

Modeling particulate complex flows using XFEM

Citation for published version (APA):

Choi, Y. J. (2011). *Modeling particulate complex flows using XFEM*. [Phd Thesis 1 (Research TU/e / Graduation TU/e), Mechanical Engineering]. Technische Universiteit Eindhoven. <https://doi.org/10.6100/IR716491>

DOI:

[10.6100/IR716491](https://doi.org/10.6100/IR716491)

Document status and date:

Published: 01/01/2011

Document Version:

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

Please check the document version of this publication:

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

[Link to publication](#)

General rights

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

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

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Modeling Particulate Complex Flows using XFEM

Modeling Particulate Complex Flows using XFEM / by Young Joon Choi.
Technische Universiteit Eindhoven, 2011.

A catalogue record is available from the Eindhoven University of Technology Library.
ISBN: 978-90-386-2690-1
Subject heading: extended finite element method, XFEM, temporary arbitrary Lagrangian-Eulerian scheme, particulate flows, viscoelastic fluids, two-phase flows

This thesis was prepared with the L^AT_EX 2_ε documentation system.
Reproduction: University Press Facilities, Eindhoven, The Netherlands.
Cover design: Young Joon Choi.

This research forms part of the research programme of the Dutch Polymer Institute (DPI),
Performance Polymers, DPI project #616.

Modeling Particulate Complex Flows using XFEM

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
rector magnificus, prof.dr.ir. C.J. van Duijn, voor een
commissie aangewezen door het College voor
Promoties in het openbaar te verdedigen
op dinsdag 18 oktober 2011 om 16.00 uur

door

Young Joon Choi

geboren te Daegu, Zuid-Korea

Dit proefschrift is goedgekeurd door de promotor:

prof.dr.ir. H.E.H. Meijer

Copromotoren:

dr.ir. M.A. Hulsen

en

dr.ir. P.D. Anderson

For my parents

Contents

Summary	xi
1 Introduction	1
1.1 Introduction	1
1.2 Review of particulate flows	1
1.2.1 Rotation of a particle in shear flow	2
1.2.2 Cross-stream lateral migration of a single particle	2
1.2.3 Shear-induced migration of concentrated suspensions	5
1.2.4 Microstructure formation of particles	5
1.3 Review of numerical methods for particulate flows	6
1.4 Objective and outline	7
2 Flow around a stationary cylinder	9
2.1 Introduction	10
2.2 Governing equations	11
2.3 Numerical methods	13
2.3.1 Mixed DEVSS-G/SUPG finite element formulation	13
2.3.2 The extended finite element method	14
2.3.3 Time integration	22
2.3.4 Comparison with other numerical methods	23
2.4 Convergence analysis	24
2.5 Results	26
2.5.1 Problem description	26
2.5.2 Accuracy of the subdomain integration	27
2.5.3 Drag coefficient	28
2.5.4 Stress along the cylinder surface	29
2.6 Conclusions	32
3 Particle migration in circular Couette flow	35
3.1 Introduction	36
3.2 Governing equations	37
3.3 Numerical methods	39
3.3.1 Mixed DEVSS-G/SUPG finite element formulation	39
3.3.2 The extended finite element method	40

3.3.3	Application to moving particles	44
3.3.4	Mesh refinements using a grid deformation method	48
3.3.5	Time integration	49
3.4	Validation	51
3.5	Results	53
3.5.1	Problem description	53
3.5.2	Mesh convergence	53
3.5.3	Initial position of a particle	57
3.5.4	Weissenberg number	57
3.5.5	Mobility parameter α	60
3.5.6	Particle size	61
3.6	Conclusions	63
4	Dynamics of particles in two-phase flows	65
4.1	Introduction	66
4.2	Mathematical formulation	67
4.2.1	Scaling of the equations	69
4.2.2	Rigid-body motion of particles	69
4.2.3	Partial wetting boundary conditions	70
4.3	Numerical methods	71
4.3.1	Weak form	71
4.3.2	Time discretization of the diffuse-interface model	72
4.3.3	XFEM formulation	72
4.3.4	Temporary ALE scheme	73
4.3.5	Time integration	75
4.4	Particle at a fluid-fluid interface	76
4.4.1	Problem description	76
4.4.2	Convergence test	78
4.4.3	Time duration of applied external force	80
4.4.4	Interfacial thickness	81
4.4.5	Surface tension	81
4.4.6	Particle size	83
4.4.7	Viscosity ratio	84
4.5	Multiple-layer configuration	84
4.6	Conclusions	86
5	Alignment of particles in confined shear flow	91
5.1	Introduction	93
5.2	Governing equations	94
5.3	Numerical methods	96
5.3.1	Mixed DEVSS-G/SUPG finite element formulation	96
5.3.2	A brief description of XFEM	99
5.3.3	Temporary ALE scheme for moving particle problems	100
5.3.4	Time integration	103

5.4	Convergence analysis of the temporary ALE scheme	105
5.5	Validation	106
5.6	Interaction of two particles	107
5.6.1	Problem description	107
5.6.2	Particle motion in Stokes flow	108
5.6.3	Particle motion in a viscoelastic fluid	109
5.7	Interaction of multiple particles	111
5.7.1	Convergence check	111
5.7.2	Alignment of three particles	112
5.7.3	Wall confinement	118
5.7.4	Viscosity ratio	120
5.7.5	Alignment of multiple particles	121
5.8	Conclusions	125
6	Particle migration in extrusion flow	127
6.1	Introduction	128
6.2	Governing equations	129
6.3	Numerical methods	131
6.3.1	Weak form	131
6.3.2	A brief description of XFEM	133
6.3.3	Temporary ALE scheme for moving boundary problems	134
6.3.4	Time integration	137
6.4	Extrudate swell	139
6.5	Particle migration in extrusion flow	141
6.5.1	Problem description	141
6.5.2	Mesh convergence	142
6.5.3	Particle motion	142
6.5.4	Weissenberg number	145
6.5.5	Particle size	145
6.6	Conclusions	146
7	Conclusions and prospects	149
7.1	Conclusions	149
7.2	Prospects	151
	References	153
	Samenvatting	165
	Acknowledgements	167
	Curriculum Vitae	169

Summary

Particulate flows arise in a wide class of research areas and industrial processes, for example, fluidized suspensions, electrophoresis, filtration, slurry transport, materials separation, rate of mixing enhancement, filled polymers, etc. In many of the applications cited, the fluid phase displays complex non-Newtonian flow behavior. In order to study the particle motion in complex fluids such as viscoelastic fluids, a numerical analysis is an essential requirement due to inherent nonlinear behavior of the fluids. In this thesis, we are aiming to develop a novel numerical scheme to simulate particulate complex flows.

If particles are suspended in a fluid, the field variables such as pressure and stress are discontinuous at the interface since no flow occurs inside the particle. The most intuitive method is using a boundary-fitted mesh. However, the generation of a boundary-fitted mesh for a complex geometry is still a challenging task, especially in three-dimensional simulations. An alternative way is using the fictitious domain method. The basic idea of this method is to fill every particle domain with the surrounding fluid, assuming and subsequently prescribing that the fluid inside the particle domain moves like a solid object. However, the motion of fluid near the interface is affected by the false mass or viscous material response from the fictitious fluid inside the particle. To overcome this problem, we incorporate an extended finite element method (XFEM) that can decouple the physical and fictitious domain completely to capture discontinuities. We develop a method that can provide the accuracy of boundary-fitted mesh solutions without any need of remeshing for the simulation of freely moving particles in complex flows. We apply the proposed XFEM to various particulate flow problems.

In Chapter 2, the flow of a viscoelastic fluid around a stationary cylinder is presented. The method is verified by comparing the solutions with those of simulations using a boundary-fitted mesh. The results are also compared with those obtained by using fictitious domain methods. Our method shows a significant improvement of local accuracy around the rigid body compared with the fictitious domain method, obtaining solutions similar to those of boundary-fitted mesh solutions.

In Chapter 3, the particle migration in circular Couette flow of a Giesekus fluid is presented. The particle migrates to a stabilized radial position near the outer cylinder regardless

of its initial position. As the Weissenberg number increases, the particle migrates more rapidly towards the outer cylinder, and the stabilized radial position of the particle shifts towards the outer cylinder. With increasing particle size, the particle migrates more rapidly towards the outer cylinder.

In Chapter 4, a model for the dynamics of particles suspended in two-phase flows is presented by coupling the Cahn-Hilliard theory with the extended finite element method. To demonstrate and validate the technique, the dynamics of a single particle at a fluid-fluid interface is studied. In particular, we are interested in the effect of interfacial thickness, surface tension, particle size and viscosity ratio of the two fluids on the particle movement. We also show the movement of a particle passing through multiple layers of fluids to demonstrate the wide applicability of the method.

In Chapter 5, the alignment of particles in confined shear flow of a viscoelastic fluid is quantitatively analyzed. The maximum obtainable length of a string of particles is limited for a certain fluid rheology. As the fluid elasticity increases, particles can form longer strings. Once particles form a string, the final state is independent of the initial particle positions and the histories to reach the steady-state. Moderate wall confinement promotes the alignment of particles, however, too strong confinement hinders the alignment by enhancing repulsive interaction between particles. The steady-state angular velocities of particles are also compared with respect to the length of strings.

In Chapter 6, the swell ratios of an upper-convected Maxwell fluid for various Deborah numbers are compared with those of the moving boundary-fitted mesh problems of the conventional ALE technique to validate the method with the presence of the free surface. The proposed XFEM combined with the temporary ALE scheme can provide similar accuracy to the boundary-fitted mesh problems for low Deborah numbers. For high Deborah numbers, the method seems to be more stable for the extrusion problem. We also investigate the migration of a single particle in extrusion flow. The presence of the particle disturbs the stress distribution and the free surface profile significantly. We found that the particle moves away from the free surface as the Weissenberg number increases. As the particle size increases, the particle moves towards the free surface.

Finally, in Chapter 7 the conclusions of this thesis are outlined together with recommendations for future research.

Chapter 1

Introduction

1.1 Introduction

The presence of particles in flowing fluids is ubiquitous both in nature and in technology, for example, air, blood, paints, cosmetics, nanocomposites, pharmaceuticals, and filled polymers. More examples are well summarized in the book by Chhabra [1]. In many of the applications involving particulate flows, the suspending fluids display complex non-Newtonian, e.g. viscoelastic and/or shear-thinning, flow behavior. Adding particles in a complex fluid further complicates the flow behavior.

In particulate flows, the spatial distribution of particles plays a dominant role on the apparent properties of a suspension such as viscosity, normal stress, heat conductivity, electric conductivity, magnetic permeability, and so on. Usually the distribution of particles in fluid flow is not uniform even if initially particles are distributed uniformly. In this chapter, the state of the art on the motion of particles suspended in Newtonian and viscoelastic fluids is reviewed. Experimental, theoretical, and numerical observations are addressed in the following sections.

1.2 Review of particulate flows

In this section, an overview of the motion of particles in Newtonian and viscoelastic fluids is presented: rotation of a particle in shear flow in Sec. 1.2.1, lateral migration of a particle in shear, Couette and Poiseuille flows in Sec. 1.2.2, particle migration in concentrated suspensions of Newtonian fluids in Sec. 1.2.3, and structure formation of particles in viscoelastic fluids in Sec. 1.2.4. In this thesis, we only consider rigid circular, cylindrical in 2D and spherical in 3D, particles suspended in laminar flows.

1.2.1 Rotation of a particle in shear flow

If a freely rotating particle is sheared in an unbounded Stokes flow, the angular velocity of the particle is half of the shear-rate [2, 3]. Cox et al. [4] studied the streamlines around a cylinder and a sphere in shear flow. In both cases, there are open and closed streamlines separated by a limiting streamline.

The effect of inertia on the angular velocity of a cylinder [5] and a sphere [6] was studied theoretically, only valid for the Reynolds number $Re < 0.1$. The flow around a freely rotating particle in Newtonian shear flow was more rigorously studied numerically [7] and experimentally [8]. It was shown that the inclusion of inertia leads to a lower angular velocity.

The effect of fluid elasticity on the rotation of a particle in shear flow was studied numerically [9] and experimentally [10]. They also provided a quantitative comparison between experimental and numerical results [11]. The viscoelasticity of the suspending fluid leads to a substantial slowing down of the angular velocity of the particle, compared to the Newtonian case. It was found that the extent of the slowing down of the angular velocity mainly depends on the ratio of the first normal stress difference to the shear stress, not on the details of the fluid rheology. This indicates that the normal stress is primarily responsible for the reduction of the angular velocity.

The effect of wall confinement on the rotation of a particle is presented by D'Avino et al. [12]. The presence of the wall slows down the particle rotation. The stronger the confinement, the lower is the angular velocity. Shear-thinning leads to a slower rotation in a highly confined geometry, contrary to an unbounded domain where shear-thinning leads to a faster rotation.

1.2.2 Cross-stream lateral migration of a single particle

Inertia-induced migration in Newtonian fluids

If a neutrally buoyant particle is suspended in a Poiseuille flow through a circular tube or in the pressure-driven parabolic flow between two parallel plates, the particle will move perpendicular to the flow direction until it reaches an equilibrium position in the midway between the centerline and the wall. This is one of the best known experimental results for particle motion in viscous flows, which is commonly called the Segré-Silberberg effect after their first experimental observations [13, 14, 15]. The lateral migration of the particle is caused by the effect of inertia and has been studied theoretically based on perturbation theories [16, 17, 18]. For example, Ho and Leal [17] studied the lateral force on a neutrally buoyant rigid sphere in a Newtonian fluid theoretically. A stable equilibrium position for the sphere in a simple shear flow between two parallel plane walls is the centerline, where the lateral force is zero. For the case of two-dimensional Poiseuille flow, there are three positions where the lateral force is zero; the centerline, which is unstable to slight perturbations, and 60% of the channel half-width from the centerline, which are stable equilibrium positions. Note that the particle can not move laterally in Stokes flow due to its linear property [19, 20].

The inertial effect on the lateral migration of a particle is also investigated numerically

by Feng et al. [21]. They studied both neutrally buoyant and non-neutrally buoyant particles. Note that, in their simulations, the direction of gravity is opposite to the main flow direction for non-neutrally buoyant particles. In a shear flow between two plane walls, a neutrally buoyant particle always migrates to the centerline between two walls, regardless of initial position and velocity. The motion of a non-neutrally buoyant particle depends on the magnitude of the buoyancy. If the density difference is small, the particle stabilizes at a position close to either of the walls. If the density difference is large enough, the equilibrium position shifts towards the centerline, irrespective of whether the particle is lighter or heavier than the fluid. For a particle in a plane Poiseuille flow, the Segré-Silberberg effect is realized, with the equilibrium position a little outside the midpoint between the wall and the centerline. If the velocity of the channel flow is increased, i.e. the Reynolds number is increased, the particle migrates faster and the final equilibrium position moves closer to the wall. If the particle is slightly heavier than the fluid, the particle moves to an equilibrium position near the centerline. If the particle is slightly lighter than the fluid, the equilibrium position is near the wall. If the density difference is large enough, the equilibrium position shifts towards the centerline, irrespective of whether the particle is lighter or heavier than the fluid, similar to the case of shear flow. Three mechanisms are identified as responsible for the lateral migration in a shear flow: wall repulsion, inertial lift related to shear slip and a lift due to particle rotation. For a Poiseuille flow, an additional mechanism is a lift force due to the curvature of the undisturbed velocity profile.

A heavy particle sedimenting in a Newtonian fluid bounded by two plane walls is also investigated theoretically [22] and numerically [23]: the particle migrates away from the walls until it reaches an equilibrium position at the center of the channel.

Viscoelasticity-induced migration

Karnis and Mason [24] experimentally studied the migration of a sphere in a viscoelastic fluid both in Couette and Poiseuille flows. In Couette flow between concentric rotating cylinders, migration occurred towards the outer cylinder wall. The rate of migration increased with increasing particle radius and increasing the apparent viscosity gradient; and decreased with increasing the distance of the particle center from the axis of rotation of the concentric cylinders. For Poiseuille flow in a circular tube, the rigid sphere migrated from the wall to the tube center at which the velocity gradient is zero. The rate of migration increased with increasing particle radius and radial distance from the tube axis. Later, the migration of particles in viscoelastic fluids has been studied experimentally in Couette flow [25, 26], Poiseuille flow [27, 28], torsional shear [29], and oscillatory shear [30]. Interestingly, Snijkers [31] observed no migration of a particle in a highly elastic Boger fluid, contrary to a shear-thinning viscoelastic fluid. Ho and Leal [32] theoretically studied the lateral migration of a neutrally buoyant rigid sphere suspended in a second-order fluid. They demonstrated that the migration induced by normal stresses occurs in the direction of decreasing shear rate, which is towards the centerline for a plane Poiseuille flow and towards the outer cylinder wall for Couette flow. Thorough reviews, both theoretical and experimental results, on the motion of particles in viscoelastic fluids can be found in [33, 34, 35, 36].

Huang et al. [37] reported the results of direct numerical simulations of the motion of a two-dimensional circular cylinder in shear flow and in Poiseuille flow of an Oldroyd-B fluid. They investigated the effects of elasticity, inertia, blockage ratio of the channel and shear-thinning on the equilibrium position of the particle. In [37], a shear-thinning viscosity is directly added to the Oldroyd-B model by using the Carreau-Bird viscosity law. Elasticity gives rise to the normal stresses on the particle and attracts it to the wall in shear flow. In the absence of inertia, the particle migrates until it touches the nearby wall. If the blockage ratio is small, the particle in Poiseuille flow moves to the center of the channel, where the shear rate of the undisturbed flow is zero. If the blockage ratio is large enough, the particle moves towards the sidewall since the effect of the curvature of the inflow velocity profile is overwhelmed by the effect of the presence of the wall. As the Reynolds number increases, the particle is pushed further away from the sidewall. The final equilibrium position of the particle is determined by a competition between inertia and normal stresses. Shear-thinning strengthens the particle movement towards the wall both in shear flow and Poiseuille flow. Huang et al. [37] also considered non-neutrally buoyant particles in shear flow and Poiseuille flow, and Feng et al. [38] simulated the sedimentation of particles in a vertical channel filled with an Oldroyd-B fluid. However, non-neutrally buoyant particles and the sedimentation of particles will not be covered in this chapter.

The migration of a single particle in confined shear flow is also investigated by D'Avino et al. using 2D [39] and 3D [40] direct numerical simulations, in the absence of inertia. Their 2D and 3D simulations produce the same phenomenology in a qualitative way. They reported the particle migration towards the closest wall regardless of its starting position, fluid and geometrical parameters, which is in good agreement with the previous experimental [30] and numerical [37] observations. The main finding of their work is that a master curve is obtained for the migration velocity in the whole channel for a given set of geometrical and fluid parameters. As a consequence, the particle trajectories collapse on a single curve regardless of the starting position after initial transients. Three regions in the channel are characterized by different particle dynamics. If the particle is close the channel centerline, the migration velocity is found to be linear with the particle vertical position. Approaching the wall, the migration becomes faster, whereas, very close to the wall, the migration velocity abruptly decreases. Villone et al. [41] applied the same numerical method to the particle migration in Poiseuille flow as well. They found a master curve of particle trajectories for different initial positions, similar to the case of shear flow. There are two symmetric separatrices in the channel. A particle starting between the separatrix and the channel centerline moves towards the channel axis, whereas a particle starting in the region between the separatrix and the wall moves to the wall. Stronger confinements move the separatrix position towards the channel axis, i.e. increasing the channel portion where the particle migrates to the wall. An interesting observation is that, as the particle approaches the wall, the translational velocity in the flow direction is found to become equal to the linear velocity corresponding to the rolling motion over the wall without slip. If the particle size is comparable to the channel gap, the separatrix coincides with the centerline, i.e. the particle migrates towards the wall regardless of its initial position; if the particle is very small, the attraction towards channel

axis extends to the whole channel, which are in qualitative agreement with the observation of Huang et al. [37].

1.2.3 Shear-induced migration of concentrated suspensions

Particles in concentrated suspensions, which usually means that the volume fraction of particles is higher than 30%, migrate from the regions of higher particle concentration to the regions of lower concentration, and from the regions of higher shear rates to the regions of lower shear rates. It is often referred to as a shear-induced migration. Karnis et al. [42] reported blunted velocity profiles for suspensions flowing through a tube, however, the particle volume fraction profiles were not quantitatively determined. Indeed, the occurrence of such blunting velocity profiles in the flow of concentrated suspensions was due to the migration of particles towards the center of the tube, where the shear rate is lowest. Eckstein et al. [43] determined self-diffusion coefficients experimentally for lateral dispersion of spherical and disk-like particles in a concentric Couette apparatus. Later, Galada-Maria and Acrivos [44] reported that the apparent shear viscosity of a concentrated suspension of neutrally buoyant spheres flowing in a Couette viscometric device decreases with time after long period of shearing. Leighton and Acrivos [45] demonstrated that this decrease of shear viscosity is due to the shear-induced migration of particles out of the sheared Couette gap, which reduces the particle concentration in the Couette gap. Nuclear magnetic resonance (NMR) imaging was used to observe the evolution of radial concentration and velocity profiles of initially well-mixed concentrated suspensions of spheres in viscous Newtonian liquid undergoing flow between rotating concentric cylinders [46, 47] and eccentric cylinders [48]. The shear-induced particle migrations can be explained in terms of the irreversible interparticle interactions that occur in suspensions [45, 49]. Phillips et al. [49] proposed a phenomenological constitutive equation for computing particle concentration and velocity fields in concentrated suspensions. Pan and Glowinski [50] reported direct numerical simulations of the motion of concentrated suspensions in a pressure-driven Poiseuille flow: particles concentrate in the central region of the low shear rate and the distributions of the horizontal speeds are a little bit flat around the center of the channel, well agreed with the experimental observations. They also presented a power law for the horizontal speed of the suspension.

1.2.4 Microstructure formation of particles

If particles are added to polymeric fluids, microstructures of multiple particles can be induced by flow, which generally affects apparent properties of suspension such as viscosity, normal stress, etc. In particular, Michele et al. [51] reported alignment of particles in a confined oscillatory shear flow of a shear-thinning viscoelastic fluid, only if the ratio of the normal stress difference to the shear stress exceeds a certain critical value. They also observed that the rotation of particles almost ceased if the particles are lined up. The phenomenon is usually called alignment, string formation or chaining of particles. Later, Lyon et al. [52] confirmed the observation of Michele et al. [51], and reported a quantitative reduction in the shear stress owing to the string formation, which shows a relevant link between the rheological

properties of suspensions and the formation of microstructures. Won and Kim [53] reported that shear-thinning plays an important role in the formation of a string-like structure, even though the driving force for the movement across the main flow direction is the first normal stress difference. Particles are not aligned in non-shear-thinning viscoelastic liquids under shear flow. Also, Scirocco et al. [54] observed no alignment in highly elastic Boger fluids, which supports that shear-thinning is an essential factor in string formation. Moreover, they demonstrated that the string formation occurs in the bulk of the liquid rather than at the walls. Walls seemed to hinder alignment of particles. Contrary to [54], Pasquino et al. [55] observed alignment only at the walls, after having migrated from the bulk to the walls. The alignment may depend on the details of the rheology of the suspending fluid and the size of the particles relative to the gap thickness. Hwang and Hulsen [56] qualitatively analyzed the alignment of particles in a sliding bi-periodic frame using a direct numerical simulation. They also observed a significant reduction in average angular velocity of particles as they form a string. In colloidal suspensions, the imposition of flow can also induce the formation of microstructures [57], however, we will not consider colloidal effects in this thesis.

1.3 Review of numerical methods for particulate flows

In order to study the particle motion in complex fluids such as viscoelastic fluids, a numerical analysis is an essential requirement due to inherent nonlinear behavior of the fluids. In this section, an overview of numerical methods for the simulation of particulate flows is presented.

In particulate flows, field variables such as pressure and stress are discontinuous over the interface since no flow occurs inside particles. The most intuitive method to tackle this problem is using a mesh that is boundary-fitted, which means that the interface is aligned with element boundaries [58, 59, 60]. In this method, the governing equations are solved only in the fluid domain, taking into account the interface conditions on the boundaries of particles. Since the problems are solved only in the fluid domain, the solutions automatically contain discontinuous characteristics at the interface. Hence, no further considerations to handle jumps over the interface are necessary. To handle moving particles, this approach incorporates the arbitrary Lagrangian-Eulerian (ALE) technique [61] that uses a moving mesh scheme. The generation of a new mesh is needed if the old mesh becomes too distorted, and the solution must be projected onto the new mesh. The generation of boundary-fitted meshes is, however, still a challenging task in view of algorithms needed and computational costs if complex geometries are involved, especially in three-dimensional simulations.

An alternative approach is the fictitious domain method developed by Glowinski et al. [62, 63, 64]. The basic idea of this method is to fill each domain of every particle with the surrounding fluid, assuming and subsequently prescribing that the fluid inside the particle domain moves like a solid object. Hence, the problem is transferred from a geometrically complex fluid domain to a simpler domain including both fluid and particles, which eliminates the need of remeshing. In this method, particles move in a Lagrangian sense on a fixed Eulerian mesh. However, the motion of fluid near the interface is affected by the false mass,

or the viscous material response, from the fictitious fluid inside the particle. Field variables at the interface are interpolated between the values possessed by the physical fluid and the fictitious fluid and, hence, discontinuities required at the interface are lost.

Some of applications of these numerical methods are already discussed in the previous Section 1.2. For example, the ALE mesh movement scheme is applied to sedimentation of particles in a Newtonian fluid [23] and in an Oldroyd-B fluid [38], particle migration in a Newtonian fluid [21] and in a viscoelastic fluid [37, 39, 40, 41], and rotation of a particle in a viscoelastic fluid [9, 11, 12]. The ALE scheme is also applied to lift-off of a particle in Newtonian and viscoelastic fluids [65, 66, 67, 68]. The fictitious domain method is applied to sedimentation of particles [64, 69, 70, 71, 72], lateral migration of particles [50, 71], particle suspension in sliding bi-periodic frames [73, 74], flow-induced crystallization of particle-filled polymers [75], concentrated suspensions in an elongational flow [76, 77], structure formation of particles in a viscoelastic fluid [56], and the motion of magnetic particles suspended in a non-magnetic Newtonian fluid [78, 79]. Yang et al. [80] compared the ALE scheme and the fictitious domain method for the lateral migration of a sphere in tube flow.

Other numerical methods that are applied to particulate flows are Stokesian dynamics [81], immersed boundary method [82, 83], finite volume method [84, 85, 86], and lattice Boltzmann method [87, 88]. However, in this thesis, we will not cover these methods in detail.

1.4 Objective and outline

In this thesis, we are aiming to develop a novel numerical scheme to simulate particulate complex flows, which can provide the accuracy of boundary-fitted mesh solutions, however, without any need of remeshing. To achieve the objective, we incorporate an extended finite element method (XFEM) that can decouple the physical and fictitious domain completely to capture discontinuities.

The content of the thesis is as follows. In Chapter 2, the flow of a viscoelastic fluid around a stationary cylinder is presented to demonstrate the basic ideas of an extended finite element method. The accuracy and convergence of the method has been verified by comparing with boundary-fitted mesh problems. In Chapter 3, the extended finite element method is coupled with a temporary arbitrary Lagrangian-Eulerian scheme to cope with moving particles, and applied to particle migration in circular Couette flow of a viscoelastic fluid. In Chapter 4, a model for the dynamics of particles suspended in two-phase flows is presented by coupling the Cahn-Hilliard theory with the extended finite element method. In Chapter 5, the alignment of particles in confined shear flow of a viscoelastic fluid is quantitatively analyzed. Also, a weak boundary condition is developed for the imposition of the no-slip boundary condition on the particle surface. In Chapter 6, the extended finite element method is applied to flows with the presence of a free surface. The swell ratio of an upper-convected Maxwell fluid is compared with those of the moving boundary-fitted mesh problems to validate the method. The migration of a particle near the free surface of an extrusion flow is investigated by using a combined extended finite element method both for the particle and the free surface. Finally, in Chapter 7 the conclusions of this thesis are outlined together with recommendations for

future research.

Chapter 2

Flow around a stationary cylinder

We present direct numerical simulations for the flow of an Oldroyd-B fluid around a stationary cylinder using an extended finite element method (XFEM) combined with the DEVSS-G/SUPG formulation. In this method, the finite element shape functions are extended through the partition of unity method (PUM) by using virtual degrees of freedom as enrichment for the description of discontinuities across an interface. For the whole computational domain, including both fluid and rigid body, we use a regular mesh which is not boundary-fitted. The fluid domain and the rigid body domain are fully decoupled by using the XFEM enrichment procedures. The no-slip boundary condition on the interface between fluid and rigid body is realized by using constraints implemented with Lagrangian multipliers. The accuracy and convergence are verified by comparing the solutions with those of simulations using a boundary-fitted mesh. The results are also compared with those obtained by using fictitious domain methods. Our method shows a significant improvement of local accuracy around the cylinder when compared with the fictitious domain method, obtaining solutions similar to those of boundary-fitted mesh solutions.

The content of this Chapter is based on:

Young Joon Choi, Martien A. Hulsen, and Han E.H. Meijer. Simulation of the flow of a viscoelastic fluid around a stationary cylinder using an extended finite element method. *Computers & Fluids*, under review.

2.1 Introduction

We present numerical simulations of the flow of a viscoelastic fluid around a stationary rigid body. In fluid-rigid body systems, field variables such as pressure and stress are discontinuous over the interface since no flow occurs inside the rigid body. The most intuitive method to tackle this problem is using a mesh that is boundary-fitted, which means that the interface is aligned with element boundaries [58, 59]. In this classical method, the governing equations are solved only in the fluid domain, taking into account the interface conditions on the boundaries of the rigid body. Since the problems are solved only in the fluid domain, the solutions automatically contain discontinuous characteristics at the interface. Hence, no further considerations to handle jumps over the interface are necessary. The generation of boundary-fitted meshes is, however, still a challenging task in view of algorithms needed and computational costs if complex geometries are involved, especially in three-dimensional simulations.

An alternative approach is the fictitious domain method developed by Glowinski et al. [62, 63]. The basic idea of the method is to fill each domain of every rigid body with the surrounding fluid, assuming and subsequently prescribing that the fluid inside the rigid body domain moves like a solid object. Hence, the problem is transferred from a geometrically complex fluid domain to a simpler domain including both fluid and rigid bodies. In this method, the motion of fluid near the interface is affected by the false mass, or the viscous material response, from the fictitious fluid inside the rigid body. Field variables at the interface are interpolated between the values possessed by the physical fluid and the fictitious fluid and, hence, discontinuities required at the interface are lost.

Recently, a new numerical method has been developed to capture discontinuous characteristics at the interface while still using a mesh which is not boundary-fitted. The method is based on the partition of unity method (PUM), which is a generalization of the standard Galerkin finite element method. The basic ideas and the mathematical foundations of the partition of unity method were introduced by Babuška and Melenk [89] and the method represents a powerful technique to integrate locally defined approximations into a global approximation. In the original partition of unity method, the approximation spaces are assumed to be continuous, while the use of discontinuous functions within a partition of unity method was introduced for solving crack growth problems without the need of remeshing [90, 91]. The extension, or application, of the partition of unity method to arbitrary discontinuities is usually called the extended finite element method (XFEM), because the finite element shape functions are extended, or enriched, by using additional degrees of freedom to handle discontinuities. Although the extended finite element method was originally introduced for the simulations of cracks in solids, applications to flow problems have been proposed as well. Wagner et al. [92] applied XFEM to problems of rigid particles in Stokes flows using analytic solutions as the partition of unity enrichment. Gerstenberger and Wall [93] applied XFEM to problems of fluid-structure interaction (FSI) in Newtonian fluids using a Heaviside function as the enrichment. More applications of XFEM, demonstrating its history of development, can be found in [94]. Here we interpret the XFEM enrichment scheme using virtual degrees of

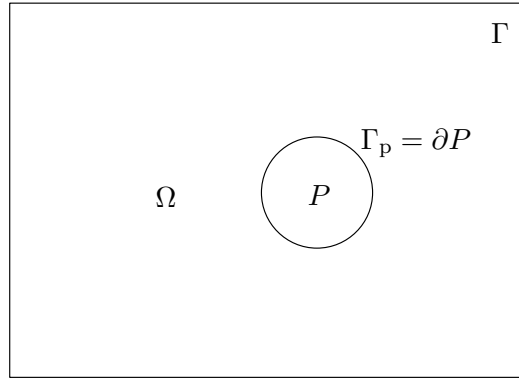


Figure 2.1: Schematic description of the fluid-rigid body system.

freedom and the method is applied to viscoelastic flows around a stationary cylinder confined between two parallel walls, which is a typical benchmark problem to investigate accuracy and stability of viscoelastic fluid flow simulations.

