/|\nabla\phi|$, so that, assuming a nonzero $|\nabla\phi|$, $a > 0$ if $\phi > 0$ and $a < 0$ if $\phi < 0$.

2.5   GENERAL DESCRIPTION OF THE COMPLETE METHOD

With all components of the numerical scheme presented in the previous section, this section aims to describe how all the methods are combined in a computational code. In this work, the aforementioned methods are combined in the computational code FENRir (Ferrofluid Emulsion Numerical Rheometer), as described in Algorithm 1. The solution methods employed to solve the linear systems of equations are described in Chapter 3. Note that, since the Level Set function is extrapolated for the calculation of the forcing terms in Eq. 2.8, it is reinitialized on every iteration, in order to avoid extrapolating across a discontinuity.

Algorithm 1: General description of the computational code FENRir

1   Read simulation parameters from a configuration file
2   Allocate and initialize all variables, including the Level Set function in accordance
     to the initial droplet shape
3   Define the interface location and flatten the Level Set function, as described in
     Sec. 2.2.5
4   **if** *ferrofluid droplet* **then**
5   Solve the magnetic potential equation (Eq. 1.1)
6   Calculate the magnetic field from the magnetic potential

7   **for** $it = 1$ **to** nit **do**
8   Solve the equations for the tentative velocity field (Eq. 2.8)
9   Solve the pressure Poisson equation (Eq. 2.10)
10   Perform the projection step to determine the final velocity field (Eq. 2.9)
11   Advect the Level Set function with Eq. 2.20)
12   Reinitialize the Level Set function with Eq. 2.31
13   Update the interface location and flatten the Level Set function, as described
       in Sec. 2.2.5
14   **if** *ferrofluid droplet* **then**
15   Solve the magnetic potential equation (Eq. 1.1), using
        $\tilde{\psi} = 2\psi^{it} - \psi^{it-1}$ as an initial approximation of the solution
16   Calculate the magnetic field from the magnetic potential

17   If applicable, perform any post-processing routines to extract flow data, and
       write relevant data to output files

3 DISCRETE EQUATION SOLVERS

This chapter presents a general outline of the solvers used in this work, ranging from basic relaxation methods, such as Jacobi and Gauss-Seidel, to more advanced and efficient solvers, such as Conjugate Gradients, Fast Poisson Solvers, and Multigrid methods. Also included are discussions regarding the implementation of these different solvers, and a comparison of their performances. When discretized, the corresponding partial differential equations become linear systems of equations, which can be written in matrix form as

$$A\mathbf{x} = \mathbf{b}\,. \tag{3.1}$$

In order to test the accuracy of a given solution $\tilde{\mathbf{x}}$, one can define the residual as

$$\mathbf{r} = \mathbf{b} - A\tilde{\mathbf{x}}\,, \tag{3.2}$$

and accept $\tilde{\mathbf{x}}$ as a solution if the residual is sufficiently small. Note that, as the residual approaches zero, the solution $\tilde{\mathbf{x}}$ approaches the exact solution $\mathbf{x}$. The definition of what is a sufficiently small residual is entirely arbitrary, but one of the most usual alternatives, which is the one used in this work, is to check if the L^2 (Euclidean) norm of the residual is smaller than a given value, referred to as the tolerance.

3.1 BASIC RELAXATION METHODS

Relaxation methods include some of the simplest solvers for the discrete elliptic equations described in the previous chapter, such as the Jacobi and Gauss-Seidel methods, which consist of very simple iterative procedures. In order to describe these methods, let the matrix A in Eq. 3.1 be split into a lower triangular component L, an upper triangular component A, and a diagonal component D, such that $A = L + D + U$. Let also the individual components of A be referred to as $a_{i,j}$, the individual components of $\mathbf{x}$ be referred to as x_i, and the individual components of $\mathbf{b}$ be reffered to as b_i. Then, for a given approximate solution $\mathbf{x}^n$, the Jacobi iteration can be defined in matrix form as [91]

$$\mathbf{x}^{n+1} = D^{-1}(\mathbf{b} - (L + U)\mathbf{x}^n)\,, \tag{3.3}$$

where the inversion of the diagonal matrix D is trivial, or, alternatively, in element-based form as

$$x_i^{n+1} = \frac{1}{a_{i,i}} \left(b_i - \sum_{j \neq i} a_{i,j} x_j^n \right) . \tag{3.4}$$

It can be seen that, for computational implementations of the Jacobi method, an additional array is required to store the values of $\mathbf{x}^{n+1}$, since the values of $\mathbf{x}^n$ cannot be overwritten until the entire iteration is complete. The Gauss-Seidel method is very similar to the Jacobi method, and can be defined as [91]

$$\mathbf{x}^{n+1} = (L + D)^{-1}(\mathbf{b} - U\mathbf{x}^n) , \tag{3.5}$$

or, in element-based form, as

$$x_i^{n+1} = \frac{1}{a_{i,i}} \left(b_i - \sum_{j < i} a_{i,j} x_j^{n+1} - \sum_{j > i} a_{i,j} x_j^n \right) . \tag{3.6}$$

It can be seen that, unlike the Jacobi method, the computations of a Gauss-Seidel iteration must be made in a specific order, which makes parallelization, although still possible, more complicated and inefficient, whereas the Jacobi method is easily and efficiently parallelizable. The advantages of the Gauss-Seidel method, however, are that it does not require an additional array, since the values of $\mathbf{x}^n$ can be overwritten by the values of $\mathbf{x}^{n+1}$ as they are computed, and that it usually has faster convergence. It is important to note, however, that the Gauss-Seidel method has a preferential direction of information transfer, which, whenever possible, should be aligned with the preferential direction of information transfer of a given problem, in order to accelerate convergence. When a given problem doesn't have a clear preferential direction of information transfer, it can be advantageous to perform Gauss-Seidel iterations using alternating directions.

Both of the iterative methods previously discussed can be improved upon by using a weighting parameter, ω, in order to obtain more desirable convergence properties. For the case of the Jacobi method, this results in the weighted, or damped, Jacobi method [91]:

$$x_i^{n+1} = (1 - \omega)x_i^n + \frac{\omega}{a_{i,i}} \left(b_i - \sum_{j \neq i} a_{i,j} x_j^n \right) , \tag{3.7}$$

and, for the Gauss-Seidel method, this results in Successive Over-Relaxation (SOR) [91]:

$$x_i^{n+1} = (1 - \omega)x_i^n + \frac{\omega}{a_{i,i}} \left(b_i - \sum_{j < i} a_{i,j} x_j^{n+1} - \sum_{j > i} a_{i,j} x_j^n \right) . \tag{3.8}$$

31

Note that, when $\omega = 1$, the damped Jacobi reduces to the standard Jacobi method, and SOR reduces to the standard Gauss-Seidel method. It is clear that different values of ω can have significant impacts on the convergence properties of the aforementioned methods, and, thus, an appropriate choice of this relaxation parameter is essential. In the case of the damped Jacobi method, it is usual to have $\omega < 1$ when it is used as a smoother for multigrid methods [91] since, although it decreases the overall convergence speed, it does improve the smoothing properties of the method. In the case of SOR, significant performance improvements can be obtained with $1 < \omega < 2$, with the optimal value of ω depending on the problem. For Poisson equations, this optimal value is $\omega_{opt} = 2/(1 + \sin(\pi h))$, where h is the mesh spacing of the corresponding problem [92].
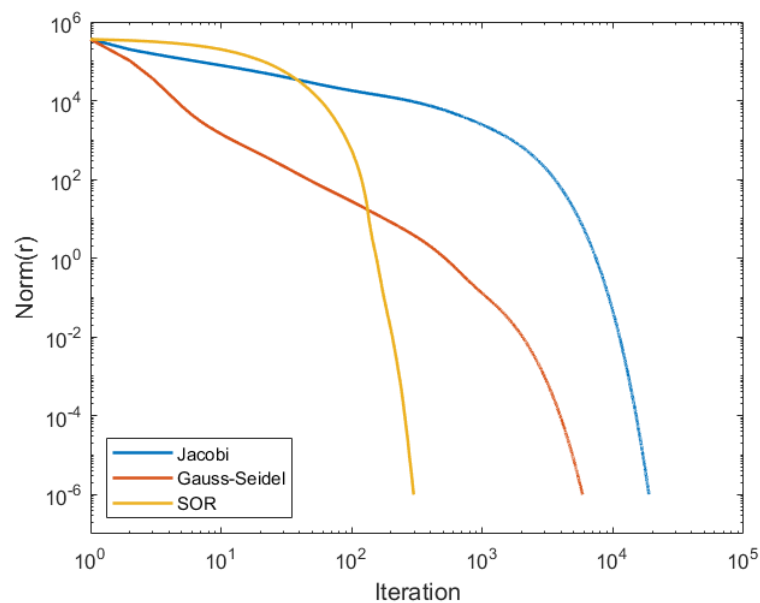
Figure 3.1 – Convergence of Jacobi, Gauss-Seidel, and SOR methods.

Figure 3.1 presents a comparison of the convergence speeds for the Jacobi, Gauss-Seidel and SOR methods for a 2D Poisson problem, on a 64×64 grid with homogeneous Dirichlet boundary conditions and random initial conditions. Note that, even for the faster SOR method, convergence still requires a large number of iterations for this simple model problem. As a remark, it should be noted that over-relaxation implies that $\omega > 1$, with the cases when $\omega < 1$ being referred to as under-relaxation. Additionally, it can be seen that SOR (and thus also Gauss-Seidel) iterations are not a symmetric procedure. They can be made a symmetric process, however, simply by performing two iterations with opposing directions, one forward and one backward (i.e. switching L and U in Eq. 3.5), resulting in Symmetric SOR (SSOR).

3.2 KRYLOV SUBSPACE METHODS

Krylov subspace methods are a class of highly successful iterative solvers for large linear algebra problems, based on the use of Krylov subspaces [93]. Some of the most successful modern methods for the solution of large systems of linear equations are Krylov subspace methods, such as Conjugate Gradient (CG) [94], Biconjugate Gradient Stabilized (Bi-CGSTAB) [95] and Generalized Minimal Residual (GMRES) [96], as well as variations and further developments of such methods. Since an in-depth discussion on Krylov subspace methods is outside the scope of this work, this section only presents a brief discussion regarding the Conjugate Gradient method and it's preconditioned version, which were implemented for the solution of linear systems of equations.

3.2.1 Conjugate Gradient

The Conjugate Gradient method is one of the most popular iterative methods for the solution of large, sparse linear systems of equations, defined as $A\mathbf{x} = \mathbf{b}$ and with an exact solution $\mathbf{x}^*$, an initial approximation $\mathbf{x}_0$, and A a symmetric and positive definite matrix. In order to solve the system, the method attempts to minimize the quadratic function

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T A \mathbf{x} - \mathbf{x}^T \mathbf{b}\,, \tag{3.9}$$

since the minimum of this function is located on the exact solution of the linear system of equations. Note that the gradient of this quadratic function is $\nabla f(\mathbf{x}) = A\mathbf{x} - \mathbf{b} = -\mathbf{r}$. To this end, the method iteratively seeks the minimum of the function alongside a search direction $\mathbf{d}$, such that, for a given approximate solution $\mathbf{x}_i$, the next approximation is obtained as $\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_i \mathbf{d}_i$, with α_i a scalar defined such that $\mathbf{x}_{i+1}$ is the location of the minimum of $f(\mathbf{x})$ alongside the search direction. Substituting the expression for $\mathbf{x}_{i+1}$ in Eq. 3.9 and minimizing it with respect to α_i, one finds that

$$\alpha_i = \frac{\mathbf{d}_i^T \mathbf{r}_i}{\mathbf{d}_i^T A \mathbf{d}_i}\,. \tag{3.10}$$

By substituting the expression for $\mathbf{x}_{i+1}$ in the definition of the residual, one also finds that $\mathbf{r}_{i+1} = \mathbf{r}_i - \alpha_i A \mathbf{d}_i$, a definition that reduces the demand for matrix-vector products, since $A\mathbf{d}_i$ is previously computed for α_i.

For the first iteration, it is natural to choose the search direction as the opposite of the gradient of $f(\mathbf{x})$, so that $\mathbf{d}_0 = \mathbf{r}_0$. For the following iterations, however, the method imposes that the search directions must be A-orthogonal. This ensures that the method converges in, at most, n iterations for an $n \times n$ matrix A. To this end, the search directions are constructed

by conjugation of the residuals using the Gram-Schmidt process, with the fact that the search directions are conjugated gradients giving rise to the method's name. The search directions are then defined as

$$\mathbf{d}_{i+1} = \mathbf{r}_{i+1} + \sum_{k=0}^{i} \beta_{ik}\mathbf{d}_k \,, \tag{3.11}$$

with

$$\beta_{ik} = -\frac{\mathbf{r}_i^T A \mathbf{d}_k}{\mathbf{d}_k^T A \mathbf{d}_k} \,. \tag{3.12}$$

However, due to the orthogonality relations between $\mathbf{r}$ and $\mathbf{d}$, this reduces to

$$\mathbf{d}_{i+1} = \mathbf{r}_{i+1} + \beta_{i+1}\mathbf{d}_i \,, \tag{3.13}$$

$$\beta_{i+1} = \frac{\mathbf{r}_{i+1}^T \mathbf{r}_{i+1}}{\mathbf{r}_i^T \mathbf{r}_i} \,. \tag{3.14}$$

The resulting method is described in Algorithm 2 (see [97] for a detailed description).

Algorithm 2: Conjugate Gradient method

1   $\mathbf{r} = \mathbf{b} - A\mathbf{x}$
2   $\mathbf{d} = \mathbf{r}$
3   $\delta_{new} = \mathbf{r}^T\mathbf{r}; \delta_{old} = \delta_{new}$
4   **while** $\sqrt{\delta_{new}} > $ *tolerance* **do**
5   $\quad$ $\mathbf{q} = A\mathbf{d}$
6   $\quad$ $\alpha = \delta_{new}/\mathbf{d}^T\mathbf{q}$
7   $\quad$ $\mathbf{x} = \mathbf{x} + \alpha\mathbf{d}$
8   $\quad$ $\delta_{old} = \delta_{new}, \delta_{new} = \mathbf{r}^T\mathbf{r}$
9   $\quad$ $\beta = \delta_{new}/\delta_{old}$
10 $\quad$ $\mathbf{d} = \mathbf{r} + \beta\mathbf{d}$

3.2.2   Preconditioned Conjugate Gradient

When using the Conjugate Gradient method to solve large linear systems, it is often desirable to use preconditioning, in order to accelerate the convergence. With preconditioning, the linear system then becomes

$$M^{-1}A\mathbf{x} = M^{-1}\mathbf{b} \,, \tag{3.15}$$

for a given preconditioner matrix M. The matrix M should be an approximation of A but easier to invert, in such a way that the work performed in the preconditioning step is offset by the faster convergence of the Conjugate Gradient method.

Assuming that M is a constant, symmetric and positive-definite matrix, the preconditioned Conjugate Gradient method is then described in Algorithm 3 [97].

Algorithm 3: Preconditioned Conjugate Gradient method

---

1   $\mathbf{r} = \mathbf{b} - A\mathbf{x}$
2   $\mathbf{d} = M^{-1}\mathbf{r}$
3   $\delta_{new} = \mathbf{r}^T\mathbf{d}$, $\delta_{old} = \delta_{new}$
4   **while** $\sqrt{\delta_{new}} > tolerance$ **do**
5      $\mathbf{q} = A\mathbf{d}$
6      $\alpha = \delta_{new}/\mathbf{d}^T\mathbf{q}$
7      $\mathbf{x} = \mathbf{x} + \alpha\mathbf{d}$
8      $\mathbf{s} = M^{-1}\mathbf{r}$
9      $\delta_{old} = \delta_{new}$, $\delta_{new} = \mathbf{r}^T\mathbf{s}$
10      $\beta = \delta_{new}/\delta_{old}$
11      $\mathbf{d} = \mathbf{s} + \beta\mathbf{d}$

In this algorithm, $\mathbf{s} = M^{-1}\mathbf{r}$ corresponds to applying the preconditioner to $\mathbf{r}$. Although this step is represented in matrix form, it does not necessarily need to be applied this way. This algorithm also requires that M is a constant, symmetric and positive-definite matrix, although preconditioners that don't satisfy these requirements are possible with the use of Flexible Conjugate Gradients [98], which allow for variable and even non-symmetric preconditioners [99]. Several different preconditioners are possible, such as SSOR, incomplete Cholesky factorization [100] and multigrid methods (see Sec. 3.4).
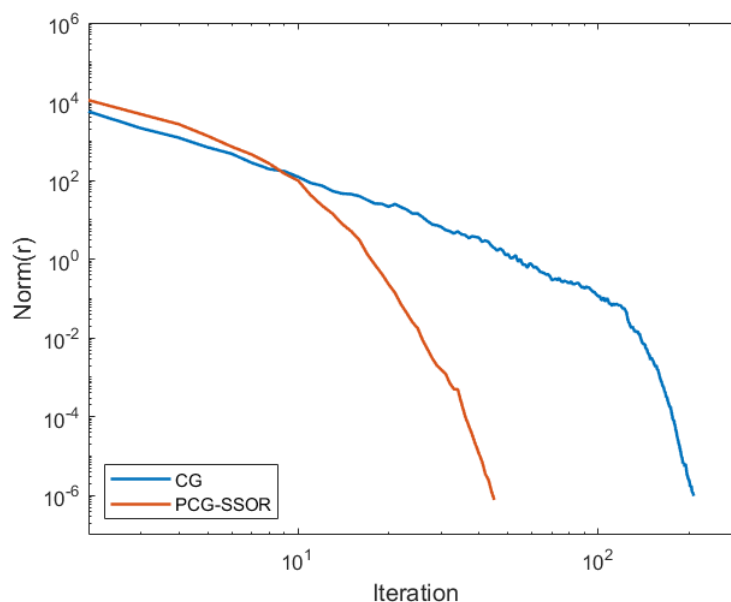
Figure 3.2 – Convergence of the unpreconditioned Conjugate Gradient method and the Conjugate Gradient method with SSOR preconditioning.

Figure 3.2 presents a comparison of the convergence speeds for the unpreconditioned

Conjugate Gradient method and the Conjugate Gradient method with SSOR preconditioning for a 2D Poisson problem, on a 64×64 grid with homogeneous Dirichlet boundary conditions and random initial conditions. Note that, for the same model problem, the unpreconditioned Conjugate Gradient converges in fewer iterations than the SOR method with optimal relaxation (Fig. 3.1), and that SSOR preconditioning provides a major increase in convergence speed. It can also be seen that, in contrast to the relaxation methods previously discussed, Conjugate Gradient methods do not lead to monotonic decreases in the norm of the residual.

3.3   DIRECT SOLUTION VIA FOURIER ANALYSIS

In the context of solving discrete elliptic equations, one of the most efficient classes of solvers available are those based on Fourier analysis, requiring $O(N \log N)$ operations to solve equations with N unknowns. For example, in [101], a solver of this class was found to be around 5 times faster than a multigrid solver, with other publications reporting even larger performance increases [55]. Since these solvers were initially developed to solve Poisson's equation, they are often referred to as Fast Poisson Solvers, although they are not limited to this class of equations, and can also be used to solve Helmholtz equations, for example. Given their extremely high efficiency in solving Poisson equations, solvers of this class are often a core part of programs designed for Direct Numerical Simulation of turbulent flows [102, 103].

These methods can be used to solve the governing equations in one of two ways: either by performing Fourier analysis on the discretized equations, as will be presented here, or by using a pseudo-spectral method, in which Fourier analysis is performed on the continuous equations (see e.g. [104] for a brief description of this method). For brevity, a deduction of the solver used in this work is presented only for the two-dimensional case, given that the extension to three spatial dimensions is straightforward. For this section, i is reserved to represent the imaginary unit.

A general linear system of equations, arising from a finite difference discretization of the 2D governing equations, can be written in stencil form as

$$a_{p,q} u_{p-1,q} + b_{p,q} u_{p,q-1} + c_{p,q} u_{p,q} + d_{p,q} u_{p+1,q} + e_{p,q} u_{p,q+1} - f_{p,q} = 0 \,, \qquad (3.16)$$

for $p = [1, P]$, $q = [1, Q]$, and the discrete inverse Fourier transform, in one direction, is defined as

$$u_p = \frac{1}{M} \sum_{m=0}^{M-1} \hat{u}_m e^{i \frac{2\pi}{M} mp} \,, \qquad (3.17)$$

with $M = P$.

It is worth noting that some Fourier transform algorithms use a slightly different definition, with m varying between 1 and M, and the m term in the exponential substituted by $m - 1$. The following deduction is also applicable to these cases, with only the small associated change in the exponential functions. By applying an inverse Fourier transform in the x direction to Eq. 3.16, and assuming that all coefficients (a, b, c, d, e) are constant in the transform direction, we get

$$\frac{1}{M} \sum_{m=0}^{M-1} \left(a_q \hat{u}_{m,q} e^{i\frac{2\pi}{M}m(p-1)} + b_q \hat{u}_{m,q-1} e^{i\frac{2\pi}{M}mp} + c_q \hat{u}_{m,q} e^{i\frac{2\pi}{M}mp} + d_q \hat{u}_{m,q} e^{i\frac{2\pi}{M}m(p+1)} \right.$$
$$\left. + e_q \hat{u}_{m,q+1} e^{i\frac{2\pi}{M}mp} - \hat{f}_{m,q} e^{i\frac{2\pi}{M}mp} \right) = 0 \,. \tag{3.18}$$

Thus, for $m = [0, M-1]$, $q = [1, Q]$, we have that

$$a_q \hat{u}_{m,q} e^{i\frac{2\pi}{M}m(p-1)} + b_q \hat{u}_{m,q-1} e^{i\frac{2\pi}{M}mp} + c_q \hat{u}_{m,q} e^{i\frac{2\pi}{M}mp} + d_q \hat{u}_{m,q} e^{i\frac{2\pi}{M}m(p+1)}$$
$$+ e_q \hat{u}_{m,q+1} e^{i\frac{2\pi}{M}mp} = \hat{f}_{m,q} e^{i\frac{2\pi}{M}mp} \,. \tag{3.19}$$

Splitting the exponential functions in the terms containing a_q and d_q, respectively, we have that

$$e^{i\frac{2\pi}{M}m(p-1)} = e^{i\frac{2\pi}{M}mp} e^{-i\frac{2\pi}{M}m} \,,$$
$$e^{i\frac{2\pi}{M}m(p+1)} = e^{i\frac{2\pi}{M}mp} e^{i\frac{2\pi}{M}m} \,, \tag{3.20}$$

and, by imposing that $a_q = d_q$, we find that

$$a_q \hat{u}_{m,q} e^{i\frac{2\pi}{M}m(p-1)} + d_q \hat{u}_{m,q} e^{i\frac{2\pi}{M}m(p-1)} = a_q \hat{u}_{m,q} e^{i\frac{2\pi}{M}mp} \left(e^{-i\frac{2\pi}{M}m} + e^{i\frac{2\pi}{M}m} \right) \,. \tag{3.21}$$

Finally, using Euler's equation and recalling that cosines and sines are even and odd functions, respectively, we have that

$$e^{-i\frac{2\pi}{M}m} + e^{i\frac{2\pi}{M}m} = \cos\left(-\frac{2\pi}{M}m\right) + i\sin\left(-\frac{2\pi}{M}m\right) + \cos\left(\frac{2\pi}{M}m\right) + i\sin\left(\frac{2\pi}{M}m\right) \,,$$
$$e^{-i\frac{2\pi}{M}m} + e^{i\frac{2\pi}{M}m} = \cos\left(\frac{2\pi}{M}m\right) - i\sin\left(\frac{2\pi}{M}m\right) + \cos\left(\frac{2\pi}{M}m\right) + i\sin\left(\frac{2\pi}{M}m\right) \,,$$
$$e^{-i\frac{2\pi}{M}m} + e^{i\frac{2\pi}{M}m} = 2\cos\left(\frac{2\pi}{M}m\right) \,.$$
$$\tag{3.22}$$

Substituting Eq. 3.22 in Eq. 3.21, we find that

$$a_q \hat{u}_{m,q} e^{i\frac{2\pi}{M}m(p-1)} + d_q \hat{u}_{m,q} e^{i\frac{2\pi}{M}m(p-1)} = 2a_q \cos\left(\frac{2\pi}{M}m\right) \hat{u}_{m,q} e^{i\frac{2\pi}{M}mp} \,, \tag{3.23}$$

and, by substituting this result in Eq. 3.19, we find that

$$e^{i\frac{2\pi}{M}mp} \left(2a_q \cos\left(\frac{2\pi}{M}m\right) \hat{u}_{m,q} + b_q \hat{u}_{m,q-1} + c_q \hat{u}_{m,q} + e_q \hat{u}_{m,q+1} \right) = \hat{f}_{m,q} e^{i\frac{2\pi}{M}mp}. \quad (3.24)$$

Thus, the original system of equations is reduced to the following set of tridiagonal systems of equations:

$$b_q \hat{u}_{m,q-1} + \left(2a_q \cos\left(\frac{2\pi}{M}m\right) + c_q \right) \hat{u}_{m,q} + e_q \hat{u}_{m,q+1} = \hat{f}_{m,q}. \quad (3.25)$$

In order to solve the above tridiagonal system of equations, one can use the following methods:

1. Apply a Fourier transform in the y direction, in a similar way to what was previously described, with similar restrictions to the coefficients. This reduces the problem to a set of diagonal systems of equations, with trivial solution.

2. Solve the system by using Thomas' algorithm for tridiagonal matrices, which is a simplified form of Gaussian elimination. This alternative is more efficient than employing an additional Fourier transform and doesn't impose such strict restrictions on the coefficients of the linear system, and is thus used in this work. Although Thomas' algorithm is not universally stable, it is so under some specific conditions, such as when the matrix associated to the linear system is symmetric positive definite [105]. All corresponding systems of equations in this work satisfy these stability requirements.

As such, the solution process of the method described above is:

1. Apply a Fourier transform in the x direction to the right-hand side of Eq. 3.16

2. Solve the M tridiagonal systems described in Eq. 3.25, in order to determine the values of $\hat{u}$

3. Apply an inverse Fourier transform in the x direction to $\hat{u}$, in order to determine the values of the original unknowns u.

When solving three-dimensional systems of equations, the above procedure can be extended simply by performing an additional Fourier transform in the z direction and performing the analogous simplifications. It is important, however, to add some remarks with regards to the implementation of such an algorithm, and to the associated boundary conditions of the solution. The Fourier transform of real values is always symmetric, and some efficient Fast Fourier Transform (FFT) algorithms take advantage of this by only outputting half of the transform (the non-redundant part), thereby saving the additional storage space that would

otherwise be required to accommodate the full complex-valued transform. This also means that, when performing an inverse Fourier transform, the output will be real if, and only if, the complex input data possesses this same symmetry. As such, when using one of these FFT algorithms, it is not necessary to unpack the data (fill-in the redundant parts of the transform). Rather, it is more efficient to use the output from the transform directly, and only solve half of the tridiagonal systems, which are those associated with the non-redundant part of the transform.