The content of this chapter is as follows. In Section 2.2, we give a brief review of the governing equations for the motion of incompressible viscoelastic fluids. In Section 2.3, we present the numerical algorithm of the extended finite element method combined with the DEVSS-G/SUPG formulation. Section 2.4 contains a convergence analysis by solving a Stokes problem with a known exact solution. In Section 2.5, we present numerical results for the flow around a stationary cylinder with the Oldroyd-B model. The accuracy of the method proposed is compared with boundary-fitted mesh problems and fictitious domain methods. A discussion follows in Section 2.6.

2.2 Governing equations

We consider the motion of an incompressible viscoelastic fluid around a stationary rigid body, see Fig. 2.1, realizing that the single body problem investigated is easily extended to a multiple body problem. Let Ω be the entire domain, including fluid and rigid body, and let P be the embedded domain of the rigid body. Boundaries are denoted by $\Gamma = \partial\Omega$ and $\Gamma_p = \partial P$. We will neglect inertia and body forces, which is common in viscoelastic fluid simulations, especially for polymer melts.

The equations of motion for an incompressible fluid are given by

$$\nabla \cdot \boldsymbol{\sigma} = \mathbf{0} \quad \text{in } \Omega \setminus P, \quad (2.1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \setminus P, \quad (2.2)$$

where $\boldsymbol{\sigma}$ is the stress tensor and \mathbf{u} is the velocity vector. Eqs. (2.1) and (2.2) are the balance equations of momentum and mass, respectively. For a viscoelastic fluid, the total stress $\boldsymbol{\sigma}$ can be split into a pressure part, a Newtonian viscous stress and a polymer stress:

$$\boldsymbol{\sigma} = -p\mathbf{I} + 2\eta_s\mathbf{D} + \boldsymbol{\tau}(\mathbf{c}), \quad (2.3)$$

where p is the pressure, \mathbf{I} is the unity tensor, η_s is the Newtonian solvent viscosity, $\mathbf{D}(\mathbf{u}) = (\nabla\mathbf{u} + (\nabla\mathbf{u})^T)/2$ is the rate-of-deformation tensor and $\boldsymbol{\tau}(\mathbf{c})$ is the polymer stress which can be represented as a function of the conformation tensor \mathbf{c} . We will use the Oldroyd-B constitutive model for the conformation tensor \mathbf{c} :

$$\lambda \overset{\nabla}{\mathbf{c}} + \mathbf{c} - \mathbf{I} = \mathbf{0}, \quad (2.4)$$

where λ is the relaxation time. The triangle (∇) denotes the upper-convected time derivative, defined as:

$$\overset{\nabla}{\mathbf{c}} = \frac{\partial \mathbf{c}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{c} - (\nabla \mathbf{u})^T \cdot \mathbf{c} - \mathbf{c} \cdot \nabla \mathbf{u}. \quad (2.5)$$

In the Oldroyd-B model, the polymer stress can be written as:

$$\boldsymbol{\tau}(\mathbf{c}) = \frac{\eta_p}{\lambda} (\mathbf{c} - \mathbf{I}), \quad (2.6)$$

where η_p is the polymer viscosity. Furthermore, the equation for the conformation tensor \mathbf{c} , Eq. (2.4), can be transformed to an equivalent log-conformation equation for $\mathbf{s} = \log \mathbf{c}$:

$$\frac{\partial \mathbf{s}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{s} - \mathbf{g}((\nabla \mathbf{u})^T, \mathbf{s}) = \mathbf{0}. \quad (2.7)$$

Once \mathbf{s} has been solved, the conformation tensor \mathbf{c} can be computed from $\mathbf{c} = \exp(\mathbf{s})$. A detailed explanation of the log-conformation representation (LCR) can be found in [95, 96]. Solving the equation for \mathbf{s} instead of the equation for \mathbf{c} leads to major stability improvements for high Weissenberg number problems. For all numerical simulations in this chapter, we use the log-conformation representation. Hence, we will present the weak formulation, and the initial and boundary conditions only in the LCR.

The boundary conditions are given by:

$$\mathbf{u} = \bar{\mathbf{u}}(t) \quad \text{on } \Gamma_D, \quad (2.8)$$

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \mathbf{t} \quad \text{on } \Gamma_N, \quad (2.9)$$

$$\mathbf{u} = \mathbf{0} \quad \text{on } \Gamma_P, \quad (2.10)$$

$$\mathbf{s} = \mathbf{s}_{\text{in}} \quad \text{on } \Gamma_{\text{in}}. \quad (2.11)$$

Eqs. (2.8) and (2.9) express that the boundary conditions on $\Gamma = \Gamma_D \cup \Gamma_N$ can be split into Dirichlet boundary conditions on Γ_D and Neumann boundary conditions on Γ_N . In Eq. (2.9), \mathbf{t} is the traction and \mathbf{n} is the outwardly directed unit normal vector on the fluid. Eq. (2.10) represents the no-slip boundary condition on the stationary rigid body surface. Due to the hyperbolic character of the constitutive equation (Eq. (2.7)), we also need a boundary condition for \mathbf{s} (Eq. (2.11)) at the inflow boundary Γ_{in} where $\mathbf{u} \cdot \mathbf{n} < 0$.

Since the evolution equation of the conformation tensor is time-dependent, we also need an initial condition for \mathbf{s} :

$$\mathbf{s}(\mathbf{x}, 0) = \mathbf{s}_0(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega \setminus P. \quad (2.12)$$

In our simulations, we use a stress-free state as initial condition over the whole fluid domain, i.e. $\mathbf{s}_0(\mathbf{x}) = \mathbf{0}, \forall \mathbf{x} \in \Omega \setminus P$. Since we neglect inertia, an initial condition for the velocity is not necessary.

2.3 Numerical methods

2.3.1 Mixed DEVSS-G/SUPG finite element formulation

For the spatial discretization of the governing equations, we use the finite element method. One of the problems associated with Galerkin methods is the compatibility between discretization spaces for mixed problems. For the velocity and pressure, we will use the Taylor-Hood Q_2Q_1 element which satisfies the Ladyzhenskaya-Babuška-Brezzi (LBB) condition [97]. Even after using the proper velocity/pressure spaces, there still remains a compatibility problem between conformation space and velocity space. To resolve this problem, we use the DEVSS-G method which is a combination of the EVSS-G method by Brown et al. [98] and the DEVSS method by Guénette and Fortin [99]. These methods are reviewed by Baaijens [100]. In this chapter, we will employ the particular form of the DEVSS-G method proposed by Bogaerds et al. [101], using the full velocity gradient for a stabilization of the momentum balance. Furthermore, we will use a bi-linear Q_1 interpolation for each component of the log-conformation tensor \mathbf{s} and velocity gradient projection tensor \mathbf{G} . Also, in Galerkin methods, the convection term in Eq. (2.7) can generate spurious oscillations. The most common method to stabilize the convection term is the SUPG method developed by Brooks and Hughes [102]. The weak formulation of the governing equations, with the DEVSS-G/SUPG stabilization technique, can be stated as follows: Find \mathbf{u} , p , \mathbf{s} , \mathbf{G} and λ such that

$$-(\nabla \cdot \mathbf{v}, p) + ((\nabla \mathbf{v})^T, 2\eta_s \mathbf{D} + \theta(\nabla \mathbf{u} - \mathbf{G}^T) + \boldsymbol{\tau}(\mathbf{c})) + (\mathbf{v}, \boldsymbol{\lambda})_{\Gamma_p} = (\mathbf{v}, \mathbf{t})_{\Gamma_N}, \quad (2.13)$$

$$(q, \nabla \cdot \mathbf{u}) = 0, \quad (2.14)$$

$$(\mathbf{d} + \tau \mathbf{u} \cdot \nabla \mathbf{d}, \frac{\partial \mathbf{s}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{s} - \mathbf{g}(\mathbf{G}, \mathbf{s})) = 0, \quad (2.15)$$

$$(\mathbf{H}, -\nabla \mathbf{u} + \mathbf{G}^T) = 0, \quad (2.16)$$

$$(\boldsymbol{\mu}, \mathbf{u})_{\Gamma_p} = 0, \quad (2.17)$$

for all admissible test functions \mathbf{v} , q , \mathbf{d} , \mathbf{H} and $\boldsymbol{\mu}$, where (\cdot, \cdot) , $(\cdot, \cdot)_{\Gamma_N}$ and $(\cdot, \cdot)_{\Gamma_p}$ are proper inner products on the fluid domain $\Omega \setminus P$, on the Neumann boundary Γ_N and on the interface Γ_p , respectively. The no-slip boundary condition on the rigid body surface is imposed by using constraints implemented with Lagrangian multipliers. In Eq. (2.13), the DEVSS-G parameter θ is chosen equal to the polymer viscosity, $\theta = \eta_p$. In Eq. (2.15), the SUPG parameter τ is given by $\tau = \bar{\beta}h/2U_c$, where $\bar{\beta}$ is a non-dimensional parameter, h is a typical size of the element in the direction of velocity and U_c is a characteristic velocity magnitude for the flow problem. The choice of τ will be discussed in Section 2.3.2.

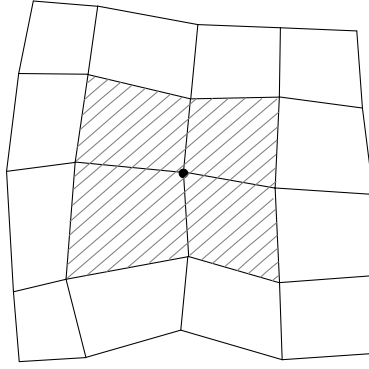


Figure 2.2: A patch connected to a node (\bullet).

2.3.2 The extended finite element method

Enrichment strategies

We introduce the enrichment of the finite element approximations through the partition of unity method (PUM) [89]. The partition of unity method is about extending the local approximation space by including functions of the expected solution in the finite element space. For defining the enriched approximation space in the PUM, we need a set of shape functions that form a partition of unity. In the standard finite element method, the finite element shape functions ϕ_k form a partition of unity, i.e.

$$\sum_k \phi_k(\mathbf{x}) = 1, \quad \text{for any } \mathbf{x} \text{ in a domain.} \quad (2.18)$$

In the following, we will use the finite element shape functions ϕ_k for the partition of unity. Then, the partition of unity approximation V is defined as

$$V = \sum_k \phi_k V_k \quad (2.19)$$

where V_k are the local approximation functions defined on the patches (Fig. 2.2). Note that the standard FEM is obtained by $V_k = 1$ at each node k .

Let's consider discontinuities across an interface. The easiest case of discontinuities is the "one-sided" discontinuity, which is basically a domain Ω^+ in which a problem is to be solved. Outside this domain, thus in Ω^- , the field variables are nonexistent, but are taken to be zero for simplicity. The outside domain Ω^- can be considered as holes or voids, similarly to [103]. Therefore, there is a jump at the interface between Ω^+ and Ω^- . The approximation can be written as:

$$V = \sum_k \phi_k(\mathbf{x}) H(s) a_k, \quad \forall \mathbf{x} \in \Omega, \quad (2.20)$$

where a_k are degrees of freedom and $H(s)$ is a Heaviside function defined by a scalar level set

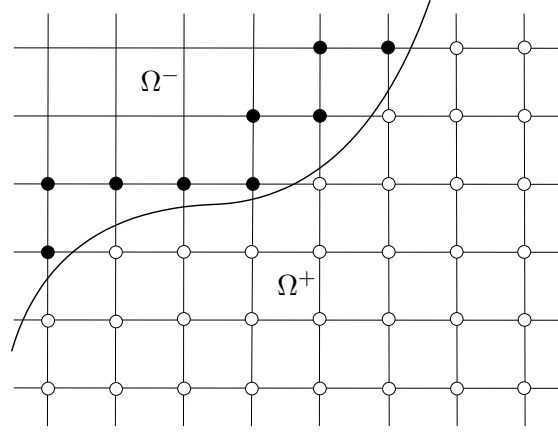


Figure 2.3: A one-sided discontinuity problem. The degrees of freedom (a_k) are composed of standard degrees of freedom (\circ) and virtual degrees of freedom (\bullet).

function s :

$$H(s) = \begin{cases} +1 & \text{if } s \geq 0, \\ 0 & \text{if } s < 0, \end{cases} \quad (2.21)$$

$$s(\mathbf{x}) \begin{cases} > 0 & \text{if } \mathbf{x} \in \Omega^+, \\ < 0 & \text{if } \mathbf{x} \in \Omega^-, \\ = 0 & \text{if } \mathbf{x} \text{ is on the interface.} \end{cases} \quad (2.22)$$

The interface is represented as the zero level contour of a level set function. The local approximation functions are now $V_k = H(s)a_k$. Since elements fully outside the domain have no contribution to the system, nodes connected only to such elements can be discarded from the system. All other nodes retain the degree of freedom a_k (Fig. 2.3). The degrees of freedom inside the domain have the same meaning as the standard finite element method, whereas the degrees of freedom outside the domain are *virtual* degrees of freedom, which are defining the spatial distribution of field variables inside the domain from the outside. They do not define field variables at that point. The integration of the weak form is only performed on the inside domain Ω^+ .

Although we will not use “two-sided” discontinuities in this thesis, it is useful to show that the previous description can be extended to that case in an easy way. If there is a fluid on both sides of the interface, the approximation can be written as:

$$V = \sum_k \phi_k(\mathbf{x})H(s)a_k + \sum_k \phi_k(\mathbf{x})H(\hat{s})b_k, \quad (2.23)$$

where a_k and b_k are degrees of freedom and $\hat{s} = -s$, then, $H(\hat{s}) = H(-s) = 1 - H(s)$. The local approximation functions are now $V_k = H(s)a_k + H(\hat{s})b_k$. Nodes in elements intersected by the interface have both the degrees of freedom a_k and b_k , whereas other nodes have only a_k or b_k (Fig. 2.4). Hence, the nodes connected to intersected elements have twice as many

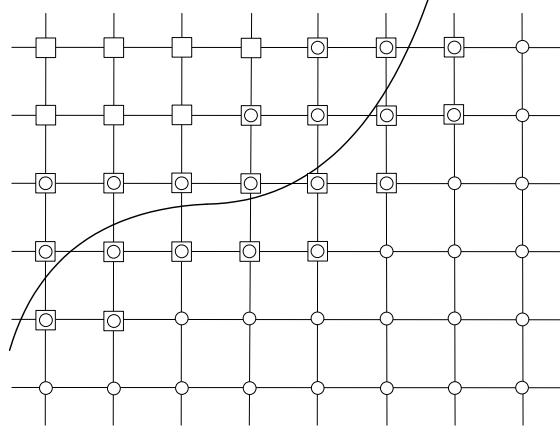


Figure 2.4: A two-sided discontinuity problem. (\circ : degrees of freedom a_k , \square : degrees of freedom b_k .)

degrees of freedom than other nodes. Furthermore, the two material sides need to be coupled. This can be done, for example, by using constraints implemented with Lagrangian multipliers. Eq. (2.23) is essentially the same method as proposed by Hansbo and Hansbo [104] for a linear elasticity problem. The degrees of freedom used near the discontinuity are different from the conventional XFEM using enriched degrees of freedom [90, 91], but Areias and Belytschko [105] showed that both approaches are equivalent.

Application to flow around rigid bodies

For our fluid-rigid body system depicted in Fig. 2.1, we have a jump at the interface from physical quantities of the velocity, pressure and conformation in fluid domain $\Omega^+ = \Omega \setminus P$ to zero or essentially nothing in rigid body domain $\Omega^- = P$, as no flow occurs inside the rigid body. Hence, the velocity, pressure, log-conformation and velocity gradient projection can be discretized as:

$$\mathbf{u}_h(\mathbf{x}) = \sum_k \varphi_k(\mathbf{x}) H(s) \mathbf{u}_k, \quad (2.24)$$

$$p_h(\mathbf{x}) = \sum_k \psi_k(\mathbf{x}) H(s) p_k, \quad (2.25)$$

$$\mathbf{s}_h(\mathbf{x}) = \sum_k \psi_k(\mathbf{x}) H(s) \mathbf{s}_k, \quad (2.26)$$

$$\mathbf{G}_h(\mathbf{x}) = \sum_k \psi_k(\mathbf{x}) H(s) \mathbf{G}_k, \quad (2.27)$$

where

$$H(s) = \begin{cases} +1 & \text{if } s \geq 0, \\ 0 & \text{if } s < 0, \end{cases} \quad (2.28)$$

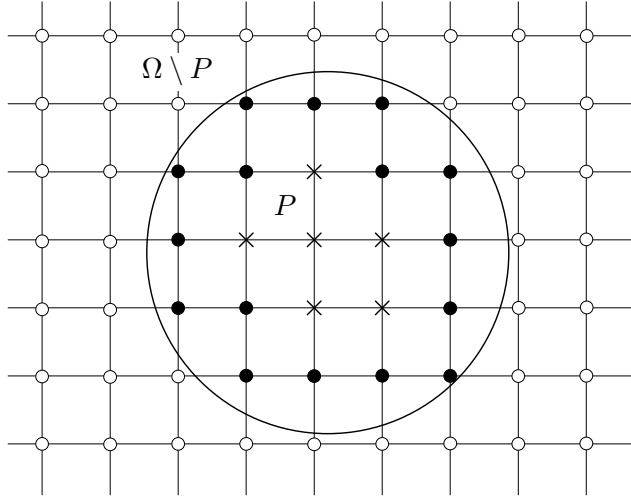


Figure 2.5: An enrichment scheme for a fluid-rigid body system. (\circ : standard degree of freedom, \bullet : virtual degree of freedom, \times : removed degree of freedom)

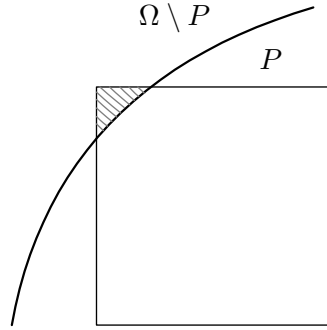


Figure 2.6: An intersected element. The shaded region represents the integration area of the element.

is defined by a level set function s :

$$s(\mathbf{x}) \begin{cases} = 0 & \text{if } \mathbf{x} \text{ is on } \Gamma_p, \\ > 0 & \text{if } \mathbf{x} \text{ is in } \Omega \setminus P, \\ < 0 & \text{if } \mathbf{x} \text{ is in } P. \end{cases} \quad (2.29)$$

We use a bi-quadratic Q_2 interpolation (φ_k) for the velocity \mathbf{u} ; and a bi-linear Q_1 interpolation (ψ_k) for the pressure p , log-conformation tensor \mathbf{s} and velocity gradient projection \mathbf{G} . In this chapter, we use a signed distance function d as a level set function s . For example, in two dimensions, a circular interface with radius r_c and the center in (x_c, y_c) is defined by the zero level of

$$s(\mathbf{x}) = d(x, y) = \sqrt{(x - x_c)^2 + (y - y_c)^2} - r_c. \quad (2.30)$$

A simple interpretation of the XFEM enrichment procedure for fluid-rigid body system is shown in Fig. 2.5. Elements fully inside the fluid domain $\Omega \setminus P$ have only standard degrees of

freedom (\circ) and are fully integrated element by element. The standard degrees of freedom (\circ) are the same as the standard finite element method and they define the field variable at that point. The elements intersected by the interface have both standard degrees of freedom and virtual degrees of freedom (\bullet) for the jump enrichment. Intersected elements are integrated only on the fluid domain part as shown in Fig. 2.6. The integration on the part of an element will be discussed in the next section. The virtual degrees of freedom (\bullet) do not define the field variable at that point, but define the spatial distribution of the field variable in the part of an intersected element. In other words, they are only used for interpolating values at the fluid domain part of an intersected element. Elements fully inside the rigid body are ignored and they are not assembled into the system matrix. Therefore, the degrees of freedom only belonging to elements fully inside the rigid body domain are the removed degrees of freedom (\times). A practical implementation of the removed degrees of freedom (\times) is treating them like Dirichlet boundary conditions and setting them to zero.

For elements intersected by the interface, the imposition of a no-slip boundary condition (Eq. (2.10)) is needed. This can be done by using constraints implemented with Lagrangian multipliers as shown in Eqs. (2.13) and (2.17). For the implementation of Lagrangian multipliers, the rigid bodies are discretized only on their boundaries by using the rigid ring description [73]. The inner product $(\cdot, \cdot)_{\Gamma_p}$ is the standard inner product in $L^2(\Gamma_p)$:

$$(\boldsymbol{\mu}, \boldsymbol{u})_{\Gamma_p} = \int_{\Gamma_p} \boldsymbol{\mu} \cdot \boldsymbol{u} \, ds. \quad (2.31)$$

For the integration in Eq. (2.31), we use a linear shape function P_1 for the discretization of Lagrangian multipliers and a quadratic shape function P_2 for the geometrical shape of each element as shown in Fig 2.7. Each element on the boundary of a rigid body has two end nodes (\bullet) and one midpoint node (\circ). Only end nodes (\bullet) have degrees of freedom

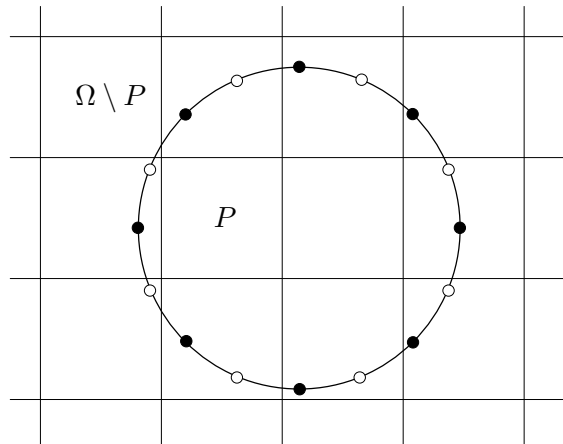


Figure 2.7: A mesh for the discretization of the boundary of a rigid body. P_1 interpolation for the discretization of Lagrangian multipliers and P_2 interpolation for the geometrical shape. Each element has two end nodes (\bullet) and one midpoint node (\circ). Midpoint nodes (\circ) do not have a degree of freedom for Lagrangian multipliers.

for Lagrangian multipliers. Midpoint nodes (\circ) together with end nodes are used for the geometrical description of an element such as an isoparametric mapping of a real element into a reference element. We choose the size of rigid body elements similar to the size of fluid elements near the interface, which will be verified in Section 2.4. The integration in each rigid body element can be performed exactly, if the intersections between rigid body elements and the fluid element boundaries are computed. Alternatively, to simplify the implementation, a rigid body element is divided into a number of subintervals and the integration is carried on each subinterval, combined with a subsequent composite integration rule applied on the element. In this chapter, we use 20 subintervals and a midpoint rule in each subinterval.

Numerical integration on a part of an element

The integration on parts of an element is crucial for the robustness and accuracy of XFEM. The elements intersected by an interface are divided into subdomains for the integration on parts of the element. First, intersected elements are divided into progressively smaller subdomains using a quadtree. Then, the quads at the interface are further divided into triangles aligned with the interface. A subdivision process is shown in Fig. 2.8 with a two level quadtree subdivision and triangulation of the smallest quads near the interface. The algorithm of triangulation is based on the level set function for the interface and can be found in [106]. Since triangular subdomains are aligned with the interface, we can easily integrate on the fluid domain only. In our simulations, we use a 3×3 Gauss integration rule for undivided elements and quadrilateral subdomains, and a 6-point Gauss integration for triangular subdomains. We test the accuracy of integration with two, three, five, eight and ten level quadtree subdivisions. This will be discussed in Section 2.5.2.

If the area for the integration on a part of an element (the shaded region in Fig. 2.6) is extremely small compared to the whole area of the element, the system matrix can be ill-conditioned and the solution may become unstable. To avoid this problem, the nodes

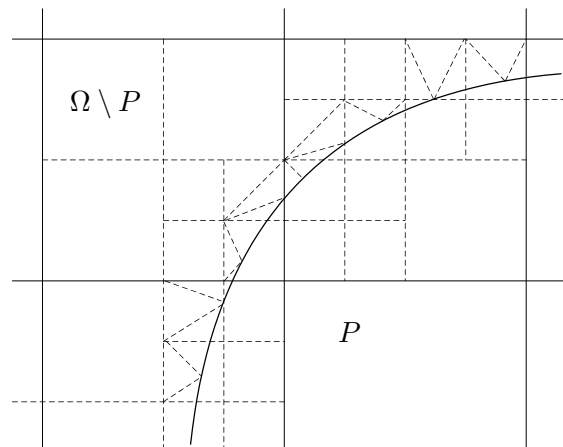


Figure 2.8: Subdivision of elements: two level quadtree subdivision and triangulation of the smallest quads near the interface.

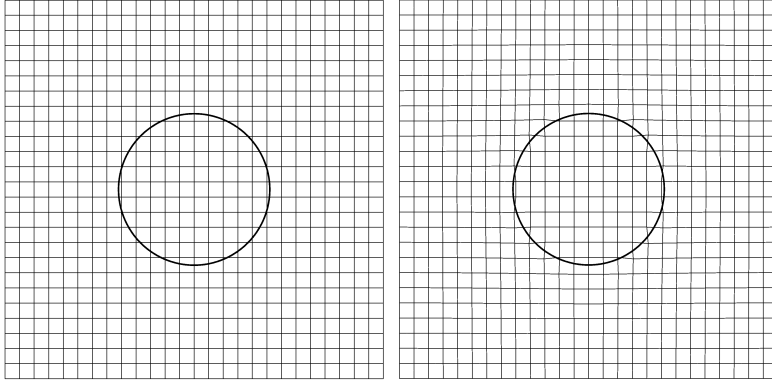


Figure 2.9: Nodal point displacement. Left figure: initial mesh, right figure: displaced mesh.

outside the rigid body are displaced along the radial direction of the rigid body.[†] We follow the following procedures. First, we compute a mesh velocity field \mathbf{u}_m by solving the Poisson problem:

$$\nabla^2 \mathbf{u}_m = \mathbf{0} \text{ in } \Omega, \quad (2.32)$$

$$\mathbf{u}_m = \mathbf{0} \text{ on } \Gamma, \quad (2.33)$$

$$\mathbf{u}_m = \hat{\mathbf{n}} \text{ on } \Gamma_p, \quad (2.34)$$

where $\hat{\mathbf{n}}$ is the outwardly directed unit normal vector on the rigid body surface. Then, each mesh point \mathbf{x}_m moves according to the following advection equations:

$$\frac{d\mathbf{x}_m}{dt_m} = \begin{cases} \mathbf{u}_m & \text{if } \mathbf{x}_m \in \Omega \setminus P, \\ \mathbf{0} & \text{if } \mathbf{x}_m \in P, \end{cases} \quad (2.35)$$

$$\mathbf{x}_m(t_m = 0) = \mathbf{x}_{m,0}, \quad (2.36)$$

where $\mathbf{x}_{m,0}$ is the initial position of mesh point. This ODE system can be easily solved using an explicit scheme such as Runge-Kutta methods. For an intersected element boundary, a node in-between element vertices is set to the midpoint of vertex nodes after each Runge-Kutta step to avoid large local distortion of elements. Fig. 2.9 shows the movement of nodal points. In Fig. 2.9, the nodal points outside the rigid body are displaced until each area of the subdomain integration is larger than 3% of the entire element area for a better distinction between the initial and displaced meshes. In our simulations, nodal points are moved until each area of integration is larger than 0.5% of the element area. For this subdomain element ratio, the difference between the initial and displaced meshes is small and at most two or three Runge-Kutta steps are needed. Note that the nodes are displaced before the simulation

[†] If the subdivided elements at the interface are exactly integrated using higher order Gauss integration rules, the mesh optimization scheme introduced in this chapter, which requires an additional Poisson problem, can be avoided. In Chapter 5 and Chapter 6, we use higher order Gauss integration rules for the integration on a part of an element.

starts and the physical quantities are calculated on the fixed displaced mesh. Hence, the mesh velocity \mathbf{u}_m is not related to the advection of physical field variables. Therefore, we do not need to incorporate an arbitrary Lagrangian-Eulerian (ALE) algorithm.

Note on the SUPG parameter τ

There are many possibilities for choosing a characteristic velocity U_c in an element [107].

Case 1. The magnitude of the velocity in the integration points:

$$U_c = \|\mathbf{u}_j\|, \quad j = 1, \dots, J \quad (2.37)$$

where J is the number of integration points in the element. The characteristic velocity is different for each integration point.

Case 2. The average of the magnitude of the velocities in all integration points in the element:

$$U_c = \frac{1}{J} \sum_{j=1}^J \|\mathbf{u}_j\| \quad (2.38)$$

The characteristic velocity is different for each element.

If the rigid body is stationary, the velocity of the fluid very close to the interface tends to zero due to the no-slip boundary condition. Hence, the characteristic velocity can be very small near the interface for both Case 1 and Case 2. If this happens, the SUPG parameter τ becomes very large since τ is inversely proportional to U_c . Then, even for very small velocity (then, the convection is also small), the SUPG parameter τ tends to put more weight to the upwind direction of the streamlines in Eq. (2.15). This can lead to numerical instabilities and solutions can become unstable. To prevent this problem, we propose a $\bar{\beta}$ parameter as a function of the Courant number $C = U_c \Delta t / h$ (see Fig. 2.10):

$$\bar{\beta} = \begin{cases} +1 & \text{if } C \geq 1, \\ C & \text{if } C < 1, \end{cases} \quad (2.39)$$

where Δt is the computational time step. As the characteristic velocity becomes smaller, $\bar{\beta}$

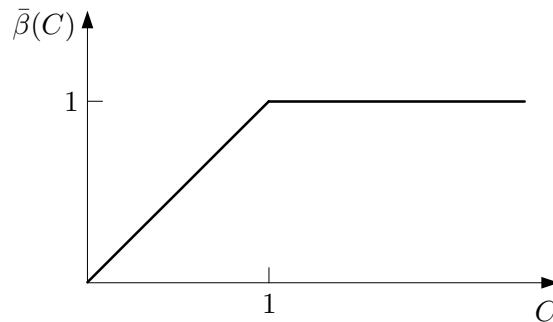


Figure 2.10: Definition of $\bar{\beta}(C)$.

imposes a lesser weight on the convection in Eq. (2.15). This leads to stability improvements. Note that

$$\tau = \begin{cases} h/2U_c & \text{if } C \geq 1, \\ \Delta t/2 & \text{if } C < 1. \end{cases} \quad (2.40)$$

This SUPG parameter τ is somewhat similar to the one proposed by Shakib et al. [108]:

$$\tau^* = [(2/\Delta t)^2 + (2U_c/h)^2]^{-1/2} \quad (2.41)$$

in which τ^* is asymptotic to $h/2U_c$ for large C and $\Delta t/2$ for small C . An alternative way is

Case 3. We can use a global velocity scale U_{glob} :

$$U_c = U_{\text{glob}} \quad (2.42)$$

In this case, the characteristic velocity is the same for all elements and we can simply set $\bar{\beta} = 1$. This procedure can be useful for flow around a rigid body in a channel flow [109]. We do not notice any significant difference between Cases 1 and 2 with proposed $\bar{\beta}$ in Eq. (2.39) and Case 3 in the aspects of accuracy and stability. Case 3 can also be interpreted such that the characteristic velocity is the magnitude of the velocity in the integration points with $\bar{\beta}_j = \|\mathbf{u}_j\|/U_{\text{glob}}$ for each integration point.

2.3.3 Time integration

We use a semi-implicit stress formulation for the time integration of the evolution equation of the log-conformation tensor \mathbf{s} (Eq. (2.15)) decoupled from the momentum balance. Initially, the viscoelastic polymer stress is set to zero over the whole fluid domain. So we can solve Eqs. (2.13), (2.14), (2.16) and (2.17) to get the distribution of the fluid velocity and pressure at the initial time step $t = 0$. Then, we apply the following procedure at every time step.

Step 1. Solve the log-conformation tensor \mathbf{s}^{n+1} by replacing the evolution equation of the log-conformation tensor (Eq. (2.7)) using a second-order semi-implicit Gear scheme:

$$\begin{aligned} (\mathbf{d} + \tau \hat{\mathbf{u}}^{n+1} \cdot \nabla \mathbf{d}, \frac{3}{2} \frac{\mathbf{s}^{n+1}}{\Delta t} + \hat{\mathbf{u}}^{n+1} \cdot \nabla \mathbf{s}^{n+1}) = \\ (\mathbf{d} + \tau \hat{\mathbf{u}}^{n+1} \cdot \nabla \mathbf{d}, \frac{2\mathbf{s}^n - \frac{1}{2}\mathbf{s}^{n-1}}{\Delta t} + 2\mathbf{g}(\mathbf{G}^n, \mathbf{s}^n) - \mathbf{g}(\mathbf{G}^{n-1}, \mathbf{s}^{n-1})), \end{aligned} \quad (2.43)$$

where $\hat{\mathbf{u}}^{n+1} = 2\mathbf{u}^n - \mathbf{u}^{n-1}$. At the first time step, we use a first-order Euler scheme:

$$(\mathbf{d} + \tau \mathbf{u}^n \cdot \nabla \mathbf{d}, \frac{\mathbf{s}^{n+1}}{\Delta t} + \mathbf{u}^n \cdot \nabla \mathbf{s}^{n+1}) = (\mathbf{d} + \tau \mathbf{u}^n \cdot \nabla \mathbf{d}, \frac{\mathbf{s}^n}{\Delta t} + \mathbf{g}(\mathbf{G}^n, \mathbf{s}^n)). \quad (2.44)$$

Note, that the system is linear in the unknown \mathbf{s}^{n+1} .

Step 2. Compute \mathbf{c}^{n+1} with $\mathbf{c}^{n+1} = \exp(\mathbf{s}^{n+1})$. Then, compute \mathbf{u}^{n+1} , p^{n+1} , \mathbf{G}^{n+1} and $\boldsymbol{\lambda}^{n+1}$ from the momentum balance, continuity equation and gradient projection equation at time t^{n+1} :

$$\begin{aligned}
& - (\nabla \cdot \mathbf{v}, p^{n+1}) + ((\nabla \mathbf{v})^T, 2\eta_s \mathbf{D}(\mathbf{u}^{n+1}) + \theta(\nabla \mathbf{u}^{n+1} - (\mathbf{G}^T)^{n+1})) \\
& \quad + (\mathbf{v}, \boldsymbol{\lambda}^{n+1})_{\Gamma_p} = -((\nabla \mathbf{v})^T, \boldsymbol{\tau}(\mathbf{c}^{n+1})) + (\mathbf{v}, \mathbf{t})_{\Gamma_N}, \quad (2.45)
\end{aligned}$$

$$(q, \nabla \cdot \mathbf{u}^{n+1}) = 0, \quad (2.46)$$

$$(\mathbf{H}, -\nabla \mathbf{u}^{n+1} + (\mathbf{G}^T)^{n+1}) = 0, \quad (2.47)$$

$$(\boldsymbol{\mu}, \mathbf{u}^{n+1})_{\Gamma_p} = 0. \quad (2.48)$$

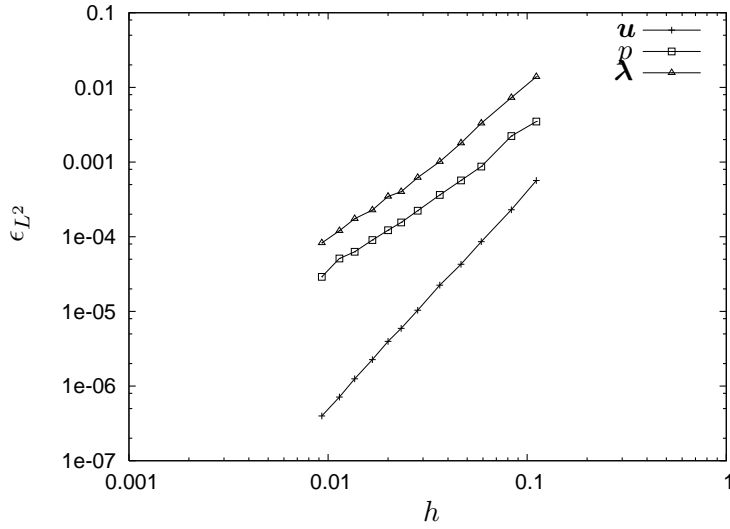
Note, that the system is linear in the unknowns $(\mathbf{u}^{n+1}, p^{n+1}, \boldsymbol{\lambda}^{n+1})$.

It should be noted that the constitutive equation is decoupled from the momentum balance. This limits the time step. Also, the constitutive equation is time discretized by a semi-implicit scheme, where the convection is implicit, but the deformation and relaxation terms are explicit. This also limits the time step that can be used. Fully implicit schemes would allow for larger time steps, but are not an option for practical problems since multiple mode models need to be used to describe the broad relaxation spectrum of real polymeric fluids.

2.3.4 Comparison with other numerical methods

In this section, we briefly introduce other numerical methods frequently used for the direct numerical simulations of fluid-rigid body systems. We will compare the results of XFEM with these methods in the next section. The most direct way is solving the fluid motion using a boundary-fitted mesh by taking into account the interface conditions on the boundary of rigid bodies [58, 59]. The whole computational domain Ω coincides with the fluid domain $\Omega \setminus P$, since the domain of rigid bodies is discarded from the system. Hence, we can easily integrate in the fluid domain. For the imposition of the no-slip boundary condition, nodes on the interface can be regarded as point collocations or element boundaries on the interface can be used as the discretization of Lagrangian multipliers.

Another method for direct numerical simulations of fluid-rigid body systems is the fictitious domain method [62, 63, 73]. In this method, the rigid body domain is filled with the surrounding fluid and it is assumed that the fluid inside each rigid body moves like a solid object. The motion of the fluid inside the rigid body domain can be imposed using constraints implemented with Lagrangian multipliers. Hence, the problem is extended from the fluid domain $\Omega \setminus P$ to the whole computational domain Ω . Since we neglect inertia, the weak formulation of the fictitious domain method has the same form as Eqs. (2.13)-(2.17) except that integrations are carried on the whole domain Ω , not on the fluid domain $\Omega \setminus P$. Hence, we can simply integrate element by element in Ω . We do not need a special integration technique such as the one explained in Section 2.3.2. The no-slip boundary condition can be imposed in the same way as in XFEM, by using Eq. (2.31).

Figure 2.11: Convergence in relative L^2 -norm.

2.4 Convergence analysis

In order to study the convergence behavior of the proposed method, in which the interfacial condition is realized by using a constraint implemented with Lagrangian multipliers, we solve a Stokes problem with a known exact solution. The problem we choose is the flow around a cylinder and the exact solution is given in [92]:

$$\bar{u}_x = \frac{(R^2 - r^2) \cos^2 \theta + r^2 \ln(r/R) + (1/2)(r^2 - R^2)}{r^2}, \quad (2.49)$$

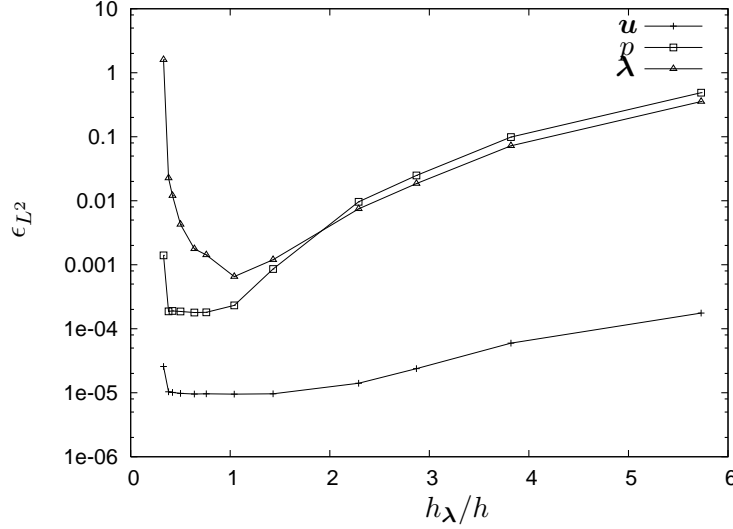
$$\bar{u}_y = \frac{(R^2 - r^2) \cos \theta \sin \theta}{r^2}, \quad (2.50)$$

$$\bar{p} = -\frac{2 \cos \theta}{r} + 10. \quad (2.51)$$

Note, that compared to [92] we added a constant pressure level to avoid a constant traction vector of $(10, 0)$ on the cylinder boundary. The problem is solved on a square domain of side length 2, with a cylinder of radius 0.2 centered at the origin, as proposed in [92]. We apply the exact value of the velocity on the boundary of the domain as a Dirichlet boundary condition, and on the cylinder surface as a constraint implemented with Lagrangian multipliers. We specify the pressure in a single point of the domain.

The relative L^2 -errors in the velocity, pressure and Lagrangian multipliers are plotted against the element size h in Fig. 2.11. The relative L^2 -errors are defined as:

$$\epsilon_{L^2, \mathbf{u}} = \frac{\|\mathbf{u}_h - \bar{\mathbf{u}}\|_{L^2}}{\|\bar{\mathbf{u}}\|_{L^2}} = \frac{(\int_{\Omega \setminus P} |\mathbf{u}_h - \bar{\mathbf{u}}|^2 dx)^{1/2}}{(\int_{\Omega \setminus P} |\bar{\mathbf{u}}|^2 dx)^{1/2}}, \quad (2.52)$$

Figure 2.12: Effect of h_λ/h on the relative L^2 -errors.

$$\epsilon_{L^2,p} = \frac{\|p_h - \bar{p}\|_{L^2}}{\|\bar{p}\|_{L^2}} = \frac{\left(\int_{\Omega \setminus P} |p_h - \bar{p}|^2 dx\right)^{1/2}}{\left(\int_{\Omega \setminus P} \bar{p}^2 dx\right)^{1/2}}, \quad (2.53)$$

$$\epsilon_{L^2,\lambda} = \frac{\|\lambda_h - \bar{\mathbf{t}}\|_{L^2}}{\|\bar{\mathbf{t}}\|_{L^2}} = \frac{\left(\int_{\Gamma_p} |\lambda_h - \bar{\mathbf{t}}|^2 ds\right)^{1/2}}{\left(\int_{\Gamma_p} |\bar{\mathbf{t}}|^2 ds\right)^{1/2}}, \quad (2.54)$$

where $\bar{\mathbf{t}} = (-\bar{p}\mathbf{I} + 2\eta_s \mathbf{D}(\bar{\mathbf{u}})) \cdot \mathbf{n}$ on the cylinder boundary Γ_p and the subscript ‘ h ’ indicates the discretized field variables. Note that the Lagrangian multipliers represent the traction required to maintain the rigid body motion constraints on the boundary [73]. Fig. 2.11 shows optimal convergence rates - third order in the velocity; second order in the pressure and Lagrangian multipliers. Note that the Lagrangian multipliers defined by the intersection of the interface with the background fluid mesh lead to oscillations in the Lagrangian multipliers and produce sub-optimal convergence rates [110, 111], probably caused by an over-constrained primary variable space. Recovering optimal convergence rates was reported either by reducing the Lagrangian multiplier space [111], or by using a stabilized method employing bubble functions [112, 113] while using the “conventional” choice for the Lagrangian multipliers. Our choice of a Lagrangian multiplier space also shows optimal convergence rates for this particular problem.

We also investigate the effect of the size of line elements for the discretization of Lagrangian multipliers h_λ to the size of fluid elements h . We fix the size of the fluid element $h = 0.0274$, then solve the problem with a different size of rigid body elements h_λ . The effect of h_λ/h on the relative L^2 -errors is plotted in Fig. 2.12. Choosing the size of the rigid body elements similar to the size of fluid element ($h_\lambda/h \approx 1$) seems optimal. We choose the size of rigid body elements similar to the size of fluid elements near the interface for all computations in this chapter.

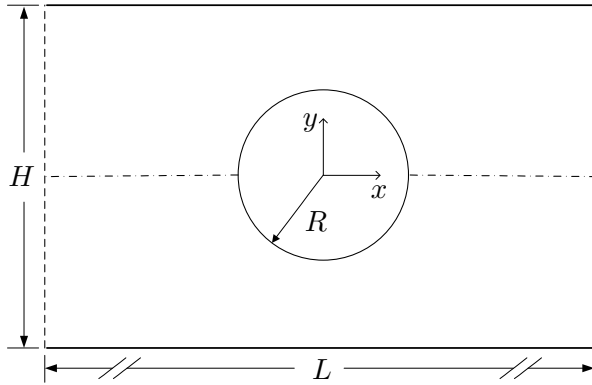


Figure 2.13: Geometry for the flow around a cylinder, confined between two parallel plates.

Table 2.1: Meshes used for the simulations of XFEM and FDM

	M0	M1	M2	M3	M4
Number of elements	1000	23256	84000	110390	258302
Element size near the interface	0.2	0.0351	0.0191	0.0151	0.00709

2.5 Results

2.5.1 Problem description

We consider a planar flow past a stationary cylinder of radius R , confined between two parallel plates, see Fig. 2.13. The length of the channel $L = 30R$ and height of the channel $H = 4R$. The flow is generated by specifying a flow rate Q that is constant in time. Thus the average velocity of the fluid is $U = Q/H = Q/4R$. We assume no-slip boundary conditions on the cylinder and on the channel walls. We also assume the flow to be periodic which means that we extend the flow domain periodically in x -direction such that cylinders are positioned $30R$ apart. Since the flow is periodic, the inflow boundary condition for the conformation (Eq. (2.11)) is not necessary.

The dimensionless parameters governing the problem are the Weissenberg number $Wi = \lambda U/R$ and the viscosity ratio η_s/η_0 , where $\eta_0 = \eta_s + \eta_p$ is the zero-shear-rate viscosity. We fix the viscosity ratio $\eta_s/\eta_0 = 0.59$ and also take a time step of $\Delta t = 0.003$ for all computations, for a fair comparison with the results of mesh M4 and $\Delta t = 0.003$ in Hulsén et al. [96]. We can use larger time steps (up to $\Delta t = 0.01$) without a significant difference. For even larger time steps the computations become numerically unstable.

We solve the problem with a boundary-fitted mesh (BFM), a fictitious domain method (FDM), and the extended finite element method (XFEM). For the simulations of XFEM and FDM, we use four meshes M1, M2, M3 and M4 summarized in Table 2.1. The base mesh M0 is shown in Fig. 2.14. Note that our meshes M2, M3 and M4 have similar resolution near the cylinder surface as the meshes M3, M4 and M6, respectively, in Hulsén et al. [96]. We also use a boundary-fitted mesh with 9410 elements which is about 5 times finer than the mesh

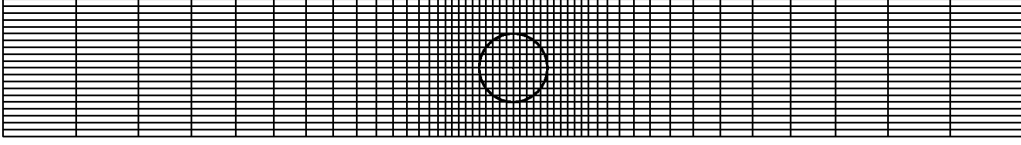


Figure 2.14: Base mesh M0 from which the other meshes are derived.

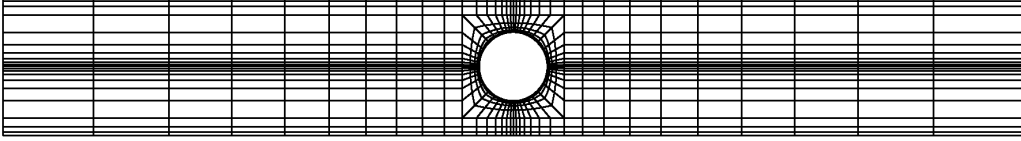


Figure 2.15: Boundary-fitted mesh. Note that the actual mesh used in the computations is about 5 times finer than the mesh shown here.

shown in Fig. 2.15. The boundary-fitted mesh has similar resolution as the mesh M4 along the radial direction on the cylinder. In the BFM, the cylinder surface is treated as a Dirichlet boundary. Note that the boundary-fitted mesh has a much lower number of elements than the mesh M4. This originates from efficient element refinements around the cylinder for the boundary-fitted mesh.

In order to judge whether we have obtained a steady-state, we monitor the values of the velocity \mathbf{u} , and the log-conformation \mathbf{s} , in the domain and the drag on the cylinder as a function of time. We solve the problem until these values do not change up to an order of $O(10^{-6})$, for example, $|u_x - u_{x,\text{prev}}| < 10^{-6}$ for all nodes, where u_x is the x -directional velocity component at the current time and $u_{x,\text{prev}}$ is the velocity component at the previous time. The time needed to obtain a steady-state increases as the Weissenberg number Wi increases.

2.5.2 Accuracy of the subdomain integration

Since we integrate only on the fluid domain $\Omega \setminus P$ using the XFEM, we can easily calculate the area of the fluid domain $A_{\Omega \setminus P}$. Then, the area of the cylinder is $A_P = A_\Omega - A_{\Omega \setminus P}$, where $A_\Omega = L \times H = 120R^2$. If $R = 1$, the exact area of the cylinder is $A_P = \pi R^2 = \pi$ (≈ 3.1415926536). We test the accuracy of the subdomain integration with computed A_P using mesh M2. In Table 2.2, we give the values of A_P for several levels of subdivisions, and the corresponding CPU times for the integration of the flow problem. Note that the CPU times are for the integration of the flow problem in Eqs. (2.45)-(2.48), not for the calculation of the area $A_{\Omega \setminus P}$. As the level of subdivision increases, we obtain more accurate values of A_P . In this problem, we use eight level subdivisions since those give accurate integrations with manageable computational times required. The CPU time for solving the system matrix equation is approximately 390 s which is much larger than the CPU time for the integration using eight level subdivisions.

Table 2.2: Accuracy of the subdomain integrations

level of subdivision	A_P	CPU times (in seconds) for the flow problem
2	3.141638457	9.12
3	3.141604508	9.31
5	3.141593336	10.39
8	3.141592664	21.44
10	3.141592654	76.13
exact value	3.1415926536	-

2.5.3 Drag coefficient

The drag F_x on the rigid body is defined by

$$F_x = \int_{\Gamma_p} \boldsymbol{\sigma} \cdot \hat{\mathbf{n}} \, ds \cdot \mathbf{e}_x = \int_{\Gamma_p} (-p\mathbf{I} + 2\eta_s \mathbf{D} + \boldsymbol{\tau}) \cdot \hat{\mathbf{n}} \, ds \cdot \mathbf{e}_x, \quad (2.55)$$

where $\hat{\mathbf{n}}$ is the outwardly directed unit normal vector on the rigid body and \mathbf{e}_x is the unit vector in x -direction. The dimensionless steady-state drag coefficient K is defined as

$$K = \frac{F_x}{\eta_0 UR}. \quad (2.56)$$

Table 2.3 shows the values of K for different meshes using XFEM. As the mesh becomes refined, we get higher values of K . Even though we do not obtain a fully convergent solution of K both for $Wi = 0.3$ and 0.6 in Table 2.3, the stabilization schemes developed for solving the numerical problems in viscoelastic fluid flows using the standard finite element behave quite well when used together with XFEM. The convergence behavior of the stresses on the cylinder surface will be further investigated in the following section 2.5.4.

In Table 2.4, we give the values of K for various Weissenberg numbers. Note that, in Table 2.4, we use the mesh M3 and compare the results with those of mesh M4 in [96], since the resolution of our mesh M3 is similar to M4 in [96]. We also show the dimensionless drag coefficient K_L computed by using the Lagrangian multipliers:

$$K_L = \frac{F_{L,x}}{\eta_0 UR}, \quad (2.57)$$

for XFEM and FDM, where $F_{L,x} = \int_{\Gamma_p} \boldsymbol{\lambda} \, ds \cdot \mathbf{e}_x$. As pointed out by Hwang et al. [73], Lagrangian multipliers represent the traction required to maintain the rigid body motion constraints on the interface. Hence, the integration of Lagrangian multipliers on the interface is the net hydrodynamic force acting on the rigid body. We also tested the accuracy of subdomain integration with the drag coefficient. As the level of subdivision increases, we obtain convergence of the drag coefficient. The difference between K 's obtained by using higher than three level subdivisions is small, similarly to the area calculations in Section 2.5.2.

Table 2.3: Dimensionless steady-state drag coefficient K

Wi	M1	M2	M3	M4
0.3	123.097	123.122	123.137	123.154
0.6	117.522	117.711	117.722	117.744

Table 2.4: Dimensionless steady-state drag coefficient for various Weissenberg numbers

Wi	FDM		XFEM		BFM	Ref. [96]
	K	K_L	K	K_L	K	K
0.0	80.262	132.426	132.302	132.357	132.358	132.358
0.1	74.368	130.354	130.306	130.360	130.362	130.363
0.2	72.350	126.690	126.569	126.622	126.623	126.626
0.3	70.399	123.303	123.137	123.187	123.188	123.193
0.4	68.883	120.722	120.538	120.588	120.588	120.596
0.5	67.819	118.953	118.773	118.824	118.822	118.836
0.6	67.142	117.872	117.722	117.774	117.771	117.792
0.7	66.780	117.352	117.259	117.314	117.309	117.340

A low level of subdivision does not seem to cause a stability problem even though the solution may be less accurate.

The drag coefficients obtained by BFM for various Wi show good agreement with the results of Hulsen et al. [96], which validates the proposed DEVSS-G/SUPG formulations. The values of K by XFEM are just slightly smaller than those of BFM for the entire range of Wi, but the difference seems negligible. However, the values of K by FDM are about 60 % of those of BFM. In FDM, the field variables - velocity, pressure and conformation, change continuously from the fluid domain to the rigid body. Hence, the values at the interface are somewhere in between the actual value and zero. This is the reason of the inaccurately predicted drag coefficient with FDM.

The values of K_L by XFEM are almost the same as those given by BFM. The values of K_L by FDM also show good agreement with the results of BFM, contrary to the values of K .

2.5.4 Stress along the cylinder surface

Let's consider the stresses on the upper cylinder surface and along the centerline in the wake. First, we show the results of mesh convergence. The coordinate ξ along the cylinder surface and the centerline is shown in Fig. 2.16; $\xi = 0$ at the front stagnation point, $\xi = \pi$ at the back stagnation point and $\xi = [\pi, 3\pi]$ along the centerline. In Fig. 2.17, we have plotted the non-dimensional polymer stress component $\tau_{xx}^* = \tau_{xx}/(\eta_0 U/R)$ along the coordinate ξ for $Wi = 0.3$. We solve the problem using three different meshes M1, M2 and M3, and using a boundary-fitted mesh (BFM) for a comparison. The stress obtained by using M1 shows small oscillations along the cylinder surface and a larger maximum value compared to that of BFM. By refining mesh resolutions to M2 and M3, we obtain smooth profiles of the stress component

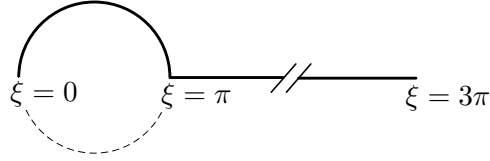


Figure 2.16: The curved coordinate ξ .

and convergence to the result of BFM. The results of M2, M3 and BFM are indistinguishable in Fig. 2.17. For a better comparison of the stress profiles using different meshes, we show the details of the stress profile on the top of the cylinder in Fig. 2.18, which clearly shows mesh convergence and also validates the proposed method.

We also solve the problem for a higher Weissenberg number $Wi=0.6$. In Fig. 2.19, we have plotted the non-dimensional polymer stress component τ_{xx}^* along the coordinate ξ for $Wi = 0.6$ obtained with three different meshes M2, M3 and M4. As the mesh becomes refined, the maximum stress on the cylinder decreases and the stress in the wake increases. It is the same trend as reported in [96], using boundary-fitted meshes. The results of the mesh M2 show small oscillations along the cylinder surface. They are reduced with mesh refinement. However, we cannot obtain a fully smooth profile for $Wi=0.6$ up to our most refined mesh, contrary to $Wi=0.3$. Obtaining a fully converged solution for a high Weissenberg number problem is a challenging task due to very thin boundary layers around the cylinder. This requires special meshing techniques such as an adaptive remeshing used by Gu enette et al. [114], which were able to obtain a converged solution up to a slightly higher value of $Wi = 0.7$ using boundary-fitted meshes. Such a special meshing technique for a high Weissenberg number problem is beyond the scope of this thesis. Nevertheless, the oscillations we find are minute and the local accuracy around the cylinder seems quite good with both meshes M3 and M4. Along the centerline of the wake, the results of M3 and M4 are almost the same and the two curves are indistinguishable in Fig 2.19.

Let's compare the results of XFEM with other numerical methods for $Wi=0.6$. Fig. 2.20 shows the stress component τ_{xxx}^* along the curved coordinate ξ using the XFEM, FDM and BFM. Note that both the results of XFEM and FDM in Fig. 2.20 are obtained using mesh M4. The result of XFEM shows good agreement with the result of BFM on the cylinder surface. Along the centerline of the wake, the results of XFEM and BFM are almost identical and the two curves are indistinguishable. The result of FDM shows low values and significant oscillations on the cylinder surface because of the continuous interpolation from the fluid domain into the cylinder domain. Moreover, the magnitude of the stress component is smaller, even in the wake. In the previous section, the accuracy of Lagrangian multipliers in FDM was verified but that does not guarantee the accuracy of stresses away from the interface. The stresses in the wake are probably affected by the inaccurate local solution on the interface.

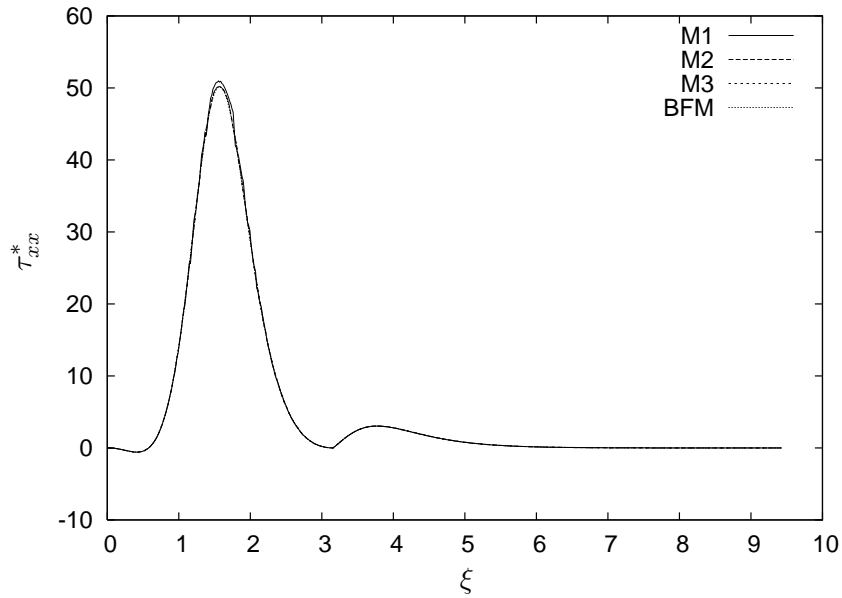


Figure 2.17: The non-dimensional polymer stress component τ_{xx}^* as a function of curved coordinate ξ for $Wi = 0.3$ using different meshes.

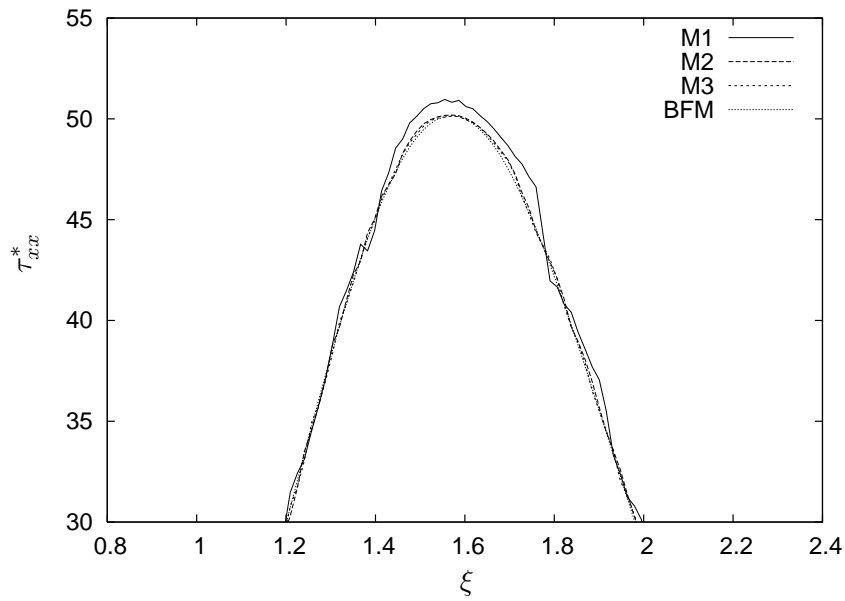


Figure 2.18: The magnified view of Fig. 2.17 on the top of the cylinder.

Hence, if closely positioned multiple rigid bodies are simulated using FDM, the prediction of interactions might be inaccurate because of under-estimated stresses around rigid bodies. The interactions of multiple rigid bodies are beyond the scope of this chapter.

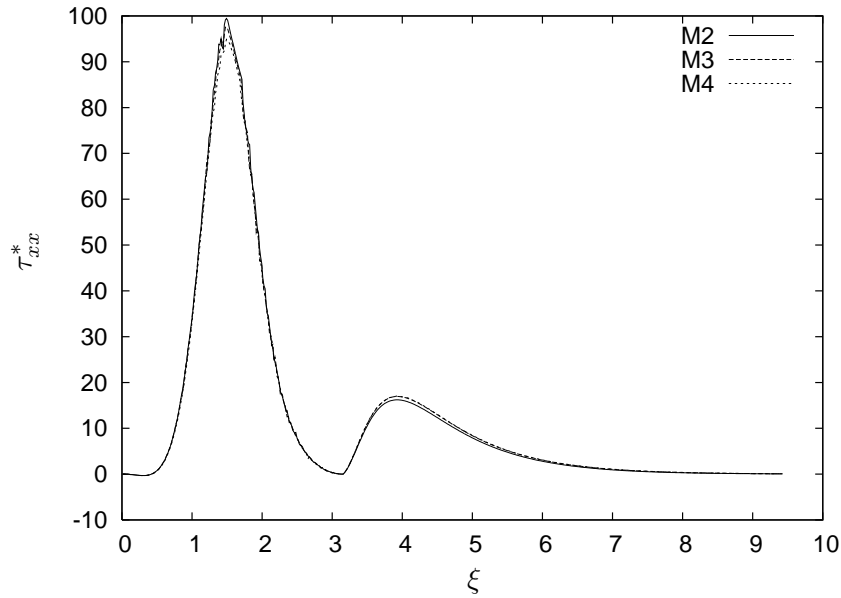


Figure 2.19: The non-dimensional polymer stress component τ_{xx}^* as a function of curved coordinate ξ for $Wi = 0.6$ using three different meshes.