With regards to the boundary conditions of the solutions, it must be noted that the Fourier transform assumes that the data is periodic and, thus, the resulting solution of the method described above will be periodic in the directions in which a Fourier transform is applied. Boundary conditions in the final direction, associated to the tridiagonal systems, is arbitrary and determined by the coefficients of the corresponding linear systems. As such, since the linear systems associated to the velocities and pressure field are periodic in the x and z directions, with either Neumann or Dirichlet boundary conditions in the y direction, using the method described above, with Fourier transforms in the x and z directions, and solving tridiagonal systems in the y direction, is ideal. In this way, it allows for a single solver to be used for all of the corresponding systems. Since the system of equations associated with the magnetic field doesn't have constant coefficients, the Fourier analysis algorithm presented here is not applicable to it. It is worth mentioning that different boundary conditions for the solution can be obtained by using different transforms, such as sine and cosine transforms instead of the Fourier transform (see e.g. [106, 104]), and that it is possible to increase the efficiency of the solver by performing cyclic reduction before the Fourier analysis, resulting the FACR algorithm [106]. However, such an improvement was deemed unnecessary for this work, and is thus not presented.

3.4   MULTIGRID METHODS

3.4.1   General outline

When direct solution via Fourier Analysis is unfeasible, such as when variable coefficients are required, multigrid methods present themselves as one of the most efficient classes of solvers for general elliptic equations, capable of solving discrete equations with N unknowns in $O(N)$ operations. It is worth noting that, although multigrid methods scale better than Fast Poisson Solvers, they are not faster for usual values of N. For example, if we assume that, for constant C_M and C_F, a multigrid solver requires $C_M N$ operations and a Fast Poisson Solver requires $C_F N \log N$, we have that, for practical values of N, $C_M > C_F \log N$ [107].

The core idea of multigrid methods arises from observing the behaviour of traditional basic relaxation methods, such as Jacobi and Gauss-Seidel, when solving elliptic equations. Although these methods initially have fast convergence, this quickly deteriorates, leading to a very large number of iterations being required before a reasonably small residual can be achieved. Upon closer examination, one can see that this behaviour is due to the diffuse nature of these methods, which quickly reduce the high frequency components of the residual but struggle to reduce the the low frequency components. This behaviour can be seen clearly in Fig. 3.3, which presents a comparison of residuals before and after 3 iterations of the Gauss-Seidel method.

(a)   (b)
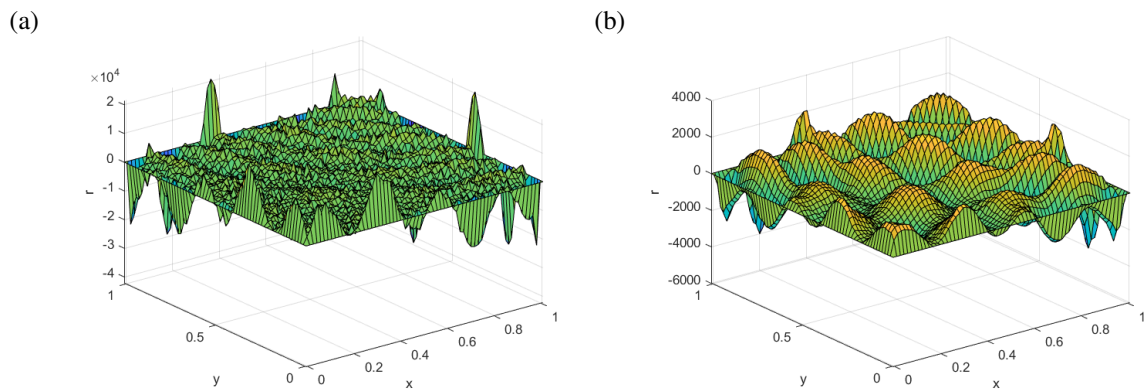
Figure 3.3 – Residuals for a 2D Poisson problem with homogeneous Dirichlet boundary conditions on a 64×64 grid. Represented are the residuals before (a) and after 3 iterations of the Gauss-Seidel method (b).

The key realization is that, as the high frequency components of the residual vanish, it becomes smooth, and, if the residual is smooth, it can be represented on a coarser grid without significant loss of information. Furthermore, components of the residual that had lower frequencies in the finer grid become of proportionately higher frequency when represented on a coarser grid, meaning that the traditional relaxation methods become more efficient in reducing them, while also having much cheaper iterations due to the reduced number of unknowns. This process can then be applied recursively, until the lowest frequency components of the residual are eliminated. Such a recursive process is the essence of multigrid methods.

It is worth noting that, on a sufficiently coarse grid, a solution of the problem can be obtained with minimal computational cost. Once a solution is obtained on the coarsest grid, this solution can then be interpolated to the finer grids, in order to obtain a final solution in the original (finest) grid, while also performing additional iterations of a relaxation method on each finer grid, in order to ensure than any higher frequency components that may have arisen are reduced.

In the context of multigrid methods, the relaxation methods (or any other method with similar characteristics) are referred to as smoothers, the representation of a residual on a coarser grid is referred to as restriction, and the interpolation of a solution to a finer grid

is referred to as prolongation. Using these terms, a general outline of a multigrid solver is then described in Algorithm 4. Note that this algorithm uses the simplest possible multigrid schedule, referred to as a V-cycle. Other schedules are possible, and are briefly discussed later in this section.

Algorithm 4: General outline of a multigrid V-cycle

---
1   **while** *solution not satisfactory* **do**
2   **while** *coarsest grid not reached* **do**
3   Perform n_1 smoothing iterations (pre-smoothing)
4   Restrict the solution to a coarser grid

5   Solve the problem on the coarsest grid
6   **while** *finest grid not reached* **do**
7   Prolongate the solution to a finer grid
8   Perform n_2 smoothing iterations (post-smoothing)

3.4.2 Specifics

When implementing a multigrid solver, several choices must be made with regards to each component of the method, including

- Which method to use as a smoother, and the number of iterations for pre-smoothing and post-smoothing

- Which restriction and prolongation operators to use

- How to define the system coefficients on the coarser grids

- How many coarse grids to use, and how to solve the problem on the coarsest one

- On what schedule the different grids are visited

This section presents a discussion regarding each of these topics, as well as detailing the fundamental aspects of the method. The discussions presented here are, however, limited to the implementations used in this work. More advanced aspects of multigrid, such as it's application to anisotropic equations and the use of adaptive methods, can be found in e.g. [108]. It is also worth mentioning that everything discussed in this work is regarding geometric multigrid, i.e. multigrid methods which are based on the geometric characteristics of the problem, which is discretized on each of the different grids. In contrast to geometric multigrid, there are the algebraic multigrid (AMG) methods, which rely solely on the problem coefficients (the matrix A for the system $A\mathbf{x} = \mathbf{b}$), and thus do not rely on any geometric characteristics of the problem [109]. Thus, Algebraic Multigrid methods, are more general

and more robust, although at the cost of a reduced efficiency. Since all linear systems solved in this work are defined on structured Cartesian grids, with solvers designed specifically for a given problem, the use of geometric multigrid is more appropriate.

With regards to the choice of smoothers, the most common choices are basic relaxation methods, although the use of more complex methods is possible, such as incomplete LU decomposition [108]. Since the smoothers do not solve the problem directly, the choice of smoother should not be based on their convergence properties, but rather on their smoothing properties, i.e. how much they smooth the residual, rather than how much they reduce the residual. Some insight regarding the smoothing characteristics of a given method can be gained by performing a smoothing analysis, such as via local mode analysis [110]. The most common smoother alternatives are damped Jacobi, lexicographic Gauss-Seidel, and red-black Gauss-Seidel. Lexicographic corresponds to the traditional Gauss-Seidel method, in which operations are performed on points adjacent to each other, following a given direction. Red-black, on the other hand, corresponds to a Gauss-Seidel method in which operations are performed in a checkerboard pattern, first for all points corresponding to cells of one color, and then for all points corresponding to cells of the other color. With this ordering of operations, red-black Gauss-Seidel becomes well suited for parallelization, since operations on points of the same color are order-independent, whereas the lexicographic Gauss-Seidel is not well suited for parallelization, since the only order-independent operations are located on a single diagonal line [108]. This altered order of operations, however, can hinder compiler optimizations such as cache prefetching, a topic that is further discussed in section 3.5. Damped Jacobi is trivially parallelizable, since all operations are order-independent, although the Gauss-Seidel alternatives offer better smoothing characteristics.

In two spacial dimensions, red-black Gauss-Seidel also displays significantly better smoothing properties when compared to the lexicographic version, in addition to being better suited for parallelization. This large difference in smoothing is not present in two spacial dimensions, however, although it is mentioned in [108] that red-black Gauss-Seidel with over-relaxation can lead to significant improvements in smoothing characteristics for 3D problems. With regards to the number of pre- and post-smoothing iterations, the most straight-forward approach is simply to test different parameters and choose the ones that result in the best performance. It is also possible to perform additional local smoothing on known problematic areas, improving the smoothness of the residuals while avoiding the overhead of performing additional smoothing iterations across the entire computational domain.

When defining the prolongation and restriction operators, special attention must be given to the type of grid used in the discretization. With the domain divided in cells, the discretization can be either vertex-centered, with the variables defined on the edges of the cells, or cell-centered, with the variables defined at the center of the cells. Since the coarser grids must use the same type of discretization as the finer ones, the inter-grid operators must be

defined accordingly [111]. Note that, with the staggered grids used in this work, the velocity components are vertex-centered in one direction and cell-centered on the others, and so the inter-grid operators must be specifically defined for each one of them. The pressure and the magnetic potential, meanwhile, use the standard cell-centered discretization.
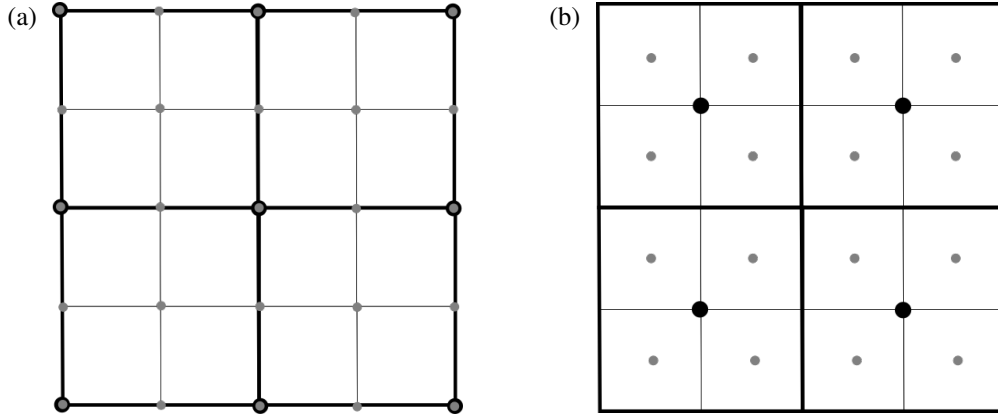
Figure 3.4 – Grid point location on fine and coarse grids for vertex-centered (a) and cell-centered (b) discretizations. Grey circles represent grid point locations on the fine grid, while black circles represent grid point locations on the coarse grid.

Figure 3.4 presents the grid point locations for the fine and coarse grids for vertex-centered and cell-centered discretizations. For vertex-centered grids, the description of the inter-grid operators is simpler, since coarse grid points coincide with fine grid points, with only one fine grid point situated between coarse grid points in each direction. For cell-centered grids, meanwhile, the coarse grid points don't align with any fine grid points, and there are two fine grid points situated between coarse grid points in each direction.

For the restriction operator, the simplest possible choice is to use a piecewise constant restriction, whereby vertex-centered coarse grid points are simply assigned the value of the fine grid point in the same location, and cell-centered coarse grid points are assigned the average value of the neighbouring fine grid points. This simple operator, however, is only first order accurate. Higher-orders of accuracy can be obtained by defining the restriction operator as the adjoint of a higher-order prolongation operator, such as a linear interpolation, which is of second order accuracy. Defining the restriction operator as the adjoint of the prolongation operator, the influence that each fine grid point has on a given coarse grid point during restriction is directly proportional to the influence that this coarse grid point has on the respective fine grid points during prolongation. For example, in the two-dimensional, vertex-centered case, the influence that a given coarse grid point has on the neighbouring fine grid points on a bi-linear prolongation is represented by the stencil

$$
P_{2h}^h = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_{2h}^h ,
\tag{3.26}
$$

43

with the central value 4 corresponding to the fine grid point with the same location of the coarse grid point. Defining a restriction operator as the adjoint of the aforementioned prolongation, it's stencil representation is then

$$R_h^{2h} = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_h^{2h} .$$ (3.27)

In the case of a cell-centered discretization, the bi-linear prolongation operator is

$$P_{2h}^h = \frac{1}{16} \begin{bmatrix} 1 & 3 & & 3 & 1 \\ 3 & 9 & & 9 & 3 \\ & & * & & \\ 3 & 9 & & 9 & 3 \\ 1 & 3 & & 3 & 1 \end{bmatrix}_{2h}^h ,$$ (3.28)

with the corresponding adjoint restriction operator defined as

$$R_h^{2h} = \frac{1}{64} \begin{bmatrix} 1 & 3 & & 3 & 1 \\ 3 & 9 & & 9 & 3 \\ & & * & & \\ 3 & 9 & & 9 & 3 \\ 1 & 3 & & 3 & 1 \end{bmatrix}_h^{2h} ,$$ (3.29)

with the $*$ representing the location of the coarse grid point.

These inter-grid operators can also be used to determine the coarse-grid coefficient matrix A^{2h}, using the Galerkin coarse grid operator $A^{2h} = R_h^{2h} A^h P_{2h}^h$. This, however, leads to an increased stencil for A^{2h} and, in turn, to reduced performance. As discussed in [112], this problem is even more pronounced in the case of cell-centered discretizations, although it can be alleviated with the use of inter-grid operators with a reduced stencil, such as the one presented in [113]. A more efficient alternative is to discretize the governing equations on the coarser grids, in the same way that was performed on the finest grid. Although this requires code adaptations for each specific implementation, it preserves the stencil sizes of A, resulting in a more efficient algorithm.

In order to solve the problem on coarser grids, multigrid methods rely on the residual equation. With the residual defined as $\mathbf{r} = \mathbf{b} - A\mathbf{x}$, it can be shown that

$$A\mathbf{e} = \mathbf{r} ,$$ (3.30)

using the error vector $\mathbf{e} = \mathbf{x}^* - \mathbf{x}$, corresponding to the error between an approximate

solution $\mathbf{x}$ and the exact solution $\mathbf{x}^*$. Referred to as the residual equation, it is clear that Eq. 3.30 does not directly help with the solution of the problem, since solving it would still require inverting the matrix A. This equation does, however, provide a basis for the multigrid algorithm. The basic, two-grid version of this algorithm is presented in Algorithm 5 [110].

Algorithm 5: Basic two-grid cycle

1 Relax $A\mathbf{x} = \mathbf{b}$ on the fine grid with the pre-smoother
2 Calculate the residual $\mathbf{r} = \mathbf{b} - A\mathbf{x}$
3 Restrict the residual to the coarse grid
4 Solve the residual equation $A^{2h}\mathbf{e}^{2h} = \mathbf{r}^{2h}$ on the coarse grid
5 Prolongate the error to the fine grid
6 Update the fine grid solution as $\mathbf{x} \leftarrow \mathbf{x} + \mathbf{e}$

To extend the two-grid cycle for an arbitrary number of grids, one can simply recursively use this two-grid algorithm when solving $A^{2h}\mathbf{e}^{2h} = \mathbf{r}^{2h}$. When the coarsest grid is reached, the residual equation is then solved exactly (or nearly exactly), using an arbitrary solver. If a sufficiently coarse grid is reached, the smoothers themselves can be used to solve the residual equation, due to how computationally inexpensive each iteration becomes. The optimal number of grids is dependent on a wide variety of factors, including characteristics of both the problem and the solver. Thus, the most straightforward way to find an optimal value is to test several different alternatives and choose the one with the best performance. Note that the maximum number of grids allowed is determined by the grid size of the finest grid, with multiples of large powers of two being ideal.

The aforementioned recursive application of the two-grid scheme results in a multigrid algorithm that only visits each grid once during the coarsening and refining parts of the algorithm. If plotted with respect to time, with the coarsest grid on the bottom, this schedule would result in a V shape. Thus, it is referred to as the V-cycle, and is the simplest possible multigrid schedule. More complex schedules exist, and may lead to improved performance. The W-cycle revisits the different grids several times during both the coarsening and refining parts of the algorithm, while the F-cycle is a hybrid, using the V-cycle schedule during the coarsening part of the algorithm and the W-cycle schedule during the refining part. The interested reader is referred to [108], Section 2.4, for a more detailed description of the three different cycles.

Another important aspect of multigrid methods is the accurate treatment of boundary conditions, both with regards to how they affect the restriction and prolongation operators, and with regards to how the boundary conditions are transported to the coarse grid problems. In order to accurately define the restriction and prolongation operators on the domain boundaries, these operators must be modified in accordance to the boundary conditions imposed on the given boundaries [108]. This can be performed in a straightforward fashion with the use of ghost grid cells located outside the domain. If values assigned to the ghost cells reflect the

imposed boundary conditions, the same restriction and prolongation operators used inside the domain can be accurately used on the domain boundaries, with the ghost cells included as part of the stencil.

Boundary conditions on the coarse grids are of the same type of the corresponding fine grid boundary conditions, but homogeneous since they are applied to the residuals, and not to the solution itself. The reasoning for this is clearer in the case of Dirichlet boundary conditions, given that the residual of a given grid point is equal to zero if the value of this grid point is the exact solution of the problem, and Dirichlet boundary conditions impose this on the boundary. An analogous reasoning is also used in the case of Neumann boundary conditions. Thus, inhomogeneous Dirichlet boundary conditions become homogeneous Dirichlet boundary conditions on the coarse grids, and inhomogeneous Neumann boundary conditions become homogeneous Neumann boundary conditions on the coarse grids. Periodic boundary conditions remain periodic boundary conditions across all grids.

Special attention must be given in the case of singular problems. Singular problems arise if some partial differential equations, such as Poisson's equation, are only subjected to Neumann and/or periodic boundary conditions. If a given problem of the type $A\mathbf{x} = \mathbf{b}$ is singular, it does not have a single, exact solution. It may not have a solution at all, or, if $\sum \mathbf{b} = 0$, it has infinitely many solutions, which can only be determined up to an arbitrary constant. In order to ensure that a solution exists, one alternative is to solve the approximate problem [108]

$$A\mathbf{x} = \tilde{\mathbf{b}}\,, \tag{3.31}$$

$$\tilde{\mathbf{b}} = \mathbf{b} - \frac{1}{N}\sum \mathbf{b}\,, \tag{3.32}$$

where N is the number of grid points in the domain. Since N is inversely proportional to the grid spacing h, with $\frac{1}{N} \propto h^2$ and $\frac{1}{N} \propto h^3$ in two and three spacial dimensions, respectively, the solution of this approximate problem will be a solution to the original problem up to discretization error, if a second-order discretization is used.

3.4.3 Conjugate Gradient acceleration of Multigrid

One powerful class of solvers arises from the combination of multigrid and Krylov subspace methods, referred to either as Krylov subspace acceleration of multigrid or as multigrid preconditioning for a Krylov subspace method. Although such methods do not lead to increased performance if an optimal multigrid algorithm is used, they do lead to a significantly improved robustness [108]. Thus, such methods are especially well-suited for the solution of complex problems for which the development of optimal multigrid solvers would be difficult, since Krylov subspace acceleration allows for simple multigrid algorithms to achieve

similar levels of performance. The reasoning for this is that, in such complex cases, specific error components arise that are not reduced by either the smoothing or the coarse grid correction of a simple multigrid algorithm, leading to substantial reductions in performance or even an outright failure to converge to a solution. Since these specific components will be the largest ones after a multigrid cycle, they are then the first ones targeted by methods such as Conjugate Gradients. Thus, by combining the two methods, their characteristics are leveraged in a way that they compliment each other, resulting in robust, high-performance solvers.

Since, in this case, multigrid methods are used as preconditioners, they must fulfill the preconditioner requirements for the specific method they are used alongside of. In the case of the Conjugate Gradient method, these requirements are that the preconditioner matrix must be constant, symmetric and positive-definite, as discussed in section 3.2.2. Note that, in this case, the preconditioning step $\mathbf{s} = M^{-1}\mathbf{r}$ in Algorithm 3 consists in performing one multigrid iteration on the system of equations $A\mathbf{s} = \mathbf{r}$, with zero initial guess and homogeneous boundary conditions [114]. The necessary requirements that a multigrid cycle must satisfy in order to be a valid preconditioner for the Conjugate Gradient method were first described in [115]. A sufficient, but not exhaustive set of conditions is that [116]:

- The restriction and prolongation operators are the adjoint of one another

- The multigrid schedule is symmetric, i.e. either a V-cycle or a W-cycle

- The pre- and post-smoothers use the same method and number of iterations, and are performed in opposite directions. For the lexicographic Gauss-Seidel method, for example, this means that if the pre-smoother is applied in the forward direction, the post-smoother must be applied in the backward direction. Jacobi methods are unaffected by this condition, since they are order-independent.

- The coarse grid solution must either be exact, or be obtained by a method that approximates the inverse of the coefficient matrix A, on the coarsest grid, as a symmetric and definite matrix.

The multigrid solver used in this work uses Conjugate Gradient acceleration, and is based on the algorithm presented in [116], using a V-cycle multigrid preconditioner. Tests were performed with different iteration numbers for damped Jacobi and lexicographic Gauss-Seidel for the pre- and post-smoothers, with a single iteration of the Gauss-Seidel method resulting in the best performance. The coarse grid solution is obtained using the unpreconditioned Conjugate Gradient method, which is an efficient solver for the smaller coarse grid problems and results in the solver achieving peak performance with a reduced number of coarse grids (3, for grid sizes in the order of $128 \times 128 \times 128$). The prolongation operator is a trilinear interpolation, with the restriction operator defined as it's adjoint. In case the solver is used

for a singular problem, the preconditioning step is changed from $\mathbf{s} = M^{-1}\mathbf{r}$ to $\mathbf{s} = M^{-1}\tilde{\mathbf{r}}$, in accordance to Eqs. 3.31 and 3.32.

Due to the far simpler boundaries used in this work in comparison to the use cases presented in [116], the boundary smoothing proposed in the aforementioned work did not result in any improvements in convergence. However, a similar reasoning behind the procedure was employed in the form of interface smoothing, whereby additional smoothing iterations are performed on grid points neighbouring the droplet's interface, obtained from the Local Level Set scheme presented in section 2.2.5. A slight improvement in performance was observed with interface smoothing consisting of a single Gauss-Seidel iteration, once after the main pre-smoother and once before the main post-smoother, and in opposite directions, in order to maintain the symmetry of the multigrid preconditioner.

3.5   IMPLEMENTATION AND OPTIMIZATION

Given the high computational cost associated with three-dimensional modelling of fluid flows, code optimization is very important, since it can provide savings in computational time that can amount to several days or weeks, and, depending on the optimization strategy, be simple to implement. This section presents a brief discussion regarding some of these strategies, but, since it is not the primary focus of this work, it is by no means extensive. Rather, it is only an introduction to the interested reader. More in-depth information is readily available in the internet, such as in the computer science literature and compiler documentations, for example.

In order to understand the reasons behind program bottlenecks, optimization strategies and, ultimately, program performance, one must first understand the fundamental concepts regarding how a modern computer works. All instructions, including data computations, are executed by the Central Processing Unit (CPU, or processor), but, since the CPU itself does not store data, it must be stored in a separate location. Unused data is placed on a storage device, typically either a Hard Disk Drive (HDD) or Solid State Drive (SSD), which possesses very high capacity, but only provides slow access to the data. When executing a program, relevant data is moved to memory (RAM), which has lower capacity, but provides much faster access to data, i.e. it has a much larger bandwidth. When executing an instruction, the CPU will read the required data from the memory, execute the instruction and then replace the corresponding data on memory with the newly computed one.

Although the data transfer speed from memory to processor is fast, it is finite, and can become a bottleneck for fast processors. In order to alleviate possible bottlenecks, modern processors introduce a cache, similar to memory but located on the CPU chip itself and with much faster access speeds, but with small capacity. The core idea to the use of cache is that,

when performing several instructions on a small set of data, it can be stored in cache, so that computations are not limited by memory speed. Some processors also possess more than one cache, described by levels. Level 1 cache is closest to the actual processing cores, and is the fastest to access, lowest capacity one. The following levels of cache will be progressively slower to access and have higher capacities. As an example, the computer in which most of the simulations presented in this work were performed (with an AMD Ryzen 7 4800h processor) possesses 32KB of Level 1 data cache per core, 512KB of Level 2 cache per core, and 8MB of Level 3 cache shared among 8 cores, with 16GB of RAM. For reference, a single $128 \times 128 \times 128$ array of double-precision variables (64 bits or 8 bytes each) requires 16MB of memory.

When running programs that require several instructions executed on a small data set, cache implementations are extremely effective, and allow for processors to achieve peak performance without being limited by memory bandwidth. However, when using data sets larger than the cache, and performing few instructions per data point, memory bandwidth becomes a significant bottleneck, and limits program performance to a fraction of what would be possible based on processor performance alone. In such cases, great care must be taken with managing data access patterns, in order to minimize memory bandwidth bottlenecks and maximize code performance, and a failure to do so can result in a program that is several times slower than it potentially could have been.

In order to determine whether a given program is compute-bound (limited by processor performance) or memory-bound (limited by memory bandwidth), one can use a Roofline Model [117]. Figure 3.5 presents a Roofline Model of the program developed in this work, obtained using the Intel® Advisor profiling tool. The horizontal axis is the ratio between number of compute operations and memory access required, and the vertical axis is the number of compute operations per second. The diagonal line that bounds the left side of the colored region represents memory bandwidth, and the horizontal line that bounds the right side of the colored region represents maximum compute performance of the processor. Colored circles represent the more time-consuming loops of the program, with the red ones the most time consuming (where optimization efforts should be focused) and the green ones the least time-consuming. The black cross represents the program as a whole. Loops in the region bounded by the diagonal line are memory-bound, while loops in the region bounded by the horizontal line are compute-bound, and it is clear that the program is fundamentally memory-bound. Note that, due to cache effects, some loops are able to achieve performances that exceed the memory bandwidth limit.