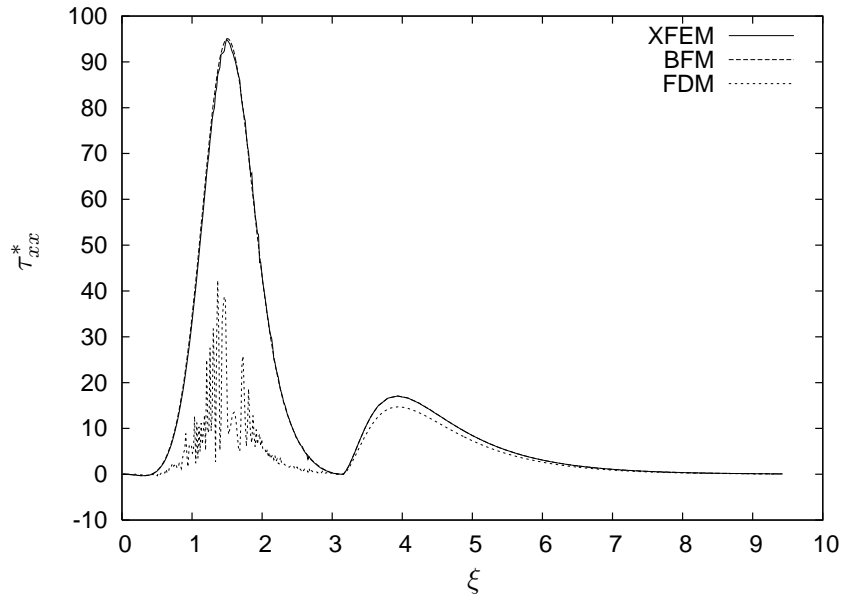


Figure 2.20: Comparison of τ_{xx}^* with different numerical methods for $Wi=0.6$.

2.6 Conclusions

In this chapter, we have presented the direct numerical simulation of viscoelastic flows around a stationary cylinder using an extended finite element method (XFEM) combined with the DEVSS-G/SUPG formulation. In this method, a regular (not boundary-fitted) mesh is

used for the whole computational domain including both fluid and rigid body. The finite element shape functions are extended by using virtual degrees of freedom for the description of the discontinuities around the interface. Then, intersected elements are integrated only on the fluid domain part, and degrees of freedoms associated to elements fully inside the rigid body are removed from the system. The no-slip boundary condition is imposed by using constraints implemented with Lagrangian multipliers. The accuracy and convergence of this scheme have been verified by comparing with boundary-fitted mesh problems. The results are also compared with fictitious domain methods.

The proposed XFEM shows significant improvement of local accuracy around the rigid body, compared with fictitious domain methods. An important difference with the fictitious domain method is that the degrees of freedom belonging to elements, or parts of elements, outside the fluid domain are eliminated from the system. This means that there is no fluid inside the rigid body and, therefore, also no false mass or viscous material responses from a fictitious fluid inside the rigid body. In the fictitious domain methods, significant oscillations of field variables occur around the interface, and local values are always under-estimated, contrary to Lagrangian multipliers. However, even if the Lagrangian multipliers are accurate, the stresses away from the rigid body can be inaccurate, as shown in the confined cylinder problem. In this situation, FDM may not give accurate results for the interactions between multiple rigid bodies. The motion of freely moving rigid bodies, and interactions of rigid bodies, are beyond the scope of this chapter.

The results of XFEM are almost the same as those of boundary-fitted mesh problems. For a fair comparison between XFEM and BFM, we need to note the computational time as well. For the problem of the channel flow around a cylinder, a boundary-fitted mesh can provide the benefit of minimal computational time because of easily applicable effective local mesh refinements. However, boundary-fitted meshes are not always preferable. For complicated flow geometries, caused e.g. by multiple or complex shaped rigid bodies, the generation of a boundary-fitted mesh is not an easy task, especially in three-dimensional flow and in the presence of free surfaces. When rigid bodies are moving in the flow field, remeshing is required at every time step to fit the new interface, and the solution should be projected from the previous mesh to the current mesh. For these cases, the proposed extended finite element method, with a regular mesh for the whole domain is easier to implement.

Chapter 3

Particle migration in circular Couette flow

In this chapter, we present an extended finite element method (XFEM) for the direct numerical simulation of the flow of viscoelastic fluids with suspended particles. To cope with the movement of particles, we devise a temporary arbitrary Lagrangian-Eulerian (ALE) scheme which defines the mapping of field variables at previous time levels onto the computational mesh at the current time level. In this method, a regular mesh is used for the whole computational domain including both fluid and particles, and particles are moving on a fixed Eulerian mesh without any need of remeshing. For mesh refinements around the interface, we combine XFEM with the grid deformation method, in which nodal points are redistributed close to the interface while preserving the mesh topology. Our method is verified by comparing with the results of boundary-fitted mesh problems combined with the conventional ALE scheme. The proposed method shows similar accuracy compared to boundary-fitted mesh problems and superior accuracy compared with the fictitious domain method. If the grid deformation method is combined with XFEM, the required computational time is reduced significantly compared with uniform mesh refinements, while providing mesh convergent solutions. We apply the proposed method to the particle migration in circular Couette flow of a Giesekus fluid. We investigate the effect of initial particle positions, the Weissenberg number, the mobility parameter of the Giesekus model and the particle size on the particle migration.

The content of this Chapter is based on:

Young Joon Choi, Martien A. Hulsen, and Han E.H. Meijer. An extended finite element method for the simulation of particulate viscoelastic flows. *Journal of Non-Newtonian Fluid Mechanics*, 165:607-624, 2010.

We also presented two-particle interactions in confined shear flow of a viscoelastic fluid in the original paper. However, the content is moved to Chapter 5 for a consistent presentation of the topic.

3.1 Introduction

Particulate flows arise in a wide class of research areas and industrial processes, for example, fluidized suspensions, slurry transport, materials separation, rate of mixing enhancement, etc. Particularly, in the processing of polymers, particles are often added to polymeric fluids to improve processing and/or product properties. In order to study the particle motion in viscoelastic fluids, numerical analysis is a basic requirement due to inherent nonlinear behavior of fluids. In the last decades, direct numerical simulation (DNS) techniques have been developed to predict the evolution of multi-particle structures from the imposition of flow. In DNS, hydrodynamic interactions are not modeled by empirical correlations but computed as a part of the solution.

In fluid-particle systems, field variables such as pressure and stress are discontinuous over the interface since no flow occurs inside particles. The most intuitive method to tackle this problem is using a mesh that is boundary-fitted, which means that the interface is aligned with element boundaries [58, 59, 60]. In this method, the governing equations are solved only in the fluid domain, taking into account the interface conditions on the boundaries of particles. Since the problems are solved only in the fluid domain, the solutions automatically contain discontinuous characteristics at the interface. Hence, no further considerations to handle jumps over the interface are necessary. To handle moving particles, this approach incorporates the arbitrary Lagrangian-Eulerian (ALE) technique [61] that uses a moving mesh scheme. The generation of a new mesh is needed if the old mesh becomes too distorted, and the solution must be projected onto the new mesh. The generation of boundary-fitted meshes is, however, still a challenging task in view of algorithms needed and computational costs if complex geometries are involved, especially in three-dimensional simulations.

An alternative approach is the fictitious domain method developed by Glowinski et al. [62, 63, 64]. The basic idea of this method is to fill each domain of every particle with the surrounding fluid, assuming and subsequently prescribing that the fluid inside the particle domain moves like a solid object. Hence, the problem is transferred from a geometrically complex fluid domain to a simpler domain including both fluid and particles, which eliminates the need of remeshing. In this method, particles move in a Lagrangian sense on a fixed Eulerian mesh. However, the motion of fluid near the interface is affected by the false mass, or the viscous material response, from the fictitious fluid inside the particle. Field variables at the interface are interpolated between the values possessed by the physical fluid and the fictitious fluid and, hence, discontinuities required at the interface are lost.

Recently, a new numerical method has been developed to capture discontinuous characteristics at the interface while still using a mesh which is not boundary-fitted. The method is usually called the extended finite element method (XFEM) because the finite element shape functions are locally extended, or enriched, by using additional degrees of freedom to handle discontinuities. Originally, XFEM was developed for the simulations of cracks in solids without the need of remeshing [90, 91], then it was applied to flow problems as well [92, 93, 115]. A recent review on extended finite element methods applied to material modeling is presented in [94]. In the present chapter, we consider the motion of a viscoelastic fluid with freely sus-

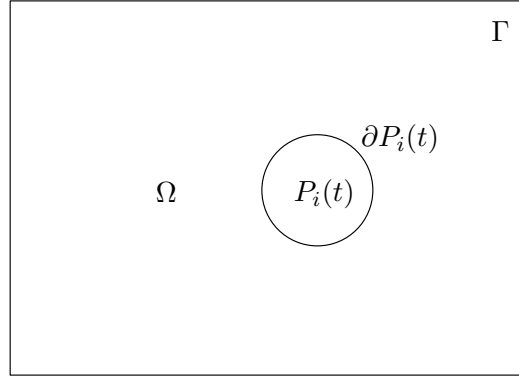


Figure 3.1: Schematic description of the fluid-particle system.

pendent rigid particles. We propose an XFEM with temporary arbitrary Lagrangian-Eulerian scheme to handle moving particles without any need of remeshing throughout the whole computations.

The content of this chapter is as follows. In Section 3.2, we give a brief review of the governing equations for the motion of incompressible viscoelastic fluids and rigid particles. In Section 3.3, we present the numerical algorithm of the extended finite element method combined with a temporary arbitrary Lagrangian-Eulerian scheme and the grid deformation method. In Section 3.4, we present a validation of the proposed method by comparing with the results of boundary-fitted mesh problem of particle migration in shear flow. In Section 3.5, we present numerical results for the particle migration in rotating Couette flow. A discussion follows in Section 3.6.

3.2 Governing equations

We consider the motion of rigid particles suspended in an incompressible viscoelastic fluid as shown in Fig. 3.1. Let Ω be the entire domain including the fluid and particles, and let $P_i(t)$ ($i = 1, \dots, N$) be the embedded domain of the i -th particle at time t with the number of particles N . The collective particle region at a certain time t is denoted by $P(t) = \bigcup_{i=1}^N P_i(t)$. Boundaries are denoted by $\Gamma = \partial\Omega$ and $\partial P_i(t)$. We will neglect inertia and body forces for both the fluid and the particles.

The equations of motion for an incompressible fluid are given by

$$\nabla \cdot \boldsymbol{\sigma} = \mathbf{0} \quad \text{in } \Omega \setminus P(t), \quad (3.1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \setminus P(t), \quad (3.2)$$

where $\boldsymbol{\sigma}$ is the stress tensor and \mathbf{u} is the velocity vector. Eqs. (3.1) and (3.2) are the balance equations of momentum and mass, respectively. For a viscoelastic fluid, the total stress $\boldsymbol{\sigma}$ can be split into a pressure part, a Newtonian viscous stress and a polymer stress:

$$\boldsymbol{\sigma} = -p\mathbf{I} + 2\eta_s\mathbf{D} + \boldsymbol{\tau}(\mathbf{c}), \quad (3.3)$$

where p is the pressure, \mathbf{I} is the unity tensor, η_s is the Newtonian solvent viscosity, $\mathbf{D} = (\nabla\mathbf{u} + (\nabla\mathbf{u})^T)/2$ is the rate-of-deformation tensor and $\boldsymbol{\tau}(\mathbf{c})$ is the polymer stress which can be represented as a function of the conformation tensor \mathbf{c} . We will use the Giesekus constitutive model for the conformation tensor \mathbf{c} :

$$\lambda \overset{\nabla}{\mathbf{c}} + \mathbf{c} - \mathbf{I} + \alpha(\mathbf{c} - \mathbf{I})^2 = \mathbf{0}, \quad (3.4)$$

where λ is the relaxation time and α is the mobility parameter. The triangle (∇) denotes the upper-convected time derivative, defined as:

$$\overset{\nabla}{\mathbf{c}} = \frac{\partial \mathbf{c}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{c} - (\nabla \mathbf{u})^T \cdot \mathbf{c} - \mathbf{c} \cdot \nabla \mathbf{u}. \quad (3.5)$$

In the Giesekus model, the polymer stress can be written as:

$$\boldsymbol{\tau}(\mathbf{c}) = \frac{\eta_p}{\lambda}(\mathbf{c} - \mathbf{I}), \quad (3.6)$$

where η_p is the polymer viscosity. Furthermore, the equation for the conformation tensor \mathbf{c} , Eq. (3.4), can be transformed to an equivalent log-conformation equation for $\mathbf{s} = \log \mathbf{c}$:

$$\frac{\partial \mathbf{s}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{s} - \mathbf{g}((\nabla \mathbf{u})^T, \mathbf{s}) = \mathbf{0}. \quad (3.7)$$

Once \mathbf{s} has been solved, the conformation tensor \mathbf{c} can be computed from $\mathbf{c} = \exp(\mathbf{s})$. A detailed explanation of the log-conformation representation (LCR) can be found in [95, 96]. Solving the equation for \mathbf{s} instead of the equation for \mathbf{c} leads to major stability improvements for high Weissenberg number problems. For all numerical simulations in this chapter, we use the log-conformation representation. Hence, we will present the weak formulation, and the initial and boundary conditions only in the LCR.

The boundary conditions are given by:

$$\mathbf{u} = \bar{\mathbf{u}}(t) \quad \text{on } \Gamma_D, \quad (3.8)$$

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \mathbf{t} \quad \text{on } \Gamma_N, \quad (3.9)$$

$$\mathbf{u} = \mathbf{U}_i + \boldsymbol{\omega}_i \times (\mathbf{x} - \mathbf{X}_i) \quad \text{on } \partial P_i(t) \quad (i = 1, \dots, N), \quad (3.10)$$

$$\mathbf{s} = \mathbf{s}_{\text{in}} \quad \text{on } \Gamma_{\text{in}}. \quad (3.11)$$

Eqs. (3.8) and (3.9) express that the boundary conditions on $\Gamma = \Gamma_D \cup \Gamma_N$ can be split into Dirichlet boundary conditions on Γ_D and Neumann boundary conditions on Γ_N . In Eq. (3.9), \mathbf{t} is the traction and \mathbf{n} is the outwardly directed unit normal vector on Γ_N . Eq. (3.10) represents the no-slip boundary condition on the particle surface, where \mathbf{U}_i is the translational velocity, $\boldsymbol{\omega}_i$ is the rotational velocity and \mathbf{X}_i is the position of the center of the particle. Due to the hyperbolic character of the constitutive equation (Eq. (3.7)), we also need a boundary condition for \mathbf{s} (Eq. (3.11)) at the inflow boundary Γ_{in} where $\mathbf{u} \cdot \mathbf{n} < 0$.

Since the evolution equation of the conformation tensor is time-dependent, we also need

an initial condition for \mathbf{s} :

$$\mathbf{s}(\mathbf{x}, 0) = \mathbf{s}_0(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega \setminus P(0). \quad (3.12)$$

In our simulations, we use a stress-free state as initial condition over the whole fluid domain, i.e. $\mathbf{s}_0(\mathbf{x}) = \mathbf{0}, \forall \mathbf{x} \in \Omega \setminus P(0)$. Since we neglect inertia, an initial condition for the velocity is not necessary.

For the unknown rigid body motions $(\mathbf{U}_i, \boldsymbol{\omega}_i)$ of the particles, we need balance equations for forces and torques on particle surfaces. In the absence of inertia, and external forces and torques on the particles, the net hydrodynamic force \mathbf{F}_i and torque \mathbf{T}_i acting on the particle $P_i(t)$ are zero:

$$\mathbf{F}_i = \int_{\partial P_i(t)} \boldsymbol{\sigma} \cdot \hat{\mathbf{n}} \, ds = \mathbf{0}, \quad (3.13)$$

$$\mathbf{T}_i = \int_{\partial P_i(t)} (\mathbf{x} - \mathbf{X}_i) \times (\boldsymbol{\sigma} \cdot \hat{\mathbf{n}}) \, ds = \mathbf{0}, \quad (3.14)$$

where $\hat{\mathbf{n}}$ is the outwardly directed unit normal vector on $\partial P_i(t)$. The particle positions \mathbf{X}_i and angular orientations $\boldsymbol{\Theta}_i$ are obtained from the following kinematic equations:

$$\frac{d\mathbf{X}_i}{dt} = \mathbf{U}_i, \quad \mathbf{X}_i(t=0) = \mathbf{X}_{i,0}, \quad (3.15)$$

$$\frac{d\boldsymbol{\Theta}_i}{dt} = \boldsymbol{\omega}_i, \quad \boldsymbol{\Theta}_i(t=0) = \boldsymbol{\Theta}_{i,0}. \quad (3.16)$$

3.3 Numerical methods

3.3.1 Mixed DEVSS-G/SUPG finite element formulation

In deriving the weak form of the governing equations for a fluid-particle system, we follow the combined equation of motion approach [64], in which the fluid and the particle equation of motion are combined in one equation. The main advantage is that the hydrodynamic forces and torques acting on particles are eliminated from the equation of motion since they are internal. For the discretization of the momentum equation, we employ the DEVSS-G method [98, 99, 101]. Furthermore, the SUPG method [102] is used for the discretization of the constitutive equation. The weak formulation of the governing equations, with the DEVSS-G/SUPG stabilization technique, can be stated as follows: Find \mathbf{u} , p , \mathbf{s} , \mathbf{G} , $\boldsymbol{\lambda}_i$, \mathbf{U}_i and $\boldsymbol{\omega}_i$ such that

$$\begin{aligned} & - (\nabla \cdot \mathbf{v}, p) + ((\nabla \mathbf{v})^T, 2\eta_s \mathbf{D} + \theta(\nabla \mathbf{u} - \mathbf{G}^T) + \boldsymbol{\tau}(c)) \\ & \quad + (\mathbf{v} - (\mathbf{V}_i + \boldsymbol{\chi}_i \times (\mathbf{x} - \mathbf{X}_i)), \boldsymbol{\lambda}_i)_{\partial P_i(t)} = (\mathbf{v}, \mathbf{t})_{\Gamma_N}, \end{aligned} \quad (3.17)$$

$$(q, \nabla \cdot \mathbf{u}) = 0, \quad (3.18)$$

$$(\mathbf{d} + \tau \mathbf{u} \cdot \nabla \mathbf{d}, \frac{\partial \mathbf{s}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{s} - \mathbf{g}(\mathbf{G}, \mathbf{s})) = 0, \quad (3.19)$$

$$(\mathbf{H}, -\nabla \mathbf{u} + \mathbf{G}^T) = 0, \quad (3.20)$$

$$(\boldsymbol{\mu}_i, \mathbf{u} - (\mathbf{U}_i + \boldsymbol{\omega}_i \times (\mathbf{x} - \mathbf{X}_i)))_{\partial P_i(t)} = 0, \quad (3.21)$$

for all admissible test functions \mathbf{v} , q , \mathbf{d} , \mathbf{H} , $\boldsymbol{\mu}_i$, \mathbf{V}_i and $\boldsymbol{\chi}_i$ where (\cdot, \cdot) , $(\cdot, \cdot)_{\Gamma_N}$ and $(\cdot, \cdot)_{\partial P_i(t)}$ are proper inner products on the fluid domain $\Omega \setminus P(t)$, on the Neumann boundary Γ_N and on the interface $\partial P_i(t)$, respectively. The no-slip boundary condition on the rigid body surface is imposed by using constraints implemented with Lagrange multipliers. In Eq. (3.17), the DEVSS-G parameter θ is chosen equal to the polymer viscosity, $\theta = \eta_p$. In Eq. (3.19), the SUPG parameter τ is given by $\tau = \bar{\beta} h / 2U_c$, where $\bar{\beta}$ is a non-dimensional parameter, h is a typical size of the element in the direction of velocity and U_c is a characteristic velocity magnitude for the flow problem. In this chapter, we choose the characteristic velocity U_c as the magnitude of the velocity in the integration points:

$$U_c = \|\mathbf{u}_j\|, \quad j = 1, \dots, J \quad (3.22)$$

where J is the number of integration points in the element, and a $\bar{\beta}$ parameter as a function of the Courant number $C = U_c \Delta t / h$:

$$\bar{\beta} = \begin{cases} +1 & \text{if } C \geq 1, \\ C & \text{if } C < 1, \end{cases} \quad (3.23)$$

where Δt is the computational time step. Note that

$$\tau = \begin{cases} h/2U_c & \text{if } C \geq 1, \\ \Delta t/2 & \text{if } C < 1. \end{cases} \quad (3.24)$$

3.3.2 The extended finite element method

Enrichment strategies

We introduce the enrichment of the finite element approximations through the partition of unity method (PUM) [89]. Let's consider the "one-sided" discontinuity across an interface, which is basically a domain Ω^+ in which a problem is to be solved. Outside this domain, thus in Ω^- , the field variables are nonexistent, but are taken to be zero for simplicity. The outside domain Ω^- can be considered as holes or voids, similarly to [103]. Therefore, there is a jump at the interface between Ω^+ and Ω^- . The approximation can be written as:

$$V = \sum_k \phi_k(\mathbf{x}) H(s) a_k, \quad \forall \mathbf{x} \in \Omega, \quad (3.25)$$

where a_k are degrees of freedom and $H(s)$ is a Heaviside function defined by a scalar level set function s :

$$H(s) = \begin{cases} +1 & \text{if } s \geq 0, \\ 0 & \text{if } s < 0, \end{cases} \quad (3.26)$$

$$s(\mathbf{x}) \begin{cases} > 0 & \text{if } \mathbf{x} \in \Omega^+, \\ < 0 & \text{if } \mathbf{x} \in \Omega^-, \\ = 0 & \text{if } \mathbf{x} \text{ is on the interface.} \end{cases} \quad (3.27)$$

The interface is represented as the zero level contour of a level set function. Since elements fully outside the domain have no contribution to the system, nodes connected only to such elements can be discarded from the system. All other nodes retain the degree of freedom a_k . The degrees of freedom inside the domain have the same meaning as the standard finite element method, whereas the degrees of freedom outside the domain are *virtual* degrees of freedom, which are defining the spatial distribution of field variables inside the domain from the outside. They do not define field variables at that point. Then, the integration of the weak form is only performed on the inside domain Ω^+ .

For our fluid-particle system depicted in Fig. 3.1, we have a jump at the interface from physical quantities of the velocity, pressure and conformation in fluid domain $\Omega^+ = \Omega \setminus P(t)$ to zero or essentially nothing in particle domain $\Omega^- = P(t)$, as no flow occurs inside particles. Hence, the velocity, pressure, log-conformation and velocity gradient projection can be discretized as:

$$\mathbf{u}_h(\mathbf{x}) = \sum_k \varphi_k(\mathbf{x}) H(s) \mathbf{u}_k, \quad (3.28)$$

$$p_h(\mathbf{x}) = \sum_k \psi_k(\mathbf{x}) H(s) p_k, \quad (3.29)$$

$$\mathbf{s}_h(\mathbf{x}) = \sum_k \psi_k(\mathbf{x}) H(s) \mathbf{s}_k, \quad (3.30)$$

$$\mathbf{G}_h(\mathbf{x}) = \sum_k \psi_k(\mathbf{x}) H(s) \mathbf{G}_k, \quad (3.31)$$

where

$$H(s) = \begin{cases} +1 & \text{if } s \geq 0, \\ 0 & \text{if } s < 0, \end{cases} \quad (3.32)$$

is defined by a level set function s :

$$s(\mathbf{x}) \begin{cases} = 0 & \text{if } \mathbf{x} \text{ is on } \partial P(t), \\ > 0 & \text{if } \mathbf{x} \text{ is in } \Omega \setminus P(t), \\ < 0 & \text{if } \mathbf{x} \text{ is in } P(t). \end{cases} \quad (3.33)$$

We use a bi-quadratic Q_2 interpolation (φ_k) for the velocity \mathbf{u} ; and a bi-linear Q_1 interpolation (ψ_k) for the pressure p , log-conformation tensor \mathbf{s} and velocity gradient projection \mathbf{G} . We

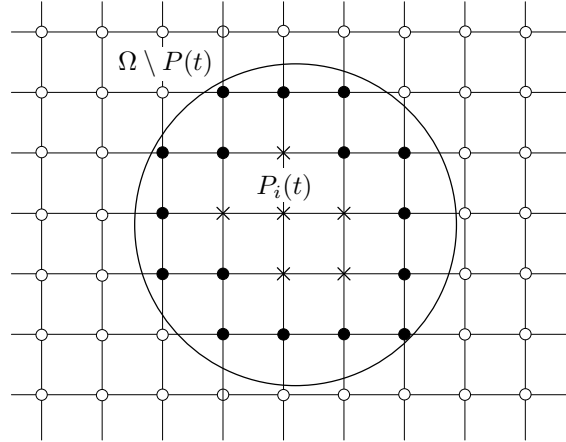


Figure 3.2: An enrichment scheme for a fluid-particle system. (o: standard degree of freedom, •: virtual degree of freedom, ×: removed degree of freedom)

use a signed distance function d as a level set function s .

A simple interpretation of the XFEM enrichment procedure for a fluid-particle system is shown in Fig. 3.2. Elements fully inside the fluid domain $\Omega \setminus P(t)$ have only standard degrees of freedom (o) and are fully integrated element by element. The standard degrees of freedom (o) are the same as the standard finite element method and they define the field variable at that point. The elements intersected by the interface have both standard degrees of freedom and virtual degrees of freedom (•) for the jump enrichment. Intersected elements are integrated only on the fluid domain part. For the integration on the part of an element, we use a quadtree subdivision and further triangulation of the smallest quads near the interface as shown in Fig. 3.3. The algorithm of triangulation is based on the level set function for the interface and can be found in [106]. In simulations, we use five level subdivisions since those give accurate integrations with manageable computational times required. The virtual degrees of freedom (•) do not define the field variable at that point, but define the spatial distribution of the field variable in the part of an intersected element. In other words, they are only used for interpolating values at the fluid domain part of an intersected element. Elements fully inside the rigid body are ignored and they are not assembled into the system matrix. Therefore, the degrees of freedom only belonging to elements fully inside the rigid body domain are the removed degrees of freedom (×). A practical implementation of the removed degrees of freedom (×) is treating them like Dirichlet boundary conditions and setting them to zero.

For elements intersected by the interface, the imposition of a no-slip boundary condition (Eq. (3.10)) is needed. This can be done by using constraints implemented with Lagrange multipliers as shown in Eqs. (3.17) and (3.21). The inner product $(\cdot, \cdot)_{\partial P_i(t)}$ is the standard inner product in $L^2(\partial P_i(t))$:

$$(\boldsymbol{\mu}_i, \mathbf{u} - (\mathbf{U}_i + \boldsymbol{\omega}_i \times (\mathbf{x} - \mathbf{X}_i)))_{\partial P_i(t)} = \int_{\partial P_i(t)} \boldsymbol{\mu}_i \cdot (\mathbf{u} - (\mathbf{U}_i + \boldsymbol{\omega}_i \times (\mathbf{x} - \mathbf{X}_i))) ds. \quad (3.34)$$

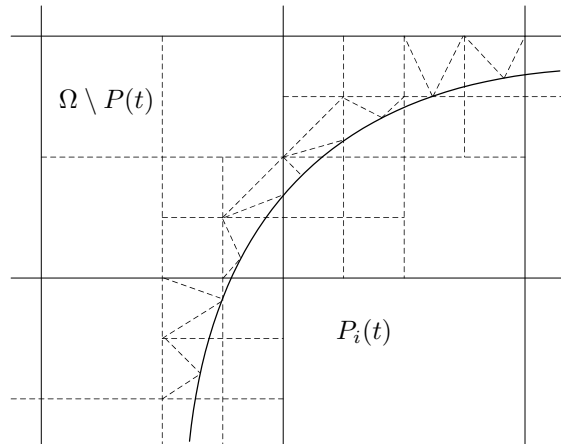


Figure 3.3: Subdivision of elements: two level quadtree subdivision and triangulation of the smallest quads near the interface.

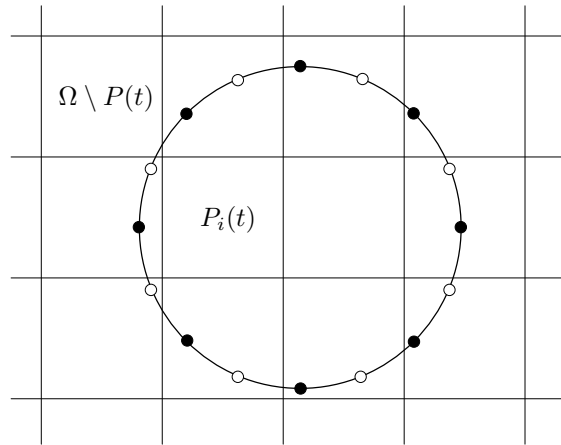


Figure 3.4: A mesh for the discretization of the boundary of a particle. P_1 interpolation for the discretization of Lagrange multipliers and P_2 interpolation for the geometrical shape. Each element has two end nodes (●) and one midpoint node (○). Midpoint nodes (○) do not have a degree of freedom for Lagrange multipliers.

For the discretization of Eq. (3.34), we use a linear shape function P_1 for the discretization of Lagrange multipliers and a quadratic shape function P_2 for the geometrical shape of each element as shown in Fig. 3.4. Each element on the boundary of a particle has two end nodes (●) and one midpoint node (○). Only end nodes (●) have degrees of freedom for Lagrange multipliers. Midpoint nodes (○) together with end nodes are used for the geometrical description of an element such as an isoparametric mapping of a real element into a reference element. We choose the size of line elements for the description of a particle similar to the size of fluid elements near the interface. A particle element is divided into a number of subintervals and the integration is carried on each subinterval, combined with a subsequent composite integration rule applied on the element. In this chapter, we use 20 subintervals and a midpoint rule in each subinterval. The accuracy of the integration using subintervals

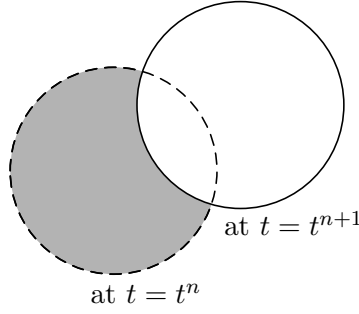


Figure 3.5: Field variables in the shaded region are undefined at time level $t = t^n$.

can be found in [116].

Since the (log-)conformation equation (Eq. (3.7)) is time-dependent, we need a time integration scheme for the conformation tensor, for example, a first-order scheme:

$$\frac{\partial \mathbf{s}}{\partial t} \approx \frac{\mathbf{s}^{n+1} - \mathbf{s}^n}{\Delta t} \quad (3.35)$$

where \mathbf{s}^{n+1} is the unknown at the current time level t^{n+1} , \mathbf{s}^n is the solution at the previous time level t^n and $\Delta t = t^{n+1} - t^n$ is the time step. However, for a moving particle problem, the field variable at the previous time level, \mathbf{s}^n , can become undefined near the boundary of the particle since there was no fluid flow at time level t^n as shown in Fig. 3.5. We will treat this problem in the next section.

3.3.3 Application to moving particles

Basic idea: first-order ALE scheme

We propose a *temporary* arbitrary Lagrangian-Eulerian technique using a mesh movement scheme similar to [59, 60]. In this method, mesh nodes near a particle follow the motion of the particle, whereas, mesh nodes far away from the particle are stationary. To guarantee a smoothly varying distribution of the nodes, a mesh velocity field \mathbf{u}_m is solved using Laplace's equation:

$$\nabla^2 \mathbf{u}_m = \mathbf{0} \quad \text{in } \Omega, \quad (3.36)$$

$$\mathbf{u}_m = \mathbf{0} \quad \text{on } \Gamma, \quad (3.37)$$

$$\mathbf{u}_m = \mathbf{U}_i + \boldsymbol{\omega}_i \times (\mathbf{x} - \mathbf{X}_i) \quad \text{on } \partial P_i(t). \quad (3.38)$$

For a circular particle, Eq. (3.38) can be replaced by $\mathbf{u}_m = \mathbf{U}_i$. Note that Eqs. (3.36)-(3.38) are solved on an Eulerian mesh including the particle domain $P(t)$ by using a similar technique as in the fictitious domain method [63]. Then, a temporary arbitrary Lagrangian-Eulerian mesh at the current time level, $\mathbf{x}_{\text{ALE}}^{n+1}$, is constructed by the following advection equation:

$$\mathbf{x}_{\text{ALE}}^{n+1} = \mathbf{x}_m^n + \mathbf{u}_m(\mathbf{x}_m^n, t^n) \Delta t \quad (3.39)$$

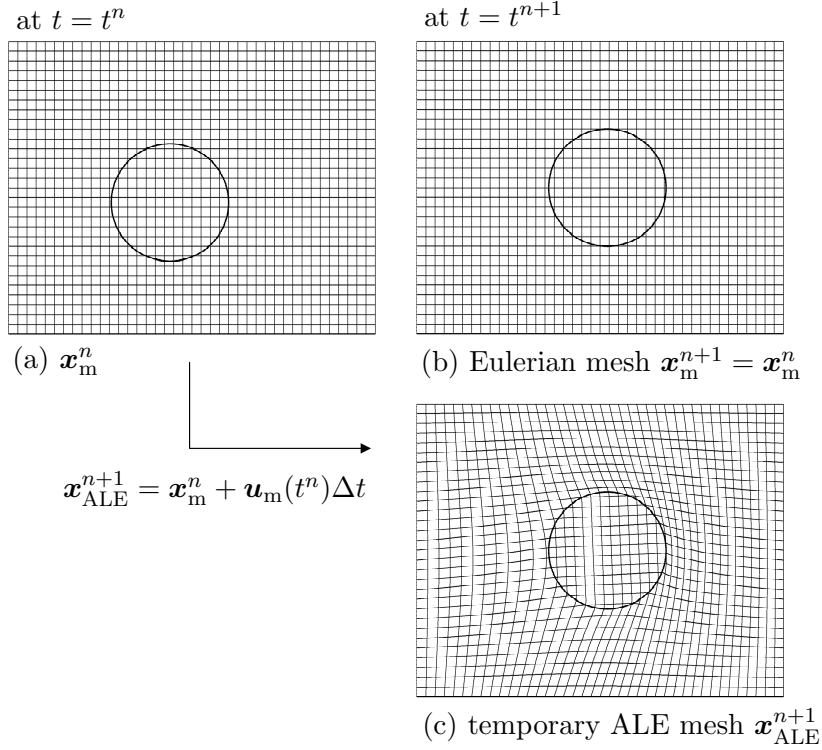
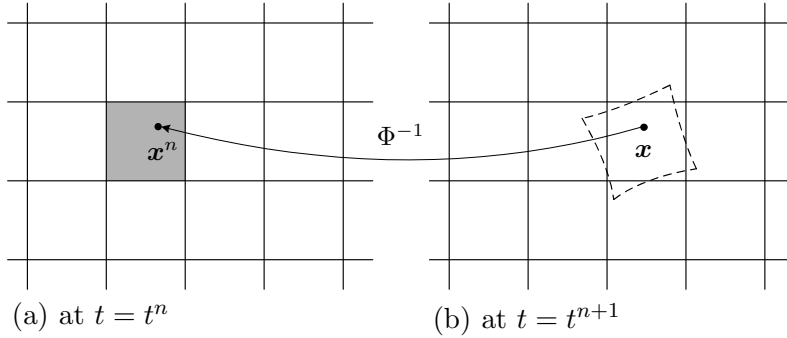


Figure 3.6: First-order ALE scheme.

Figure 3.7: Mapping using a first-order ALE scheme. The dashed element at $t = t^{n+1}$ represents the advection of the shaded element at $t = t^n$.

The construction of the Eulerian mesh \mathbf{x}_{ALE}^{n+1} is shown in Fig. 3.6. By using an ALE technique and a semi-implicit Euler scheme, the time discretization of the log-conformation equation (Eq. (3.7)) can be written as:

$$\frac{\mathbf{s}^{n+1}}{\Delta t} + (\mathbf{u}^n - \mathbf{u}_m^n) \cdot \nabla \mathbf{s}^{n+1} = \frac{\mathbf{s}^n}{\Delta t} + \mathbf{g}((\nabla \mathbf{u}^n)^T, \mathbf{s}^n). \quad (3.40)$$

In Eq. (3.40), the unknown at the current time level, \mathbf{s}^{n+1} , is computed on an Eulerian mesh, whereas, field variables at the previous time level, such as \mathbf{u}^n and \mathbf{s}^n , are mapped along with the ALE mesh \mathbf{x}_{ALE}^{n+1} . The mapping of field variables at the previous time level is shown in

Fig. 3.7. Let's consider the log-conformation tensor $\mathbf{s}^n = \mathbf{s}(\mathbf{x}^n, t^n)$, for example:

$$\mathbf{s}^n = \mathbf{s}(\mathbf{x}^n(\mathbf{x}), t^n) \quad (3.41)$$

$$= \mathbf{s}(\Phi^{-1}(\mathbf{x}), t^n) \quad (3.42)$$

The mapping Φ is the advection of the ALE mesh given by Eq.(3.39). Hence, the inverse mapping Φ^{-1} can be easily implemented in the finite element context: find the reference coordinates of \mathbf{x} in the ALE mesh $\mathbf{x}_{\text{ALE}}^{n+1}$, then interpolate the values of \mathbf{s} at the same reference coordinates in the previous computational mesh \mathbf{x}_{m}^n (see Fig. 3.6). Similarly, the fluid velocity and the mesh velocity at the previous time level is mapped by

$$\mathbf{u}^n = \mathbf{u}(\Phi^{-1}(\mathbf{x}), t^n), \quad (3.43)$$

$$\mathbf{u}_{\text{m}}^n = \mathbf{u}_{\text{m}}(\Phi^{-1}(\mathbf{x}), t^n). \quad (3.44)$$

Since we use a backward differencing time discretization of the log-conformation tensor in Eq. (3.40), the gradient of the log-conformation tensor at the previous time level, $\nabla \mathbf{s}^n$, is not necessary. Moreover, since we use a DEVSS-G method in the weak form, $\nabla \mathbf{u}^n$ is replaced by \mathbf{G}^n (see Eq. (3.19)) which is also mapped along with the movement of the ALE mesh. Hence, we do not need to calculate gradients of field variables at the previous time level, such as $\nabla \mathbf{u}^n$ and $\nabla \mathbf{s}^n$. We only need \mathbf{u}^n , \mathbf{s}^n and \mathbf{G}^n . At the next time level $t = t^{n+2}$, we delete the ALE mesh $\mathbf{x}_{\text{ALE}}^{n+1}$, and then construct a new temporary ALE mesh $\mathbf{x}_{\text{ALE}}^{n+2}$.

In the original ALE scheme for a fluid-particle system [59, 60], a boundary-fitted computational mesh itself is moved according to the mesh velocity field \mathbf{u}_{m} . Hence, after several time steps, the mesh can become too much distorted. If this happens, a generation of a new mesh is needed and the solution must be projected onto the new mesh. However, in our method, an ALE mesh is constructed separately and the computational mesh is kept unchanged throughout the whole computation. Note that particles are moving on a fixed Eulerian mesh. So there is no need for remeshing.

Extension to a second-order scheme

The idea introduced in the previous section can easily be extended to higher-order schemes. In this section, we describe a second-order scheme of a temporary ALE technique (Fig. 3.8). An ALE mesh at the previous time level $t = t^n$, $\mathbf{x}_{\text{ALE}}^n$, is constructed using a predictor-corrector method:

$$\mathbf{x}_{\text{m}}^* = \mathbf{x}_{\text{m}}^{n-1} + \mathbf{u}_{\text{m}}(\mathbf{x}_{\text{m}}^{n-1}, t^{n-1})\Delta t \quad (\text{predictor}) \quad (3.45)$$

$$\mathbf{x}_{\text{ALE}}^n = \mathbf{x}_{\text{m}}^{n-1} + \frac{1}{2}[\mathbf{u}_{\text{m}}(\mathbf{x}_{\text{m}}^*, t^n) + \mathbf{u}_{\text{m}}(\mathbf{x}_{\text{m}}^{n-1}, t^{n-1})]\Delta t \quad (\text{corrector}) \quad (3.46)$$

Note that \mathbf{u}_{m} at $t = t^n$ is defined on the fixed Eulerian mesh, and $\mathbf{u}_{\text{m}}(\mathbf{x}_{\text{m}}^*, t^n)$ is interpolated from the Eulerian mesh at the point \mathbf{x}_{m}^* . Then, the ALE mesh at the current time level,

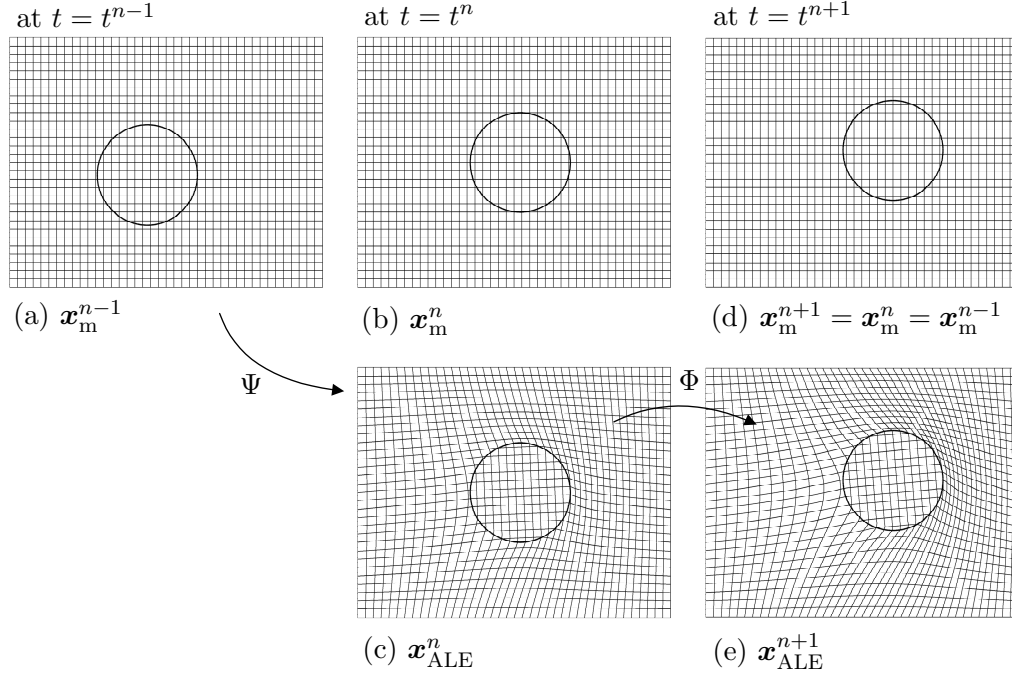


Figure 3.8: Second-order ALE scheme.

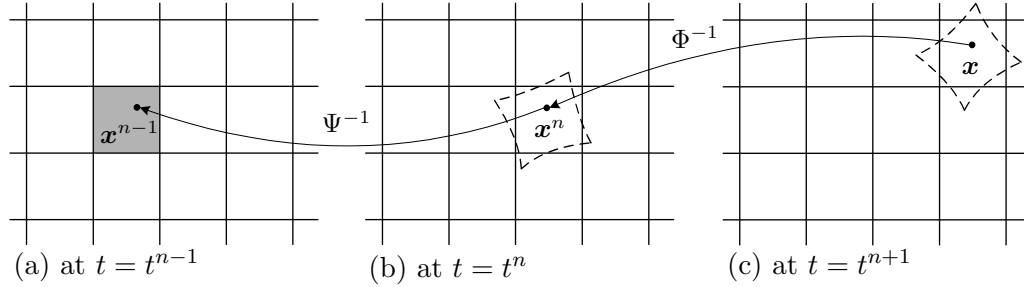


Figure 3.9: Mapping using a second-order ALE scheme.

$\mathbf{x}_{\text{ALE}}^{n+1}$, can be constructed using a second-order Adams-Bashforth method (AB2):

$$\mathbf{x}_{\text{ALE}}^{n+1} = \mathbf{x}_{\text{ALE}}^n + \left[\frac{3}{2} \mathbf{u}_m(\mathbf{x}_{\text{ALE}}^n, t^n) - \frac{1}{2} \mathbf{u}_m(\mathbf{x}_m^{n-1}, t^{n-1}) \right] \Delta t \quad (\text{AB2}) \quad (3.47)$$

Eqs. (3.45) and (3.46) define the mapping Ψ and Eq. (3.47) defines the mapping Φ . Ψ and Φ are represented in Fig. 3.8.

For the time discretization of the log-conformation equation (Eq. (3.7)), we use a semi-implicit Gear scheme:

$$\frac{3}{2} \frac{\mathbf{s}^{n+1}}{\Delta t} + \hat{\mathbf{u}}^{n+1} \cdot \nabla \mathbf{s}^{n+1} = \frac{2\mathbf{s}^n - \frac{1}{2}\mathbf{s}^{n-1}}{\Delta t} + 2\mathbf{g}((\nabla \mathbf{u}^n)^T, \mathbf{s}^n) - \mathbf{g}((\nabla \mathbf{u}^{n-1})^T, \mathbf{s}^{n-1}) \quad (3.48)$$

where $\hat{\mathbf{u}}^{n+1} = 2(\mathbf{u}^n - \mathbf{u}_m^n) - (\mathbf{u}^{n-1} - \mathbf{u}_m^{n-1})$. As in the first order scheme, in Eq. (3.48), the unknown at the current time level, \mathbf{s}^{n+1} , is computed on the Eulerian mesh, whereas, field

variables at previous time levels, such as \mathbf{u}^n , \mathbf{u}^{n-1} , \mathbf{s}^n , \mathbf{s}^{n-1} , \mathbf{G}^n and \mathbf{G}^{n-1} are mapped along with the ALE meshes (Fig. 3.9):

$$\mathbf{s}^n = \mathbf{s}(\Phi^{-1}(\mathbf{x}), t^n), \quad (3.49)$$

$$\mathbf{u}^n = \mathbf{u}(\Phi^{-1}(\mathbf{x}), t^n), \quad (3.50)$$

$$\mathbf{u}_m^n = \mathbf{u}_m(\Phi^{-1}(\mathbf{x}), t^n), \quad (3.51)$$

$$\mathbf{G}^n = \mathbf{G}(\Phi^{-1}(\mathbf{x}), t^n), \quad (3.52)$$

$$\mathbf{s}^{n-1} = \mathbf{s}(\Psi^{-1} \circ \Phi^{-1}(\mathbf{x}), t^{n-1}), \quad (3.53)$$

$$\mathbf{u}^{n-1} = \mathbf{u}(\Psi^{-1} \circ \Phi^{-1}(\mathbf{x}), t^{n-1}), \quad (3.54)$$

$$\mathbf{u}_m^{n-1} = \mathbf{u}_m(\Psi^{-1} \circ \Phi^{-1}(\mathbf{x}), t^{n-1}), \quad (3.55)$$

$$\mathbf{G}^{n-1} = \mathbf{G}(\Psi^{-1} \circ \Phi^{-1}(\mathbf{x}), t^{n-1}). \quad (3.56)$$

As explained in the previous section, since we use a DEVSS-G method, $\nabla \mathbf{u}^n$ and $\nabla \mathbf{u}^{n-1}$ are replaced by \mathbf{G}^n and \mathbf{G}^{n-1} , respectively, in the weak form (Eq. (3.19)). Still we do not need to calculate gradients of field variables at the previous time levels, such as $\nabla \mathbf{u}^n$, $\nabla \mathbf{u}^{n-1}$, $\nabla \mathbf{s}^n$ and $\nabla \mathbf{s}^{n-1}$. At the next time level $t = t^{n+2}$, we construct new temporary ALE meshes $\mathbf{x}_{\text{ALE}}^{n+1}$ and $\mathbf{x}_{\text{ALE}}^{n+2}$.

3.3.4 Mesh refinements using a grid deformation method

In the previous sections, we introduced an extended finite element method for the simulation of the flow of particulate viscoelastic fluids using a temporary ALE technique. Since the ALE mesh is advected according to the mesh velocity at previous time levels, it is not coupled with the current computational mesh. Hence, we can freely change the current computational mesh without regarding the temporary ALE mesh. In other words, we can simply apply local mesh refinements on the computational mesh without any need of further considerations on the ALE mesh.

In this chapter, for the local mesh refinements along the interface, we use a grid deformation method [117], in which mesh nodes are redistributed while preserving the mesh topology

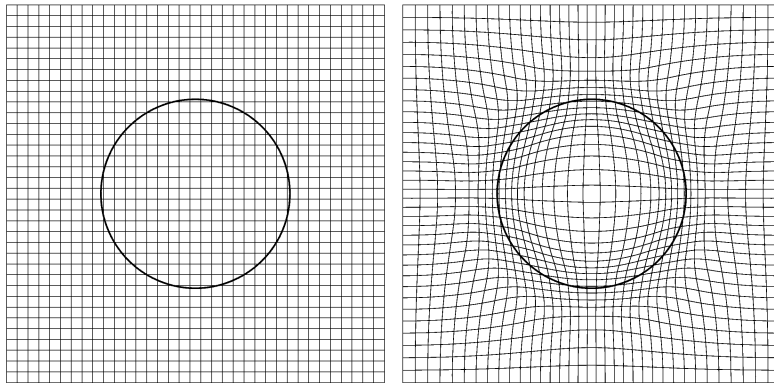


Figure 3.10: Grid deformation method. Left figure: initial mesh, right figure: deformed mesh.

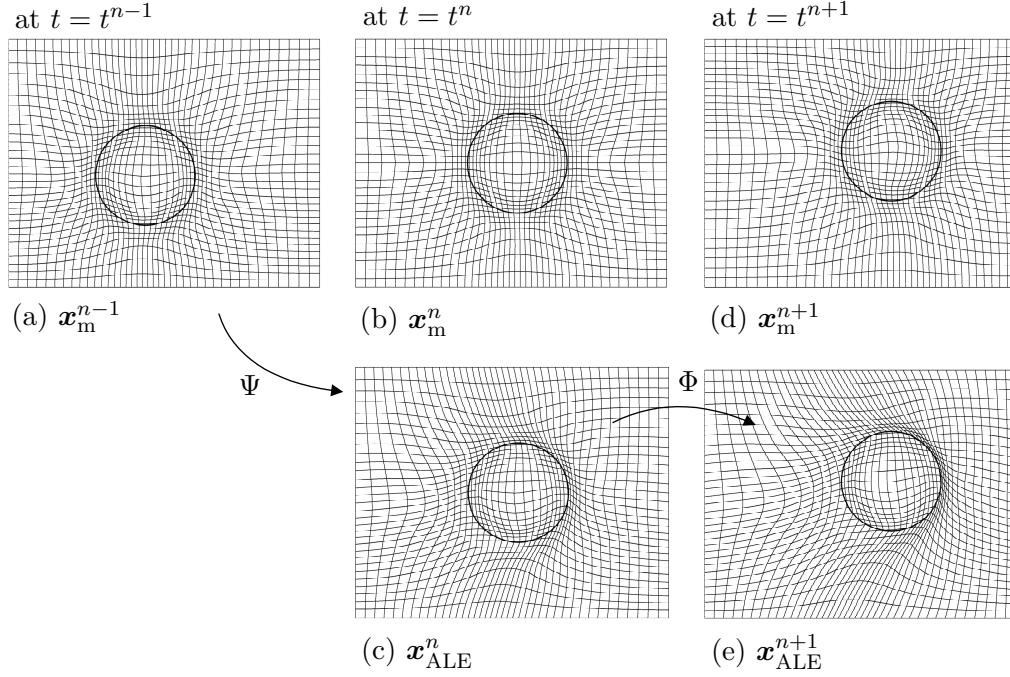


Figure 3.11: Second-order ALE scheme with grid deformation method.

(Fig. 3.10). The detailed algorithms of the grid deformation method are given in [117], and we will not repeat them here. The application of the grid deformation method on the problem of a fluid-particle system is shown in Fig. 3.11. At each time step, a deformed computational mesh is derived from an initial regular mesh, according to the location of the particle boundary. All physical quantities, such as \mathbf{s} , \mathbf{u} and \mathbf{u}_m , are solved on the deformed computational mesh.

Note that, in Fig. 3.6, 3.8 and 3.11, an unrealistically large Δt is used for a clear distinction between the computational mesh and the ALE mesh. Hence the particle moves quite a long distance at each time step, which cause the elements in the ALE mesh look rather distorted near the interface. In actual computations, the movement of the ALE mesh is small since the time step Δt is small.

3.3.5 Time integration

We use a semi-implicit stress formulation for the time integration of the evolution equation of the log-conformation tensor \mathbf{s} (Eq. (3.19)) decoupled from the momentum balance. Initially, the viscoelastic polymer stress is set to zero over the whole fluid domain. So we can solve Eqs. (3.17), (3.18), (3.20) and (3.21) to get the distribution of the fluid velocity and pressure at the initial time $t = 0$. Then, we apply the following procedure at every time step.

Step 1. Construct a temporary ALE mesh using Eqs. (3.45)-(3.47) for the interpolation of field variables at previous time levels. At the first time step, we use a first-order scheme given in Eq. (3.39).

Step 2. Update the particle configuration by integrating the kinematic equations in

Eqs. (3.15) and (3.16) using the explicit second-order Adams-Bashforth method (AB2):

$$\mathbf{X}_i^{n+1} = \mathbf{X}_i^n + \left(\frac{3}{2}\mathbf{U}_i^n - \frac{1}{2}\mathbf{U}_i^{n-1}\right)\Delta t, \quad (3.57)$$

$$\Theta_i^{n+1} = \Theta_i^n + \left(\frac{3}{2}\boldsymbol{\omega}_i^n - \frac{1}{2}\boldsymbol{\omega}_i^{n-1}\right)\Delta t. \quad (3.58)$$

For circular particles, the update of angular rotations is not necessary. At the first time step, we use an explicit Euler method:

$$\mathbf{X}_i^{n+1} = \mathbf{X}_i^n + \mathbf{U}_i^n \Delta t. \quad (3.59)$$

Step 3. Compute a deformed mesh for local mesh refinements, if necessary.

Step 4. Modify the computational mesh to avoid very small integration areas. First, compute a mesh velocity field $\hat{\mathbf{u}}_m$:

$$\nabla^2 \hat{\mathbf{u}}_m = \mathbf{0} \quad \text{in } \Omega, \quad (3.60)$$

$$\hat{\mathbf{u}}_m = \mathbf{0} \quad \text{on } \Gamma, \quad (3.61)$$

$$\hat{\mathbf{u}}_m = \hat{\mathbf{n}} \quad \text{on } \partial P_i(t), \quad (3.62)$$

where $\hat{\mathbf{n}}$ is the outwardly directed unit normal vector on the particle surface. Then, each mesh point \mathbf{x}_m moves according to the following advection equations:

$$\frac{d\mathbf{x}_m}{dt_m} = \begin{cases} \hat{\mathbf{u}}_m & \text{if } \mathbf{x}_m \in \Omega \setminus P(t), \\ \mathbf{0} & \text{if } \mathbf{x}_m \in P(t), \end{cases} \quad (3.63)$$

$$\mathbf{x}_m(t_m = 0) = \mathbf{x}_{m,0} \quad (3.64)$$

where $\mathbf{x}_{m,0}$ is the initial position of mesh point. In our simulations, we use a third-order Adams-Bashforth method (AB3) and nodal points are moved until each area of integration is larger than 0.5% of the element area.

Step 5. Solve the log-conformation tensor \mathbf{s}^{n+1} by replacing the evolution equation of the log-conformation tensor (Eq. (3.7)) using a second-order semi-implicit Gear scheme:

$$\begin{aligned} (\mathbf{d} + \tau \hat{\mathbf{u}}^{n+1} \cdot \nabla \mathbf{d}, \frac{3}{2} \frac{\mathbf{s}^{n+1}}{\Delta t} + \hat{\mathbf{u}}^{n+1} \cdot \nabla \mathbf{s}^{n+1}) = \\ (\mathbf{d} + \tau \hat{\mathbf{u}}^{n+1} \cdot \nabla \mathbf{d}, \frac{2\mathbf{s}^n - \frac{1}{2}\mathbf{s}^{n-1}}{\Delta t} + 2\mathbf{g}(\mathbf{G}^n, \mathbf{s}^n) - \mathbf{g}(\mathbf{G}^{n-1}, \mathbf{s}^{n-1})) \end{aligned} \quad (3.65)$$

where $\hat{\mathbf{u}}^{n+1} = 2(\mathbf{u}^n - \mathbf{u}_m^n) - (\mathbf{u}^{n-1} - \mathbf{u}_m^{n-1})$. At the first time step, we use a first-order Euler scheme:

$$\begin{aligned} (\mathbf{d} + \tau(\mathbf{u}^n - \mathbf{u}_m^n) \cdot \nabla \mathbf{d}, \frac{\mathbf{s}^{n+1}}{\Delta t} + (\mathbf{u}^n - \mathbf{u}_m^n) \cdot \nabla \mathbf{s}^{n+1}) = \\ (\mathbf{d} + \tau(\mathbf{u}^n - \mathbf{u}_m^n) \cdot \nabla \mathbf{d}, \frac{\mathbf{s}^n}{\Delta t} + \mathbf{g}(\mathbf{G}^n, \mathbf{s}^n)). \end{aligned} \quad (3.66)$$

Step 6. Compute \mathbf{c}^{n+1} with $\mathbf{c}^{n+1} = \exp(\mathbf{s}^{n+1})$. Then, compute \mathbf{u}^{n+1} , p^{n+1} , \mathbf{G}^{n+1} , $\boldsymbol{\lambda}_i^{n+1}$, \mathbf{U}_i^{n+1} and $\boldsymbol{\omega}_i^{n+1}$ from the momentum balance, continuity equation and gradient projection equation:

$$\begin{aligned} -(\nabla \cdot \mathbf{v}, p^{n+1}) + ((\nabla \mathbf{v})^T, 2\eta_s \mathbf{D}(\mathbf{u}^{n+1}) + \theta(\nabla \mathbf{u}^{n+1} - (\mathbf{G}^T)^{n+1})) \\ + (\mathbf{v} - (\mathbf{V}_i + \boldsymbol{\chi}_i \times (\mathbf{x} - \mathbf{X}_i^{n+1})), \boldsymbol{\lambda}_i^{n+1})_{\partial P_i(t^{n+1})} = \\ -((\nabla \mathbf{v})^T, \boldsymbol{\tau}(\mathbf{c}^{n+1})) + (\mathbf{v}, \mathbf{t})_{\Gamma_N}, \end{aligned} \quad (3.67)$$

$$(q, \nabla \cdot \mathbf{u}^{n+1}) = 0, \quad (3.68)$$

$$(\mathbf{H}, -\nabla \mathbf{u}^{n+1} + (\mathbf{G}^T)^{n+1}) = 0, \quad (3.69)$$

$$(\boldsymbol{\mu}_i, \mathbf{u}^{n+1} - (\mathbf{U}_i^{n+1} + \boldsymbol{\omega}_i^{n+1} \times (\mathbf{x} - \mathbf{X}_i^{n+1})))_{\partial P_i(t^{n+1})} = 0. \quad (3.70)$$

3.4 Validation

We will validate the proposed method by comparing with the results of boundary-fitted mesh problems of particle migration in shear flow confined between two walls. The problem is schematically shown in Fig. 3.12. The dimensionless parameters governing the problem are the Weissenberg number $\text{Wi} = \lambda\dot{\gamma} = \lambda U_w/H$, the mobility parameter α of the Giesekus model, the viscosity ratio η_s/η_p and the confinement ratio of a particle $\epsilon = 2a/H$, where a is the radius of the particle. In this problem, we choose $\text{Wi} = 1$, $\alpha = 0.2$, $\eta_s/\eta_p = 0.1$ and $\epsilon = 0.1$. We will investigate the migrational velocity V and angular velocity ω of the particle for different initial particle positions Y_0 . This problem was originally solved by D'Avino et al. [39] by using boundary-fitted meshes and a conventional ALE scheme.

In these simulations, we use a grid deformation method for the local mesh refinements around the particle surface as described in Section 3.3.4. Our mesh has a similar resolution

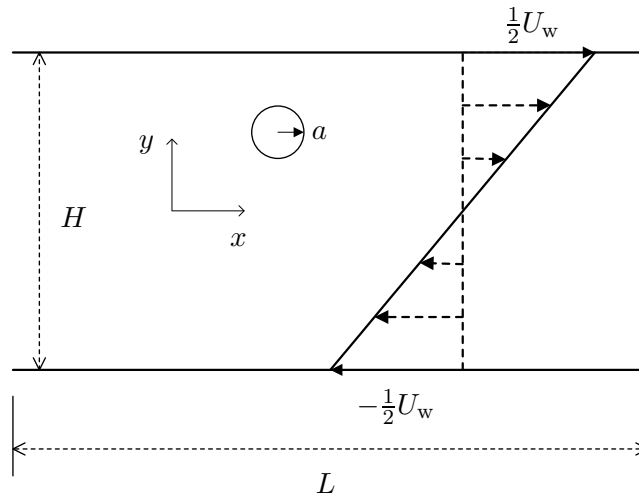


Figure 3.12: Geometry of simple shear flow.

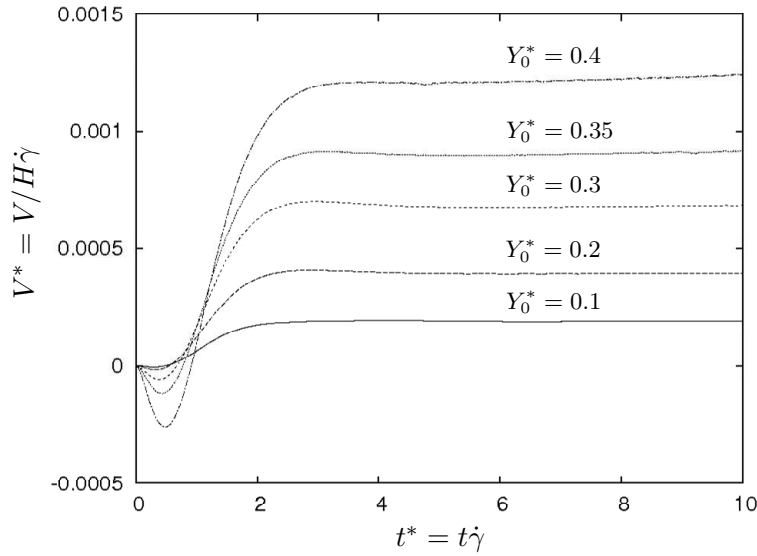


Figure 3.13: Migration velocity of a particle as a function of time for different initial positions.

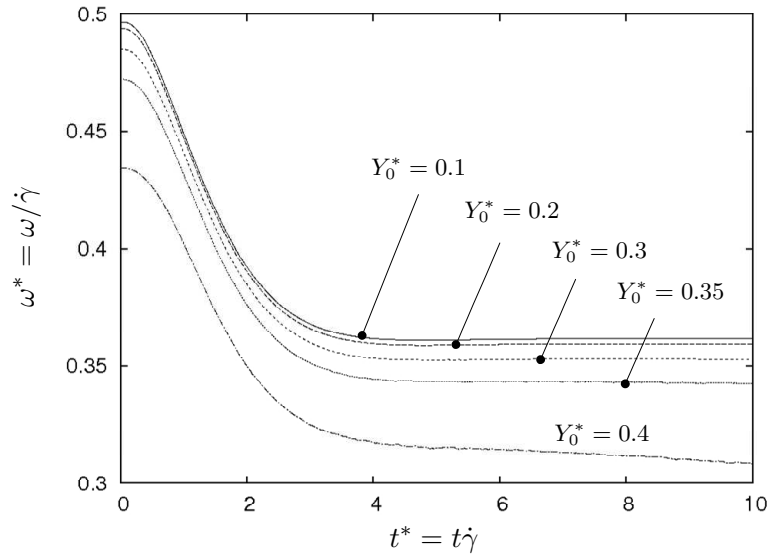


Figure 3.14: Angular velocity of a particle as a function of time for different initial positions.

near the particle surface as the mesh M4 in [39]. We also solved this problem by using a uniformly refined mesh for the whole domain without using the grid deformation method, and we do not notice any difference in results. Fig. 3.13 and Fig. 3.14 show the non-dimensional migrational velocity $V^* = V/H\dot{\gamma}$ and angular velocity $\omega^* = \omega/\dot{\gamma}$, respectively, as a function of non-dimensional time $t^* = t\dot{\gamma}$ for different initial positions $Y_0^* = Y_0/H = \{0.1, 0.2, 0.3, 0.35, 0.4\}$. Note that we define a clockwise rotation as positive throughout the chapter. Our results show good agreement with those of boundary-fitted mesh problems in [39] (see Fig. 7 and 8 in [39]). If we plot both results in the same figure, the curves overlap. The relative errors between our results and those in [39] are approximately 0.1%, which validates the proposed XFEM

combined with the temporary ALE scheme.