Since program performance is limited by memory bandwidth rather than computational power, optimization efforts must be directed towards improving memory usage, rather than reducing operation count or parallelization, especially in the case of shared-memory parallelization, i.e. when different processor cores share the same memory. Note that reductions
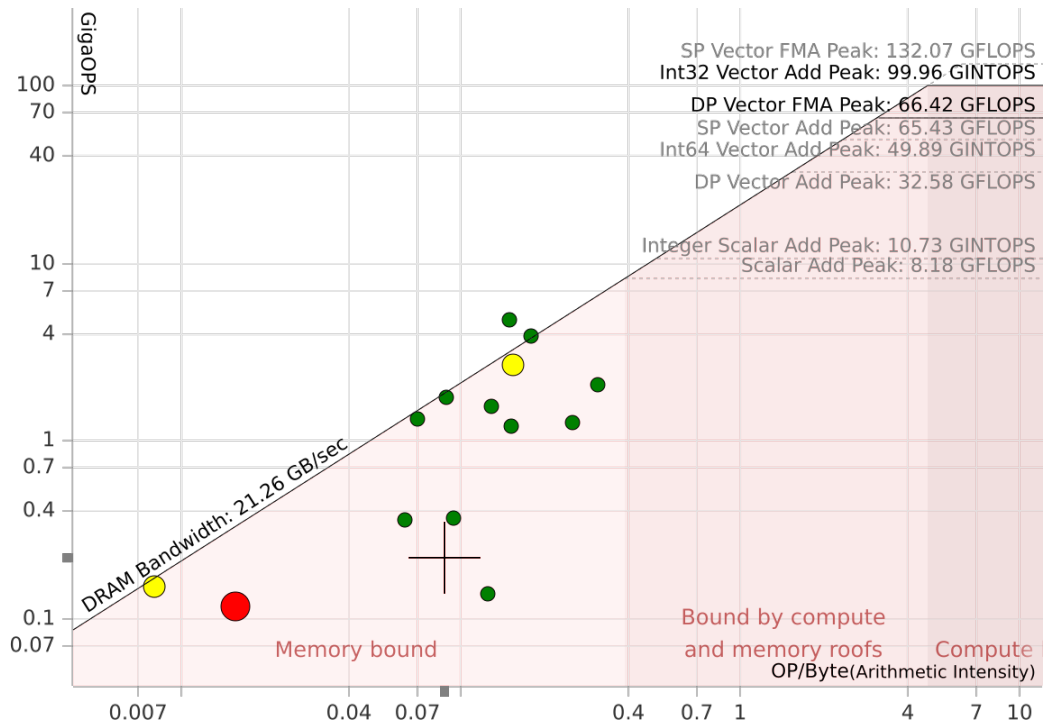
Figure 3.5 – Roofline model of the program developed for this work.

in operation count could lead to improved performance if they also result in fewer memory accesses. Distributed memory parallelization could be profitable, given that it essentially increases memory bandwidth by using different memory systems in parallel, as well as GPU (Graphics Processing Unit) offloading, since GPU memory usually has much higher bandwidth than standard system memory (RAM). However, neither of these options were feasible due to hardware availability and time constraints.

The most basic optimization of memory usage is to ensure that, whenever cache optimization strategies don't dictate otherwise, data is accessed, as much as possible, in unit strides, i.e. moving between adjacent points in an array, since it allows for faster fetching of data from memory. One must keep in mind, however, that multidimensional arrays are stored linearly in memory, and that different programming languages have different rules regarding the order in which such arrays are stored. In Fortran, arrays are stored in column-major order, with the first index being contiguous in memory. For example, in a $N_i \times N_j \times N_k$ array `u`, `u(2,1,1)` is a single stride away from `u(1,1,1)`, while `u(1,2,1)` is N_i strides away from `u(1,1,1)`, and `u(1,1,2)` is $N_i \times N_j$ strides away from `u(1,1,1)`. In C, on the other hand, arrays are stored in row-major order, so `u(1,1,2)` is adjacent to `u(1,1,1)`.

One further way to optimize memory access patterns is through cache blocking, in which loops are rearranged so as to access data in blocks small enough to fit in cache and performing all operations required for such block while it remains in cache, exploiting the benefits of data locality and reducing RAM access requirements. The more traditional form of cache blocking is the spatial one, in which blocking is performed during a single loop through

data. However, due to the very small amount of operations per loop required for each data point, spatial cache blocking is not effective for stencil operations [118], and can even decrease program performance due to increased loop overheads and hindering other compiler (or hardware) optimizations, such as prefetching, which moves data from memory to cache before the processor actually requires it, in order to avoid cache misses (when data the processor needs is not present in the cache).

A more modern approach to cache blocking, which does work well for stencil operations, is temporal blocking. It consists in attempting to reuse previously cached data not only during a single loop through data, but across multiple loops. For example, in the case of Jacobi or Gauss-Seidel iterations, this strategy performs multiple iterations over a single block of points, before moving over to a different block. Although such an approach may seem counter-intuitive at first, since updating one component to the instant $n+2$ requires that other components in the array have already been updated to the instant $n + 1$, upon closer examination it is clear that this requirement does not apply to the entire array, but only to the neighbours of this given component. This, in turn, makes it possible that, after updating a block to the instant $n + 1$, most of the components inside the block can already be updated to the instant $n + 2$, with the exception of some components located in the boundaries, while they are still stored in the cache. Such a process can also be applied recursively across several iterations, and has been shown to be efficient for stencil operations [119, 120]. However, since none of the stencil operations performed in the program developed in this work are repeated, such an approach is not applicable to it. Additionally, peak performance requires tailoring the algorithm to a specific processor architecture, and poor implementations can even cause a reduction in overall code performance [121].

Outside of memory access pattern optimization, there are also other optimization strategies which can be beneficial, such as, for example, function inlining and loop unrolling, in which functions and loops are written explicitly, rather than using calls, in order to save overheads. Note that both of these optimizations may be implemented automatically by the compiler without any changes to the source code, depending on the compiler settings. If the code is well suited for SIMD (single instruction, multiple data) operations, vectorization may provide major performance improvements, if the processor supports it. Vectorization allows for instructions to be performed on multiple elements of an array concurrently, rather than performing them on one element at a time, and may be performed automatically by the compiler, if it is informed that the target processor supports it. Support for vector operations is included on instruction sets available on most modern processors, such as SSE (Streaming SIMD Extensions) and AVX (Advanced Vector Extensions). Finally, further performance improvements may be obtained by eliminating unnecessary operations, such as some array reinitializations, for example. Another example of unnecessary operations is in the solution of the linear system for the magnetic field potential. Although the system's coefficients are

not constant in space, an so would, in principle, need to be stored as arrays covering the entire domain, they are constant (and equal, in the case of equally-spaced meshes) throughout most of the domain, outside of the droplet interface, and in these cases it is unnecessary to fetch individual coefficients from memory.

3.6 PERFORMANCE COMPARISON

This section presents a comparison between the performance of some of the methods previously discussed, namely the Conjugate Gradient method with SSOR preconditioning (PCG-SSOR), the Conjugate Gradient method with multigrid preconditioning (PCG-MG), and direct solution via Fourier analysis (Fast Poisson Solver, FPS-FFT). This is performed based on solution times for the pressure Poisson equation (Eq. 2.10), which is the most costly of the linear systems solved in this work.

Experiments were performed for a test case using a $10 \times 10 \times 10$ domain, with $Re = 0.01$, $Ca = 0.15$, $\lambda = 1$, $Ca_{mag} = 8$, $\zeta = 2$ and $\Delta t = 0.002$. Results are for a Fortran code compiled with the Intel Fortran Compiler, running on a single thread on an AMD Ryzen 4800h processor, equipped with dual-channel, 3200MHz DDR4 RAM. FFTs are performed using Intel's Math Kernel Libraray (MKL).

Grid size	$64 \times 64 \times 64$	$96 \times 96 \times 96$	$128 \times 128 \times 128$	$160 \times 160 \times 160$
PCG-SSOR	0.65s	2.87s	7.94s	17.5s
PCG-MG	0.22s	0.76s	1.75s	3.54s
FPS-FFT	0.0156s	0.0312s	0.0625s	0.115s

Table 3.1 – Solution times for the pressure Poisson equation, for different solvers and grid sizes.

The results of these experiments are presented in Table 3.1, with solution times relative to the first program iteration, averaged across 5 different runs. It can be seen that the Conjugate Gradient solver with SSOR preconditioning is significantly slower than the one with multigrid preconditioning, and that it does not scale well with increasing numbers of grid points. The solution times for the PCG-MG solver, meanwhile, scale linearly with the number of grid points, indicating that the solver is, indeed, converging in $O(N)$ operations, which is the optimal convergence rate. Despite the efficiency of the PCG-MG solver, however, it's solution times are still a full order of magnitude slower than the Fast Poisson Solver. It also appears to have scaling superior to the optimal $O(N)$, whereas the method is expected to solve the problem in $O(N \log N)$ operations. This, however, is likely due to optimizations in the FFT, performed with an external library, or even to inaccuracies in the CPU time measurements, given how quickly solutions are obtained.

Since the FPS-FFT is the only solver that benefits from external libraries, with the two other solvers entirely self-written, it is very likely that part of it's performance advantage is due to additional optimizations. However, due to how significant this performance advantage is, it is clear that it is not only due to further optimizations, but also due to the algorithm's exceptional efficiency. Thus, the Fast Poisson Solver should be the preferred choice whenever it is applicable, especially given that it is a far simpler algorithm than a multigrid solver, for example. In cases where the Fast Poisson Solver is not applicable, such as in the solution of the magnetic potential equation, the PCG-MG solver is preferable.

4 DROPLET DYNAMICS, RHEOLOGY AND MAGNETIZATION

This chapter presents the results of the numerical simulations performed. Unless otherwise specified, all results are for $Re = 0.01$, $\lambda = 1$ and $\zeta = 2$, and simulations are performed on a $10 \times 10 \times 7.5$ domain, discretized with a $128 \times 128 \times 96$ grid. This results in a dispersed phase volume fraction of $\beta \approx 0.56\%$, well within the dilute regime where interactions between droplets are negligible. In order to ensure mesh convergence and that confinement effects are negligible, tests were performed for a simple shear flow with an external magnetic field in the main velocity gradient direction, $Ca = 0.15$ and $Ca_{mag} = 12$, using a finer mesh and maintaining domain size, and increasing domain size and maintaining mesh density. Maintaining domain size and comparing the results to those with a $192 \times 192 \times 144$ mesh, relative errors for inclination angle, droplet deformation and dispersed phase shear stress $\sigma_{\mathbf{yx}}^{\mathbf{d}}$ were of 3% or less. Maintaining mesh density and comparing the results to those with a $15 \times 15 \times 11.25$ domain, relative errors for inclination angle, droplet deformation and normalized dispersed phase shear stress $\sigma_{\mathbf{yx}}^{\mathbf{d}}/\beta$ were less than 1%. Considering that this is a case of very high droplet deformation, mesh convergence can be considered as very good, and confinement effects can be assumed to be negligible. Effects of Δt were found to be negligible provided that it is small enough to ensure stability, indicating that the solution is dominated by spacial discretization errors. Changes in droplet volume, which can arise from numerical errors in the Level Set formulation, were monitored and generally kept to within 3%, although increases in droplet volume of up to 8% were observed for oscillatory shear flows in the limit of low oscillation frequency and high magnetic capillary number.

4.1 NON-MAGNETIC DROPLET IN SIMPLE SHEAR FLOWS

The simplest case studied in this work is that of a non-magnetic droplet subjected to a simple shear flow. In the limit of $Re \to 0$ and in the absence of confinement effects, the flow regime is then governed by the capillary number and the viscosity ratio between the dispersed and continuous phases. The droplet's geometry can be defined by it's deformation parameter $D = \frac{L-B}{L+B}$ and it's inclination angle θ, as described in Fig. 1.1. Here, both the droplet's deformation and inclination angle are determined based on the droplet's inertia tensor. The inclination angle is obtained as the angle between the droplet's longest principal axis (the one with the smallest associated moment of inertia), and the deformation is obtained by fitting the three principal moments of inertia to those of an ellipsoid. It has been shown

in [122] that the droplet's shape closely resembles an ellipsoid up until it is close to the point of breakup, and this measurement method has been previously validated in [123].
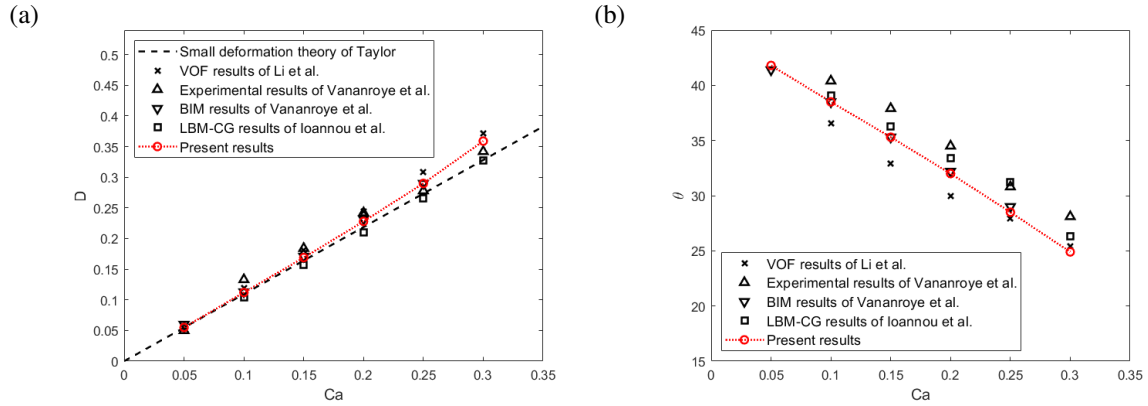

Figure 4.1 – Non-magnetic droplet deformation (a) and inclination angle, in degrees (b), as a function of capillary number. The small deformation theory of Taylor [1] in (a) corresponds to $D = (35/32)Ca$.

Figure 4.1 presents the droplet deformation and inclination angle as a function of capillary number, alongside some results available in the literature, for validation. These include the small deformation theory of Taylor [1], with $D = (35/32)Ca$, the Volume of Fluid (VOF) results of Li et al. [32], the experimental and Boundary Integral Method (BIM) results of Vananroye et al. [13], and the Lattice-Boltzmann Method with Color Gradient (LBM-CG) results of Ioannou et al. [35]. It can be seen that, although there is some significant variation between the different results, there is an agreement between the present results and the theoretical, experimental and numerical results available in the literature, indicating that the methodology presented in this work is accurate in the case of non-magnetic droplets.

With regards to the droplet geometry, it can be seen that, in the limit of $Ca \to 0$, the droplet is nearly spherical, with $D \to 0$, and has an inclination angle of $\theta \to 45°$. This elongation in a $\approx 45°$ angle is characteristic of any system subjected to a shear stress, and can also be seen in solids, for example [124]. However, as the capillary number increases and the droplet is further stretched by the shear stresses, the inclination angle decreases, as the droplet is rotated by the shear flow. Figure 4.2 presents the pressure field for the flow around a droplet with $Ca = 0.3$, where the droplet's shape can be clearly identified by the jump in pressure across the interface, caused by the surface tension stresses. Note that the droplet is significantly deformed by the shear flow, and that it's inclination angle is also significantly smaller than $45°$. It can also be seen that, due to the smeared interface used by the Level Set method, the pressure jump is not sharp, with the pressure field instead displaying abrupt but continuous changes across a thin band containing the interface (zero level set of ϕ).

In addition to the droplet's geometry, another area of interest is in characterizing the stress system of the two-phase flow, since it is the average of this stress system that governs the behaviour of the corresponding emulsions. The average stresses acting on the two-phase
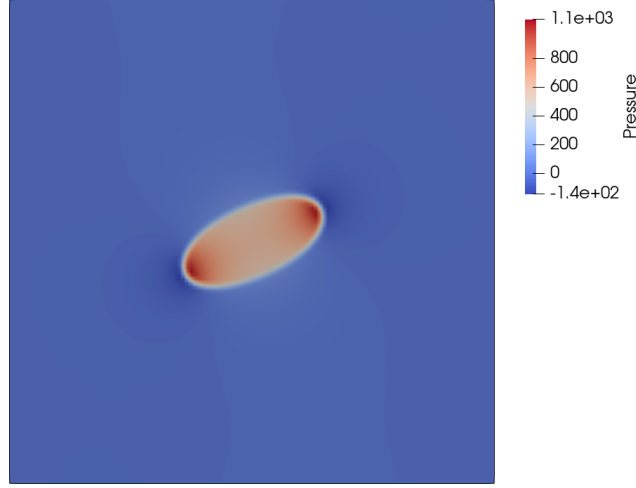
Figure 4.2 – Pressure field for a non-magnetic droplet with $Ca = 0.3$.

system can be decomposed into two parts, one associated to the viscosity and pressure field of the continuous phase, and thus Newtonian in this case, and one associated to the stresses induced by the presence of the droplet. This characterization of the stress system was first proposed in [125], with the latter, non-Newtonian component of the stresses referred to as the particle stress tensor. In the case of an emulsion of ferrofluid droplets, the particle stress tensor can be calculated by the surface integral [46]

$$\mathbf{S} = \frac{1}{V} \int_{\Gamma} \frac{1}{Ca} \left[\kappa \mathbf{x} \hat{\mathbf{n}} - \frac{Ca_{mag}}{2}(\zeta - 1)|\mathbf{H}|^2 \mathbf{x} \hat{\mathbf{n}} + (\lambda - 1)(\mathbf{u}\hat{\mathbf{n}} + \hat{\mathbf{n}}\mathbf{u}) \right] dS, \qquad (4.1)$$

where V is the total volume of the two phases and Γ is the droplet surface. In the right-hand side of Eq. 4.1, the first term corresponds to the stresses induced by surface tension, the second term corresponds to stresses induced by magnetic forces, and the third term corresponds to stresses induced by changes in viscosity between the dispersed and continuous phases. In the case of a non-magnetic droplet with the same viscosity of the continuous phase, only the first term in the right hand size is non-zero, since $\zeta - 1 = 0$ and $\lambda - 1 = 0$, and thus the only non-Newtonian stresses acting in the system are those arising from surface tension.

In the context of the rheology of emulsions, there are three quantities associated with the particle stress tensor that can be measured experimentally, and thus are the most relevant to rheological research, in the case of simple shear flows [126]. These are the shear stress $\mathbf{S}_{yx}$, the first normal stress difference $N_1 = \mathbf{S}_{xx} - \mathbf{S}_{yy}$, and the second normal stress difference $N_2 = \mathbf{S}_{yy} - \mathbf{S}_{zz}$. Note that the rheology of the emulsion is dependent on the droplet's geometry [21], and it was suggested in [127] that $N_1 = 2\cot(2\theta)\mathbf{S}_{yx}$.

Figure 4.3 presents the shear stress component of the particle stress tensor $\mathbf{S}_{yx}$ normalized by the volume fraction β, a quantity that is often referred to as the reduced viscosity of the emulsion, and Fig. 4.4 presents the first and second normal stress differences, also
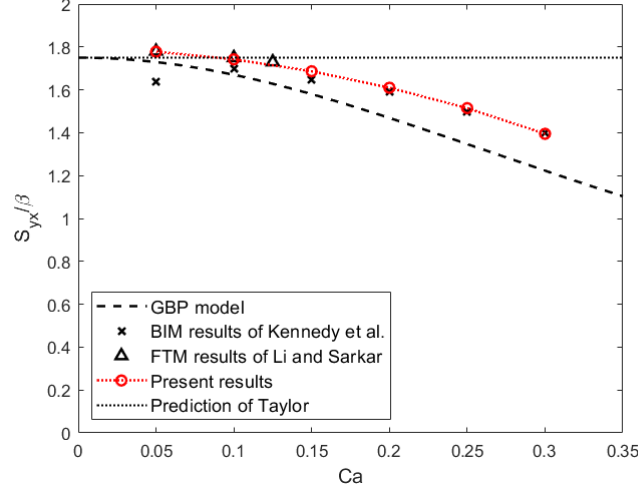
Figure 4.3 – Shear stress component of the particle stress tensor as a function of capillary number. The no-deformation prediction of Taylor [2] corresponds to $\mathbf{S}_{yx}/\beta = 7/4$.

normalized by β. The results of the Grmela-Bousmina-Palierne (GBP) model [20, 21], the BIM results of Kennedy et al. [29] and the Front-Tracking Method (FTM) results of Li and Sarkar [33] are also presented alongside the present results. Figure 4.3 also presents the no-deformation prediction of Taylor [2] that $\mathbf{S}_{yx}/\beta = 7/4$. The GBP model predicts that, for $z = (35/16)Ca$,

$$\frac{\mathbf{S}_{yx}}{\beta} = \frac{7}{4(z^2+1)}\,, \tag{4.2}$$

which converges to the no-deformation prediction of Taylor in the limit of $Ca \to 0$, and that

$$\frac{N_1}{\beta} = \frac{245Ca}{32(z^2+1)}\,, \tag{4.3}$$

$$\frac{N_2}{\beta} = -0.5\frac{N_1}{\beta}\,. \tag{4.4}$$

It can be seen in Figs.4.3 and 4.4 that, although there is some divergence between the numerical results and GBP model, there is close agreement between the present results and the BIM and FTM results, indicating that the rheological measurements presented in this work are accurate for non-magnetic droplets. Note that the low capillary number results of Kennedy et al. presented may not be accurate, due to difficulties in extracting the data points from the original figures.

In Fig. 4.3, it can be seen that the shear stress, and thus the emulsion's viscosity, is at it's maximum for $Ca \to 0$, since a nearly spherical droplet offers the most resistance to the fluid flow. However, as the capillary number increases and the droplet's morphology displays higher deformations and smaller inclination angles, the droplet offers progressively less resistance to the fluid flow, leading to reductions in the shear stress. This reduction in shear

Figure 4.4 – First (a) and second (b) normal stress differences as a function of capillary number.

stress corresponds to a decrease in the emulsion's viscosity, and leads to the characteristic shear-thinning behaviour of dilute emulsions.

With regards to the first and second normal stress differences, presented in Fig. 4.4, it can be seen that N_1 is always positive, while N_2 is negative and smaller in magnitude. For 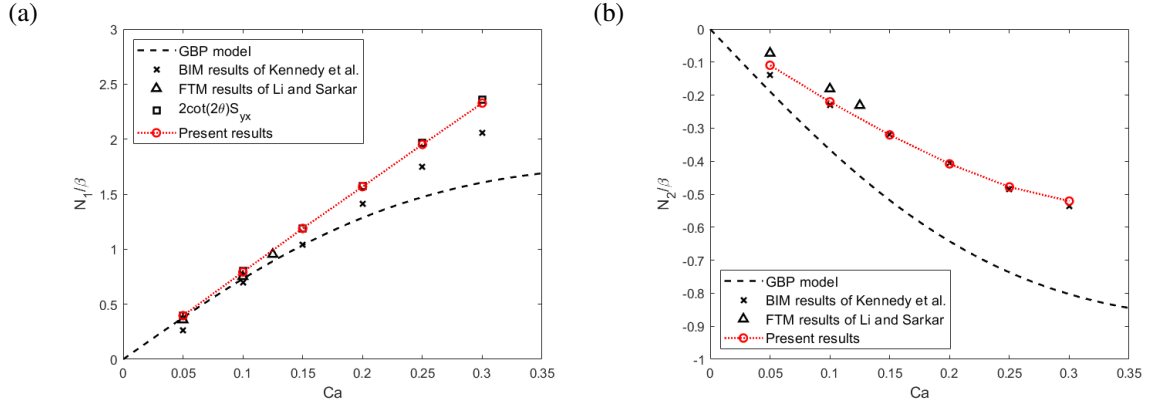the capillary number ranges tested both normal stress differences monotonically increase in magnitude with larger values of Ca, and it can be seen that the values of N_1 calculated with the geometric relationship presented in [127] are in excellent agreement with the values of N_1 obtained directly for the particle stress tensor. The reason for the existence of these two stress differences is the distribution of capillary forces across the droplet surface.

In the case of nearly spherical droplets, the capillary forces are mostly uniform, and thus $N_1, N_2 \to 0$ with $Ca \to 0$. As the droplet gets deformed, however, it's surface assumes larger curvatures on the droplet tips, resulting in larger capillary forces, and, due to the reduced inclination angles, the main component of these larger capillary forces acts in the x direction. The result is that this additional compression in the x direction, caused by the larger capillary forces at the droplet tips, causes an increase in $\mathbf{S}_{xx}$. Meanwhile, the surface curvature reaches a minimum roughly in regions normal to the y direction, while regions normal to the z direction have intermediate values of curvature. Thus, the droplet's deformation and rotation towards the flow direction creates an anisotropy of stresses, with $\mathbf{S}_{xx} > \mathbf{S}_{zz} > \mathbf{S}_{yy}$, leading to positive and negative values of N_1 and N_2, respectively. In loose terms, these values of N_1 and N_2, characteristic of dilute emulsions, indicate that the shear flow generates an additional compression in the y direction, such that a normal force must be exerted on the upper and lower walls in order to maintain the shear flow [126].

In contrast to the rheological measurements presented here, characteristic of dilute emulsions, Newtonian fluids display a constant viscosity and no normal stress differences. Since in the cases presented in this section both fluids are Newtonian and of equal viscosities, all the non-Newtonian effects described are caused solely by the presence of surface tension in

the droplet interface.

4.2 NON-MAGNETIC DROPLET IN OSCILLATORY SHEAR FLOWS

In addition to the steady-state simple shear flow analyzed previously, another important experiment in the field of rheology is the case of oscillatory shear flows. In this case, rather than subjecting the fluid to a constant shear rate $\dot{\gamma}_0$, the fluid is subjected to an oscillatory shear rate $\dot{\gamma} = \dot{\gamma}_0 \cos(\omega t)$. Since the behaviour of non-Newtonian fluids is often not dependent solely on the instantaneous shear rate, but also on other parameters, such as the shear strain γ, the stress system in the fluid may not be perfectly in phase with the imposed shear rate. Thus, the shear response of the system is characterized by a storage modulus G', in phase with the shear strain γ, and a loss modulus G'', in phase with the shear rate $\dot{\gamma}$, rather than by a single viscosity value [126]. Since the storage modulus is associated to stresses that are in phase with the strain, it corresponds to an elastic behaviour of the fluid, while the loss modulus corresponds to a viscous behaviour, resulting in a viscoelastic behaviour of the fluid.

In the case of emulsions, since the non-Newtonian behaviour is strongly associated to the surface tension forces acting on the droplet surface, the stresses acting in the system are dependent on the droplet's deformation. Since the surface tension forces are always attempting to return the droplet to a spherical shape, it is clear that shear stresses arising from surface tension will be at their peak when the droplet is in it's most deformed state. Since the droplet deformation is not only dependent on the instantaneous shear rate, but also on the imposed strain, the viscoelastic behaviour of the emulsion is associated with the phase angle between the droplet deformation and the imposed shear rate. Thus, disregarding the constant loss modulus associated with the viscosity of the Newtonian continuous phase, the stress response of the emulsion can be described as [126]

$$\mathbf{S}_{yx} = G'_u \gamma_0 \sin(\omega t) + G''_u \gamma_0 \cos(\omega t) \,, \tag{4.5}$$

where $\gamma_0 = \dot{\gamma}_0/\omega$ is the strain amplitude, and G'_u and G''_u are the storage and loss moduli nondimensionalized using the shear rate characteristic time. Note that Eq. 4.5 is only valid in the case of small amplitude oscillations, where the system response is linear and higher order harmonics are not present. These nonlinear behaviours can be studied with the use of large amplitude oscillatory shear (LAOS) and more complex material functions [128, 129], although such an analysis is outside the scope of this work.