3.5 Results

3.5.1 Problem description

We consider the migration of a particle freely suspended in a circular Couette flow of viscoelastic fluids, see Fig. 3.15. The radius of the inner cylinder is R_1 , the radius of the outer cylinder is R_2 , the radius of particle is a and the radial position of the center of a particle is r . The dimensionless radial position of the particle is defined by $r^* = (r - R_1)/(R_2 - R_1)$. The inner cylinder rotates counterclockwise at a constant angular velocity $\bar{\omega}$, and the outer cylinder is stationary. The dimensionless parameters governing the problem are the Weissenberg number $Wi = \lambda \bar{\omega} R_1 / (R_2 - R_1)$, the geometric ratio of the two cylinders R_2/R_1 , the confinement ratio of the particle $\epsilon = 2a/(R_2 - R_1)$, the initial radial position of the particle $r_0^* = (r_0 - R_1)/(R_2 - R_1)$, the mobility parameter α of the Giesekus model and the viscosity ratio $\beta = \eta_s/(\eta_s + \eta_p)$. We fix the geometric ratio $R_2/R_1 = 4$ and the viscosity ratio $\beta = 0.1$ for all computations. Note that $\frac{1}{2}\epsilon \leq r^* \leq 1 - \frac{1}{2}\epsilon$ due to the particle size. We will investigate the effects of Weissenberg number Wi , the mobility parameter α , the initial particle position r_0^* and the confinement ratio ϵ on the particle migration.

3.5.2 Mesh convergence

First, we will present the results of mesh convergence. Mesh parameters used in the simulations are given in Table 3.1, where “D” represents deformed mesh refined near the particle boundary. Fig. 3.16 shows the meshes M1 and M1D; they have same number of elements and mesh topology. We choose the size of line elements on the particle boundary for the imposition of no-slip boundary condition similar to the size of fluid elements near the

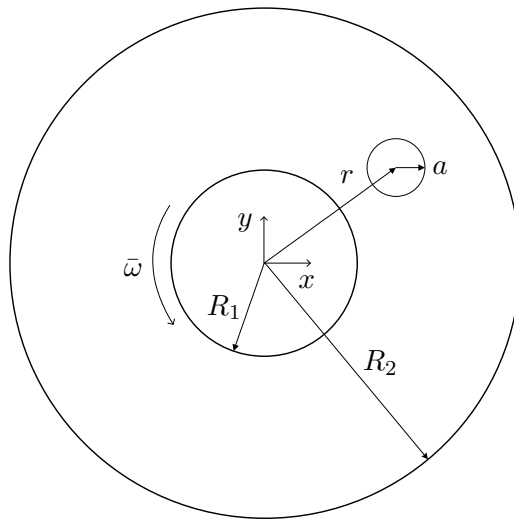


Figure 3.15: Geometry of concentric cylinders.

Table 3.1: Meshes used for the simulations

mesh	number of elements in the mesh	number of line elements on the particle boundary for $\epsilon = 4/15$
M1	11028	40
M1D		60
M2	21832	60
M2D		80
M3	33008	80
M3D		100

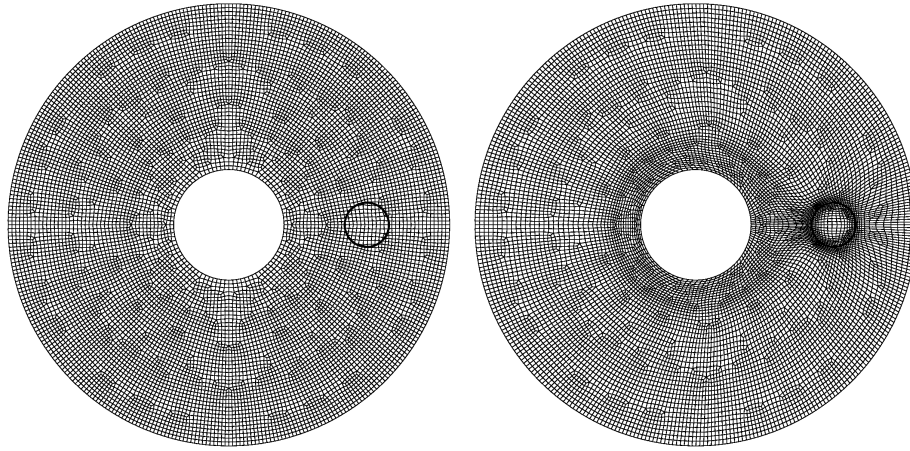


Figure 3.16: Typical meshes used for the simulations. Left figure: M1, right figure: M1D. ($\epsilon = 4/15, r^* = 1/2$)

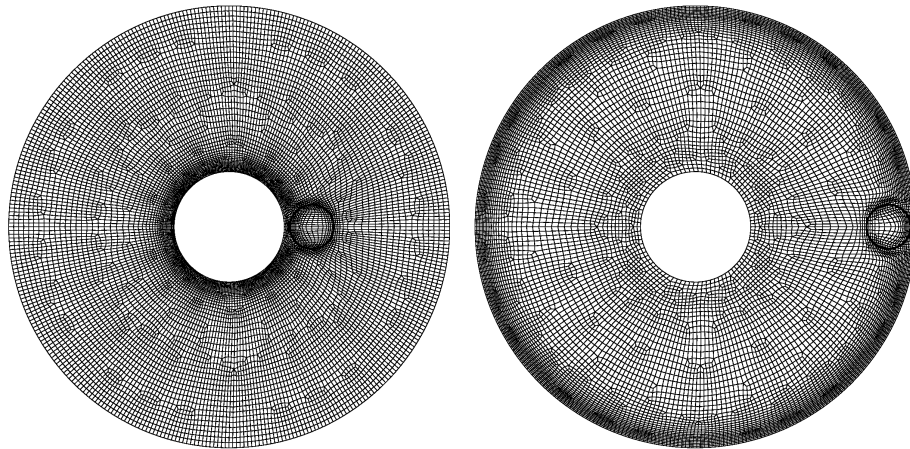


Figure 3.17: The mesh M1D with refinements around the inner or the outer cylinder. Left figure: $r^* = 1/6$, right figure: $r^* = 5/6$.

interface. If the position of the particle is close to either cylinder, we also refine elements around the cylinder as shown in Fig. 3.17.

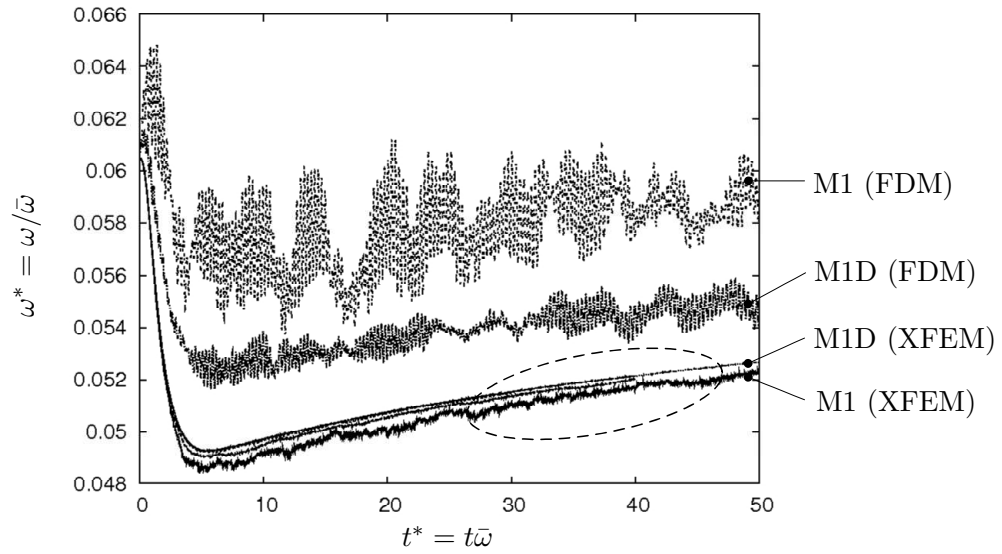


Figure 3.18: Angular velocity of a particle using different meshes. We also plot the results by using fictitious domain method (FDM) for comparison. ($Wi = 1/3, \alpha = 0.01, \epsilon = 4/15, r_0^* = 1/2$)

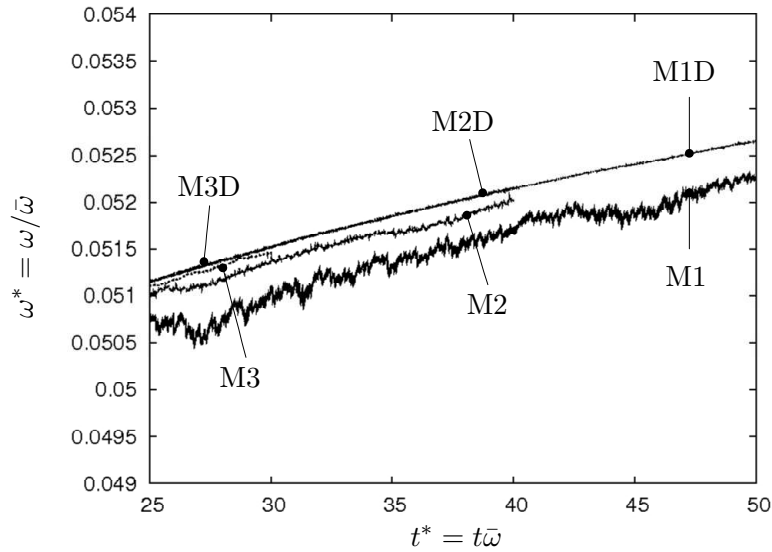


Figure 3.19: Angular velocity of a particle using different meshes. A magnified view of the encircled region in Fig. 3.18. ($Wi = 1/3, \alpha = 0.01, \epsilon = 4/15, r_0^* = 1/2$)

We solve the particle motion for $Wi = 1/3$, $\alpha = 0.01$, $\epsilon = 4/15$ and $r_0^* = 1/2$. The angular velocities of the particle $\omega^* = \omega/\bar{\omega}$ obtained by using different meshes are shown in Fig. 3.18. We also plot the results by using the fictitious domain method (FDM) [73, 74, 116] for comparison. The results of FDM show significant oscillations and larger values compared with those of XFEM. Further mesh refinements for FDM does not improve these discrepancies. The result of FDM using mesh M3D is similar to that of M1D, so we only plot the results

Table 3.2: CPU times for the simulations

mesh	CPU time in seconds (for 5 time steps)
M1	224
M1D	294
M2	607
M2D	720
M3	1135
M3D	1357

of M1 and M1D for FDM. This result shows superior accuracy of XFEM compared to FDM. Contrary to angular velocities, translational velocities of the particle show good agreements between the results obtained by XFEM and FDM. The differences in translational velocities using different meshes and methods are quite small and seem to be negligible. Hence, we only plot angular velocities.

For a better comparison of the results of XFEM, the encircled region in Fig. 3.18 is magnified in Fig. 3.19. The result of M1 shows some oscillations and those oscillations are reduced with mesh refinement (see the results of M2 and M3). The results of uniform mesh refinements from M1 to M3 seem to converge to the result of M1D. The results by using deformed meshes M1D, M2D and M3D are almost identical and indistinguishable in Fig. 3.19. The mesh resolution of M1D near the particle boundary is similar to M2, and coarser than M3 (see Table 3.1). However, assuming the result of M1D is the correct mesh converged solution, the result of M1D is even better than that of M3. This can be explained as follows. Since the mesh coordinates of M1, M2 and M3 are fixed for the whole computation, the particle moves on the fixed computational mesh. As the particle moves, an element in the ALE mesh covers several elements in the computational mesh, as shown in Fig. 3.7 and Fig. 3.9, which can cause slight oscillations and lower accuracy in solutions as in Fig. 3.19. On the other hand, if the grid deformation method is applied, the computational mesh also changes according to the position of the particle. Even though the computational mesh is not fully advected with the motion of the particle, the mesh coordinates near the particle surface can roughly follow the motion of the particle for several time steps, which makes the mapping more or less element to element. In other words, the proposed method combined with the grid deformation method resembles the original ALE mesh movement scheme [59, 60]. Hence, the deformed mesh can give smoother and more accurate results than the fixed mesh.

For a comparison between deformed and fixed meshes, we need to note the computational time as well. The CPU times required for the simulations using different meshes are given in Table 3.2. The computation time required for a deformed mesh is slightly increased compared to its own base mesh due to additional costs for the calculations of mesh coordinates. Still, the computation time of a coarse deformed mesh (such as M1D) is much lower than that of a finer uniform mesh (such as M2 or M3). The mesh convergence and computation time show the usefulness of the grid deformation method. For every simulation in this problem,

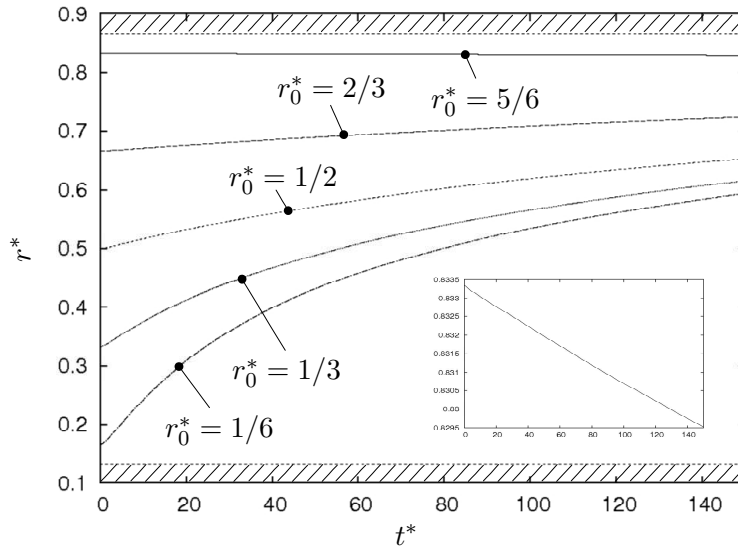


Figure 3.20: Migration of a particle with different initial positions. $r_0^* = 5/6$ is also plotted in a smaller box to show the decreasing of radial position more clearly. ($Wi = 1/3, \alpha = 0.01, \epsilon = 4/15$)

we use the grid deformation method. The mesh convergence is checked for all cases. Mostly, the mesh M1D gives a converged solution as in Fig. 3.19 of the previous paragraph. As the Weissenberg number increases and the particle moves close to either cylinders, we need a more refined mesh such as M2D.

3.5.3 Initial position of a particle

We investigate the effect of initial particle positions for a given flow condition. The histories of the radial position of a particle starting from different initial positions are shown in Fig. 3.20 for $Wi = 1/3, \alpha = 0.01$ and $\epsilon = 4/15$. The shaded regions represent the zones where the particle cannot move due to its size. The case of initial radial position at $r_0^* = 5/6$ is also plotted in a smaller box to show the decreasing of the radial position with respect to time more clearly. The closer the particle position is to the inner cylinder, the faster the radial migration is. As the particle moves towards the outer cylinder, the radial velocity slows down. If the initial particle position is very close to the outer cylinder (such as $r_0^* = 5/6$), the particle slowly migrates towards the inner cylinder. Hence, we can expect that the particle will eventually reach a stabilized radial position near the outer cylinder regardless of initial positions.

3.5.4 Weissenberg number

We investigate the effect of Weissenberg number on particle migration. Fig. 3.21 shows the radial migration of a particle initially positioned at $r_0^* = 2/3$ for different Weissenberg numbers. Other parameters are $\alpha = 0.01$ and $\epsilon = 4/15$. As the Weissenberg number increases,

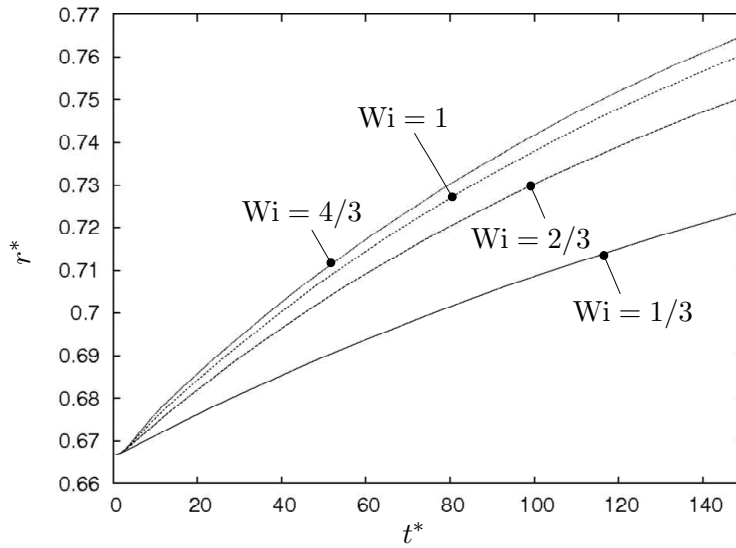


Figure 3.21: Migration of a particle with different Weissenberg numbers. ($\alpha = 0.01, \epsilon = 4/15, r_0^* = 2/3$)

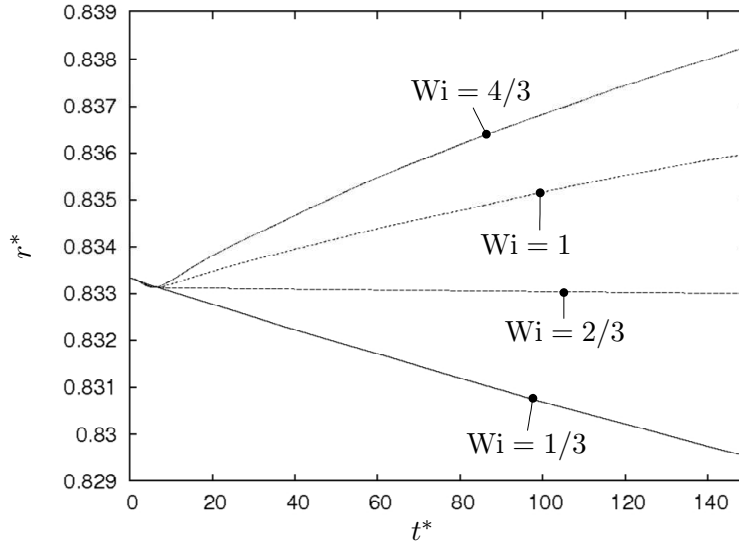


Figure 3.22: Migration of a particle with different Weissenberg numbers. ($\alpha = 0.01, \epsilon = 4/15, r_0^* = 5/6$)

the particle migrates more rapidly towards the outer cylinder. Moreover, the stabilized radial position is shifted towards the outer cylinder as shown in Fig. 3.22, in which the particle is released at $r_0^* = 5/6$. In Fig. 3.22, for $Wi = 2/3$, the particle already reached a stabilized radial position approximately at $r^* = 0.833$. We also show the radial migration of a particle initially positioned midway between the inner and outer cylinders (see Fig. 3.23, in which $r_0^* = 1/2$). From $Wi = 1/3$ to $Wi = 2/3$, the radial migration becomes faster, however, for $Wi = 4/3$, the radial migration is even slower than the case of $Wi = 1$. This is a transient phenomena,

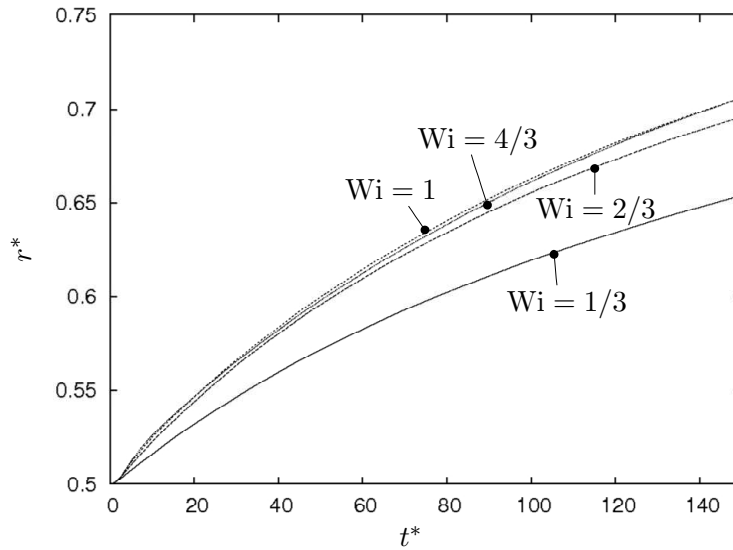


Figure 3.23: Migration of a particle with different Weissenberg numbers. ($\alpha = 0.01, \epsilon = 4/15, r_0^* = 1/2$)

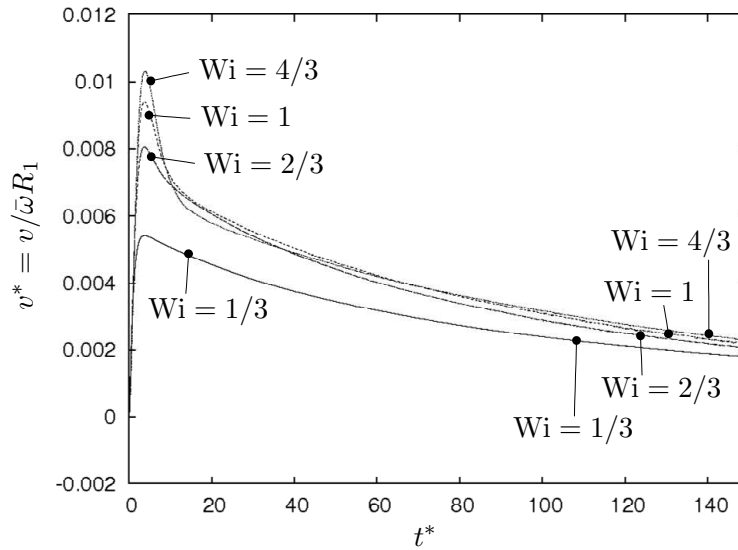


Figure 3.24: Migration velocity of a particle with different Weissenberg numbers. ($\alpha = 0.01, \epsilon = 4/15, r_0^* = 1/2$)

and the migration velocity becomes larger for larger Weissenberg numbers, eventually. This fact is clearly shown in Fig. 3.24. For $Wi = 4/3$, after the initial overshoot, the migration velocity is smaller than those of $Wi = 2/3$ and $Wi = 1$ (around $t^* = 20$). Then, as the migration goes on, the radial velocity for $Wi = 4/3$ becomes larger than those of $Wi = 2/3$ and $Wi = 1$. This transient effect becomes profounder as the particle initial position is closer to the inner cylinder, where the shear rate is largest. However, after the initial transient stage, the migration becomes faster for larger Weissenberg numbers.

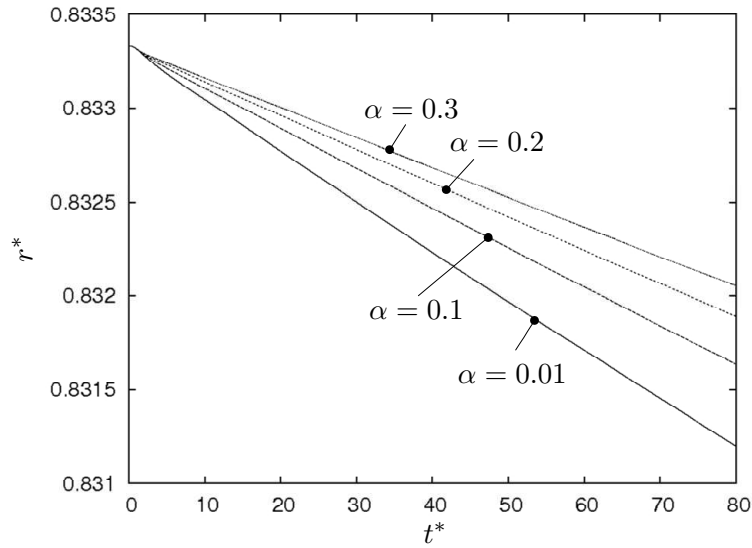


Figure 3.25: Migration of a particle with different mobility parameters. ($Wi = 1/3, \epsilon = 4/15, r_0^* = 5/6$)

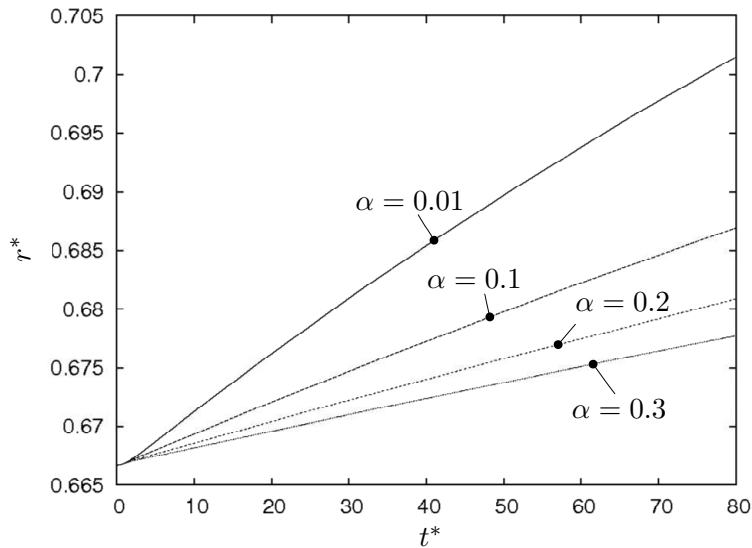


Figure 3.26: Migration of a particle with different mobility parameters. ($Wi = 1/3, \epsilon = 4/15, r_0^* = 2/3$)

3.5.5 Mobility parameter α

Fig. 3.25 and Fig. 3.26 show the effect of mobility parameter α on the migration of a particle released at $r_0^* = 5/6$ and $r_0^* = 2/3$, respectively. As the mobility parameter increases, the radial migration becomes slower. Hence it will take more time to reach a stabilized radial position for larger α . For the initial particle positions $r_0^* = 1/2$ and $r_0^* = 1/3$, we obtained similar trend of decreasing migration velocity for increasing α . So we will not show those results here. For a particle very close to the inner cylinder, more interesting phenomena

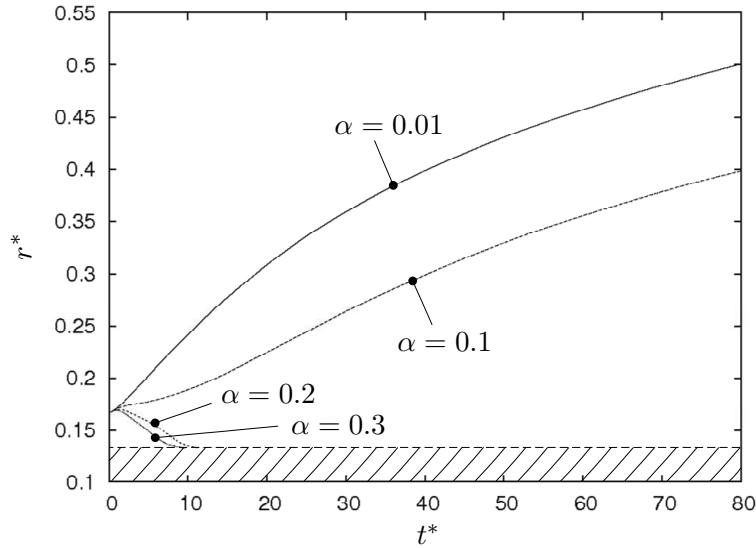


Figure 3.27: Migration of a particle with different mobility parameters. ($Wi = 1/3, \epsilon = 4/15, r_0^* = 1/6$)

are observed. Fig. 3.27 shows the radial migration of a particle released at $r_0^* = 1/6$. If α is increased from 0.01 to 0.1, the migration slows down, as before. If α is increased from 0.1 to 0.2, the migration direction is reversed and the particle moves until it touches the inner cylinder, at this moment our simulation crashes. Since the confinement ratio $\epsilon = 4/15$, when the particle is at $r^* = r/R_1 = 2/15$, the particle touches the inner cylinder. If the mobility parameter α is further increased to 0.3, the particle migrates more rapidly to the inner cylinder and touches the inner cylinder.

Our results show qualitatively good agreements with the experimental results of sphere migrations in a Couette flow of a shear-thinning elastic polymer solution by Snijkers [31]. If initial positions of spheres were distant from the inner cylinder, spheres were observed to migrate towards the outer cylinder with a constant velocity. Whereas, if initial positions of spheres were close to the inner cylinder, spheres were observed to migrate towards the inner cylinder following a roughly parabolic profile until they touched the inner cylinder.

3.5.6 Particle size

We consider the effect of particle size or the confinement ratio ϵ on the migration. Fig. 3.28 shows the migration of a particle released at $r_0^* = 1/3$ for different confinement ratios. As the particle size or the confinement ratio increases, the radial migration becomes faster. For $\epsilon = 8/15$, the migration velocity significantly decreases with respect to time. Since the particle size is big and moves fast to the outer cylinder, the gap between the particle and the outer wall reduces rapidly. Then the particle motion is highly affected by the presence of the outer cylinder. This is the reason for the depletion of migration velocity for large particles. If the initial particle position is very close to the outer cylinder, there is little room for a large particle to move radially. Hence the migration can be dramatically slowed down. This can be

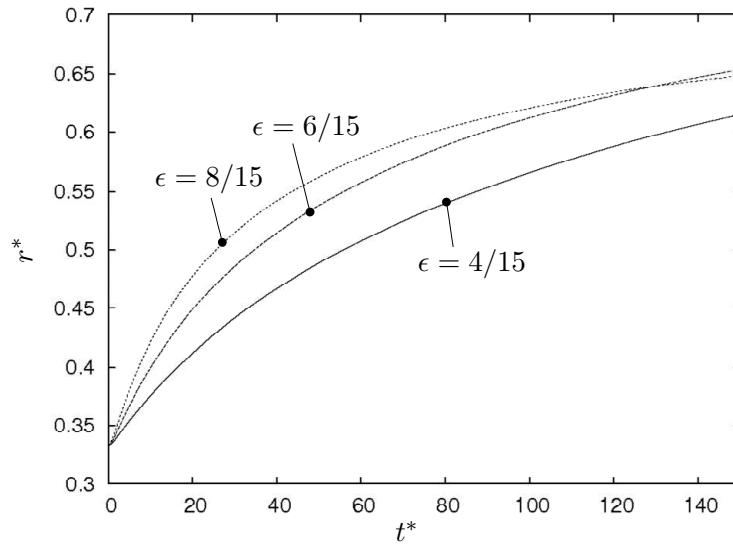


Figure 3.28: Migration of a particle with different confinement ratios. ($Wi = 1/3, \alpha = 0.01, r_0^* = 1/3$)

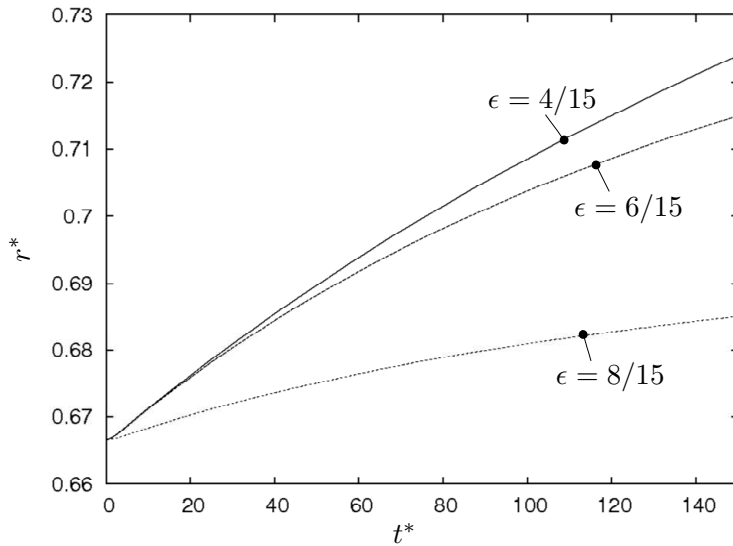


Figure 3.29: Migration of a particle with different confinement ratios. ($Wi = 1/3, \alpha = 0.01, r_0^* = 2/3$)

seen in Fig. 3.29 for $r_0^* = 2/3$, in which a larger particle migrates slower from the beginning. Hence we can conclude that a large particle migrates fast towards the outer cylinder if the particle is far from the outer cylinder. If a large particle is close to the outer cylinder, the radial migration slows down significantly due to an overwhelming wall effect.