In the analysis of the oscillatory response of an emulsion, great care must be taken with regards to the characteristic time scales used in the nondimensionalization. This is especially

true in the present work, since the solution methodology is implemented using the shear rate characteristic time $\tau_u = 1/\dot{\gamma}_0$, while results for the oscillatory behaviour of emulsions are usually based on the droplet interface characteristic relaxation time $\tau_\sigma = \eta a/\sigma$. In this case, the two characteristic times are related by the capillary number, with $\tau_\sigma = Ca\tau_u$, and the particle stress based on the interface relaxation time can be calculated as $\mathbf{S}^\sigma = Ca\mathbf{S}$.

For the results presented in this section, the shear amplitude is fixed as $\gamma_0 = 0.2$, which is achieved by fixing the shear rate-based angular velocity $\omega = 5$. In order to obtain the desired oscillation angular velocity normalized by the interface relaxation time ω_σ, the capillary number is set according to $Ca = \omega_\sigma/\omega$. The storage and loss moduli normalized by the interface relaxation time can then be obtained from Eq. 4.5, using $G' = CaG'_u$ and $G'' = CaG''_u$. In the case of oscillatory shear flows, test cases include a wide range of values for Ca, and thus it is necessary to also vary the values of the Reynolds number, to ensure that it is sufficiently small both to ensure that viscous dissipation maintains a constant shear rate across the emulsion, and to ensure that all terms in the right-hand side of Eq. 1.4, inversely proportional to $ReCa$, are much larger than those on the left-hand side, so that inertial effects are negligible. Thus, the Reynolds number is set as $Re = \min(10^{-3}, 10^{-4}/Ca)$.

(a) (b)
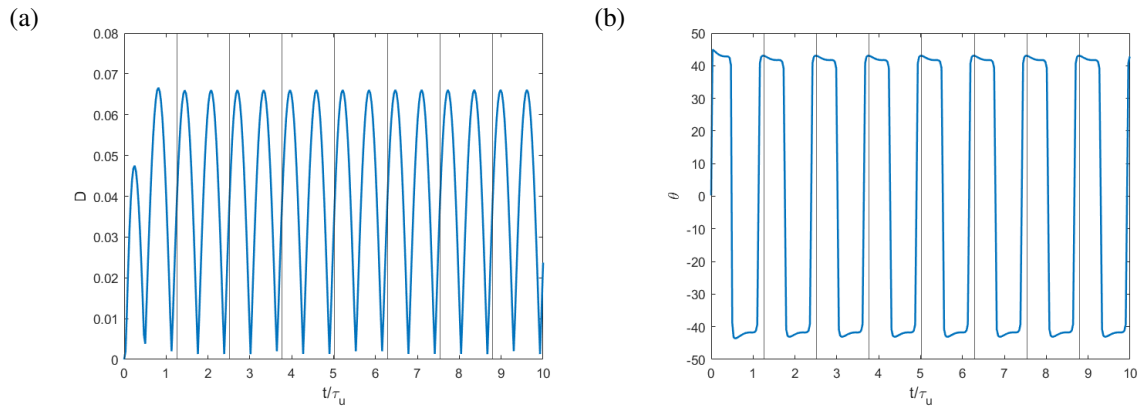
Figure 4.5 – Droplet deformation (a) and inclination angle, in degrees (b), for a droplet under oscillatory shear flow. Vertical lines separate the periods of the shear rate. $\gamma_0 = 0.2$, $\omega_\sigma = 0.4$.

Figure 4.5 presents the droplet deformation and inclination as a function of time in the case of $\omega_\sigma = 0.4$. It can seen that, after an initial start-up period, the droplet deformation follows the norm of a sinusoid, since it cannot assume negative values. Thus, a droplet with near zero deformation will then be deformed by the shear flow moving in one direction, reach a maximum, and then return to a nearly spherical shape before being deformed in the opposite direction as the shear rate reverses direction. The inclination angle, meanwhile, reaches it's peak as soon as the droplet starts to be deformed, then gradually decreases as the droplet is further deformed and rotated by the shear flow, before changing sign as the droplet deformation reverses orientation. Note that the periods of the shear rate are represented in the figures by the vertical lines, with both the droplet deformation and inclination angle being

60

out of phase with the shear rate.
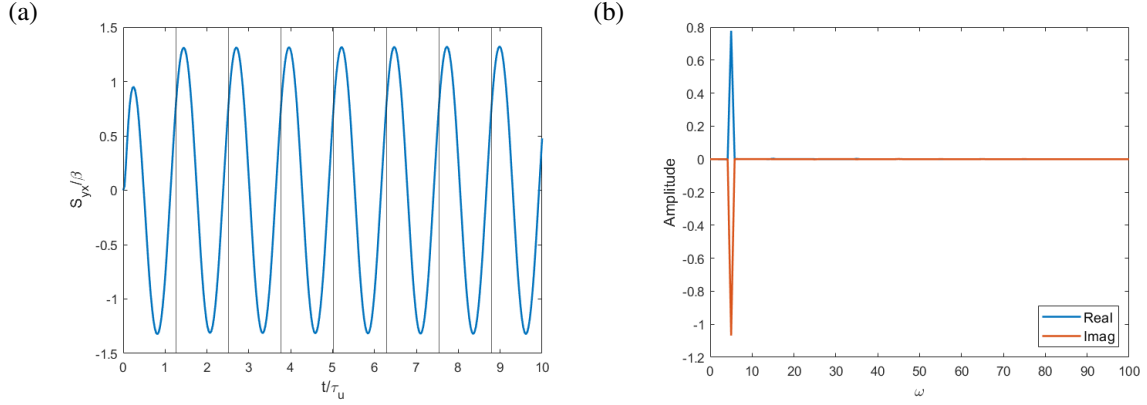
Figure 4.6 – Shear stress component of the particle stress tensor as a function of time (a) and angular velocity (b). Vertical lines in (a) separate the periods of the shear rate. Note that in (b) the transient start-up period is discarded, so that only the stationary regime is represented. $\gamma_0 = 0.2, \omega_\sigma = 0.4$.

With regards to the shear stress, Fig. 4.6 presents the shear stress component of the particle stress tensor both as a function of time and as a function of the angular velocity of the signal components, obtained by performing a Fast Fourier Transform on the original, time-based signal. In order to accurately decompose the stress signal with the Fourier transform, the transient start-up period must be discarded, so that only the stationary regime is considered, and the data points must perfectly represent an integer number of shear rate periods. Note that the shear stress is not in phase with the shear rate, but it is in phase with the droplet deformation, and that this signal only has a single component, with no higher order harmonics. By comparing Eq. 4.5 to the definition of the Fourier transform, it can be seen that the peak in the real component of the decomposed signal corresponds to $G''\gamma_0/Ca$ and the negative peak in the imaginary component corresponds to $-G'\gamma_0/Ca$. Note that both peaks occur at the exact excitation angular velocity $\omega = 5$.

Figure 4.7 presents the values of G' and G'' as a function of ω_σ, alongside the results of the model of Palierne [23]. For emulsions with $\lambda = 1$, Palierne's model reduces to $G^* = G' + iG'' = 28G_c^*/(35G_c^* + 16)$, where $G_c^* = 0 + i\omega_\sigma$ is the complex shear modulus of the continuous phase. It can be seen that there is an overall excellent agreement between the present results and those predicted by the model, indicating that the results presented here are accurate. Although there is some significant deviation between the present results and the model in the case of G'' for $\omega_\sigma > 3$, the values of G'' in this range are an order of magnitude smaller than those of G', and thus this deviation does not significantly affect the behaviour of the emulsion, since in this regime it is dominated by G'.

It can be seen in Fig. 4.7 that both moduli present equal magnitudes around $\omega_\sigma = 0.4$, which is also around the peak value of G'', with $G' > G''$ for higher oscillation frequencies and $G'' > G'$ for lower frequencies. With decreasing oscillation frequencies, the storage
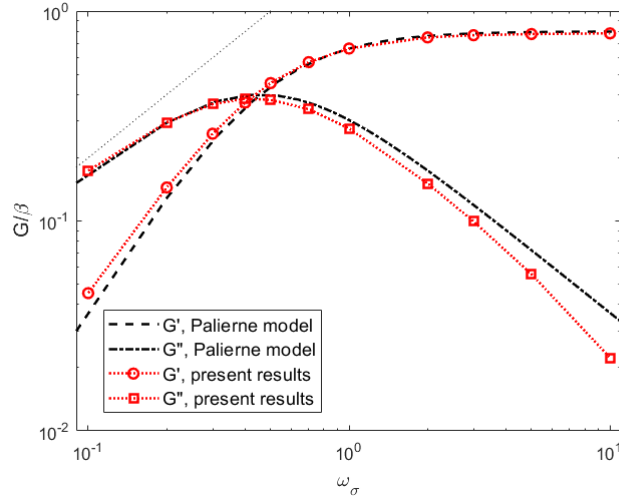
Figure 4.7 – Storage and loss moduli as a function of the oscillation angular velocity ω_σ. The dotted line corresponds to $G/\beta = 2\omega_\sigma$.

modulus quickly decays, such that the particle stress behaviour becomes dominated by the loss modulus for $\omega_\sigma < 0.1$. In this regime, the droplet relaxation time is far smaller than the oscillation periods, and thus the droplet's shape becomes more dependent on the instantaneous shear rate than the shear strain, since it has plenty of time to relax during the oscillation cycles, resulting in a particle stress behaviour that is mostly viscous. In fact, in the limit of $\omega_\sigma \to 0$, the flow would become a simple shear flow, where the shear stresses become governed solely by the viscosity G''/ω_σ. Although not confirmed by the results presented in Fig. 4.7, the results do indicate this trend, with values of G' quickly decaying and values of G'' asymptotically approaching a line of constant viscosity with $G'' \propto \omega_\sigma$.

In the case of $\omega_\sigma > 0.4$, it can be seen in Fig. 4.7 that values of G'' steadily decrease while values of G' increase until reaching a plateau around $\omega_\sigma = 2$. In these regimes of $\omega_\sigma > 2$, the droplet relaxation time is longer than the oscillation periods, so that, with minimal relaxation, the droplet's shape becomes more dependent on the shear strain than on the instantaneous shear rate. This is reflected in the particle stress behaviour, which becomes mostly elastic, with $G'' \to 0$ in the limit of $\omega_\sigma \to \infty$. The dependence of the particle stress on shear strain, rather than shear rate, is also evidenced by the values of G' becoming independent of the oscillation frequency.

4.3 FERROFLUID DROPLET IN QUIESCENT FLOWS

In the case of ferrofluid droplets under the effect of an external magnetic field, the simplest possible case is that of droplets immersed in a quiescent flow. In this case, the system dynamics are governed solely by the magnetic and capillary forces, since fluid flow is limited

to that caused by changes in the droplet's shape. Thus, the steady-state droplet dynamics are governed solely by the magnetic capillary number. Here, two quantities are of interest: the droplet's deformation, caused by the magnetic forces and resisted by capillary forces, and the bulk magnetization of the emulsion, defined as [46]

$$\langle \mathbf{M} \rangle = \frac{1}{V} \int_V (\zeta(\phi) - 1)\mathbf{H}dV \, . \tag{4.6}$$

Note that, as evidenced by Eq. 4.6, the non-magnetic continuous phase does not contribute to the bulk magnetization, since $\zeta(\phi) - 1 = 0$ outside the ferrofluid phase.
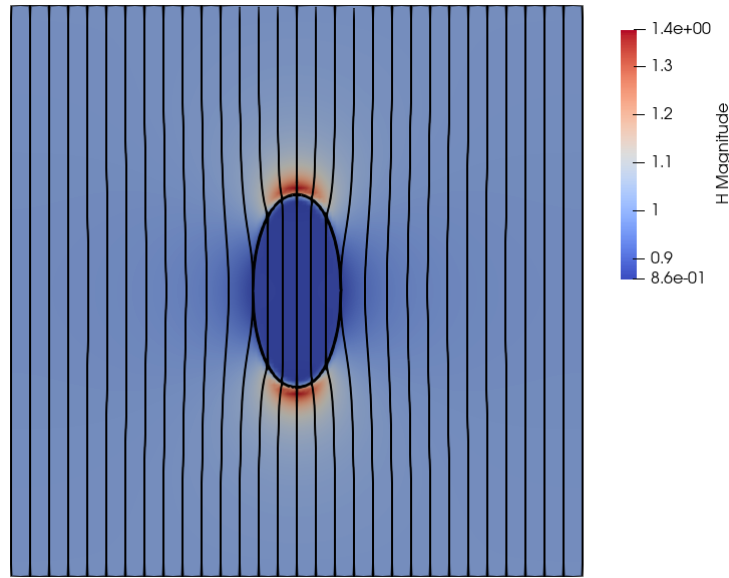
Figure 4.8 – Magnetic field for a ferrofluid droplet in a quiescent flow, $Ca_{mag} = 12$.

Figure 4.8 presents the magnetic field for a ferrofluid droplet in a quiescent flow, for $Ca_{mag} = 12$. It can be seen that the presence of the droplet, due to it's higher magnetic permeability, affects the magnetic field outside of it, attracting the field lines and increasing the magnitude of $\mathbf{H}$ near the droplet tips. Inside the droplet, meanwhile, the magnetic field is mostly uniform, with field lines parallel to the external field direction, and a lower magnitude of $\mathbf{H}$, resulting in a jump in field intensity across the interface. In accordance to Eq. 1.4, this jump in magnetic field intensity across the interface gives rise to magnetic forces, which effectively pull on the interface, stretching the droplet in the magnetic field direction.

Based on Eqs. 1.1 and 1.4, it can be expected that the magnetic field is dependent on the droplet's permeability ratio and shape, while the magnetic forces acting on the interface are dependent on the permeability ratio, field intensity jump, and magnetic capillary number. Note, however, that, while the droplet's shape is dependent on the magnetic forces, it also influences the magnetic field, which in turn determines the magnetic forces. Thus, the actual system behaviour becomes non-linear. In order to exemplify this, Fig. 4.9 presents the magnetic field for a droplet with $\zeta = 5$ and $Ca_{mag} = 3$, such that the product $(\zeta - 1)Ca_{mag}$
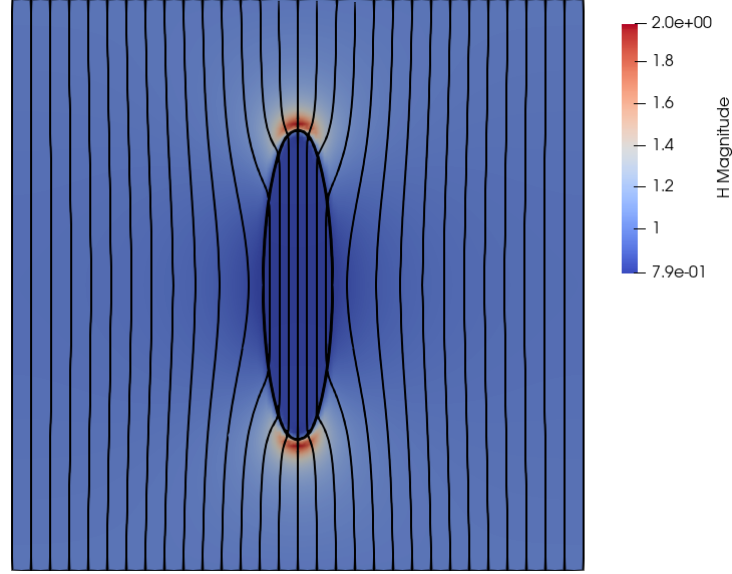
Figure 4.9 – Magnetic field for a ferrofluid droplet in a quiescent flow, $Ca_{mag} = 3$, $\zeta = 5$.

is constant between Figs. 4.8 and 4.9. Despite this similarity, the results are significantly different. It can be seen that, for $\zeta = 5$, the higher permeability ratio causes the droplet's influence on the magnetic field to be more significant, leading to larger field line deflections and changes in field intensity (note the different scales between both figures). This results in a larger field intensity jump in the case of $\zeta = 5$, and, in turn, in a much larger droplet deformation. An analysis on the influence of ζ on the emulsion's dynamics is not the focus of this work, however, and thus will not be investigated in further detail.

As previously mentioned, the two quantities of interest in the case of ferrofluid droplets in quiescent flows are the droplet deformation, to investigate how the magnetic forces influence the droplet shape, and the bulk magnetization, to investigate how changes in droplet shape affect the magnetic behaviour of the emulsion. Figure 4.10 presents both quantities as a function of magnetic capillary number, alongside the models of Afkhami et al. [40] and Jesus et al. [4], and the Lattice-Boltmann Method with Phase-Field results of Ishida and Matsunaga [3] for the droplet deformation. The model of Afkhami et al. predicts the magnetic capillary number based on the droplet's deformation, with

$$Ca_{mag} = 2 \left(\frac{1}{\zeta - 1} + k \right)^2 \left(\frac{1+D}{1-D} \right)^{\frac{1}{3}} \left[2 \left(\frac{1+D}{1-D} \right) - \left(\frac{1+D}{1-D} \right)^{-2} + 1 \right] , \qquad (4.7)$$

where

$$k = \frac{1-E^2}{2E^3} \left(\ln \frac{1+E}{1-E} - 2E \right) , \qquad (4.8)$$

$$E = \sqrt{1 - \left(\frac{1+D}{1-D} \right)^2} . \qquad (4.9)$$

64

The model of Jesus et al., meanwhile, is a small deformation theory for droplets subjected to both an external magnetic field in the y direction and a simple shear flow, with the case of quiescent flows represented by $Ca = 0$. It predicts that, for $\lambda = 1$ and $\zeta = 2$,

$$D = \frac{\sqrt{[(35/16)Ca]^2 + [(9/128)Ca_{mag}]^2}}{2 + (3/128)Ca_{mag}} .$$ (4.10)

It can be seen in Fig. 4.10 that, up to $Ca_{mag} = 16$, there is close agreement between both models and the present results, with higher values of Ca_{mag} leading to some divergence between the models, with the present results remaining close to the model of Afkhami et al. It can also be seen that there is a nearly exact match between the present results and those of Ishida and Matsunaga.
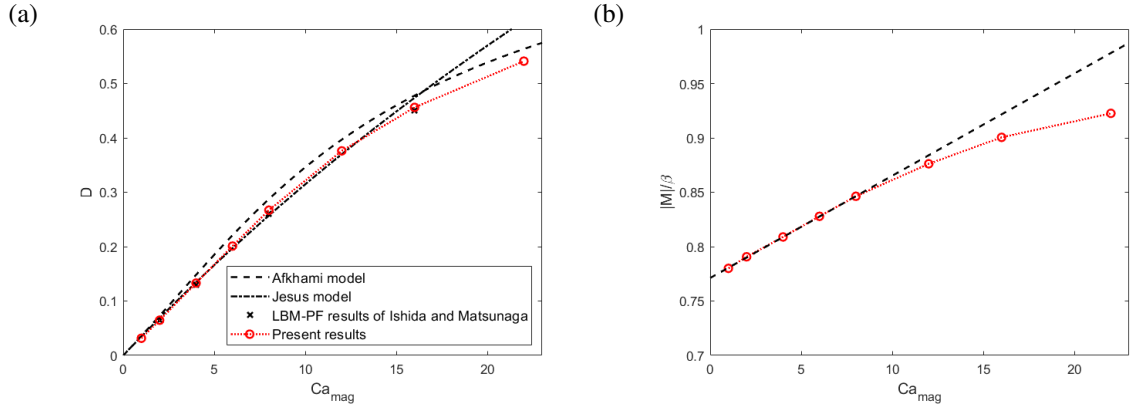
Figure 4.10 – Droplet deformation (a) and bulk emulsion magnetization (b) as a function of magnetic capillary number. The dashed line in (b) corresponds to $|\mathbf{M}|/\beta = 0.7712 + 0.0094Ca_{mag}$. Quiescent flow.

Since the aforementioned models have previously been validated against numerical and experimental results [40, 4, 43], this close agreement to the present results indicates that the magnetic components of the methodology presented in this work are accurate. As previously discussed, the droplet deformation is calculated based on the assumption that the droplet has an ellipsoidal shape. This has been shown to be the case for small and moderate droplet deformations [41], while droplets exhibiting large deformations start to deviate from an ellipsoid due to sharply increased curvatures on the droplet tips. This sharp increase in curvature has not been observed in the present results, even in the case of $Ca_{mag} = 22$. In fact, this is not clearly seen even in the case of higher permeability ratio presented in Fig. 4.9, which has a deformation of $D = 0.635$, significantly larger than any of the cases presented in Fig. 4.10.

In the case of droplet deformation, it can be seen that the droplet deformation monotonically increases with capillary number. For $Ca_{mag} < 10$, this increase is mostly linear. However, for larger magnetic capillary numbers, magnetic effects start to display some saturation, with a decreasing slope of the droplet deformation curve. The curve for bulk mag-

netization also follows a similar trend, albeit with a non-zero value at the limit $Ca_{mag} \to 0$, following the linear relationship $|\mathbf{M}|/\beta = 0.7712 + 0.0094 Ca_{mag}$ for $Ca_{mag} < 10$. In the limit of $Ca_{mag} \to 0$, where the droplet is mostly spherical, this relationship approaches the bulk magnetization of a sphere, given by $|\mathbf{M}|/\beta = 3(\zeta - 1)/(\zeta + 2) = 0.75$. Since the bulk magnetization is only a function of permeability ratio and droplet shape, this indicates that the magnetization increases with the droplet's length alongside the external field direction.

4.4 FERROFLUID DROPLET IN SIMPLE SHEAR FLOWS

After investigating the effects of an external magnetic field on a ferrofluid droplet immersed in a quiescent flow in the previous section, this section now investigates the case of a ferrofluid droplet under the combined action of an external magnetic field and a simple shear flow. In this case, magnetic and shear effects can have additive or opposed effects, leading to a far more complex behaviour, dependent on the capillary number, the magnetic capillary number, and the external magnetic field direction.

4.4.1 Droplet shape

The first step to analyzing the behaviour of the system is to investigate how the shear flow and external magnetic field influence the droplet's shape, as represented by it's deformation and inclination angle. To this end, Fig. 4.11 presents the droplet deformation and inclination angle for external magnetic fields in the x (main flow), y (main velocity gradient) and z (main vorticity) directions. In the case of external magnetic fields in the x and y directions, droplet deformation and inclination angle are calculated in the same way described previously, based on the droplet's inertia tensor. For external magnetic fields in the z direction, however, as the droplet is stretched in the direction of the external field, it's longest axis becomes the one normal to the shear plane. This causes the aforementioned methodology, as implemented, to fail to accurately measure the droplet's deformation and inclination angle, which by definition must be measured on the shear plane that crosses the droplet center. In order to accurately measure the droplet's deformation and inclination angle in this case, an ellipse is fitted to the droplet shape in the shear plane using an image recognition algorithm [130].

In addition to the present results, Fig. 4.11 also presents the Lattice-Boltzmann Method with Phase Field results of Ishida and Matsunaga [3], and the model of Jesus et al. [4] for $Ca = 0.05$ and external magnetic fields in the y direction. It can be seen that the model of Jesus et al. performs remarkably well up to $Ca_{mag} = 16$, after which it diverges from the present results, similarly to what was observed for the same model in the case of quiescent
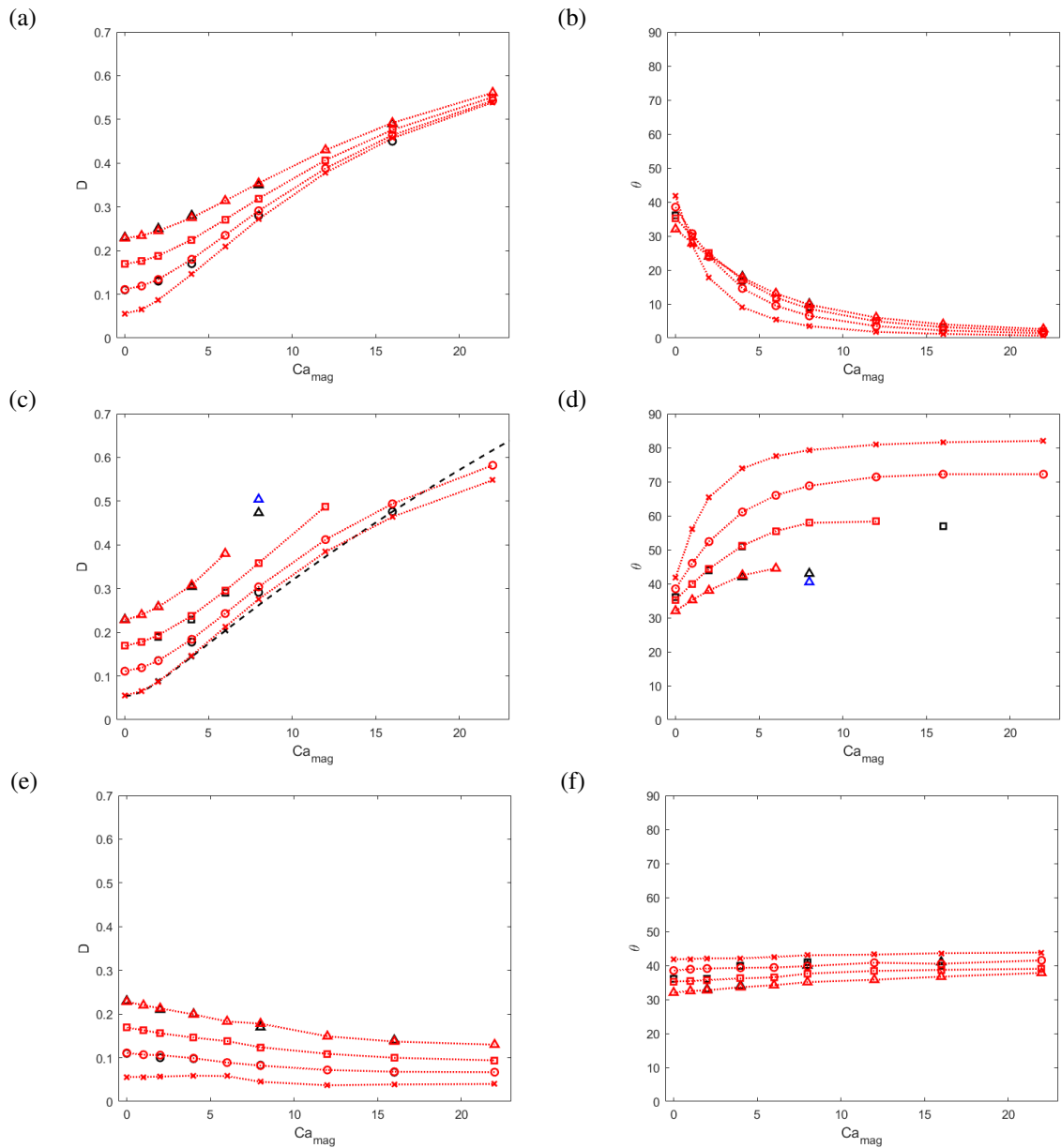
Figure 4.11 – Droplet deformation and inclination angle, in degrees, as a function of magnetic capillary number, for external magnetic fields in the x direction - (a) and (b), in the y direction - (c) and (d), and in the z direction - (e) and (f). Crosses correspond to $Ca = 0.05$, circles correspond to $Ca = 0.1$, squares correspond to $Ca = 0.15$, and triangles correspond to $Ca = 0.2$. Red markers correspond to the present results and black markers correspond to the results of Ishida and Matsunaga [3]. The dashed line in (c) corresponds to the model of Jesus et al. [4] for $Ca = 0.05$. Blue markers represent results obtained with an alternative interface curvature calculation.

flows (Fig. 4.10). There is also, in general, a close agreement between the present results and those of Ishida and Matsunaga, save for a few exceptions. These disagreements are mostly restricted to limit cases where there was droplet breakup in the present results, but not in those of Ishida and Matsunaga, and in the case of inclination angles for external magnetic fields in the z direction. For the latter, the results of Ishida and Matsunaga display a discontinuity around $Ca_{mag} = 3$, which is also around where the droplet's longest axis becomes the one in the z direction. This raises the possibility that this disagreement is due to failures in the measurements of Ishida and Matsunaga in these particular cases.

For the cases of droplet breakup, however, further investigations would be required to determine the reasons for the disagreement. In the case of $Ca = 0.2$ and $Ca_{mag} = 8$, there was a droplet breakup in simulations performed with the standard curvature calculation, although the droplet did not breakup when calculating the curvature with the reduced stencil scheme presented in section 2.3. This, however, resulted in a larger droplet deformation and smaller inclination angle, indicating that the droplet is closer to break-up than that reported by Ishida and Matsunaga. In the case of $Ca = 0.15$ and $Ca_{mag} = 16$, both curvature calculations resulted in droplet breakup. Despite the aforementioned disagreements, the overall close agreement between both results indicate that the present results are accurate, at least excluding the cases of droplet breakup which are not presented.

For external magnetic fields in the x direction, it can be seen in Fig. 4.11 that increases in magnetic capillary number lead to monotonic increases in droplet deformation and monotonic decreases in inclination angle. This is due to magnetic forces stretching the droplet in the external field direction. Thus, stronger magnetic fields stretch the droplet further, increasing it's deformation and, since it is stretched in the x direction, reducing it's inclination angle. It is worth mentioning that shear effects are still present, and can be seen both in the case of droplet deformation and inclination angle. In the case of droplet deformation, larger capillary numbers lead to larger droplet deformations, as expected. However, this effect becomes less pronounced as the magnetic capillary number increases and magnetic effects start to dominate the system's response. This effect is also compounded by the fact that the magnetic-induced droplet deformations reduce the droplet's length in the y direction, confining it to regions of lower velocity flow and thus reducing the effective shear to which it is subjected to. For $Ca_{mag} \to 0$, different capillary numbers lead to significantly different droplet deformations, while changes in droplet deformation due to changes in capillary number are only minor for $Ca_{mag} = 22$.

In the case of inclination angle, larger capillary numbers lead to smaller inclination angles in the case of $Ca_{mag} \to 0$, as previously discussed in the case of a non-magnetic droplet. As the magnetic capillary number increases, however, this trend reverses, so that larger capillary numbers lead to larger inclination angles for $Ca_{mag} > 4$. The reason for this is that, in the absence of a shear flow, the droplet would perfectly align to the external field, with

$\theta = 0$. However, the presence of a shear flow causes some deformation in the 45° direction, preventing the droplet from perfectly aligning to the external magnetic field. For lower capillary numbers, this shear-induced deformation is less significant in the system's dynamics, leading to lower inclination angles, while the opposite is true for higher capillary numbers, which lead to more significant shear-induced deformations and, in turn, to larger inclination angles.

Figure 4.12 – Shape of a ferrofluid droplet in the shear plane. External magnetic field in the x direction, $Ca = 0.15$, $Ca_{mag} = 12$.

Figure 4.12 presents the droplet shape in the case of an external magnetic field in the x direction, with $Ca = 0.15$ and $Ca_{mag} = 12$. It can be seen that the droplet is significantly deformed and nearly aligned to the external field direction, indicating that, as previously discussed, the system's response is mostly dominated by magnetic effects.

For external magnetic fields in the y direction, Fig. 4.11 shows that increases in magnetic capillary number again lead to monotonic increases in droplet deformation. In this case, however, the influences of capillary number remain significant regardless of magnetic capillary number, indicating that magnetic effects never dominate the system's response. The reason for this is that, as the droplet is stretched in the y direction by magnetic forces, it encounters higher velocity flows and is thus subjected to a stronger effective shear, which further deforms the droplet and increases the relevance of the capillary number on the system's response. The droplet's inclination angle also reflects these same effects, with influences of capillary number remaining relevant across the entire magnetic capillary number range. In this case, as the magnetic forces attempt to align the droplet to the external field, they cause an increase in inclination angle up to around $Ca_{mag} = 8$. For higher magnetic capillary numbers, however, the stronger effective shears associated to the larger droplet deformations counteract the aligning effects of the stronger magnetic forces. For $Ca = 0.05$ and $Ca = 0.1$, this results in mostly constant inclination angles for $Ca_{mag} > 12$. For $Ca = 0.15$

and $Ca = 0.2$, however, the increase in relevance of the shear effects causes a reduction in inclination angle and, eventually, an earlier onset of droplet breakup.
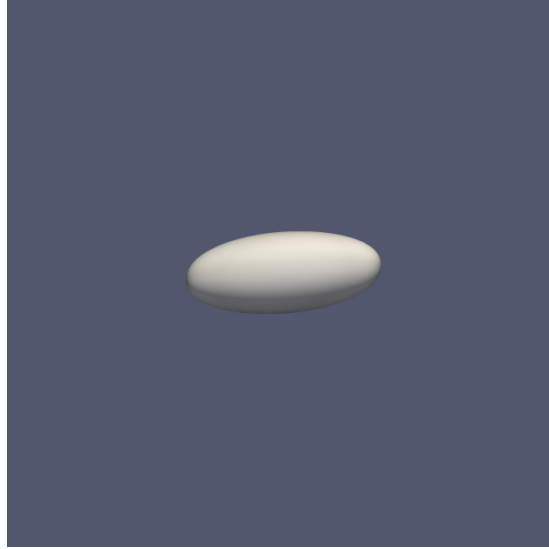

Figure 4.13 – Shape of a ferrofluid droplet in the shear plane. External magnetic field in the y direction, $Ca = 0.15$, $Ca_{mag} = 12$.

Figure 4.13 presents the droplet shape in the case of an external magnetic field in the y direction, with $Ca = 0.15$ and $Ca_{mag} = 12$. It can be seen that the droplet is significantly deformed but not aligned to the external magnetic field, instead presenting an inclination angle of $\theta = 58°$. In this case, it is clear that both magnetic and shear effects are relevant in the system's response.

In the case of external magnetic fields in the z direction, it can be seen in Fig. 4.11 that changes in magnetic capillary number lead to far less significant changes in droplet deformation and inclination angle, when compared to the previous two cases. In fact, the droplet deformation presents the opposite behaviour of the previous cases, with increases in magnetic capillary number leading to decreases in droplet deformation. The reason for this is that, in this case, the magnetic forces stretch the droplet in a direction normal to the shear plane, where droplet deformation and inclination angle are measured. Thus, the magnetic-induced deformations cause a reduction in the droplet's cross-sectional area in the shear plane, due to the conservation of the droplet's volume. This causes an increase in curvature and, in turn, of surface tension. Furthermore, this reduction in cross-sectional area also confines the droplet to regions of lower velocity flow, reducing the effective shear to which it is subjected to. Both of these effects represent a decrease in effective capillary number, leading to decreases in droplet deformation and slight increases in inclination angle.

Figure 4.14 presents the droplet shape in the shear plane, in the case of an external magnetic field in the z direction, with $Ca = 0.15$ and $Ca_{mag} = 12$. It is clear that the droplet's cross-sectional area is far smaller than the two previous cases (Figs. 4.12 and 4.13), and
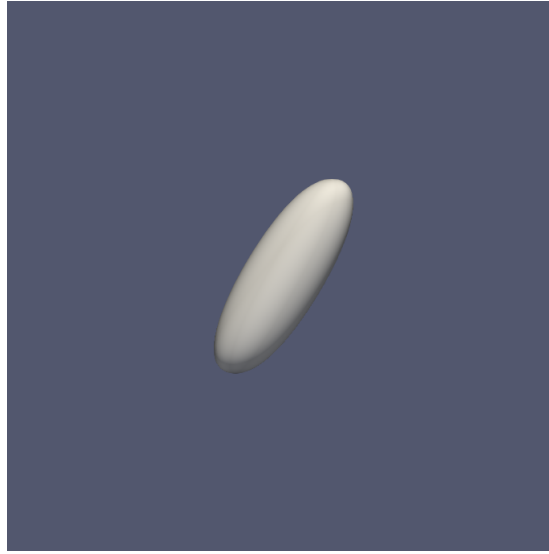
Figure 4.14 – Shape of a ferrofluid droplet in the shear plane. External magnetic field in the z direction, $Ca = 0.15$, $Ca_{mag} = 12$.

also smaller than that of a non-magnetic droplet (Fig. 4.2). Despite this reduction in cross-sectional area, the droplet's shape, in the shear plane, is otherwise similar to that of a non-magnetic droplet, with a moderate deformation and an inclination angle $\theta = 38°$. Note that, in this case, the droplet is significantly stretched in the z direction, although this is not visible in Fig. 4.14.

One interesting aspect of external magnetic fields in the z direction, however, is that shear-induced and magnetic-induced deformations happen in different planes, with shear stresses stretching the droplet in the xy plane while magnetic forces stretch the droplet in the z direction. In cases where both deformations are large and of similar magnitude, this can lead to droplets of a distinct, disc-like shape. One such case is presented in Fig. 4.15, for $Ca = 0.6$ and $Ca_{mag} = 16$.

4.4.2 Rheology

With the effects of external magnetic fields on the droplet shape investigated in the previous section, this section now investigates how these effects influence the system's rheology. Similar to the case of a non-magnetic droplet, the three main quantities of interest are the shear stress component of the particle stress tensor and the first and second normal stress differences of the system. Starting this investigation with the shear stress component of the particle stress tensor, Fig. 4.16 presents $\mathbf{S}_{yx}$ as a function of magnetic capillary number, for various capillary numbers and external magnetic field directions, as well as a comparison to some of the results of Ishida and Matsunaga [3]. The comparison between both results is similar to the case of droplet deformation and inclination angle, displaying a close agreement, with the exception of one case of droplet breakup. As discussed in the previous
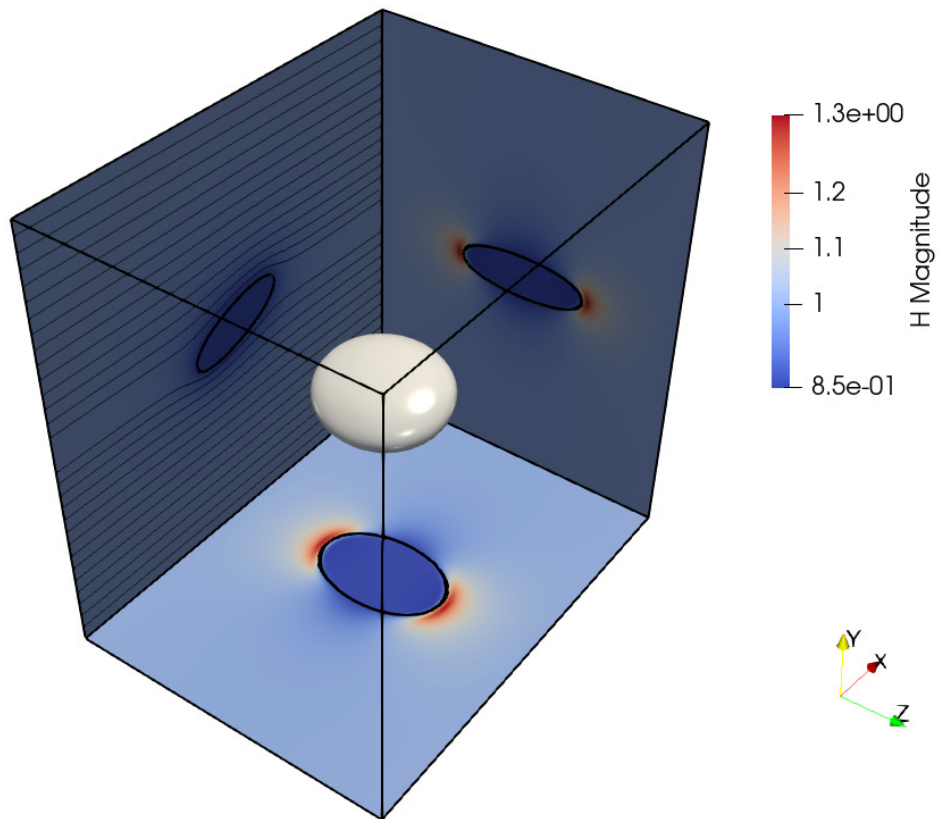
Figure 4.15 – Three-dimensional view of a ferrofluid droplet subjected to an external magnetic field in the z direction, $Ca = 0.6$, $Ca_{mag} = 16$. Slices of planes crossing the droplet center are projected to the domain boundaries, with velocity streamlines present in the shear plane. Reproduced from Phys. Fluids 33, 053313 (2021), with the permission of AIP Publishing [5].

section, a steady-state solution for this particular case could be obtained by using an alternative interface curvature calculation, although the results of this simulation are not as close to the results of Ishida and Matsunaga. However, since there is a close agreement for all remaining results, this indicates that the results presented, which exclude this breakup case, are accurate.
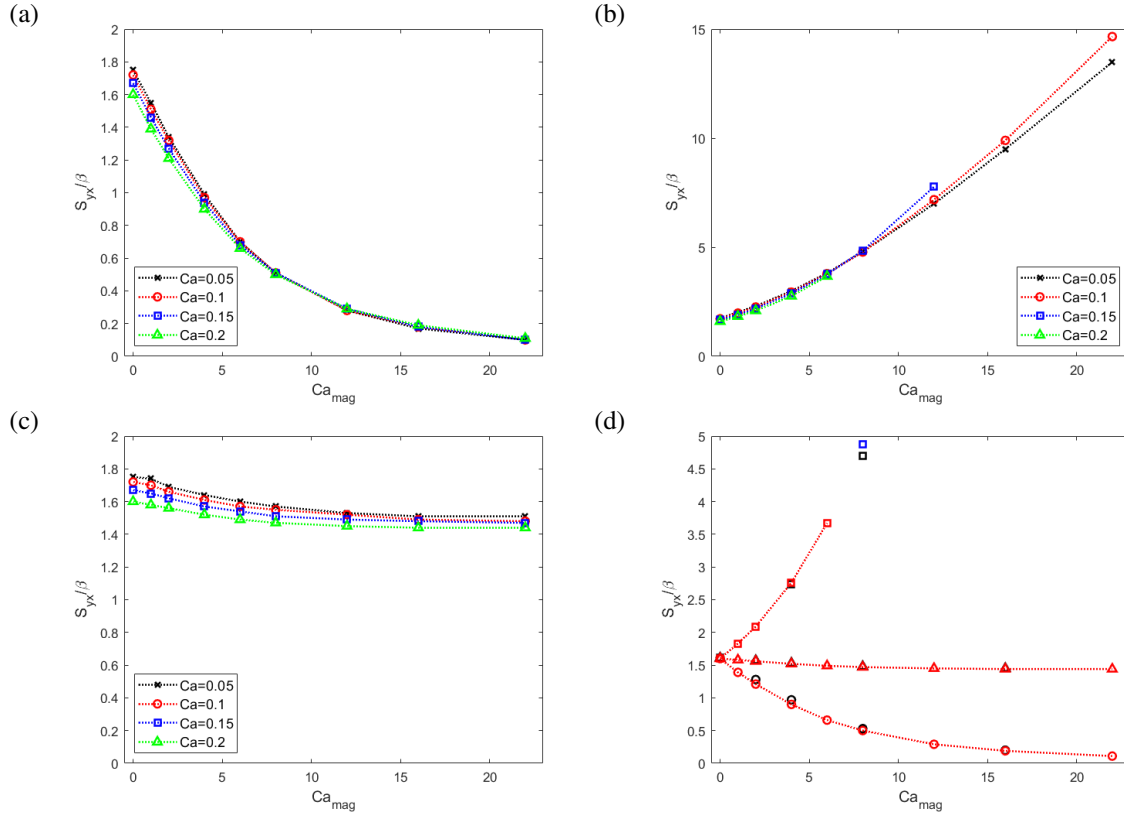
Figure 4.16 – Shear stress component of the particle stress tensor as a function of magnetic capillary number, for external magnetic fields in the x direction (a), y direction (b), and z direction (c), as well as aggregate results for $Ca = 0.2$ (d). In (d), circles correspond to external fields in the x direction, squares correspond to external fields in the y direction, and triangles correspond to external fields in the z direction. Red markers correspond to the present results and black markers correspond to the results of Ishida and Matsunaga [3]. The blue marker represents a result obtained with an alternative interface curvature calculation.

In the case of external magnetic fields in the x direction, it can be seen in Fig. 4.16 that increases in magnetic capillary number lead to major decreases in the emulsion's viscosity. The reason for this is that, as the magnetic forces stretch the droplet in the x direction, they significantly reduce the droplet's length in both the y and z directions. By reducing the droplet's length in the y direction, they reduce the effective shear to which the droplet is subjected, and by reducing it's length in the z direction, they reduce the droplet's cross-sectional area relative to the flow direction. Both of these effects reduce the resistance the droplet imposes to the flow, thereby significantly reducing the emulsion's viscosity. Additionally, in this case magnetic torques attempt to rotate the droplet towards the main flow direction, which also reduces the resistance it imposes to the flow. It is interesting to note

that, around $Ca_{mag} = 8$, the emulsion switches from a shear-thinning behaviour to a slightly shear-thickening one, although this effect is barely noticeable due to the dominance of magnetic effects in the higher magnetic capillary number range. For lower magnetic capillary numbers, the emulsion presents the characteristic shear-thinning behaviour, similar to that of a non-magnetic one. However, for larger magnetic capillary numbers, increases in capillary number lead to increases in inclination angle, as previously discussed and shown in Fig. 4.11. This increase in inclination angle causes the droplet to present slightly more resistance to the shear flow, thereby increasing the emulsion's viscosity.

For external magnetic fields in the y direction, however, increases in magnetic capillary number lead to major increases in the emulsion's viscosity. In this case, as the magnetic forces stretch the droplet in the y direction, they significantly increase the effective shear to which the droplet is subjected, while also counteracting the shear flow's tendency to rotate the droplet towards the main flow direction. The result is that, in this case, the droplet offers significantly more resistance to the shear flow, thereby increasing the emulsion's viscosity. It can also be seen that, for $Ca_{mag} > 8$, the emulsion's behaviour switches from the characteristic shear-thinning behaviour to a shear-thickening one. In this case, increases in capillary number lead to increases in droplet deformation and decreases in inclination angle, as shown in Fig. 4.11. This is qualitatively similar to the case of a non-magnetic droplet, which results in a shear-thinning emulsion. Thus, the droplet's shape alone does not provide a clear explanation for this shear-thickening behaviour. In this case, the magnetic torque attempting to rotate the droplet towards the y direction is the most probable cause for the shear-thickening behaviour. This magnetic torque will be investigated in detail in a later section, and it's effects on the emulsion's viscosity have been studied in detail in [46].

In the case of external magnetic fields in the z direction, increases in magnetic capillary number lead to slight decreases in the emulsion's viscosity since, as the droplet is stretched in the external field direction, it's cross-sectional area in the shear plane is reduced along with the effective shear to which it is subjected to. This decrease in viscosity is, however, far smaller than the one observed for external magnetic fields in the x direction. The reason for this is that, in this case, the increase in surface curvature leads to an increase in capillary forces, and that the droplet's deformation in the z direction actually increases the droplet's cross-sectional area relative to the flow direction, both of which contribute to an increase in the emulsion's viscosity. However, since the reduction in the droplet's cross-sectional area on the shear plane has the most significant effects, the emulsion's viscosity is reduced. In this case, the emulsion retains the characteristic shear-thinning behaviour across the entire magnetic capillary number range.

In addition to the emulsion's viscosity, external magnetic fields also have an influence on the emulsion's first and second normal stress differences. In order to compare the present results to those available in the literature, Fig. 4.17 presents results of N_1 and N_2 as a function
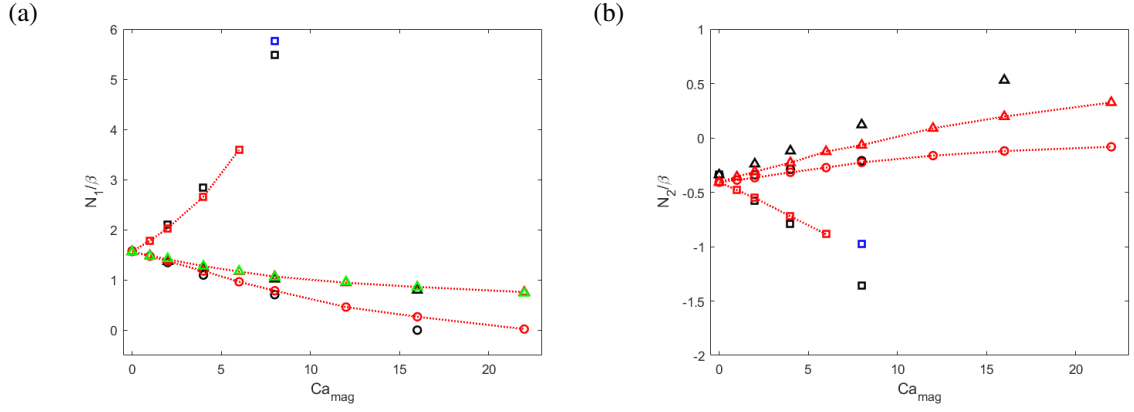
(a)

(b)

Figure 4.17 – First (a) and second (b) normal stress differences as a function of magnetic capillary number, for $Ca = 0.2$. Circles correspond to external fields in the x direction, squares correspond to external fields in the y direction, and triangles correspond to external fields in the z direction. Red markers correspond to the present results and black markers correspond to the results of Ishida and Matsunaga [3]. The blue marker represents a result obtained with an alternative interface curvature calculation, and the green markers represent values for the first normal stress difference calculated as $N_1 = 2 \cot(2\theta) \mathbf{S}_{yx}$.

of magnetic capillary number for $Ca = 0.2$, with the present results displayed alongside those of Ishida and Matsunaga [3]. For N_1, there is still a good agreement between both results, although not as close as the results for droplet shape and emulsion viscosity. For N_2, there is a good agreement for external magnetic fields in the x and y directions, although there is a significant disagreement between both results in the case of external magnetic fields in the z direction. This is likely due to the high sensitivity of the normal stress differences to numerical errors, since they are the difference of two much larger quantities. Thus, in the absence of other results for comparison, no conclusion can be made with regards to accuracy in this particular case. It is worth mentioning, however, that the results of Ishida and Matsunaga were obtained with significantly finer meshes than the present ones, and therefore this disagreement could be due to a lack of sufficient refinement in the meshes used in this work, given the higher sensitivity to numerical errors of the normal stress differences when compared to other quantities, such as the reduced viscosity. Despite this disagreement, both results present the same qualitative trends.

Fig. 4.17 also presents the approximation proposed in [127], that $N_1 = 2 \cot(2\theta) \mathbf{S}_{yx}$. In the case of external magnetic fields in the z direction, it works remarkably well, even though it was not developed for ferrofluid emulsions. For external magnetic fields in the x and y directions, however, it fails to account for magnetic forces which have a direct influence on N_1, and thus was omitted from the figure.

In the case of external magnetic fields in the x direction, Fig. 4.18 presents N_1 and N_2 as a function of magnetic capillary number, for varying capillary numbers. In can be seen that increases in magnetic capillary number lead to monotonic decreases in N_1, up to the

Figure 4.18 – First (a) and second (b) normal stress differences as a function of magnetic capillary number, for external magnetic fields in the x direction.

point of sign reversal in some cases, and to monotonic decreases in magnitude of N_2. In the absence of an external magnetic field, as previously discussed, the increased capillary forces at the droplet tips, the largest component of which is in the x direction, causes an increase in $\mathbf{S}_{xx}$ and positive values of N_1. When an external magnetic field in the x direction is applied, one could expect that this would cause an increase in N_1, since it decreases the inclination angle and, in turn, increases the component of capillary forces on the droplet tips in the x direction. This, however, is not the case, since the magnetic forces stretching the droplet in the x direction counteract the capillary forces attempting to restore the droplet to a spherical shape, reducing the values of N_1 and even changing it's sign on some cases. For N_2, the main component of the magnetic forces, acting on the x direction, has little direct effect. The magnetic forces do, however, have an indirect effect. As the droplet is stretched and becomes nearly parallel to the x direction, the differences in capillary forces in the y and z directions are reduced, since the droplet approaches an axisymmetric shape around the x axis. This leads to a reduction in magnitude of N_2, approaching zero for $Ca_{mag} \to \infty$.
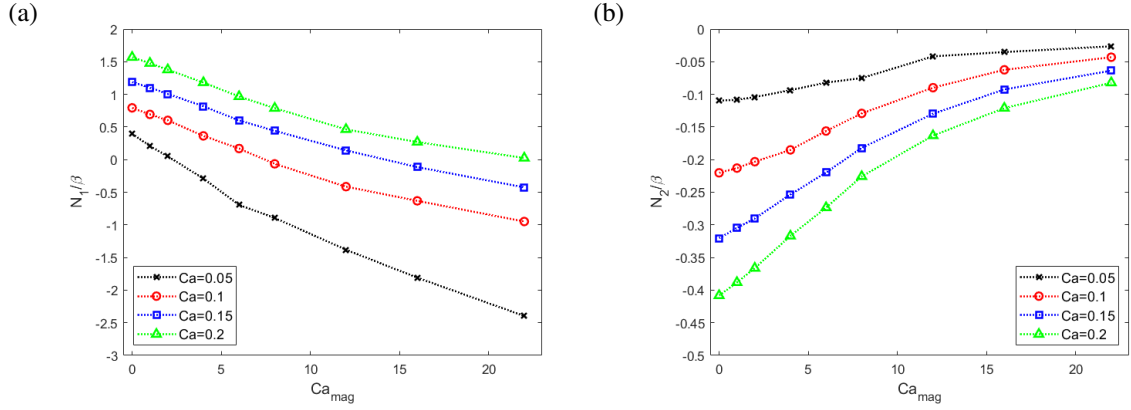

Figure 4.19 – First (a) and second (b) normal stress differences as a function of magnetic capillary number, for external magnetic fields in the y direction.