3.6 Conclusions

An extended finite element method for the direct numerical simulation of particulate viscoelastic fluids has been developed. For moving particle problems, we incorporate a temporary arbitrary Lagrangian-Eulerian scheme which defines the mapping of field variables at previous time levels onto the computational mesh at the current time level in the finite element context. In this method, a fixed, regular (not boundary-fitted) mesh is used for the whole computational domain including both fluid and particles without any need of remeshing. An ALE mesh is constructed separately and the computational mesh is kept unchanged throughout the whole computation, then particles are moving on a fixed Eulerian mesh. The proposed method shows similar accuracy compared to the boundary-fitted mesh problems combined with the conventional ALE scheme, which still require remeshing if the mesh becomes too much distorted. Our method shows superior accuracy compared with the fictitious domain method.

For mesh refinements around the interface, we combined the XFEM with the grid deformation method, in which nodal points are redistributed close to the interface while preserving the mesh topology. In the grid deformation method, the number of elements is not increased compared to its own base mesh. If the grid deformation method is applied to fluid-particle systems, particles still move on a fixed Eulerian mesh, then, at each position of particles, the interface acts like a global attractor of elements. If the grid deformation method is combined with the XFEM, the required computational time is reduced significantly compared to uniform mesh refinements, while providing mesh convergent solutions.

We applied the proposed XFEM, combined with the temporary arbitrary Lagrangian-Eulerian scheme and the grid deformation method, to the particle migration in circular Couette flow of a viscoelastic fluid. The particle moves towards the outer cylinder. The closer the particle position is to the inner cylinder, the faster the radial migration is. As the particle moves close to the outer cylinder, the radial velocity slows down, and eventually, the particle reaches to a final stabilized radial position near the outer cylinder regardless of its initial positions. As the Weissenberg number increases, the particle migrates more rapidly towards the outer cylinder, and the stabilized radial position is shifted towards the outer cylinder. With increasing mobility parameter of the Giesekus model, the rate of migration decreases. If the initial position of the particle is very close to the inner cylinder and the mobility parameter is large enough, the direction of migration is reversed and the particle moves towards the inner cylinder until it touches the inner cylinder. With increasing confinement ratio of the particle, the particle migrates more rapidly towards the outer cylinder if the particle is far from the outer cylinder. If a large particle is close to the outer cylinder, the radial migration slows down significantly due to an overwhelming wall effect.

Chapter 4

Dynamics of particles in two-phase flows

In this chapter, we present a model for the dynamics of particles suspended in two-phase flows by coupling the Cahn-Hilliard theory with the extended finite element method and temporary ALE scheme, which are already introduced in Chapter 3. In the Cahn-Hilliard model the interface is considered to have a small but finite thickness, which circumvents explicit tracking of the interface. By combining the Cahn-Hilliard model with the extended finite element method, the particle motion at an interface can be simulated on a fixed Eulerian mesh without any need of remeshing. The model is general, but to demonstrate and validate the technique, here the dynamics of a single particle at a fluid-fluid interface is studied. First, we apply a small disturbance on a particle resting at an interface between two fluids, and investigate the particle movement towards its equilibrium position. In particular, we are interested in the effect of interfacial thickness, surface tension, particle size and viscosity ratio of the two fluids on the particle movement towards its equilibrium position. Finally, we show the movement of a particle passing through multiple layers of fluids.

The content of this Chapter is based on:

Young Joon Choi and Patrick D. Anderson. Cahn-Hilliard modeling of particles suspended in two-phase flows. *International Journal for Numerical Methods in Fluids*, DOI: 10.1002/flid.2623, published online: 11 July 2011.

4.1 Introduction

Small solid particles adsorbed at liquid interfaces arise in many industrial products and processes, such as anti-foam formulations, crude oil emulsions, fluidized suspensions, slurry transport, materials separation, rate of mixing enhancement etc. In particular, if particles are suspended in emulsions, this emulsion can be stabilized by solid particles which adsorb onto the interface between the two fluids, which is usually called a Pickering emulsion [118]. They act in many ways like traditional surfactant molecules, but offer distinct advantages. Unfortunately, the understanding of how these particles operate in such systems is limited. In this chapter we present a numerical method for the dynamics of particles in two-phase flows based on the Cahn-Hilliard theory.

Diffuse-interface modeling is based on the van der Waals's approach of the interface problem [119] and developed by Cahn and Hilliard [120]. The main assumption is that the interface is not sharp, but has a thickness that is not explicitly prescribed, but follows from the governing equations that couple thermodynamic and hydrodynamic forces. The main elements of the theory, and the coupling of thermodynamics and hydrodynamics are summarized in the review paper by Anderson, McFadden and Wheeler [121] and references within.

Diffuse-interface methods have been applied to a variety of multi-phase flow problems ranging from phase separating polymer blends to simulating solid tumor growth using mixture models. For example, Prusty et al. used the Cahn-Hilliard technique to the coarsening dynamics for PMMA/SAN28 blends and a quantitative comparison between the experimental and numerically predicted phase separation kinetics was presented [122]. Wise et al. presented simulations of tumor growth in 2D and 3D that demonstrate the capabilities of the diffuse interface model in accurately and efficiently simulating the progression of tumors with complex morphologies [123]. Khatavkar et al. used the diffuse-interface method to model micron-sized drop spreading and impact on smooth and structured substrates [124, 125, 126]. Recently, Tufano et al. applied the Cahn-Hilliard theory coupled with hydrodynamic interactions to describe three-phase systems where the effects of mutual diffusion on interfacial tension, drop-drop and drop-wall interactions in quiescent conditions are investigated and compared with experimental observations [127]. Millet et al. introduced a diffuse-interface field description of each fluid phase in addition to the set of solid particles. Their model can include particles of arbitrary shapes and orientations, as well as the ability to incorporate electrostatic particle interactions [128].

In order to simulate particle movement in two-phase flows, the most intuitive method is using a boundary-fitted mesh, which means that the particle surface is aligned with element boundaries of the fluid [58, 59, 60]. In this method, the governing equations are solved only in the fluid domain, taking into account the interface conditions on the boundaries of particles. To handle moving particles, this approach incorporates the arbitrary Lagrangian-Eulerian (ALE) technique [61] that relies on a moving mesh scheme. The generation of a new mesh is needed if the old mesh becomes too distorted, and the solution must be projected onto the new mesh. The generation of boundary-fitted meshes is, however, still a challenging task in view of algorithms needed and computational costs if complex geometries are involved,

especially in three-dimensional simulations using second-order hexahedron elements.

An alternative approach is the fictitious domain method developed by Glowinski et al. [62, 63, 64]. The basic idea of this method is to fill each domain of every particle with the surrounding fluid, assuming and subsequently prescribing that the fluid inside the particle domain moves like a solid object. Hence, the problem is transferred from a geometrically complex fluid domain to a simpler domain including both fluid and particles, which eliminates the need of remeshing. In this method, particles move in a Lagrangian sense on a fixed Eulerian mesh. For single-component problems this approach has been quite successful and the dynamics of particles in complex fluids has been studied in a variety of flow conditions [78, 129]. However, if a particle is suspended in a two-component system, a fictitious domain approach would require additional constraints for the motion of fictitious fluids inside the particle, which is not trivial and in this work an alternative approach is followed.

The extended finite element method (XFEM) has been recently developed to simulate particle suspended single component fluid flows. In this method, the finite element shape functions are locally extended, or enriched, to decouple the fluid domain from the particle domain while still using a mesh which is not boundary-fitted. Originally, XFEM was developed for the simulations of cracks in solids without the need of remeshing [90, 91]; later it was applied to flow problems as well [92, 93, 115]. A recent review on extended finite element methods applied to material modeling is presented in [94]. Choi et al. proposed a temporary arbitrary Lagrangian-Eulerian scheme to handle moving particles without any need of remeshing throughout the whole computations in the extended finite element context [130].

In the present chapter, we present a numerical method for the dynamics of particles in two-phase flows by coupling the Cahn-Hilliard theory with the extended finite element method while using a fixed Eulerian mesh without any need of remeshing. Since the fluid domain is decoupled from the particle domain in XFEM, we do not need extra conditions inside the particles, contrary to the fictitious domain method. The content of this chapter is as follows. In Section 4.2 we give a brief review of the Cahn-Hilliard theory. In Section 4.3 the numerical algorithm of the extended finite element method combined with a temporary arbitrary Lagrangian-Eulerian (ALE) scheme is discussed. The introduced model is applied to study the dynamics of a single particle at a fluid-fluid interface in Section 4.4, and the movement of a particle passing through multiple layers of fluids is demonstrated in Section 4.5. Finally, a discussion follows in Section 4.6.

4.2 Mathematical formulation

The classical expression for the specific Helmholtz free energy used in diffuse interface modeling is based on the work of Cahn and Hilliard [120]:

$$f(c, \nabla c) = -\frac{1}{2}\alpha c^2 + \frac{1}{4}\beta c^4 + \frac{1}{2}\varepsilon |\nabla c|^2, \quad (4.1)$$

where α and β are positive constants and ε is the gradient-energy parameter and c is the mass fraction of one of the two components [131]. The chemical potential is obtained from

the variational derivative with respect to concentration:

$$\mu = \frac{\delta f}{\delta c} = -\alpha c + \beta c^3 - \varepsilon \nabla^2 c. \quad (4.2)$$

This generalized chemical potential allows for the description of the interface between the two materials by a continuously varying concentration profile. For example, for a planar interface at equilibrium ($\mu=0$ in Eq. (4.2)), with z being the direction normal to the interface, the analytical solution of Eq. (4.2) reads:

$$c(z) = \pm c_B \tanh\left(\frac{z}{\sqrt{2}\xi}\right), \quad (4.3)$$

where $c_B = \sqrt{\alpha/\beta}$ is the equilibrium bulk concentration and $\xi = \sqrt{\varepsilon/\alpha}$ is a measure for the interfacial thickness. In order to comply with mass conservation for both components, the balance equation should be fulfilled:

$$\frac{\partial c}{\partial t} + \mathbf{u} \cdot \nabla c = \nabla \cdot (M \nabla \mu), \quad (4.4)$$

with M the mobility, which in general is a function of the composition, but is here taken constant for simplicity. The diffusion flux is assumed to be proportional to the gradient of the chemical potential, which is more general than the common Fickian diffusion, based on the concentration gradients (∇c), that does not hold for multiphase systems, even at equilibrium. The more general expression used in Eq. (4.4) reflects Gibbs' findings that the chemical potential becomes uniform in a non-ideal mixture at equilibrium, and is known as the Cahn-Hilliard equation [132]. To obtain momentum conservation, a generalized Navier–Stokes equation can be derived for the velocity field [133]:

$$\rho \left[\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right] = -\rho \nabla g + \nabla \cdot (2\eta \mathbf{D}) + \rho \mu \nabla c, \quad (4.5)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (4.6)$$

where $\mathbf{D} = (\nabla \mathbf{u} + \nabla \mathbf{u}^T)/2$ is the rate-of-deformation tensor; g is the Gibbs free energy $g = f + p/\rho$, with p the local pressure and ρ the density. The viscosity η generally depends on c since the two fluids, in general, have different viscosities. The viscosity η , is assumed to have the following linear relationship with the concentration c ,

$$\eta = \eta_1 \left(\frac{c+1}{2} \right) - \eta_2 \left(\frac{c-1}{2} \right), \quad (4.7)$$

where η_1 and η_2 are the viscosities of the two fluids, respectively.

Compared to the Navier–Stokes equations, in Eq. (4.5) only one extra capillary term $\rho \mu \nabla c$ appears reflecting the interfacial tension. This modification accounts for hydrodynamic interactions, i.e. the influence of the concentration c or the morphology on the velocity field due to the presence of interfaces.

To focus on the coupling of the particles with the multi-phase system, without any loss of generality, we now further assume quasi-stationary flow and neglect inertia in the momentum balance. Then, the momentum balance equation Eq. (4.5) reduces to

$$-\nabla \cdot (2\eta \mathbf{D}) + \nabla g = \rho \mu \nabla c. \quad (4.8)$$

4.2.1 Scaling of the equations

Using $c^* = c/c_B$, $\mathbf{u}^* = \mathbf{u}/U$, $\mu^* = \mu \xi^2 / (\varepsilon c_B)$, $t^* = tU/L$, with $c_B = \sqrt{\alpha/\beta}$ the bulk concentration, U a characteristic velocity, and L a characteristic domain size, and omitting the asterisk notation, Eqs. (4.2), (4.4), (4.6) and (4.8) read in dimensionless form:

$$\frac{dc}{dt} = \frac{1}{\text{Pe}} \nabla^2 \mu, \quad (4.9)$$

$$\mu = c^3 - c - \text{C}^2 \nabla^2 c, \quad (4.10)$$

$$-\nabla \cdot (2\eta \mathbf{D}) + \nabla g = \frac{1}{\text{Ca} \text{C}} \mu \nabla c, \quad (4.11)$$

$$\nabla \cdot \mathbf{u} = 0. \quad (4.12)$$

Three dimensionless groups appear in the governing equations: Péclet number Pe, the capillary number Ca and the Cahn number C, defined as:

$$\text{Pe} = \frac{\xi^2 LU}{M\varepsilon}; \quad \text{Ca} = \frac{\xi \eta U}{\rho \varepsilon c_B^2}; \quad \text{C} = \frac{\xi}{L}. \quad (4.13)$$

4.2.2 Rigid-body motion of particles

We suppose that N -particles are suspended in an incompressible fluid. Let Ω be the entire domain including the fluid and particles, and let $P_i(t)$ ($i = 1, \dots, N$) be the embedded domain of the i -th particle at time t with the number of particles N . The collective particle region at a certain time t is denoted by $P(t) = \bigcup_{i=1}^N P_i(t)$. Boundaries are denoted by $\Gamma = \partial\Omega$ and $\partial P_i(t)$.

For the unknown rigid body motions $(\mathbf{U}_i, \boldsymbol{\omega}_i)$ of the particles, we need balance equations for forces and torques on particle surfaces. In the absence of inertia, and external forces $\mathbf{F}_{\text{ext},i}$ and torques $\mathbf{T}_{\text{ext},i}$ acting on the particle $P_i(t)$ are balanced by the net hydrodynamic force \mathbf{F}_i and torque \mathbf{T}_i on the particle, respectively:

$$\mathbf{F}_i = \int_{\partial P_i(t)} \boldsymbol{\sigma} \cdot \hat{\mathbf{n}} \, ds = \mathbf{F}_{\text{ext},i}, \quad (4.14)$$

$$\mathbf{T}_i = \int_{\partial P_i(t)} (\mathbf{x} - \mathbf{X}_i) \times (\boldsymbol{\sigma} \cdot \hat{\mathbf{n}}) \, ds = \mathbf{T}_{\text{ext},i}, \quad (4.15)$$

where $\hat{\mathbf{n}}$ is the outwardly directed unit normal vector on the particle surface $\partial P_i(t)$. The particle positions \mathbf{X}_i and angular orientations $\boldsymbol{\Theta}_i$ are obtained from the following kinematic

equations:

$$\frac{d\mathbf{X}_i}{dt} = \mathbf{U}_i, \quad \mathbf{X}_i(t=0) = \mathbf{X}_{i,0}, \quad (4.16)$$

$$\frac{d\boldsymbol{\Theta}_i}{dt} = \boldsymbol{\omega}_i, \quad \boldsymbol{\Theta}_i(t=0) = \boldsymbol{\Theta}_{i,0}. \quad (4.17)$$

At the fluid-particle interface, we use no-slip boundary condition:

$$\mathbf{u} = \mathbf{U}_i + \boldsymbol{\omega}_i \times (\mathbf{x} - \mathbf{X}_i) \quad \text{on } \partial P_i(t) \quad (i = 1, \dots, N). \quad (4.18)$$

4.2.3 Partial wetting boundary conditions

The particle may be neutral or preferably wetted by one of the components of the binary fluid. This effect is accounted for by following the approach given by Cahn [134], where solid-fluid interactions are assumed to be short-ranged. Due to this assumption, the total system free energy \mathcal{F} can be written as:

$$\mathcal{F} = \int_{\Omega \setminus P(t)} f \, dV + \sum_{i=1}^N \int_{\partial P_i(t)} f_p \, ds, \quad (4.19)$$

where f_p is the specific particle free energy which depends only on the concentration at the particle, $\Omega \setminus P(t)$ is the fluid domain bounded with a collective particle surface $\partial P(t) = \bigcup_{i=1}^N \partial P_i(t)$, and f is defined in Eq. (4.1). The surface integral term in Eq. (4.19) represents the contribution of solid-fluid interactions.

At equilibrium, \mathcal{F} is at its minimum. Minimizing \mathcal{F} using methods of variational calculus subject to natural boundary condition gives the following boundary condition on the particle surface $\partial P_i(t)$:

$$-\epsilon \frac{\partial c}{\partial n} + \frac{\partial f_p}{\partial c} = 0, \quad (4.20)$$

where n is the direction normal to $\partial P_i(t)$ and f_p is the specific particle free energy.

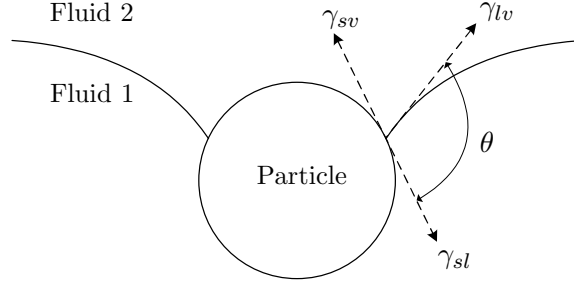
For f_p we use the form proposed by Jacqmin [135], which reads,

$$f_p = \chi \left(c - \frac{c^3}{3} \right), \quad (4.21)$$

where χ is assumed to be a constant and referred to as the wetting potential. It can be made to vary spatially to indicate chemical heterogeneity of the particle surface. With f_p of the form given by Eq. (4.21), $\frac{\partial f_p}{\partial c}$ evaluated at c_B is zero. Therefore, at equilibrium, $\frac{\partial c}{\partial n} = 0$ on the particle surface.

Eqs. (4.20) and (4.21) are non-dimensionalized using the dimensionless variables defined in Section 4.2.1 with the addition of γ_{lv} as the characteristic scale for the specific particle free energy to give:

$$-C \frac{\partial c}{\partial n} + \chi (1 - c^2) = 0, \quad (4.22)$$

Figure 4.1: Definition of contact angle θ .

$$f_p = \chi \left(c - \frac{c^3}{3} \right), \quad (4.23)$$

where $\chi = \frac{\phi_{CB}}{\gamma_{lv}}$ is the dimensionless wetting potential and C is the Cahn number defined in Eq. (4.13).

Using Young's equation which connects the contact angle θ with the interfacial tensions γ_{lv} , γ_{sv} and γ_{sl} :

$$\cos \theta = \frac{\gamma_{sv} - \gamma_{sl}}{\gamma_{lv}}, \quad (4.24)$$

which is also shown in Fig. 4.1. The parameter χ can be related to the (equilibrium) contact angle θ to yield:

$$\cos \theta = \frac{4}{3}\chi. \quad (4.25)$$

From (4.25) it is concluded that for a contact angle θ equal to 90° , χ is zero and the mixed boundary condition (4.22) reduces to the natural boundary condition $\frac{\partial c}{\partial n} = 0$. In this chapter, we only consider a contact angle $\theta = 90^\circ$.

4.3 Numerical methods

4.3.1 Weak form

In deriving the weak form of the governing equations for a fluid-particle system, we follow the combined equation of motion approach [64], in which the hydrodynamic forces and torques acting on particles are eliminated from the equation of motion since they are internal. The no-slip boundary condition on the particle surface is imposed by using constraints implemented with Lagrangian multipliers.

$$\left(r, \frac{\partial c}{\partial t} + \mathbf{u} \cdot \nabla c \right) + M(\nabla r, \nabla \mu) = 0, \quad (4.26)$$

$$(e, \alpha c - \beta c^3) - \varepsilon(\nabla e, \nabla c) + (e, \mu) = 0, \quad (4.27)$$

$$\begin{aligned} & ((\nabla \mathbf{v})^T, 2\eta \mathbf{D}) - (\nabla \cdot \mathbf{v}, g) + (\mathbf{v} - (\mathbf{V}_i + \boldsymbol{\chi}_i \times (\mathbf{x} - \mathbf{X}_i)), \boldsymbol{\lambda}_i)_{\partial P_i(t)} \\ & = (\mathbf{v}, \mathbf{t})_{\Gamma_N} + \mathbf{V}_i \cdot \mathbf{F}_{\text{ext},i} + \boldsymbol{\chi}_i \cdot \mathbf{T}_{\text{ext},i} + (\mathbf{v}, \rho \mu \nabla c), \end{aligned} \quad (4.28)$$

$$(q, \nabla \cdot \mathbf{u}) = 0, \quad (4.29)$$

$$(\boldsymbol{\mu}_i, \mathbf{u} - (\mathbf{U}_i + \boldsymbol{\omega}_i \times (\mathbf{x} - \mathbf{X}_i)))_{\partial P_i(t)} = 0, \quad (4.30)$$

for all admissible test functions r , e , \mathbf{v} , q , $\boldsymbol{\mu}_i$, \mathbf{V}_i and $\boldsymbol{\chi}_i$. (\cdot, \cdot) , $(\cdot, \cdot)_{\Gamma_N}$ and $(\cdot, \cdot)_{\partial P_i(t)}$ are proper inner products on the fluid domain $\Omega \setminus P(t)$, on the Neumann boundary Γ_N and on the particle interface $\partial P_i(t)$, respectively.

4.3.2 Time discretization of the diffuse-interface model

We solve the governing equations Eqs. (4.26)-(4.30) in a decoupled way. First, the concentration c and chemical potential μ are solved simultaneously from Eqs. (4.26) and (4.27) with the natural boundary condition $\frac{\partial c}{\partial n} = 0$ and $\frac{\partial \mu}{\partial n} = 0$, which implies a contact angle $\theta = 90^\circ$ on the particle surface. Then, we solve the Stokes-type flow problem by treating the additional $\rho \mu \nabla c$ term as a forcing.

For the time discretization of the evolution equation of the concentration (Eq. (4.26)), we use a second-order Gear scheme:

$$\left(r, \frac{\frac{3}{2}c^{n+1} - 2c^n + \frac{1}{2}c^{n-1}}{\Delta t} + \hat{\mathbf{u}}^{n+1} \cdot \nabla c^{n+1} \right) + M(\nabla r, \nabla \mu^{n+1}) = 0, \quad (4.31)$$

$$(e, \alpha c^{n+1} - \beta (c^{n+1})^3) - \varepsilon(\nabla e, \nabla c^{n+1}) + (e, \mu^{n+1}) = 0, \quad (4.32)$$

where Δt is the time step and $\hat{\mathbf{u}}^{n+1} = 2(\mathbf{u}^n - \mathbf{u}_m^n) - (\mathbf{u}^{n-1} - \mathbf{u}_m^{n-1})$. Here \mathbf{u}_m represents a mesh velocity due to a mesh movement scheme which will be defined in the following Section 4.3.4. Since these equations are non-linear, we solve them by Picard iteration at each time level $t = t^{n+1}$:

$$\left(r, \frac{3}{2} \frac{c_{j+1}}{\Delta t} + \hat{\mathbf{u}}^{n+1} \cdot \nabla c_{j+1} \right) + M(\nabla r, \nabla \mu_{j+1}) = \left(r, \frac{2c^n - \frac{1}{2}c^{n-1}}{\Delta t} \right), \quad (4.33)$$

$$(e, (\alpha - \beta c_j^2)c_{j+1}) - \varepsilon(\nabla e, \nabla c_{j+1}) + (e, \mu_{j+1}) = 0, \quad (4.34)$$

for $j = 0, 1, \dots$ until convergence, with $c_0 = c^n$. For the first time step $t = \Delta t$, we use an implicit Euler scheme instead of Eq. (4.31):

$$\left(r, \frac{c^{n+1} - c^n}{\Delta t} + (\mathbf{u}^n - \mathbf{u}_m^n) \cdot \nabla c^{n+1} \right) + M(\nabla r, \nabla \mu^{n+1}) = 0. \quad (4.35)$$

The complete time integration steps are given in Section 4.3.5.

4.3.3 XFEM formulation

For the direct numerical simulation of flows with freely suspended particles, we use the extended finite element method (XFEM) combined with the temporary arbitrary Lagrangian-

Eulerian scheme, which is presented by Choi et al. [130]. Here we briefly review the method. For a detailed description of the method, please see [130] and the references therein.

In the XFEM context, the velocity, pressure, concentration and chemical potential can be discretized as:

$$\mathbf{u}_h(\mathbf{x}) = \sum_k \varphi_k(\mathbf{x}) H(s) \mathbf{u}_k, \quad (4.36)$$

$$g_h(\mathbf{x}) = \sum_k \psi_k(\mathbf{x}) H(s) g_k, \quad (4.37)$$

$$c_h(\mathbf{x}) = \sum_k \varphi_k(\mathbf{x}) H(s) c_k, \quad (4.38)$$

$$\mu_h(\mathbf{x}) = \sum_k \varphi_k(\mathbf{x}) H(s) \mu_k, \quad (4.39)$$

where

$$H(s) = \begin{cases} +1 & \text{if } s \geq 0, \\ 0 & \text{if } s < 0, \end{cases} \quad (4.40)$$

is defined by a level set function s :

$$s(\mathbf{x}) \begin{cases} = 0 & \text{if } \mathbf{x} \text{ is on } \partial P(t), \\ > 0 & \text{if } \mathbf{x} \text{ is in } \Omega \setminus P(t), \\ < 0 & \text{if } \mathbf{x} \text{ is in } P(t). \end{cases} \quad (4.41)$$

We use a bi-quadratic Q_2 interpolation (φ_k) for the velocity \mathbf{u} , concentration c and chemical potential μ ; a bi-linear Q_1 interpolation (ψ_k) for the modified pressure g .

For elements intersected by the surface of a particle, the no-slip boundary condition (Eq. (4.18)) is imposed by using constraints implemented with Lagrangian multipliers as shown in Eqs. (4.28) and (4.30). The inner product $(\cdot, \cdot)_{\partial P_i(t)}$ is the standard inner product in $L^2(\partial P_i(t))$:

$$(\boldsymbol{\mu}_i, \mathbf{u} - (\mathbf{U}_i + \boldsymbol{\omega}_i \times (\mathbf{x} - \mathbf{X}_i)))_{\partial P_i(t)} = \int_{\partial P_i(t)} \boldsymbol{\mu}_i \cdot (\mathbf{u} - (\mathbf{U}_i + \boldsymbol{\omega}_i \times (\mathbf{x} - \mathbf{X}_i))) \, ds. \quad (4.42)$$

For the discretization of Eq. (4.42), we use a linear shape function P_1 for the discretization of Lagrangian multipliers and a quadratic shape function P_2 for the geometrical shape of each element.

4.3.4 Temporary ALE scheme

For a moving particle problem, the field variable at the previous time levels, such as \mathbf{u}^n , \mathbf{u}^{n-1} , c^n and c^{n-1} , can become undefined near the boundary of the particle since there may have been no fluid flow at time level t^n . To overcome this problem, we use a temporary ALE scheme, which defines a mapping of field variables at the previous time levels on the current time level [130].

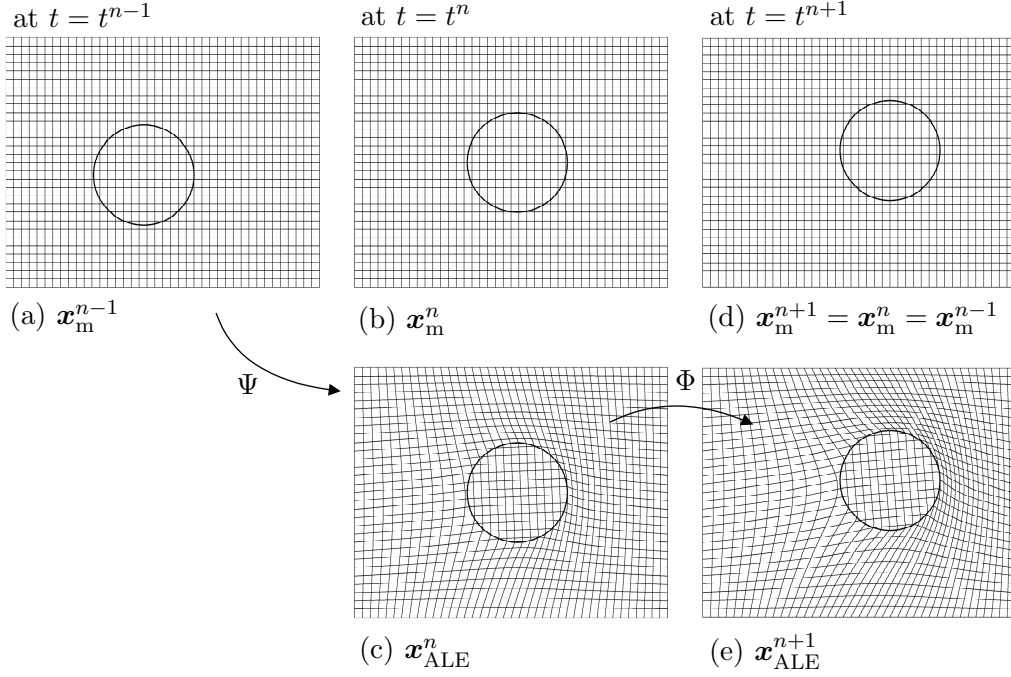


Figure 4.2: Construction of temporary ALE meshes $\mathbf{x}_{\text{ALE}}^n$ and $\mathbf{x}_{\text{ALE}}^{n+1}$ using a second order scheme, which defines the mappings Ψ and Φ .

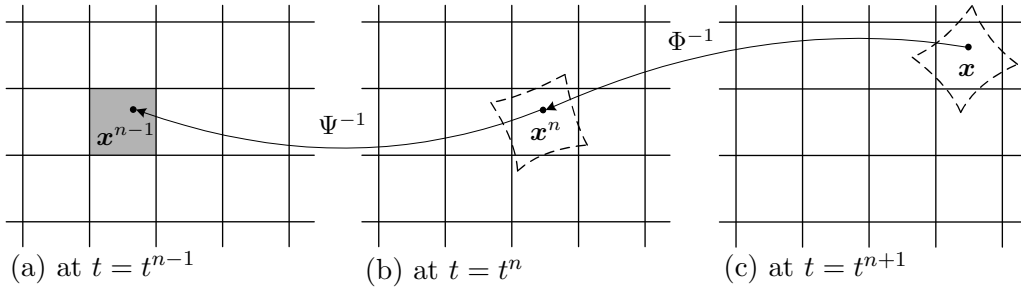


Figure 4.3: Field variable at previous time levels are mapped along with the ALE meshes. The advection of the ALE meshes defines the mappings Ψ and Φ .

In this method, mesh nodes near a particle follow the motion of the particle, whereas, mesh nodes far away from the particle are stationary. A mesh velocity field \mathbf{u}_m is solved using Laplace's equation:

$$\nabla^2 \mathbf{u}_m = \mathbf{0} \quad \text{in } \Omega, \quad (4.43)$$

$$\mathbf{u}_m = \mathbf{0} \quad \text{on } \Gamma, \quad (4.44)$$

$$\mathbf{u}_m = \mathbf{U}_i + \boldsymbol{\omega}_i \times (\mathbf{x} - \mathbf{X}_i) \quad \text{on } \partial P_i(t). \quad (4.45)$$

For a circular particle, Eq. (4.45) can be replaced by $\mathbf{u}_m = \mathbf{U}_i$. Note that Eqs. (4.43)-(4.45) are solved on an Eulerian mesh Ω , including the particle domain $P(t)$, by using a similar technique as in the fictitious domain method [62, 63, 64]. Eq. (4.45) is realized by using a

constraint implemented with Lagrangian multipliers.