For external magnetic fields in the y direction, it can be seen in Fig. 4.19 that increases

in magnetic capillary number lead to increases in magnitude for both N_1 and N_2. In this case, the magnetic forces stretching the droplet in the y direction overcome the additional capillary forces, caused by the increase in inclination angle, and cause a reduction in $\mathbf{S}_{yy}$, thereby increasing the positive values of N_1 and decreasing the negative values of N_2. It is interesting to note that, for $Ca_{mag} > 4$, the largest magnitudes of N_2 occur for $Ca = 0.05$, whereas this capillary number would result in the smallest magnitude of N_2 for $Ca_{mag} \to 0$, and that such a drastic change in N_2 with increasing values of magnetic capillary number, or a similar reversal in behaviour, are not observed for the other values of capillary number.
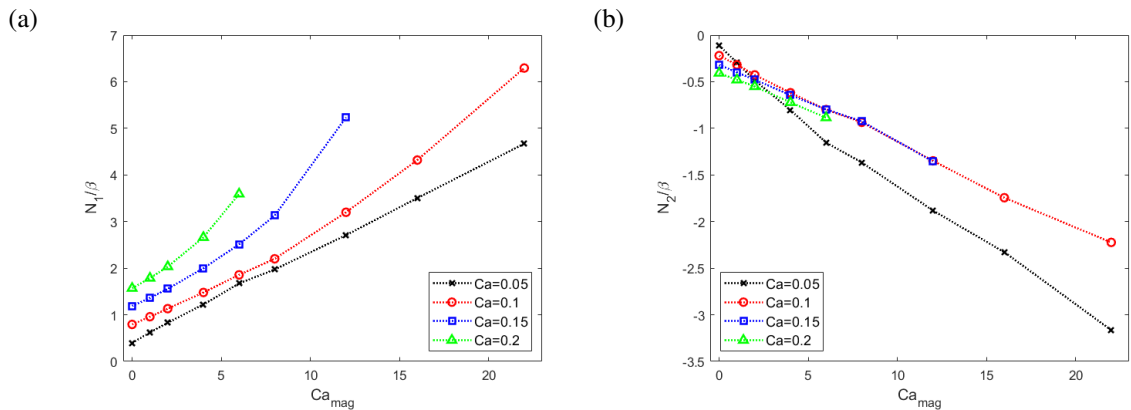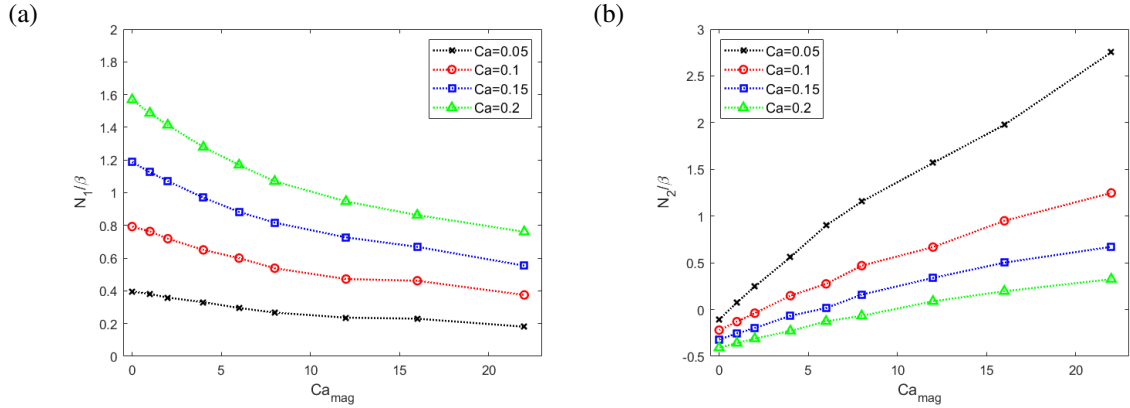

Figure 4.20 – First (a) and second (b) normal stress differences as a function of magnetic capillary number, for external magnetic fields in the z direction.

In the case of external magnetic fields in the z direction, Fig. 4.20 shows that increases in magnetic capillary numbers lead to decreases in N_1. Since the main component of the magnetic forces, in the z direction, has little direct effect on N_1, this reduction is caused by the decrease in droplet deformation and increase in inclination angle. For N_2, however, the magnetic forces have a significant effect since, as they stretch the droplet in the z direction, they cause a reduction in $\mathbf{S}_{zz}$. This, in turn, leads to marked increases in N_2 and even to a reversal of it's sign, particularly for lower capillary numbers.

4.4.3 Magnetization

In addition to the emulsion's rheology, it is also interesting to study it's bulk magnetization, in order to understand how different capillary and magnetic capillary numbers influence the magnetic properties of the emulsion. According to Eq. 4.6, the emulsion's bulk magnetization is directly dependent on the magnetic field inside the ferrofluid droplet. To better understand how the magnetic field behaves inside the droplet, and how it is influenced by the droplet shape, Fig. 4.21 presents a shear plane view of the droplet and the magnetic field, for $Ca = 0.15$, $Ca_{mag} = 12$, and external magnetic fields in the x and y directions.

In Fig. 4.21, it can be seen that the magnetic field inside the droplet is mostly uniform, such that the bulk magnetization direction, represented by the red arrows, is parallel to the
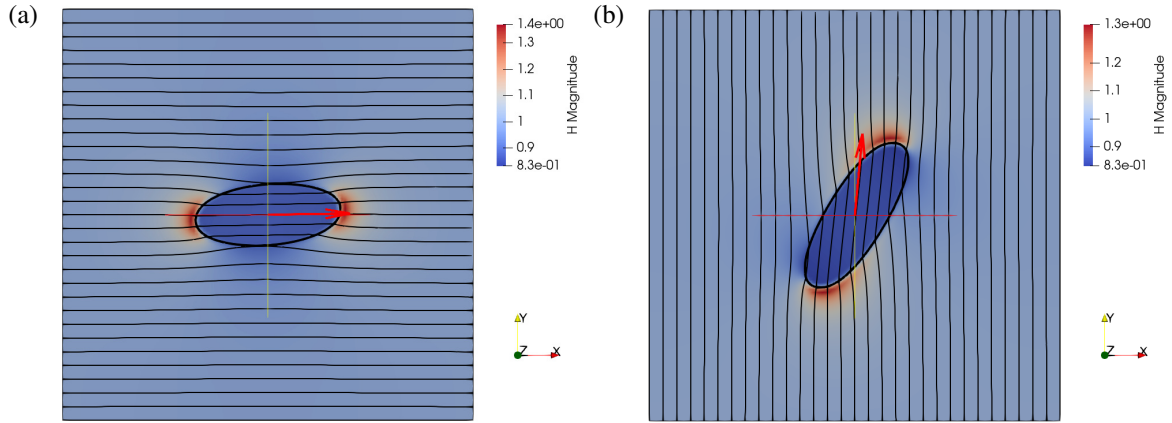
Figure 4.21 – Shear plane view of a droplet subjected to external magnetic fields in the x (a) and y (b) directions, for $Ca = 0.15$ and $Ca_{mag} = 12$. Red arrows correspond to the bulk magnetization direction. Reproduced from Phys. Fluids 33, 053313 (2021), with the permission of AIP Publishing [5].

magnetic field direction. However, despite the assumption that the ferrofluid is superparamagnetic, the magnetic field inside the droplet does not perfectly align to the external magnetic field, instead displaying a tendency to align itself to the droplet's long axis, leading to a misalignment angle θ_{mag} between the bulk magnetization direction and the external magnetic field direction. This misalignment angle is present both in the case of external magnetic fields in the x direction and in the case of external magnetic fields in the y direction, although it is significantly smaller for external magnetic fields in the x direction.

The reason for this difference in misalignment angle is that the droplet's long axis is nearly aligned to the external magnetic field direction in the case of external fields in the x direction, leading to smaller misalignment angles, whereas the droplet's long axis has a significant misalignment to the external field direction in the case of external magnetic fields in the y direction, leading to larger misalignment angles. It is also interesting to note that the regions of highest field intensity jump across the interface, which are where the droplet surface is perpendicular to the external magnetic field and near the droplet tips, are not perfectly aligned. Since the magnetic forces act in an outward direction, normal to the droplet surface, this offset in the magnetic forces gives rise to magnetic torques in the system.

In order to better understand how the droplet shape, and thus how different values of capillary and magnetic capillary numbers influence the bulk magnetization's misalignment, Fig. 4.22 presents this misalignment angle as a function of magnetic capillary number, for external magnetic fields in the x and y directions and for varying capillary numbers. It can be seen that, for external magnetic fields in the x direction, increases in magnetic capillary number lead to monotonic decreases in misalignment angle, and increases in capillary number lead to monotonic increases in misalignment angle. The reason for this is that the stronger magnetic forces associated with higher magnetic capillary numbers cause the droplet to become
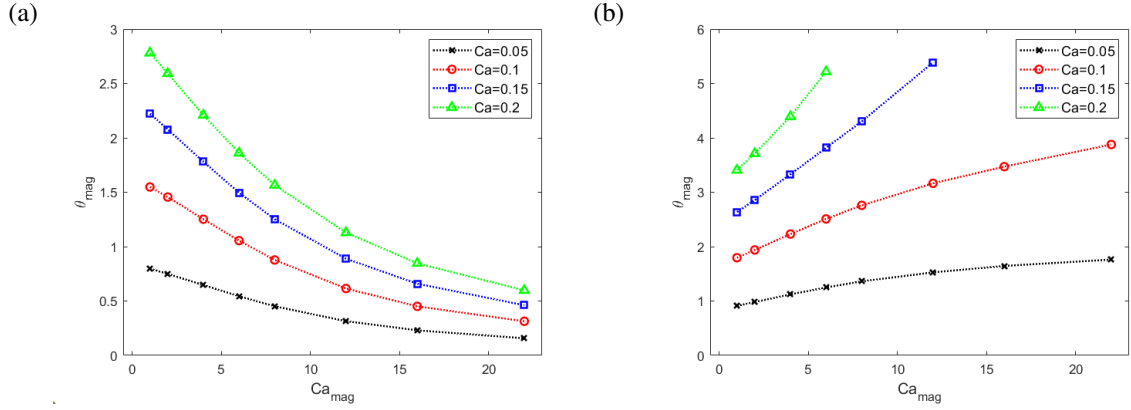
Figure 4.22 – Misalignment angle, in degrees, between the bulk magnetization and external magnetic field directions as a function of magnetic capillary number, for external magnetic fields in the x (a) and y (b) directions.

more aligned to the external magnetic field, with smaller inclination angles, while the larger shear-induced deformations associated with higher capillary numbers lead to increases in droplet inclination angle, as shown in Fig. 4.11 for $Ca_{mag} > 4$.

For $Ca_{mag} < 4$, however, the aforementioned correlation between the inclination and misalignment angles is not as clear, and is in fact reversed for $Ca_{mag} \to 0$. The reason for this is that, in such cases, the magnetic forces are not as significant, and, for lower capillary numbers, the droplet is nearly spherical. In this case, the droplet's long axis is nearly the same length as it's short axis, which results in smaller deflections of the magnetic field and, in turn, in smaller misalignment angles. Thus, it can be concluded that misalignment angles are dependent not only on droplet inclination angle, but also on droplet deformation, with more deformed droplets having a larger influence on the magnetic field direction. Therefore, for $Ca_{mag} \to 0$, even though larger capillary numbers lead to smaller inclination angles, they lead to larger misalignment angles, due to the larger droplet deformations.

In the case of external magnetic fields in the y direction, it can be seen that, similar to the case of external magnetic fields in the x direction, increases in capillary number lead to increases in misalignment angle, due to the increases in droplet deformation and the shear flow's effect of rotating the droplet away from the external magnetic field direction. However, opposite to the case of external magnetic fields in the x direction, increases in magnetic capillary number lead to monotonic increases in misalignment angle. The reason for this is that, in this case, despite the magnetic forces attempting to align the droplet to the external magnetic field direction, increases in magnetic capillary number do not lead to monotonic increases in inclination angle, as can be seen in Fig. 4.11 (in this case, the droplet would be aligned to the external magnetic field with an inclination angle of $\theta = 90°$). Thus, with these less significant changes in inclination angle, the increases in droplet deformation associated with larger magnetic capillary numbers lead to increases in misalignment angle.

For external magnetic fields in the z direction, misalignment angles are always equal to zero, up to numerical accuracy of the results. In this case, the external magnetic field is normal to the shear plane, and thus shear-induced deformations do not rotate the droplet's long axis, which is always parallel to the external magnetic field.

In addition to the misalignment angle, another important property of the emulsion's bulk magnetization is it's magnitude $|\mathbf{M}|$. As discussed in section 4.3, the magnitude of the magnetization is dependent on the droplet's shape, with increases in droplet length alongside the external magnetic field direction leading to increases in magnetization magnitude. Therefore, it is to be expected that the shear flow has an influence on magnetization magnitude, since it leads to changes in the droplet's shape. To investigate this influence, Fig. 4.23 presents the magnitude of the emulsion's magnetization as a function of magnetic capillary number, for varying capillary numbers and external magnetic fields in the x, y and z directions. It can be seen that, regardless of external magnetic field direction and capillary number, all curves have a very similar behaviour to that of a ferrofluid droplet immersed in a quiescent flow. However, changes in capillary number do have an influence in the magnetization's magnitude, which varies depending on external magnetic field direction.

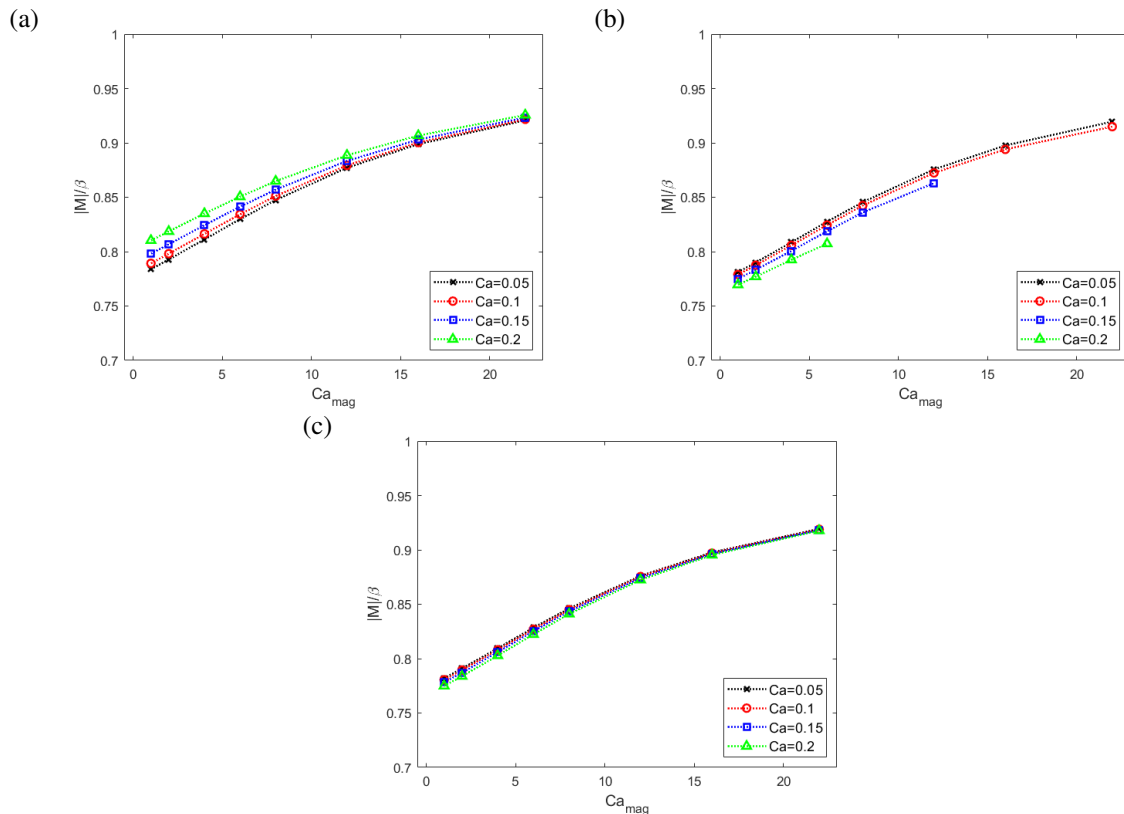(a)   (b)

(c)

Figure 4.23 – Bulk emulsion magnetization as a function of magnetic capillary number, for external magnetic fields in the x (a), y (b) and z (c) directions.

For external magnetic fields in the x direction, it can be seen that increases in capillary number lead to increases in magnetization magnitude, although this effect is reduced with

increases in magnetic capillary number. In this case, due to the small inclination angles assumed by the droplet, increases in shear-induced deformation, associated with higher capillary numbers, lead to increases in the droplet's length alongside the x axis, while changes in inclination angle don't have a significant effect on this length.

In the case of external magnetic fields in the y direction, the opposite behaviour is observed, with increases in capillary number leading to decreases in magnetization magnitude. In this case, even though increases in capillary number lead to increases in droplet deformation, they also lead to significant decreases in inclination angle, rotating the droplet away from the y axis and, in turn, reducing the droplet's length alongside the external magnetic field direction. For external magnetic fields in the z direction, increases in capillary number lead to minor decreases in magnetization magnitude, with small decreases in the droplet's length alongside the z direction caused by the conservation of volume, as the droplet is stretched in the shear plane due to capillary effects.

It is interesting to note that, due to the misalignment between the external magnetic field and the emulsion's bulk magnetization, a magnetic torque arises in the system. This magnetic torque is given by [131]

$$\tau_{mag} = \frac{Ca_{mag}}{Ca} \langle \mathbf{M} \rangle \times \mathbf{H}_0 \,, \tag{4.11}$$

where it is clear that τ_{mag} is directly proportional to both the sine of the misalignment angle $\sin(\theta_{mag})$ and the bulk magnetization magnitude $|\mathbf{M}|$. Moreover, the conservation of angular momentum in the system dictates that an opposite hydrodynamic torque τ_{hyd} must arise in the system, such that $\tau_{mag} + \tau_{hyd} = \mathbf{0}$. This hydrodynamic torque can be obtained by splitting the particle stress tensor into a symmetric part $\mathbf{S}_s$ and an antisymmetric part $\mathbf{S}_a$, which are respectively referred to as the stresslet and couplet [125]. The hydrodynamic torque can then be obtained as [46]

$$\tau_{hyd} = -\epsilon : \mathbf{S}_a \,, \tag{4.12}$$

where ϵ is the Levi-Civita permutation symbol.

Figure 4.24 presents the magnetic and hydrodynamic torques as a function of magnetic capillary number, for $Ca = 0.1$ and external magnetic fields in the x and y directions. It can be seen that there is a close agreement between τ_{mag} and $-\tau_{hyd}$, indicating that both magnetic and hydrodynamic measurements are accurate. It can also be seen that, following the coordinate systems in Figs. 1.1 and 4.21, both torques act in the z direction, with magnetic torques attempting to rotate the droplet into alignment with the external magnetic field, while hydrodynamic torques attempt to rotate the droplet away from it. For external magnetic fields in the z direction, both torques are always equal to zero, since $\langle \mathbf{M} \rangle$ and $\mathbf{H}_0$ are always parallel.

To better understand the influence of different capillary numbers and external magnetic field directions on the magnetic torques arising in the system, Fig. 4.25 presents such torques
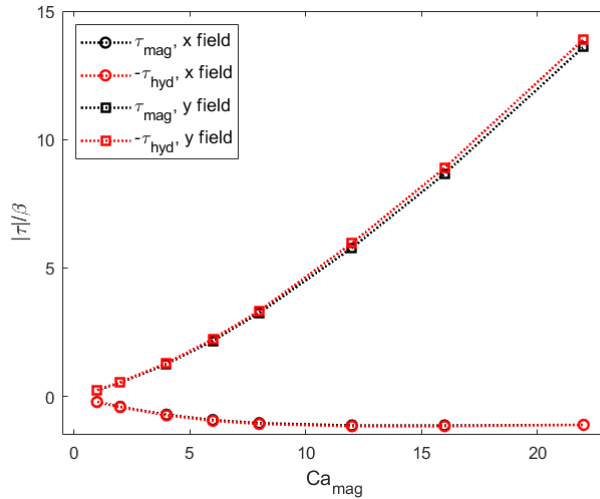
Figure 4.24 – Magnetic and hydrodynamic torques as a function of magnetic capillary number, for $Ca = 0.1$ and external magnetic fields in the x and y directions.

as a function of magnetic capillary number, for varying capillary numbers and external magnetic fields in the x and y directions. For external magnetic fields in the x direction, it can be seen that magnetic torques are always negative, corresponding to a clockwise rotation in Fig. 4.21, and increase in magnitude up to $Ca_{mag} \approx 12$. In this regime, increases in magnetic capillary number and in magnetization magnitude are more significant than the decreases in misalignment angle, leading to increases in magnetic torque. For $Ca_{mag} > 12$, however, the decreases in misalignment angle become more significant, resulting in magnetic torques that do not vary with magnetic capillary number, or even decrease in magnitude with increases in Ca_{mag}, especially for $Ca = 0.05$. It can also be seen that larger capillary numbers lead to decreases in magnetic torque. Although larger capillary numbers are associated with larger misalignment angles and magnetization magnitudes, these effects are not as significant as the direct, inversely proportional role of the capillary number in Eq. 4.11.

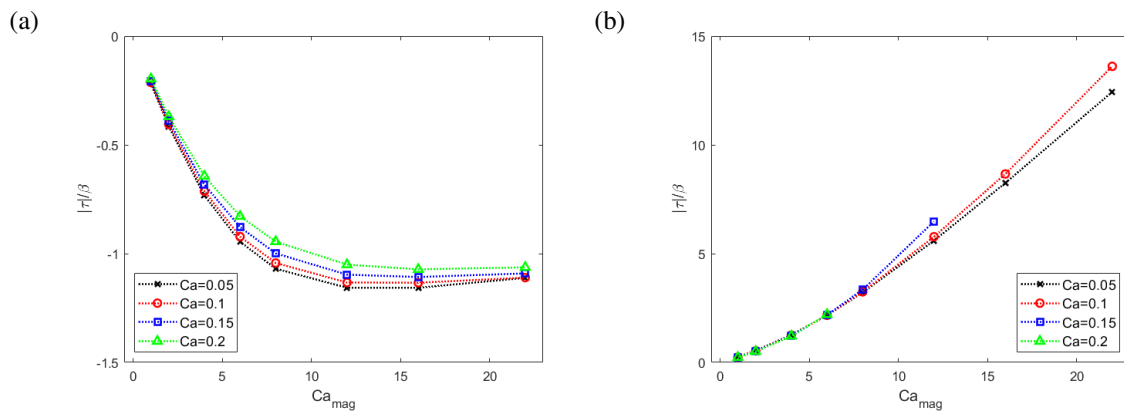(a)   (b)

Figure 4.25 – Magnetic torques as a function of magnetic capillary number, for external magnetic fields in the x (a) and y (b) directions.

In the case of external magnetic fields in the y direction, it can be seen that magnetic

torques are always positive, corresponding to a counter-clockwise rotation in Fig. 4.21, and that increases in magnetic capillary number lead to monotonic increases in magnetic torque, with magnetic torques having up to ten times the magnitude of magnetic torques for external magnetic fields in the x direction. In this case, in addition to the direct influence in Eq. 4.11, increases in magnetic capillary number also lead to increases in magnetization magnitude and to significant increases in misalignment angle, all of which have a compounding effect on magnetic torque magnitude. It can also be seen that, although variations in capillary number have little effect for $Ca_{mag} < 8$, for $Ca_{mag} > 8$ increases in capillary number lead to increases in magnetic torque, opposed to what is observed for external magnetic fields in the x direction. Here, the increases in misalignment angle are large enough to overcome the reductions in magnetization magnitude and the direct influence of the capillary number in Eq. 4.11.

4.5   FERROFLUID DROPLET IN OSCILLATORY SHEAR FLOWS

The last case studied in this work is that of a ferrofluid droplet immersed in oscillatory shear flows, in order to investigate the influence of varying external magnetic field directions and intensities on the emulsion's storage and loss moduli, G' and G'', respectively. As discussed in section 4.2, the relationship between these two moduli is largely dependent on the ratio between the droplet's relaxation time and the shear flow's frequency. It can also be expected that external magnetic fields lead to significant increases or decreases in the magnitude of both moduli, by increasing or decreasing the droplet's overall influence on the flow, in a similar way to that of ferrofluid droplets in simple shear flows discussed in section 4.4.2. Thus, it is possible that external magnetic fields influence the emulsion's response to oscillatory shear flows both by changing the droplet's overall influence on the flow, and by changing the droplet's relaxation time. All results presented in this section are for the same flow parameters used in section 4.2, with a shear amplitude of $\gamma_0 = 0.2$ and $Re = \min(10^{-3}, 10^{-4}/Ca)$.

Figure 4.26 presents the storage and loss moduli as a function of the oscillatory flow's angular velocity, for varying magnetic capillary numbers and external magnetic fields in the x direction, and Fig. 4.27 presents both moduli as a function of magnetic capillary number for $\omega_\sigma = 0.1$ and $\omega_\sigma = 10$. It can be seen that increases in magnetic capillary number lead to significant decreases in both the storage and loss moduli, across the entire range of oscillation frequencies. This is analogous to the behaviour observed in section 4.4.2 for the case of simple shear flows, and is due to similar reasons. As the external magnetic fields stretch the droplet in x direction, the droplet's length in both the y and z directions is reduced, reducing it's cross-sectional area relative to the flow direction and confining it to regions of

83

Figure 4.26 – Storage modulus G' (circles) and loss modulus G'' (squares) as a function of the oscillation angular velocity ω_σ, for external magnetic fields in the x direction and varying magnetic capillary numbers.

lower velocity flow. Both of these effects cause a reduction on the influence the droplet has on the overall flow dynamics, and therefore also cause a reduction in the magnitude of the shear component of the particle stress tensor. This, in turn, leads to the observed reductions in the storage and loss moduli, both of similar magnitude.

Figure 4.27 – Storage modulus G' and loss modulus G'' as a function of magnetic capillary number, for $\omega_\sigma = 0.1$ (a) and $\omega_\sigma = 10$ (b). External magnetic fields in the x direction.

Another important characteristic of the emulsion is the crossover point, corresponding to the oscillation frequency in which the storage modulus becomes larger than the loss modulus, and is thus associated to the change between the mostly viscous behaviour observed for lower oscillation frequencies and the mostly elastic behaviour observed for higher frequencies. For external magnetic fields in the x direction, this point moves from $\omega_\sigma = 0.42$ for $Ca_{mag} = 0$ to $\omega_\sigma = 0.37$ for $Ca_{mag} = 16$. Since this crossover point is associated to

the oscillation frequency in which the time the droplet takes to relax from the shear-induced deformations becomes larger than the oscillation period, the aforementioned changes in this point indicate that, due to the additional deformations in the shear plane caused by the external magnetic field, the droplet takes slightly more time to relax than that of the non-magnetic case. However, this change in the crossover point is not particularly relevant when compared to the range of oscillation frequencies observed and the magnitude of the changes observed in the storage and loss moduli. Moreover, this change in the crossover point is smaller than the intervals in angular velocity between the results presented and thus relies on interpolation to be measured, rendering it more vulnerable to numerical and measurement errors.
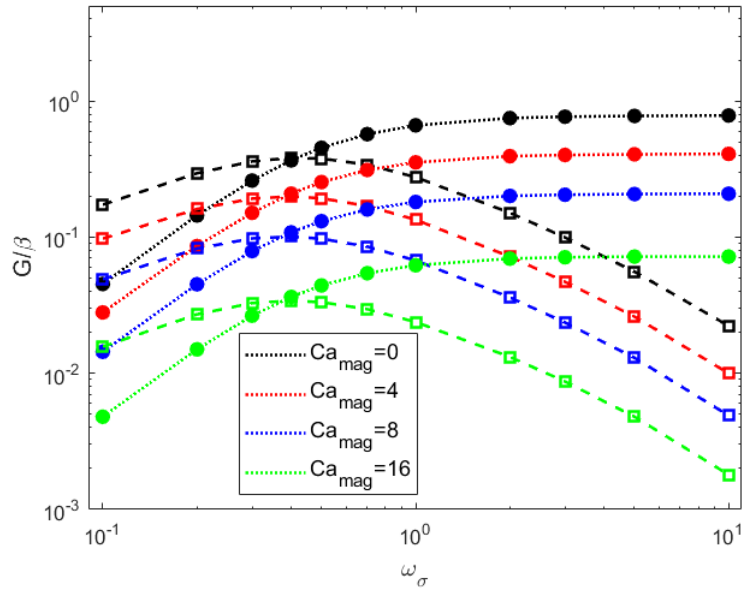

Figure 4.28 – Storage modulus G' (circles) and loss modulus G'' (squares) as a function of the oscillation angular velocity ω_σ, for external magnetic fields in the y direction and varying magnetic capillary numbers.

In the case of external magnetic fields in the y direction, it can be seen in Figs. 4.28 and 4.29 that increases in magnetic capillary number lead to significant increases in the storage and loss moduli across the entire oscillation frequency range, opposite to what is observed for external magnetic fields in the x direction. In this case, as the external magnetic fields stretch the droplet in the y direction, it's cross-sectional area relative to the flow direction is increased and it interacts with regions of higher velocity flow, with both of these effects leading to an increase in the influence the droplet has on the overall dynamics of the flow. This behaviour is analogous to that of the simple shear case discussed in section 4.4.2, and leads to increases in magnitude of the shear component of the particle stress tensor and, in turn, to increases of similar magnitude in both the storage and loss moduli.

In this case, the presence of the external magnetic field has similar effects on the crossover point to those for external magnetic fields in the x direction, with the crossover point again
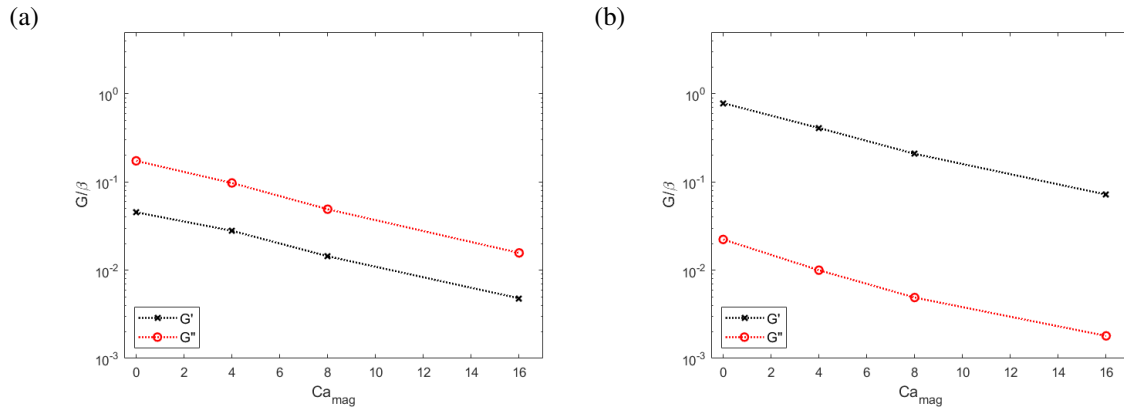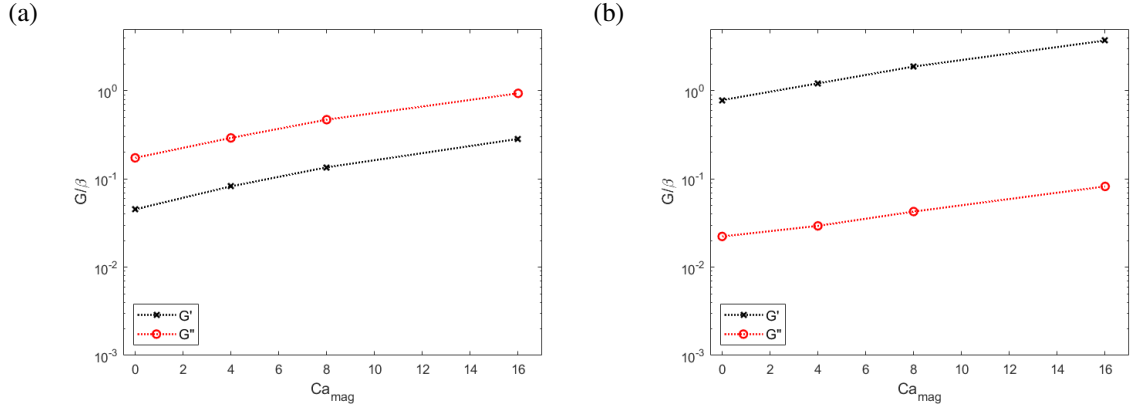
Figure 4.29 – Storage modulus G' and loss modulus G'' as a function of magnetic capillary number, for $\omega_\sigma = 0.1$ (a) and $\omega_\sigma = 10$ (b). External magnetic fields in the y direction.

moving from $\omega_\sigma = 0.42$ for $Ca_{mag} = 0$ to $\omega_\sigma = 0.37$ for $Ca_{mag} = 16$. Although the droplet deformations for external magnetic fields in the x and y directions occur in different directions, the similar magnitude of these deformations results in similarly minor changes in the time the droplet takes to relax, at least for the small amplitude flows analyzed in this section. For both of these cases, the changes in the crossover point are far less significant than the changes in magnitude observed for the storage and loss moduli, indicating that external magnetic fields in the x and y directions can be used to control the magnitude of the emulsion's response to an oscillatory shear flow, but not how this response varies across different oscillation frequencies. In other words, this indicates that external magnetic fields in the x and y directions can be used to control the emulsion's storage and loss moduli for a given flow condition, but not the relationship between them.

For external magnetic fields in the z direction, however, it can be seen in Figs. 4.30 and 4.31 that the effects of different magnetic capillary numbers on the storage and loss moduli are completely different to those for external magnetic fields in the x and y directions. Rather than leading to increases or decreases in the magnitude of the moduli across the entire oscillation frequency range, increases in magnetic capillary number lead to decreases in both moduli for lower frequencies, and to increases in both moduli for higher frequencies.

In the limit of $\omega_\sigma \to 0$, the emulsion's behaviour is similar to the simple shear case discussed in section 4.4.2, with the reduction in the droplet's cross-sectional area in the shear plane, and the associated reduction in the effective shear the droplet is subjected to, leading to decreases in the magnitude of the shear component of the particle stress tensor, and in turn to decreases in both the storage and loss moduli. However, this behaviour changes for $\omega_\sigma > 0.2$, with increases in magnetic capillary number leading to increases in the loss modulus G'', but to decreases in the storage modulus G', and for $\omega_\sigma > 0.8$, with increases in magnetic capillary number leading to increases in both moduli.

As previously discussed in section 4.4.2, the rheology of a ferrofluid emulsion subjected
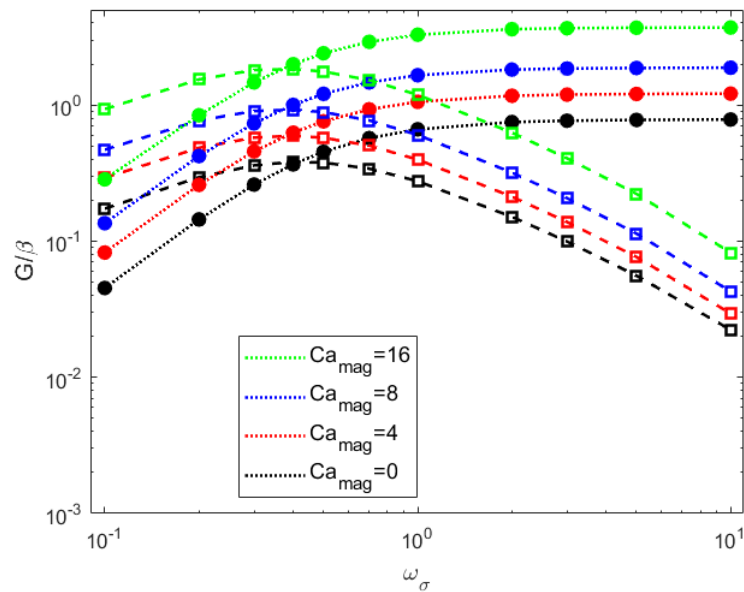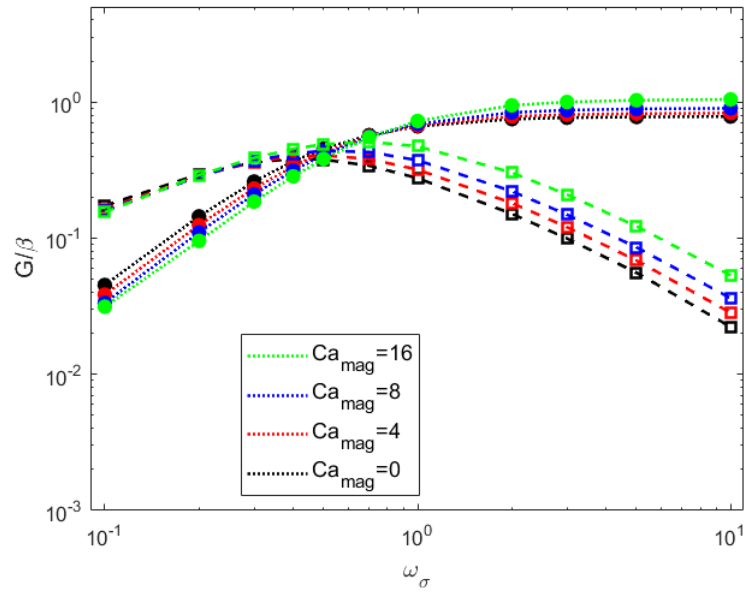
Figure 4.30 – Storage modulus G' (circles) and loss modulus G'' (squares) as a function of the oscillation angular velocity ω_σ, for external magnetic fields in the z direction and varying magnetic capillary numbers.
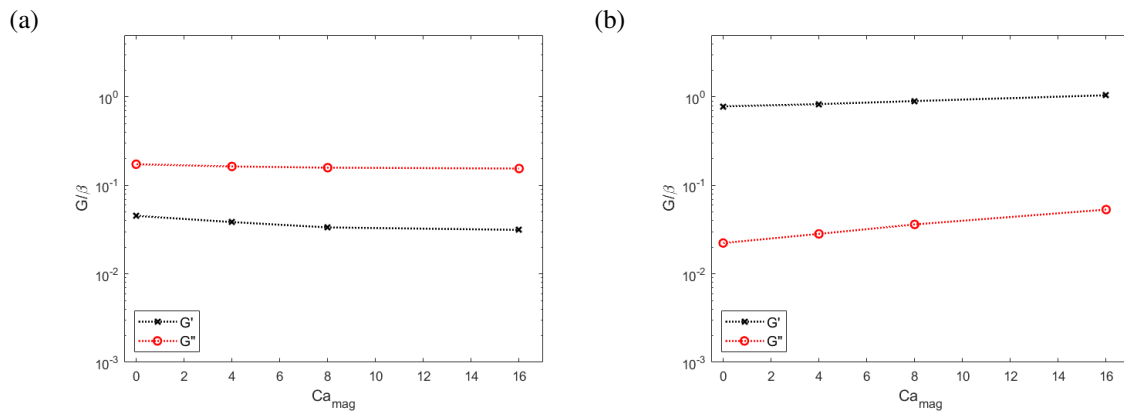

Figure 4.31 – Storage modulus G' and loss modulus G'' as a function of magnetic capillary number, for $\omega_\sigma = 0.1$ (a) and $\omega_\sigma = 10$ (b). External magnetic fields in the z direction.

to external magnetic fields in the z direction is significantly influenced by the reduction in the droplet's cross-sectional area in the shear plane, with two opposing effects: the confinement of the droplet to regions of lower velocity flow, which leads to a decrease in the influence the droplet has on the overall flow dynamics, and the increase in surface curvature, which reduces the droplet's deformation and increases the influence the droplet has on the flow dynamics. In the simple shear case, and for oscillatory flows with $\omega_\sigma < 0.2$, the former dominates, with increases in magnetic capillary number leading to decreases in the rheological moduli. For higher oscillation frequencies, however, it is possible that the latter becomes more dominant, leading to increases in magnetic capillary number causing increases in the storage and loss moduli. The increase in the droplet's length in the z direction, caused by the external magnetic fields, is also expected to increase the influence the droplet has on the flow dynamics, since it increases the droplet's cross-sectional area relative to the flow direction. However, further investigations would be required in order to confidently state the reasons for such a peculiar behaviour.

It can also be seen that, in this case, increases in magnetic capillary number lead to increases in the oscillation frequency of the crossover point, which moves from $\omega_\sigma = 0.42$ for $Ca_{mag} = 0$ to $\omega_\sigma = 0.62$ for $Ca_{mag} = 16$, a change that is far more significant than those observed for external magnetic fields in the x and y directions. In this case, as the droplet is stretched by the external magnetic field, it's cross-sectional area in the shear plane is reduced, reducing the time it takes to relax from the shear-induced deformations and leading to the increase in the oscillation frequency of the crossover point. This indicates that, opposite to the case of external magnetic fields in the x and y directions

5 CONCLUSIONS

This work presented a numerical analysis of the rheological and magnetic behaviour of dilute ferrofluid emulsions, when subjected to an external, uniform magnetic field. Analyses were performed for external magnetic fields parallel to the main flow direction (x), to the main velocity gradient direction (y) and to the main vorticity direction (z), spanning several different combinations of capillary and magnetic capillary numbers, and for simple shear and oscillatory shear flows. A detailed description of the computational code FENRir, which was developed to perform the simulations presented in this work, was also presented.

For the model problems studied in this work, both the continuous and dispersed phases of the emulsion were assumed to be Newtonian fluids of the same viscosity and density, with the continuous phase assumed to be non-magnetic and the dispersed phase assumed to be superparamagnetic. The magnetic relaxation times were assumed to be negligible, such that the magnetic problem could be treated as a magnetostatic one, governed by a magnetic potential equation, and the fluid flow problem was modelled by solving the Navier-Stokes equations with the addition of capillary and magnetic forcing terms.

The computational code FENRir is based on a second-order, finite difference discretization of the governing equations, using a staggered (MAC), regular grid. The temporal discretization of the Navier-Stokes equations is based on the second-order Crank-Nicolson scheme, coupled to a projection method to impose the incompressibility constraint, with a semi-implicit (fully implicit for fluids of equal viscosity) treatment of the viscous term and an explicit treatment of all other terms. The droplet interface is captured with the Level Set method, using a fifth-order WENO scheme for spacial discretization and a third-order SSP Runge-Kutta scheme for temporal discretization, with the Level Set function reinitialized by solving the reinitialization equation with a volume-preserving constraint. Linear systems of equations associated with the Navier-Stokes equations are solved directly by Fourier analysis, and the linear system of equations associated with the magnetic problem is solved by the Conjugate Gradient method with Multigrid preconditioning. Since it uses the Level Set method to capture the interface, the computational code FENRir is able to deal with problems involving droplet breakup in an automatic fashion, although such results were not presented in this work.

The results presented in this work first focused on the case of non-magnetic droplets under simple shear and oscillatory shear flows, with regards to both the emulsion's morphology, defined by the droplet's shape and inclination angle, and to it's rheology. For simple shear flows, it was observed that increases in capillary number lead to increases in droplet deformation and decreases in inclination angle, which in turn lead to a reduction in the emulsion's

viscosity, giving it it's characteristic shear-thinning behaviour. It was also observed that increases in capillary number lead to increases in magnitude for both the positive first normal stress difference and the negative second normal stress difference. For oscillatory shear flows, the storage and loss moduli were obtained as functions of the oscillation frequency, with the emulsion displaying a mostly viscous behaviour for lower frequencies and a mostly plastic behaviour for higher frequencies. All of the aforementioned behaviours are well established in the literature, with the present results displaying an overall close agreement with the results of previous works.

For ferrofluid droplets subjected to external magnetic fields, the first case studied was that of droplets in quiescent flows, in order to focus on the magnetic effects. The droplets were observed to stretch in the external magnetic field direction, with deformations proportional to the magnetic capillary number, and displaying a close agreement to results of previous works. It was also observed that increases in magnetic capillary number lead to increases in the emulsion's bulk magnetization, indicating that the magnetization is proportional to the droplet's length in the external magnetic field direction.

When also subjected to a simple shear flow, the ferrofluid emulsion's behaviour varied significantly with different external magnetic field directions. For external magnetic fields in the x direction , increases in magnetic capillary number lead to increases in droplet deformation and decreases in inclination angle, with the influence of varying capillary numbers decreasing with increasing values of the magnetic capillary number. For higher magnetic capillary numbers, it was also observed that increases in capillary number lead to increases, rather than to decreases in inclination angle. For external magnetic fields in the y direction, increases in magnetic capillary number lead to increases in droplet deformation and inclination angle, with the influence of varying capillary numbers remaining relevant across the entire magnetic capillary number range. It was also observed that the inclination angle, as a function of magnetic capillary number, stabilizes in the higher magnetic capillary number range, and that, for external magnetic fields in the y direction, the magnetic effects lead to an earlier onset of droplet breakup. For external magnetic fields in the z direction, increases in magnetic capillary number lead to slight decreases in droplet deformation and slight increases in inclination angle. The present results were also found to be in close agreement to other results available in the literature.

The aforementioned changes in the emulsion's morphology, as well as the magnetic forces that cause them, also lead to significant changes in the emulsion's rheology. In the case of the emulsion's reduced viscosity, it was observed that, for external magnetic fields in the x direction, increases in magnetic capillary number lead to significant decreases in the emulsion's viscosity, while for external magnetic fields in the y direction, increases in magnetic capillary number lead to significant increases in the emulsion's viscosity. For external magnetic fields in the z direction, however, increases in magnetic capillary number only lead

to minor decreases in the emulsion's reduced viscosity.

With regards to the first and second normal stress differences, however, the effects of the external magnetic fields on the emulsion's behaviour were observed to be more complex. For external magnetic fields in the x direction, increases in magnetic capillary number lead to significant decreases in the first normal stress difference, even changing it's sign from positive to negative. The second normal stress difference reduces in magnitude with increases in magnetic capillary number, although it's sign always remains negative. For external magnetic fields in the y direction, increases in magnetic capillary number lead to significant increases in magnitude of both normal stress differences, although they retain the positive and negative signs for the first and second normal stress differences, respectively. Meanwhile, in the case of external magnetic fields in the z direction, increases in magnetic capillary number were observed to lead to decreases in magnitude of the first normal stress difference, and to significant increases to the second normal stress difference, eventually causing it's sign to change from negative to positive. The present results for the emulsion's reduced viscosity and first normal stress difference were found to be in an overall close agreement with results previously reported in the literature. Although the results for the second normal stress difference did not display such a close agreement, this is likely due to their sensibility to numerical errors.

Similar to the case of ferrofluid droplets in quiescent flows, the magnitude of the emulsion's bulk magnetization was found to increase with increases in magnetic capillary number, although in this case the shear flow also has some influence on the magnetization magnitude. This results in increases in capillary number leading to slight increases in magnetization magnitude, in the case of external magnetic fields in the x direction, and to slight decreases in magnitude in the case of external magnetic fields in the y direction. For external magnetic fields in the z direction, the influence of the capillary number was found to be negligible. The most interesting aspect, however, is that, when subjected to a shear flow, the emulsion's magnetization does not perfectly align to the external magnetic field, even though the droplets are assumed to be superparamagnetic, leading to the existence of a misalignment angle between the bulk magnetization and the external magnetic field directions.

For external magnetic fields in both x and y directions, increases in capillary number were found to lead to significant increases in misalignment angle, although the different magnetic field directions display opposite behaviours with regards to the magnetic capillary number. For external magnetic fields in the x direction, increases in magnetic capillary number were found to lead to significant decreases in misalignment angle, while for external magnetic fields in the y direction, increases in magnetic capillary number were found to lead to significant increases in misalignment angle. For external magnetic fields in the z direction, however, the emulsion's bulk magnetization was found to always be parallel to the external magnetic field direction.

Due to the aforementioned misalignment between the bulk magnetization and the external magnetic field, a magnetic torque is introduced in the system and, due to the conservation of angular momentum, an equal and opposite hydrodynamic torque must also be present in the system. Both torques were measured and found to be in very close agreement with each other. For external magnetic fields in the x direction, the magnetic torque was found to increase in magnitude with increases in magnetic capillary number, although this effect plateaus at higher magnetic capillary numbers, and to decrease in magnitude with increases in capillary number. For external magnetic fields in the y direction, the magnetic torques were found to be much stronger than those observed for magnetic fields in the x direction, and to increase with increases in both capillary and magnetic capillary numbers. For both external magnetic field directions, the magnetic torque attempts to rotate the droplet into alignment with the external magnetic field.

In the case of ferrofluid droplets immersed in oscillatory shear flows, it was found that external magnetic fields in all three directions had distinct, but significant impacts on the emulsion's rheology. For external magnetic fields in the x directions, increases in magnetic capillary number were found to lead to significant decreases in both the storage and loss moduli across the entire oscillation frequency range, while external magnetic fields in the y direction were found to have the opposite behaviour, with increases in magnetic capillary number leading to increases in both moduli. In both cases, it was found that the crossover point, in which the storage modulus becomes larger than the loss modulus, happens at lower frequencies for higher magnetic capillary numbers, although this change in the crossover point is far less significant than the changes observed for the storage and loss moduli.

For external magnetic fields in the z direction, the emulsion's response was found to be quite distinct from the other two cases. In the lower range of oscillation frequencies, increases in magnetic capillary number were found to lead to decreases in both the storage and loss moduli, similarly to the reduction in emulsion viscosity observed for the simple shear case. In the higher range of oscillation frequencies, however, it was found that increases in magnetic capillary number have the opposite effect, leading to increases in both moduli. It was also found that the crossover point moves to higher oscillation frequencies with increases in magnetic capillary number, indicating that that the magnetic-induced deformations lead to a significant decrease in the time the droplet takes to relax from the shear-induced deformations.

It is worth mentioning that there are plenty of further research possibilities arising from the work presented here. Examples of such further developments include a more in-depth analysis of a ferrofluid emulsion's response in oscillatory shear flows, particularly with respect to the emulsion's bulk magnetization, as well as a detailed analysis of droplet breakup and how it is affected by different magnetic field directions and intensities. Both of these analyses could also be extended to the case of fluids with different viscosities, which could

lead to interesting results, particularly in the case of high viscosity rations, in which the droplet's dynamics are significantly altered in the non-magnetic case. Despite the highly efficient solvers already employed, there are still significant improvement opportunities in the computational code FENRir, both with regards to computational efficiency and memory allocation. One of the most prominent of these opportunities is the use of non-uniform grids in the y direction, which is possible with both solvers currently employed, and would allow for significantly finer grids in the droplet region while saving computational resources in the monotonous areas closer to the domain walls.

BIBLIOGRAPHY

1 TAYLOR, G. I. The formation of emulsions in definable fields of flow. *Proc. R. Soc. Lond. A*, The Royal Society, v. 146, n. 858, p. 501–523, 1934.

2 TAYLOR, G. I. The viscosity of a fluid containing small drops of another fluid. *Proc. R. Soc. Lond. A*, The Royal Society, v. 138, n. 834, p. 41–48, 1932.

3 ISHIDA, S.; MATSUNAGA, D. Rheology of a dilute ferrofluid droplet suspension in shear flow: Viscosity and normal stress differences. *Physical Review Fluids*, APS, v. 5, n. 12, p. 123603, 2020.

4 JESUS, W. C.; ROMA, A. M.; CENICEROS, H. D. Deformation of a sheared magnetic droplet in a viscous fluid. *Commun. Comput. Phys*, v. 24, p. 332–355, 2018.

5 ABICALIL, V. G. e.; ABDO, R. F.; CUNHA, L. H. P. da; OLIVEIRA, T. F. de. On the magnetization of dilute ferrofluid emulsions in shear flows. *Physics of Fluids*, AIP Publishing LLC, v. 33, n. 5, p. 053313, 2021.

6 EINSTEIN, A. A new determination of molecular dimensions. *Ann. Phys.*, v. 19, p. 289–306, 1906.

7 RUMSCHEIDT, F.-D.; MASON, S. Particle motions in sheared suspensions xii. deformation and burst of fluid drops in shear and hyperbolic flow. *Journal of Colloid Science*, Elsevier, v. 16, n. 3, p. 238–261, 1961.

8 TORZA, S.; COX, R.; MASON, S. Particle motions in sheared suspensions xxvii. transient and steady deformation and burst of liquid drops. *Journal of Colloid and Interface science*, Elsevier, v. 38, n. 2, p. 395–411, 1972.

9 BENTLEY, B.; LEAL, L. G. An experimental investigation of drop deformation and breakup in steady, two-dimensional linear flows. *Journal of Fluid Mechanics*, Cambridge University Press, v. 167, p. 241–283, 1986.

10 SIBILLO, V.; PASQUARIELLO, G.; SIMEONE, M.; CRISTINI, V.; GUIDO, S. Drop deformation in microconfined shear flow. *Physical review letters*, APS, v. 97, n. 5, p. 054502, 2006.

11 VANANROYE, A.; PUYVELDE, P. V.; MOLDENAERS, P. Effect of confinement on droplet breakup in sheared emulsions. *Langmuir*, ACS Publications, v. 22, n. 9, p. 3972–3974, 2006.

12 VANANROYE, A.; PUYVELDE, P. V.; MOLDENAERS, P. Effect of confinement on the steady-state behavior of single droplets during shear flow. *Journal of rheology*, SOR, v. 51, n. 1, p. 139–153, 2007.

13 VANANROYE, A.; JANSSEN, P. J.; ANDERSON, P. D.; PUYVELDE, P. V.; MOLDENAERS, P. Microconfined equiviscous droplet deformation: Comparison of experimental and numerical results. *Physics of Fluids*, American Institute of Physics, v. 20, n. 1, p. 013101, 2008.

14   GUIDO, S. Shear-induced droplet deformation: Effects of confined geometry and viscoelasticity. *Current Opinion in Colloid & Interface Science*, Elsevier, v. 16, n. 1, p. 61–70, 2011.

15   WANNABORWORN, S.; MACKLEY, M. R.; RENARDY, Y. Experimental observation and matching numerical simulation for the deformation and breakup of immiscible drops in oscillatory shear. *Journal of Rheology*, The Society of Rheology, v. 46, n. 5, p. 1279–1293, 2002.

16   CAVALLO, R.; GUIDO, S.; SIMEONE, M. Drop deformation under small-amplitude oscillatory shear flow. *Rheologica acta*, Springer, v. 42, n. 1, p. 1–9, 2003.

17   JANPAEN, V.; NIAMLANG, S.; LERDWIJITJARUD, W.; SIRIVAT, A. Oscillatory shear induced droplet deformation and breakup in immiscible polymer blends. *Physics of Fluids*, American Institute of Physics, v. 21, n. 6, p. 063102, 2009.

18   SCHOWALTER, W.; CHAFFEY, C.; BRENNER, H. Rheological behavior of a dilute emulsion. *Journal of colloid and interface science*, Elsevier, v. 26, n. 2, p. 152–160, 1968.

19   BARTHES-BIESEL, D.; ACRIVOS, A. Deformation and burst of a liquid droplet freely suspended in a linear shear field. *Journal of Fluid Mechanics*, Cambridge University Press, v. 61, n. 1, p. 1–22, 1973.

20   GRMELA, M.; BOUSMINA, M.; PALIERNE, J.-F. On the rheology of immiscible blends. *Rheologica acta*, Springer, v. 40, n. 6, p. 560–569, 2001.

21   YU, W.; BOUSMINA, M.; GRMELA, M.; PALIERNE, J.-F.; ZHOU, C. Quantitative relationship between rheology and morphology in emulsions. *Journal of Rheology*, The Society of Rheology, v. 46, n. 6, p. 1381–1399, 2002.

22   FRANKEL, N.; ACRIVOS, A. The constitutive equation for a dilute emulsion. *Journal of Fluid Mechanics*, Cambridge University Press, v. 44, n. 1, p. 65–78, 1970.

23   PALIERNE, J. Linear rheology of viscoelastic emulsions with interfacial tension. *Rheologica acta*, Springer, v. 29, n. 3, p. 204–214, 1990.

24   YU, W.; BOUSMINA, M.; GRMELA, M.; ZHOU, C. Modeling of oscillatory shear flow of emulsions under small and large deformation fields. *Journal of Rheology*, The Society of Rheology, v. 46, n. 6, p. 1401–1418, 2002.

25   RALLISON, J. The deformation of small viscous drops and bubbles in shear flows. *Annual review of fluid mechanics*, Annual Reviews 4139 El Camino Way, PO Box 10139, Palo Alto, CA 94303-0139, USA, v. 16, n. 1, p. 45–66, 1984.

26   DERKACH, S. R. Rheology of emulsions. *Advances in colloid and interface science*, Elsevier, v. 151, n. 1-2, p. 1–23, 2009.

27   PAL, R. Rheology of simple and multiple emulsions. *Current opinion in colloid & interface science*, Elsevier, v. 16, n. 1, p. 41–60, 2011.

28   RALLISON, J. A numerical study of the deformation and burst of a viscous drop in general shear flows. *Journal of Fluid Mechanics*, Cambridge University Press, v. 109, p. 465–482, 1981.

29   KENNEDY, M.; POZRIKIDIS, C.; SKALAK, R. Motion and deformation of liquid drops, and the rheology of dilute emulsions in simple shear flow. *Computers & fluids*, Elsevier, v. 23, n. 2, p. 251–278, 1994.

30   JANSSEN, P.; ANDERSON, P. Boundary-integral method for drop deformation between parallel plates. *Physics of Fluids*, American Institute of Physics, v. 19, n. 4, p. 043602, 2007.

31   CUNHA, F.; COUTO, H. A new boundary integral formulation to describe three-dimensional motions of interfaces between magnetic fluids. *Applied mathematics and computation*, Elsevier, v. 199, n. 1, p. 70–83, 2008.

32   LI, J.; RENARDY, Y. Y.; RENARDY, M. Numerical simulation of breakup of a viscous drop in simple shear flow through a volume-of-fluid method. *Physics of Fluids*, AIP, v. 12, n. 2, p. 269–282, 2000.

33   LI, X.; SARKAR, K. Effects of inertia on the rheology of a dilute emulsion of drops in shear. *Journal of Rheology*, The Society of Rheology, v. 49, n. 6, p. 1377–1394, 2005.

34   XI, H.; DUNCAN, C. Lattice boltzmann simulations of three-dimensional single droplet deformation and breakup under simple shear flow. *Physical Review E*, APS, v. 59, n. 3, p. 3022, 1999.

35   IOANNOU, N.; LIU, H.; ZHANG, Y. Droplet dynamics in confinement. *Journal of Computational Science*, Elsevier, v. 17, p. 463–474, 2016.

36   GOUNLEY, J.; BOEDEC, G.; JAEGER, M.; LEONETTI, M. Influence of surface viscosity on droplets in shear flow. *Journal of Fluid Mechanics*, Cambridge University Press, v. 791, p. 464–494, 2016.

37   TORRES-DÍAZ, I.; RINALDI, C. Recent progress in ferrofluids research: novel applications of magnetically controllable and tunable fluids. *Soft matter*, Royal Society of Chemistry, v. 10, n. 43, p. 8584–8602, 2014.

38   SHERWOOD, J. Breakup of fluid droplets in electric and magnetic fields. *Journal of Fluid Mechanics*, Cambridge University Press, v. 188, p. 133–146, 1988.

39   SERO-GUILLAUME, O.; ZOUAOUI, D.; BERNARDIN, D.; BRANCHER, J. The shape of a magnetic liquid drop. *Journal of Fluid Mechanics*, Cambridge University Press, v. 241, p. 215–232, 1992.

40   AFKHAMI, S. a.; TYLER, A.; RENARDY, Y.; RENARDY, M.; PIERRE, T. S.; WOODWARD, R.; RIFFLE, J. S. Deformation of a hydrophobic ferrofluid droplet suspended in a viscous medium under uniform magnetic fields. *Journal of Fluid Mechanics*, Cambridge University Press, v. 663, p. 358–384, 2010.

41   ROWGHANIAN, P.; MEINHART, C.; CAMPÀS, O. Dynamics of ferrofluid drop deformations under spatially uniform magnetic fields. *Journal of Fluid Mechanics*, Cambridge University Press, v. 802, p. 245, 2016.

42   CAPOBIANCHI, P.; LAPPA, M.; OLIVEIRA, M. S. Deformation of a ferrofluid droplet in a simple shear flow under the effect of a constant magnetic field. *Computers & Fluids*, Elsevier, v. 173, p. 313–323, 2018.

43   HASSAN, M. R.; ZHANG, J.; WANG, C. Deformation of a ferrofluid droplet in simple shear flows under uniform magnetic fields. *Physics of Fluids*, AIP Publishing, v. 30, n. 9, p. 092002, 2018.

44   CUNHA, L. H.; SIQUEIRA, I. R.; OLIVEIRA, T. F.; CENICEROS, H. D. Field-induced control of ferrofluid emulsion rheology and droplet break-up in shear flows. *Physics of Fluids*, AIP Publishing, v. 30, n. 12, p. 122110, 2018.

45   HASSAN, M. R.; WANG, C. Magnetic field induced ferrofluid droplet breakup in a simple shear flow at a low reynolds number. *Physics of Fluids*, AIP Publishing LLC, v. 31, n. 12, p. 127104, 2019.

46   CUNHA, L.; SIQUEIRA, I.; CUNHA, F.; OLIVEIRA, T. Effects of external magnetic fields on the rheology and magnetization of dilute emulsions of ferrofluid droplets in shear flows. *Physics of Fluids*, AIP Publishing LLC, v. 32, n. 7, p. 073306, 2020.

47   PAWLOWSKI, R. P.; SHADID, J. N.; SIMONIS, J. P.; WALKER, H. F. Globalization techniques for newton–krylov methods and applications to the fully coupled solution of the navier–stokes equations. *SIAM review*, SIAM, v. 48, n. 4, p. 700–721, 2006.

48   SPOTZ, W.; CAREY, G. High-order compact scheme for the steady stream-function vorticity equations. *International Journal for Numerical Methods in Engineering*, Wiley Online Library, v. 38, n. 20, p. 3497–3512, 1995.

49   CHORIN, A. J. Numerical solution of the navier-stokes equations. *Mathematics of computation*, v. 22, n. 104, p. 745–762, 1968.

50   GUERMOND, J.-L.; MINEV, P.; SHEN, J. An overview of projection methods for incompressible flows. *Computer methods in applied mechanics and engineering*, Elsevier, v. 195, n. 44-47, p. 6011–6045, 2006.

51   BROWN, D. L.; CORTEZ, R.; MINION, M. L. Accurate projection methods for the incompressible navier–stokes equations. *Journal of computational physics*, Elsevier, v. 168, n. 2, p. 464–499, 2001.

52   KIM, J.; MOIN, P. Application of a fractional-step method to incompressible navier-stokes equations. *Journal of Computational Physics*, v. 59, n. 2, p. 308 – 323, 1985. ISSN 0021-9991. Disponível em: <http://www.sciencedirect.com/science/article/pii/0021999185901482>.

53   BADALASSI, V. E.; CENICEROS, H. D.; BANERJEE, S. Computation of multiphase systems with phase field models. *Journal of computational physics*, Elsevier, v. 190, n. 2, p. 371–397, 2003.

54   DONG, S.; SHEN, J. A time-stepping scheme involving constant coefficient matrices for phase-field simulations of two-phase incompressible flows with large density ratios. *Journal of Computational Physics*, Elsevier, v. 231, n. 17, p. 5788–5804, 2012.

55   DODD, M. S.; FERRANTE, A. A fast pressure-correction method for incompressible two-fluid flows. *Journal of Computational Physics*, Elsevier, v. 273, p. 416–434, 2014.

56   HOU, T. Y.; LOWENGRUB, J. S.; SHELLEY, M. J. Boundary integral methods for multicomponent fluids and multiphase materials. *Journal of Computational Physics*, Elsevier, v. 169, n. 2, p. 302–362, 2001.

57   PESKIN, C. S. The immersed boundary method. *Acta numerica*, Cambridge University Press, v. 11, p. 479–517, 2002.

58   HIRT, C. W.; NICHOLS, B. D. Volume of fluid (vof) method for the dynamics of free boundaries. *Journal of computational physics*, Elsevier, v. 39, n. 1, p. 201–225, 1981.

59   JACQMIN, D. Calculation of two-phase navier–stokes flows using phase-field modeling. *Journal of Computational Physics*, Elsevier, v. 155, n. 1, p. 96–127, 1999.

60   SUSSMAN, M.; SMEREKA, P.; OSHER, S. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational physics*, Elsevier, v. 114, n. 1, p. 146–159, 1994.

61   SUSSMAN, M. A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles. *Journal of Computational Physics*, Elsevier, v. 187, n. 1, p. 110–136, 2003.

62   CENICEROS, H. D.; ROMA, A. M.; SILVEIRA-NETO, A.; VILLAR, M. M. et al. A robust, fully adaptive hybrid level-set/front-tracking method for two-phase flows with an accurate surface tension computation. *Commun Comput Phys*, v. 8, n. 1, p. 51–94, 2010.

63   OSHER, S.; SETHIAN, J. A. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *Journal of computational physics*, Elsevier, v. 79, n. 1, p. 12–49, 1988.

64   OSHER, S.; FEDKIW, R. *Level set methods and dynamic implicit surfaces*. [S.l.]: Springer Science & Business Media, 2006. v. 153.

65   SUSSMAN, M.; FATEMI, E.; SMEREKA, P.; OSHER, S. An improved level set method for incompressible two-phase flows. *Computers and Fluids*, v. 27, n. 5, p. 663 – 680, 1998. ISSN 0045-7930. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0045793097000534>.

66   GIBOU, F.; FEDKIW, R.; OSHER, S. A review of level-set methods and some recent applications. *Journal of Computational Physics*, Elsevier, v. 353, p. 82–109, 2018.

67   ZHU, J.; SETHIAN, J. Projection methods coupled to level set interface techniques. *Journal of computational physics*, Elsevier, v. 102, n. 1, p. 128–138, 1992.

68   KANG, M.; FEDKIW, R. P.; LIU, X.-D. A boundary condition capturing method for multiphase incompressible flow. *Journal of Scientific Computing*, Springer, v. 15, n. 3, p. 323–360, 2000.

69   LI, C.; XU, C.; GUI, C.; FOX, M. D. Level set evolution without re-initialization: a new variational formulation. In: IEEE. *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. [S.l.], 2005. v. 1, p. 430–436.

70   SETHIAN, J. A. Fast marching methods. *SIAM review*, SIAM, v. 41, n. 2, p. 199–235, 1999.

71 DETRIXHE, M.; GIBOU, F.; MIN, C. A parallel fast sweeping method for the eikonal equation. *Journal of Computational Physics*, Elsevier, v. 237, p. 46–55, 2013.

72 ROYSTON, M.; PRADHANA, A.; LEE, B.; CHOW, Y. T.; YIN, W.; TERAN, J.; OSHER, S. Parallel redistancing using the hopf–lax formula. *Journal of Computational Physics*, Elsevier, v. 365, p. 7–17, 2018.

73 PENG, D.; MERRIMAN, B.; OSHER, S.; ZHAO, H.; KANG, M. A pde-based fast local level set method. *Journal of computational physics*, Elsevier, v. 155, n. 2, p. 410–438, 1999.

74 SUSSMAN, M.; FATEMI, E. An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow. *SIAM Journal on scientific computing*, SIAM, v. 20, n. 4, p. 1165–1191, 1999.

75 FERZIGER, J. H.; PERIĆ, M.; STREET, R. L. *Computational methods for fluid dynamics*. [S.l.]: Springer, 2002. v. 3.

76 PATANKAR, S. *Numerical heat transfer and fluid flow*. [S.l.]: Taylor & Francis, 2018.

77 HARLOW, F. H.; WELCH, J. E. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *The physics of fluids*, American Institute of Physics, v. 8, n. 12, p. 2182–2189, 1965.

78 TRYGGVASON, G.; BUNNER, B.; EBRAT, O.; TAUBER, W. Computations of multiphase flows by a finite difference/front tracking method. i. multi-fluid flows. *Lecture Series-von Karman Institute For Fluid Dynamics*, Von Karman Institute for Fluid Dynamics, p. 7–7, 1998.

79 COWARD, A. V.; RENARDY, Y. Y.; RENARDY, M.; RICHARDS, J. R. Temporal evolution of periodic disturbances in two-layer couette flow. *Journal of Computational Physics*, Elsevier, v. 132, n. 2, p. 346–361, 1997.

80 GODLEWSKI, E.; RAVIART, P.-A. *Numerical approximation of hyperbolic systems of conservation laws*. [S.l.]: Springer Science & Business Media, 2013. v. 118.

81 FEISTAUER, M.; FEISTAUER, M.; FELCMAN, J.; FELCMAN, J.; STRAŠKRABA, I. *Mathematical and computational methods for compressible flow*. [S.l.]: Oxford University Press on Demand, 2003.

82 GOTTLIEB, S.; KETCHESON, D. I.; SHU, C.-W. *Strong stability preserving Runge-Kutta and multistep time discretizations*. [S.l.]: World Scientific, 2011.

83 SHU, C.-W.; OSHER, S. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of computational physics*, Elsevier, v. 77, n. 2, p. 439–471, 1988.

84 BREHM, C.; BARAD, M. F.; HOUSMAN, J. A.; KIRIS, C. C. A comparison of higher-order finite-difference shock capturing schemes. *Computers & Fluids*, Elsevier, v. 122, p. 184–208, 2015.

85 ZHAO, G.; SUN, M.; MEMMOLO, A.; PIROZZOLI, S. A general framework for the evaluation of shock-capturing schemes. *Journal of Computational Physics*, Elsevier, v. 376, p. 924–936, 2019.

86   CASTRO, M.; COSTA, B.; DON, W. S. High order weighted essentially non-oscillatory weno-z schemes for hyperbolic conservation laws. *Journal of Computational Physics*, Elsevier, v. 230, n. 5, p. 1766–1792, 2011.

87   JIANG, G.-S.; SHU, C.-W. Efficient implementation of weighted eno schemes. *Journal of computational physics*, Elsevier, v. 126, n. 1, p. 202–228, 1996.

88   JIANG, G.-S.; PENG, D. Weighted eno schemes for hamilton–jacobi equations. *SIAM Journal on Scientific computing*, SIAM, v. 21, n. 6, p. 2126–2143, 2000.

89   OSHER, S.; SHU, C.-W. High-order essentially nonoscillatory schemes for hamilton–jacobi equations. *SIAM Journal on numerical analysis*, SIAM, v. 28, n. 4, p. 907–922, 1991.

90   ROUY, E.; TOURIN, A. A viscosity solutions approach to shape-from-shading. *SIAM Journal on Numerical Analysis*, SIAM, v. 29, n. 3, p. 867–884, 1992.

91   SAAD, Y. *Iterative methods for sparse linear systems*. [S.l.]: siam, 2003. v. 82.

92   YANG, S.; GOBBERT, M. K. The optimal relaxation parameter for the sor method applied to the poisson equation in any space dimensions. *Applied Mathematics Letters*, Elsevier, v. 22, n. 3, p. 325–331, 2009.

93   LIESEN, J.; STRAKOS, Z. *Krylov subspace methods: principles and analysis*. [S.l.]: Oxford University Press, 2013.

94   HESTENES, M. R.; STIEFEL, E. et al. *Methods of conjugate gradients for solving linear systems*. [S.l.]: NBS Washington, DC, 1952. v. 49.

95   VORST, H. A. Van der. Bi-cgstab: A fast and smoothly converging variant of bi-cg for the solution of nonsymmetric linear systems. *SIAM Journal on scientific and Statistical Computing*, SIAM, v. 13, n. 2, p. 631–644, 1992.

96   SAAD, Y.; SCHULTZ, M. H. Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, SIAM, v. 7, n. 3, p. 856–869, 1986.

97   SHEWCHUK, J. R. et al. *An introduction to the conjugate gradient method without the agonizing pain*. [S.l.]: Carnegie-Mellon University. Department of Computer Science, 1994.

98   NOTAY, Y. Flexible conjugate gradients. *SIAM Journal on Scientific Computing*, SIAM, v. 22, n. 4, p. 1444–1460, 2000.

99   BOUWMEESTER, H.; DOUGHERTY, A.; KNYAZEV, A. V. Nonsymmetric preconditioning for conjugate gradient and steepest descent methods. *Procedia Computer Science*, Elsevier, v. 51, p. 276–285, 2015.

100   CONCUS, P.; GOLUB, G. H.; MEURANT, G. Block preconditioning for the conjugate gradient method. *SIAM Journal on Scientific and Statistical Computing*, SIAM, v. 6, n. 1, p. 220–252, 1985.

101   SAN, O.; STAPLES, A. E. A coarse-grid projection method for accelerating incompressible flow computations. *Journal of Computational Physics*, Elsevier, v. 233, p. 480–508, 2013.

102   DODD, M. S.; FERRANTE, A. On the interaction of taylor length scale size droplets and isotropic turbulence. *Journal of Fluid Mechanics*, Cambridge University Press, v. 806, p. 356–412, 2016.

103   COSTA, P. A fft-based finite-difference solver for massively-parallel direct numerical simulations of turbulent flows. *Computers & Mathematics with Applications*, Elsevier, v. 76, n. 8, p. 1853–1862, 2018.

104   FUKA, V. Poisfft–a free parallel fast poisson solver. *Applied Mathematics and Computation*, Elsevier, v. 267, p. 356–364, 2015.

105   HIGHAM, N. J. *Accuracy and stability of numerical algorithms*. [S.l.]: SIAM, 2002.

106   PRESS, W. H.; TEUKOLSKY, S. A.; FLANNERY, B. P.; VETTERLING, W. T. *Numerical recipes in Fortran 77: volume 1, volume 1 of Fortran numerical recipes: the art of scientific computing*. [S.l.]: Cambridge university press, 1992.

107   GHOLAMI, A.; MALHOTRA, D.; SUNDAR, H.; BIROS, G. Fft, fmm, or multigrid? a comparative study of state-of-the-art poisson solvers for uniform and nonuniform grids in the unit cube. *SIAM Journal on Scientific Computing*, SIAM, v. 38, n. 3, p. C280–C306, 2016.

108   TROTTENBERG, U.; OOSTERLEE, C. W.; SCHULLER, A. *Multigrid*. [S.l.]: Elsevier, 2000.

109   NOTAY, Y.; NAPOV, A. A massively parallel solver for discrete poisson-like problems. *Journal of computational physics*, Elsevier, v. 281, p. 237–250, 2015.

110   BRIGGS, W. L.; HENSON, V. E.; MCCORMICK, S. F. *A multigrid tutorial*. [S.l.]: SIAM, 2000.

111   WESSELING, P. *An Introduction to Multigrid Methods*. R.T. Edwards, 2004. (An Introduction to Multigrid Methods). ISBN 9781930217089. Disponível em: <https://books.google.com.br/books?id=RoRGAAAAYAAJ>.

112   MOHR, M.; WIENANDS, R. Cell-centred multigrid revisited. *Computing and Visualization in Science*, Springer, v. 7, n. 3-4, p. 129–140, 2004.

113   KHALIL, M.; WESSELING, P. Vertex-centered and cell-centered multigrid for interface problems. *Journal of Computational Physics*, Elsevier, v. 98, n. 1, p. 1–10, 1992.

114   ASHBY, S. F.; FALGOUT, R. D. A parallel multigrid preconditioned conjugate gradient algorithm for groundwater flow simulations. *Nuclear science and engineering*, Taylor & Francis, v. 124, n. 1, p. 145–159, 1996.

115   TATEBE, O. The multigrid preconditioned conjugate gradient method. In: NASA. *NASA conference publication*. [S.l.], 1993. p. 621–621.

116   MCADAMS, A.; SIFAKIS, E.; TERAN, J. A parallel multigrid poisson solver for fluids simulation on large grids. In: *Symposium on Computer Animation*. [S.l.: s.n.], 2010. p. 65–73.

117   WILLIAMS, S.; WATERMAN, A.; PATTERSON, D. Roofline: an insightful visual performance model for multicore architectures. *Communications of the ACM*, ACM New York, NY, USA, v. 52, n. 4, p. 65–76, 2009.

118   DATTA, K.; KAMIL, S.; WILLIAMS, S.; OLIKER, L.; SHALF, J.; YELICK, K. Optimization and performance modeling of stencil computations on modern microprocessors. *SIAM review*, SIAM, v. 51, n. 1, p. 129–159, 2009.

119   TREIBIG, J.; WELLEIN, G.; HAGER, G. Efficient multicore-aware parallelization strategies for iterative stencil computations. *Journal of Computational Science*, Elsevier, v. 2, n. 2, p. 130–137, 2011.

120   MURANUSHI, T.; MAKINO, J. Optimal temporal blocking for stencil computation. *Procedia Computer Science*, Elsevier, v. 51, p. 1303–1312, 2015.

121   KAMIL, S.; DATTA, K.; WILLIAMS, S.; OLIKER, L.; SHALF, J.; YELICK, K. Implicit and explicit optimizations for stencil computations. In: *Proceedings of the 2006 workshop on Memory system performance and correctness*. [S.l.: s.n.], 2006. p. 51–60.

122   GUIDO, S.; VILLONE, M. Three-dimensional shape of a drop under simple shear flow. *Journal of rheology*, The Society of Rheology, v. 42, n. 2, p. 395–415, 1998.

123   ABICALIL, V. G. Modelagem numérica de escoamentos multifásicos tridimensionais. 2018.

124   PARDIS, N.; EBRAHIMI, R.; KIM, H. S. Equivalent strain at large shear deformation: Theoretical, numerical and finite element analysis. *Journal of applied research and technology*, Elsevier, v. 15, n. 5, p. 442–448, 2017.

125   BATCHELOR, G. The stress system in a suspension of force-free particles. *Journal of fluid mechanics*, Cambridge University Press, v. 41, n. 3, p. 545–570, 1970.

126   BIRD, R. B.; ARMSTRONG, R. C.; HASSAGER, O. Dynamics of polymeric liquids. vol. 1: Fluid mechanics. 1987.

127   JANSSEUNE, T.; MEWIS, J.; MOLDENAERS, P.; MINALE, M.; MAFFETTONE, P. Rheology and rheological morphology determination in immiscible two-phase polymer model blends. *Journal of non-newtonian fluid mechanics*, Elsevier, v. 93, n. 1, p. 153–165, 2000.

128   HYUN, K.; WILHELM, M.; KLEIN, C. O.; CHO, K. S.; NAM, J. G.; AHN, K. H.; LEE, S. J.; EWOLDT, R. H.; MCKINLEY, G. H. A review of nonlinear oscillatory shear tests: Analysis and application of large amplitude oscillatory shear (laos). *Progress in Polymer Science*, Elsevier, v. 36, n. 12, p. 1697–1753, 2011.

129   EWOLDT, R. H. Defining nonlinear rheological material functions for oscillatory shear. *Journal of Rheology*, The Society of Rheology, v. 57, n. 1, p. 177–195, 2013.

130   XIE, Y.; JI, Q. A new efficient ellipse detection method. In: IEEE. *Object recognition supported by user interaction for service robots*. [S.l.], 2002. v. 2, p. 957–960.

131   ROSENSWEIG, R. E. *Ferrohydrodynamics*. [S.l.]: Courier Corporation, 2013.