An ALE mesh at the previous time level $t = t^n$, $\mathbf{x}_{\text{ALE}}^n$, is constructed using a predictor-corrector method:

$$\mathbf{x}_m^* = \mathbf{x}_m^{n-1} + \mathbf{u}_m(\mathbf{x}_m^{n-1}, t^{n-1})\Delta t \quad (\text{predictor}) \quad (4.46)$$

$$\mathbf{x}_{\text{ALE}}^n = \mathbf{x}_m^{n-1} + \frac{1}{2}[\mathbf{u}_m(\mathbf{x}_m^*, t^n) + \mathbf{u}_m(\mathbf{x}_m^{n-1}, t^{n-1})]\Delta t \quad (\text{corrector}) \quad (4.47)$$

Then, the ALE mesh at the current time level, $\mathbf{x}_{\text{ALE}}^{n+1}$, can be constructed using a second-order Adams-Bashforth method (AB2):

$$\mathbf{x}_{\text{ALE}}^{n+1} = \mathbf{x}_{\text{ALE}}^n + \left[\frac{3}{2}\mathbf{u}_m(\mathbf{x}_{\text{ALE}}^n, t^n) - \frac{1}{2}\mathbf{u}_m(\mathbf{x}_m^{n-1}, t^{n-1})\right]\Delta t \quad (\text{AB2}) \quad (4.48)$$

Eqs. (4.46) and (4.47) define the mapping Ψ and Eq. (4.48) defines the mapping Φ . The mappings Ψ and Φ are represented in Fig. 4.2. The field variables at previous time levels are mapped along with the ALE meshes (Fig. 4.3):

$$c^n = c(\Phi^{-1}(\mathbf{x}), t^n), \quad (4.49)$$

$$\mathbf{u}^n = \mathbf{u}(\Phi^{-1}(\mathbf{x}), t^n), \quad (4.50)$$

$$\mathbf{u}_m^n = \mathbf{u}_m(\Phi^{-1}(\mathbf{x}), t^n), \quad (4.51)$$

$$c^{n-1} = c(\Psi^{-1} \circ \Phi^{-1}(\mathbf{x}), t^{n-1}), \quad (4.52)$$

$$\mathbf{u}^{n-1} = \mathbf{u}(\Psi^{-1} \circ \Phi^{-1}(\mathbf{x}), t^{n-1}), \quad (4.53)$$

$$\mathbf{u}_m^{n-1} = \mathbf{u}_m(\Psi^{-1} \circ \Phi^{-1}(\mathbf{x}), t^{n-1}). \quad (4.54)$$

Note that the unknowns at the current time level, such as c^{n+1} and μ^{n+1} , are computed on the fixed Eulerian mesh.

4.3.5 Time integration

At the initial time $t = 0$ we solve the flow equations without the right-hand side term $\rho\mu\nabla c$ to obtain an initial flow solution. Also the initial concentration field $c^0 = c(t = 0)$ should be specified. Then, we apply the following procedure at every time step.

Step 1. Construct a temporary ALE mesh using Eqs. (4.46)-(4.48) for the interpolation of field variables at previous time levels. At the first time step, we use a first-order scheme given by

$$\mathbf{x}_{\text{ALE}}^{n+1} = \mathbf{x}_m^n + \mathbf{u}_m(\mathbf{x}_m^n, t^n)\Delta t. \quad (4.55)$$

Step 2. Update the particle configuration by integrating the kinematic equations in Eqs. (4.16) and (4.17) using the explicit second-order Adams-Bashforth method (AB2):

$$\mathbf{X}_i^{n+1} = \mathbf{X}_i^n + \left(\frac{3}{2}\mathbf{U}_i^n - \frac{1}{2}\mathbf{U}_i^{n-1}\right)\Delta t, \quad (4.56)$$

$$\Theta_i^{n+1} = \Theta_i^n + \left(\frac{3}{2}\omega_i^n - \frac{1}{2}\omega_i^{n-1}\right)\Delta t. \quad (4.57)$$

For circular particles, the update of angular rotations is not necessary. At the first time step, we use an explicit Euler method:

$$\mathbf{X}_i^{n+1} = \mathbf{X}_i^n + \mathbf{U}_i^n \Delta t. \quad (4.58)$$

Step 3. Modify the computational mesh to avoid very small integration areas [130]. First, compute a mesh velocity field $\hat{\mathbf{u}}_m$:

$$\nabla^2 \hat{\mathbf{u}}_m = \mathbf{0} \quad \text{in } \Omega, \quad (4.59)$$

$$\hat{\mathbf{u}}_m = \mathbf{0} \quad \text{on } \Gamma, \quad (4.60)$$

$$\hat{\mathbf{u}}_m = \hat{\mathbf{n}} \quad \text{on } \partial P_i(t), \quad (4.61)$$

where $\hat{\mathbf{n}}$ is the outwardly directed unit normal vector on the particle surface. Then, each mesh point \mathbf{x}_m moves according to the following advection equations:

$$\frac{d\mathbf{x}_m}{dt_m} = \begin{cases} \hat{\mathbf{u}}_m & \text{if } \mathbf{x}_m \in \Omega \setminus P(t), \\ \mathbf{0} & \text{if } \mathbf{x}_m \in P(t), \end{cases} \quad (4.62)$$

$$\mathbf{x}_m(t_m = 0) = \mathbf{x}_{m,0} \quad (4.63)$$

where $\mathbf{x}_{m,0}$ is the initial position of mesh point. In our simulations, we use a third-order Adams-Bashforth method (AB3) and nodal points are moved until each area of integration is larger than 0.5% of the element area.

Step 4. compute c^{n+1} and μ^{n+1} by solving Eqs. (4.33) and (4.34) iteratively.

Step 5. compute \mathbf{u}^{n+1} , g^{n+1} , $\boldsymbol{\lambda}_i^{n+1}$, \mathbf{U}_i^{n+1} and $\boldsymbol{\omega}_i^{n+1}$ from the momentum balance and continuity equation:

$$\begin{aligned} & ((\nabla \mathbf{v})^T, 2\eta \mathbf{D}(\mathbf{u}^{n+1})) - (\nabla \cdot \mathbf{v}, g^{n+1}) + (\mathbf{v} - (\mathbf{V}_i + \boldsymbol{\chi}_i \times (\mathbf{x} - \mathbf{X}_i^{n+1})), \boldsymbol{\lambda}_i^{n+1})_{\partial P_i(t^{n+1})} \\ & = (\mathbf{v}, \mathbf{t})_{\Gamma_N} + \mathbf{V}_i \cdot \mathbf{F}_{\text{ext},i} + \boldsymbol{\chi}_i \cdot \mathbf{T}_{\text{ext},i} + (\mathbf{v}, \rho \mu^{n+1} \nabla c^{n+1}), \end{aligned} \quad (4.64)$$

$$(q, \nabla \cdot \mathbf{u}^{n+1}) = 0, \quad (4.65)$$

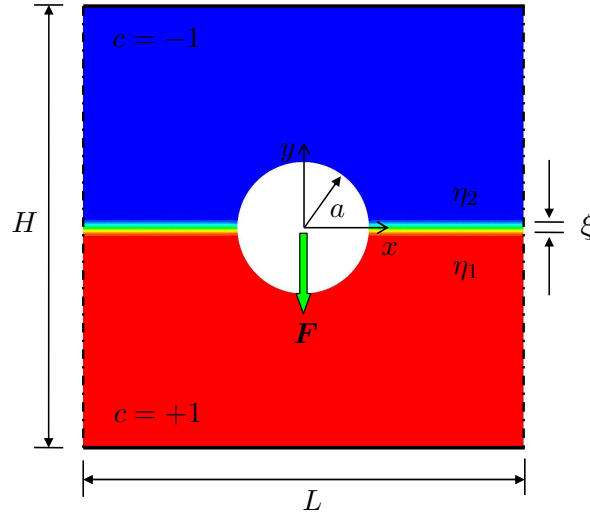
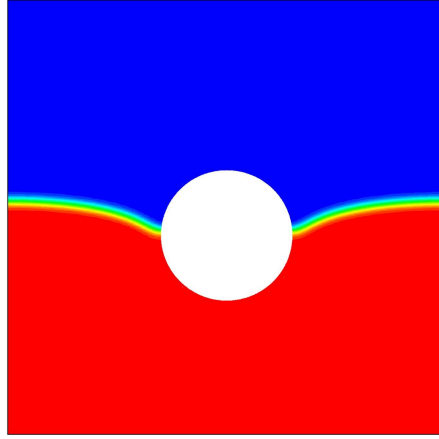
$$(\boldsymbol{\mu}_i, \mathbf{u}^{n+1} - (\mathbf{U}_i^{n+1} + \boldsymbol{\omega}_i^{n+1} \times (\mathbf{x} - \mathbf{X}_i^{n+1})))_{\partial P_i(t^{n+1})} = 0. \quad (4.66)$$

Eqs. (4.33)-(4.34) are solved by using a direct solver HSL MA41, and Eqs. (4.64)-(4.66) are solved by using a direct symmetric solver HSL MA57 [136].

4.4 Particle at a fluid-fluid interface

4.4.1 Problem description

As a model problem a particle is placed at a fluid-fluid interface, confined between two parallel plates. Initially, we assume the steady-state condition, i.e. the fluids are stationary and the particle is at rest in the middle of the fluid-fluid interface (see Fig. 4.4). The particle

Figure 4.4: Geometry for a particle at a fluid-fluid interface at $t = 0$.Figure 4.5: Concentration contours at $t = 0.1$ with forcing time $t_{\mathbf{F}} = 0.1$ and $F_y = 10$.

radius is denoted by a , the thickness of the fluid-fluid interface by ξ , and the viscosity of lower and upper fluids are η_1 and η_2 , respectively. The effective viscosity η is assumed to have a linear relationship given by Eq. (4.7). At $t = 0$, we disturb the flow by applying an external force $\mathbf{F} = (0, -F_y)$ on the particle in y -direction for a certain time duration $t_{\mathbf{F}}$. For $t > t_{\mathbf{F}}$, the external force \mathbf{F} on the particle is removed, and the particle freely moves to its equilibrium position as a result of the acting surface tension forces. Note that dependent on the value of $t_{\mathbf{F}}$ the contact position of the interface with the particle may change, but as long as the particle remains at the interface for $t < t_{\mathbf{F}}$ it will return to its equilibrium position.

In this problem, the scaling of equations by dimensionless groups given in Eq. (4.13) is not trivial since a characteristic velocity U is unknown prior to solving the fluid and particle velocities. Instead of stating the dimensionless groups given in Eq. (4.13), we provide the actual values used in our simulations and in principle one could determine the characteristic velocity U to estimate the magnitude of the characteristic dimensionless groups. We fix the

Table 4.1: Meshes used for the simulations.

mesh	number of elements	number of nodes
M1	4900	19881
M2	10000	40401
M3	15625	63001
M4	22500	90601

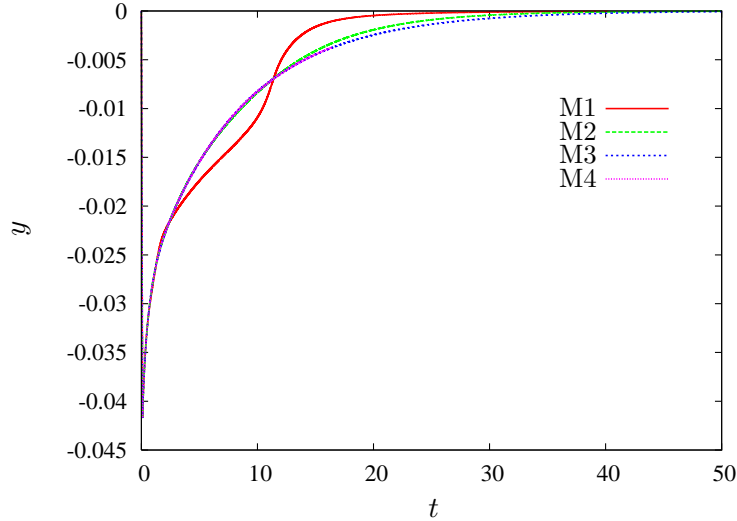


Figure 4.6: Mesh convergence showing the position of the particle as a function of time $t \leq 50$ with forcing time $t_{\mathbf{F}} = 0.1$.

channel height $H = 1$ and channel length $L = 1$, assuming that the top and bottom walls are stationary and the flow is periodic in x -direction. Otherwise stated, the particle radius $a = 0.15$, the viscosities $\eta_1 = \eta_2 = 1$, the magnitude of the external force $F_y = 10$, and $\alpha = 1$, $\beta = 1$, $\varepsilon = 0.0001$, $M = 0.01$, $\rho = 100$ for material parameters used in governing equations (4.2), (4.4) and (4.8). Note that the interfacial thickness $\xi = \sqrt{\varepsilon/\alpha} = 0.01$ for the given values.

Fig. 4.5 shows the concentration profile at $t = 0.1$ for forcing time duration $t_{\mathbf{F}} = 0.1$ with $F_y = 10$, where the particle position is lowest for the given forcing condition. After validation of the computational scheme, we will investigate the particle motion to its equilibrium position for various parameters - $t_{\mathbf{F}}$, a , η_2/η_1 , and so on.

4.4.2 Convergence test

Before we study the effect of the different material and process parameters on the dynamics of the particle, we first demonstrate the convergence of the method by mesh and time step refinements. Four meshes are defined with decreasing element size - M1 (70×70 elements), M2 (100×100 elements), M3 (125×125 elements) and M4 (150×150 elements) and the mesh parameters are summarized in Table 4.1. Fig. 4.6 shows the position of the center of the

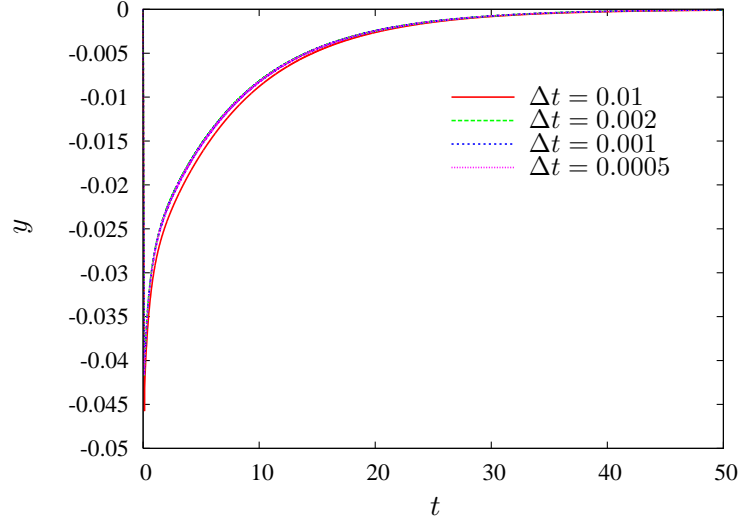


Figure 4.7: Time convergence showing the position of the particle as a function of time $t \leq 50$.

particle as a function of time for the case of forcing time duration $t_{\mathbf{F}} = 0.1$ with the external force $F_y = 10$. The time step $\Delta t = 0.001$ is used for all meshes. The result of M1 shows non-monotonic behavior of particle movement due to un-resolved mesh resolution; the results of M3 and M4 are fully overlapped and can not be distinguished.

The time step convergence is checked by using mesh M3 as it demonstrated a fully resolved mesh resolution. Fig. 4.7 shows the histories of the particle position obtained by using various time steps. For $\Delta t = 0.01$, the particle motion is predicted slightly slower than the other cases. For all other cases, $\Delta t \leq 0.002$, the particle positions fully overlap, but if we use a time step $\Delta t \geq 0.02$, the simulation becomes unstable. Hence, for all our simulations in the chapter, we use the mesh M3 in combination with the time step $\Delta t = 0.001$.

Finally, to obtain more information about the induced flow by the retracting particle, we study the vorticity of the flow as a function of time. Note that the vorticity is defined by

$$\zeta = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}, \quad (4.67)$$

where u and v are x - and y -directional velocity of the fluids, respectively. Figure 4.8 shows the vorticity in the domain at times $t = 1$ and $t = 5$ with forcing time $t_{\mathbf{F}} = 0.1$. The figure shows that the magnitude of the vorticity is larger below the particle compared to above the particle and a flow is induced to push the particle towards its equilibrium position. Figure 4.9 shows the corresponding pressure plots at the same time levels; these figures also show a higher pressure below the particle that pushes the particle in the direction back to equilibrium. Fig. 4.10 shows the maximum of the absolute value of the vorticity in the fluid domain $|\zeta|_{\max}$ as a function of time. During $0 \leq t \leq t_{\mathbf{F}}$, the maximum vorticity increases due to external force \mathbf{F} , then decreases to zero as time goes on. The maximum vorticity obtained by using M1 and M2 shows fluctuations in time especially when we pull down the particle during $0 \leq t \leq t_{\mathbf{F}}$. By further refining to M3 and M4, we can obtain mesh convergence for

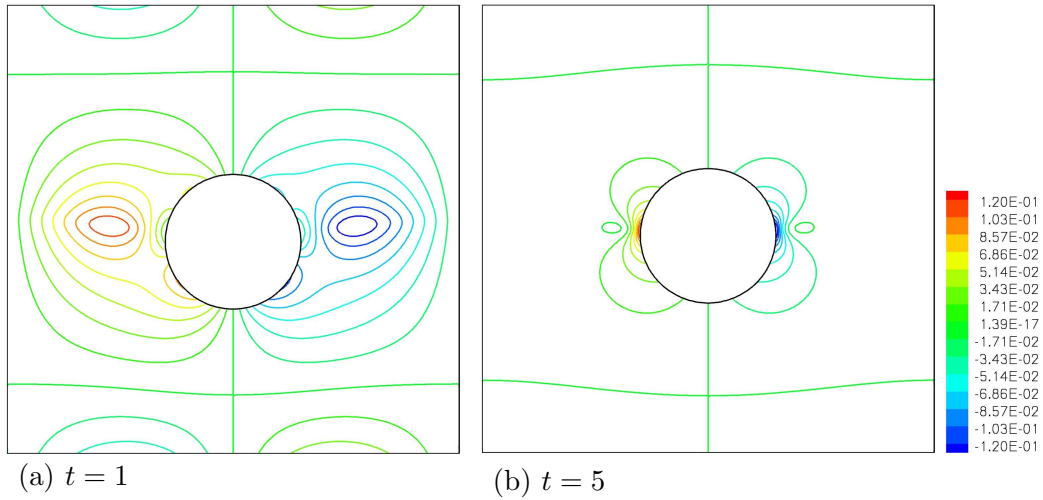


Figure 4.8: Vorticity contours at time $t = 1$ and $t = 5$ with forcing time $t_F = 0.1$.

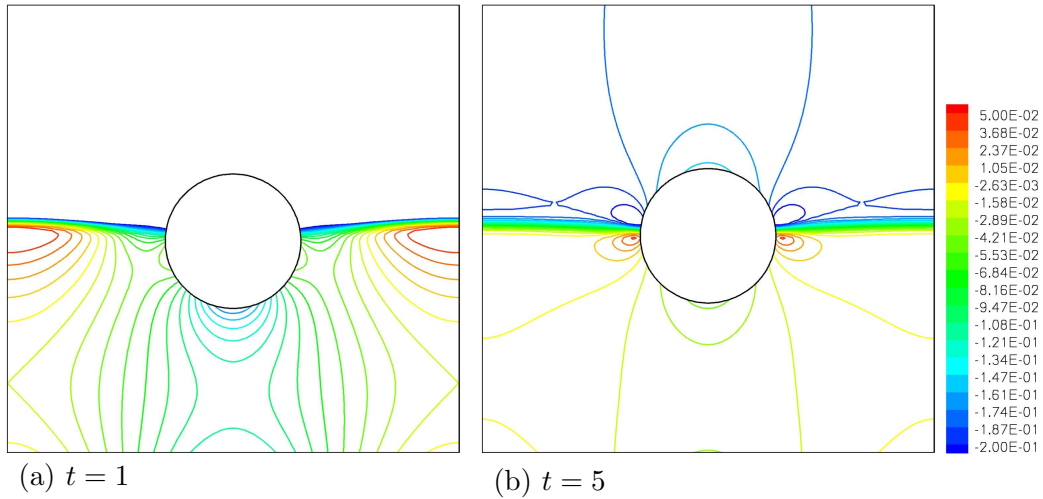


Figure 4.9: Pressure contours at time $t = 1$ and $t = 5$ with forcing time $t_F = 0.1$.

maximum vorticity in time.

4.4.3 Time duration of applied external force

We investigate the effect of the time duration of applied external force on the particle. For time $0 \leq t \leq t_F$, the particle moves downward due to the action of external force \mathbf{F} (see Fig. 4.11), then the particle moves freely under the influence of the surface tension. Eventually, the particle reaches to its equilibrium position in the middle of the channel as shown in Fig. 4.12, as long as the particle stays in between the fluid-fluid interface during the external disturbance.

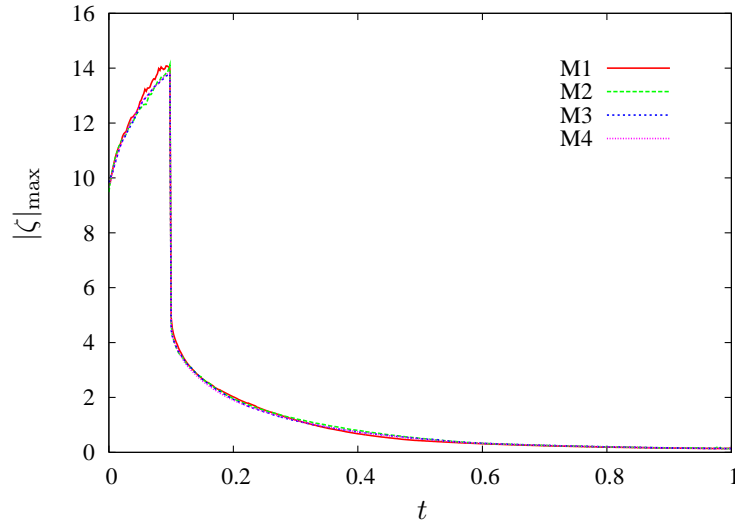


Figure 4.10: Maximum of absolute vorticity in the fluid domain as a function of time $t \leq 1$ with forcing time $t_{\mathbf{F}} = 0.1$.

4.4.4 Interfacial thickness

In the diffuse-interface model, the interfacial thickness is defined by $\xi = \sqrt{\varepsilon/\alpha}$, as explained in Section 4.2. The non-dimensional measure for the interface thickness is the Cahn number, defined by $C = \xi/H$. Since we scaled every length with channel height H , i.e. $H = 1$, the interfacial thickness ξ is already non-dimensional ($\xi = C$). We change the interfacial thickness ξ by changing ε values while keeping $\varepsilon = M^2$ relationship. Note that $C = \xi = \sqrt{\varepsilon} = M$ since we fix $\alpha = 1$. Fig. 4.13 shows the interface of Cahn number 0.01 and 0.04, which means the interfacial thickness is 1% and 4% of the channel height, respectively. Note that the particle radius is $a = 0.15$.

Fig. 4.14 shows the histories of the particle position for various Cahn numbers where $t_{\mathbf{F}} = 0.05$ with $F_y = 10$. As the Cahn number increases, i.e. the interfacial thickness increases, the particle moves rapidly towards the equilibrium position.

Here we varied ε values to change the interfacial thickness. However this changes not only the Cahn number but also the Capillary number and Péclet number (Eq. (4.13)). Hence, Fig. 4.14 is manifested by the combination of these non-dimensional parameters. In the Cahn-Hilliard model, it is not trivial to change the Cahn number only without affecting other parameters.

4.4.5 Surface tension

The non-dimensional measure of the surface tension is described by the capillary number $\text{Ca} = \xi\eta U/\rho\varepsilon c_B^2$, which contains a characteristic velocity U . Since the fluid and particle velocities are unknowns and part of the solution, it is not trivial to define a characteristic velocity U , prior to solving the given problem. In our simulations, we change the value of ρ to change the surface tension, while fixing the other values $\xi = 0.01$, $\eta = 1$, $\varepsilon = 10^{-4}$ and

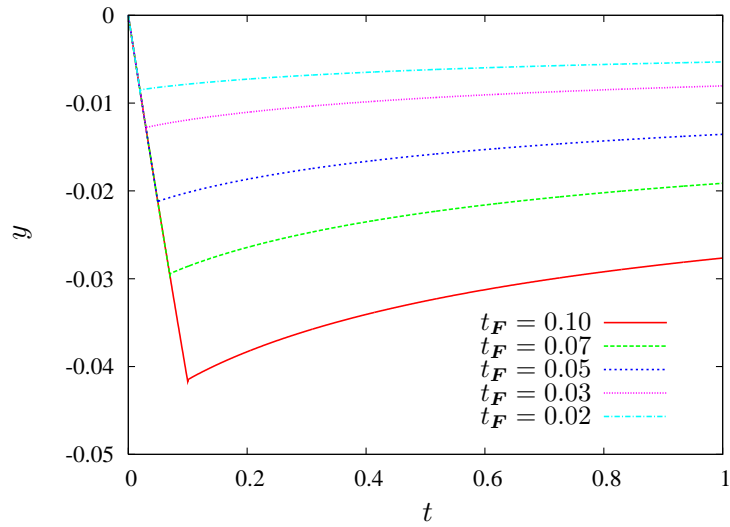


Figure 4.11: The position of the particle as a function of time $t \leq 1$ for different forcing times t_F .

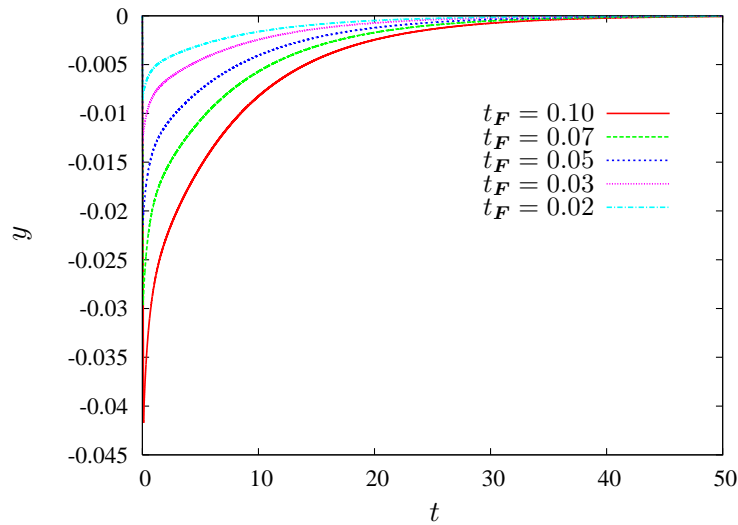


Figure 4.12: The position of the particle as a function of time $t \leq 50$ for different forcing times t_F .

$M = 10^{-2}$. By increasing ρ , the surface tension increases (the capillary number Ca decreases).

Fig. 4.15 shows the histories of particle position for various ρ values where $t_F = 0.05$ with $F_y = 10$. As ρ increases, i.e. the surface tension increases, the particle moves rapidly towards the equilibrium position. The results are quite intuitive: since the driving force pulling the particle back to its original equilibrium position is the surface tension, particle will return faster under higher surface tension.

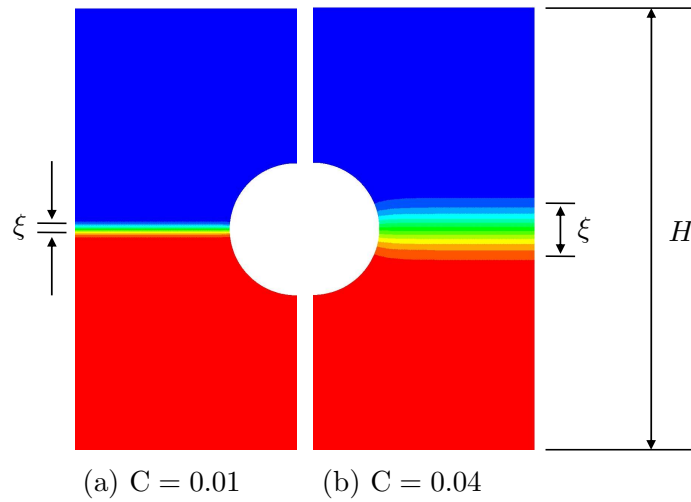


Figure 4.13: Interfacial thickness for the Cahn number $C = 0.01$ and $C = 0.04$.

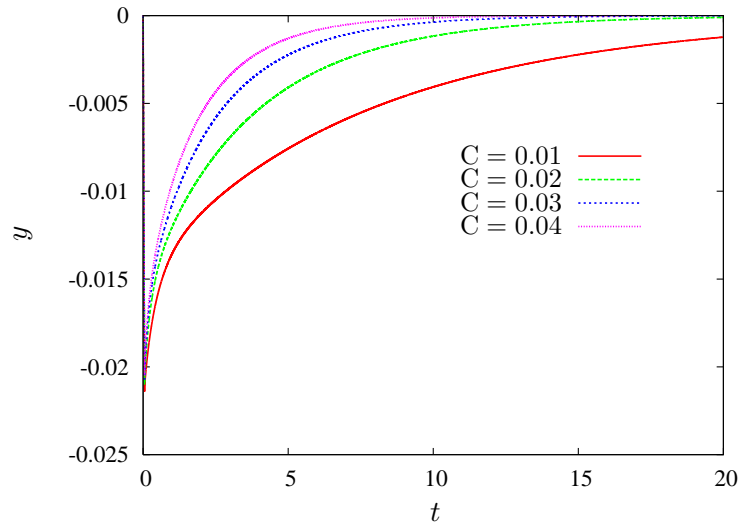


Figure 4.14: Effect of the Cahn number on the position of the particle as a function of time $t \leq 20$.

4.4.6 Particle size

Fig. 4.16 shows the histories of the particle position for various particle radii where $t_{\mathbf{F}} = 0.05$ with $F_y = 10$. As the particle radius a decreases, the particle moves further downward when external force \mathbf{F} is applied since smaller particle experiences less drag than larger one. After the external force is released, smaller particle moves faster than larger one, which can be seen in Fig. 4.17 where the y -directional translational velocity of the particle V is shown.

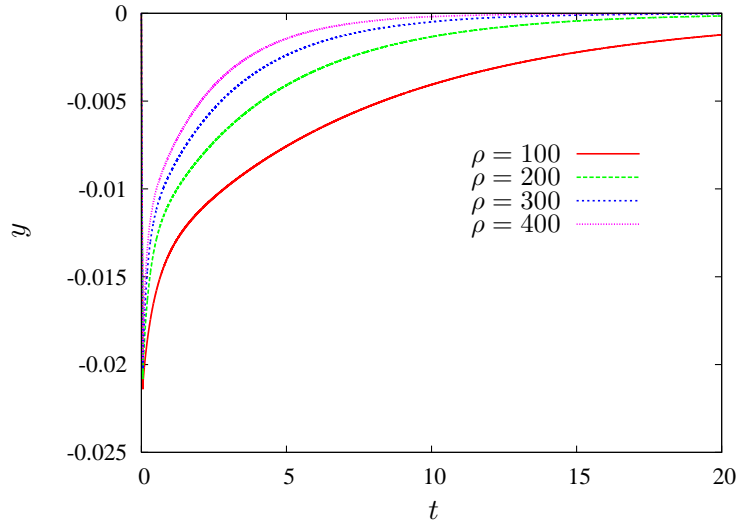


Figure 4.15: Effect of the surface tension on the position of the particle as a function of time $t \leq 20$. As ρ increases, the surface tension increases.

4.4.7 Viscosity ratio

We now consider the effect of different viscosities of upper and lower fluids. We define the viscosity ratio $\lambda = \eta_2/\eta_1$. Fig. 4.18 shows the histories of the particle position for different viscosity ratios where $t_{\mathbf{F}} = 0.05$ with $F_y = 10$. As the viscosity ratio decreases, i.e. the upper fluid is thinner than the lower fluid, the particle moves further downward when external force \mathbf{F} is applied. After the external force is released, the particle moves faster for lower viscosity ratios, eventually the particle reaches the equilibrium position earlier. The fast movement of the particle for lower viscosity ratios is clearly seen in Fig. 4.19, where the y -directional translational velocity of the particle V is plotted after the external force \mathbf{F} is released.

4.5 Multiple-layer configuration

In this section, we show dynamics of a particle passing through multiple layers of fluids, confined between two parallel plates. A schematic description of the problem is shown in Fig. 4.20. The length of the channel is $L = 1$, the positions of the uppermost and lowest fluids are $H_1 = H_5 = 0.6$ and the positions of the fluids in-between are $H_2 = H_3 = H_4 = 0.3$. The viscosity of the fluid layers is chosen $\eta = 1$, but the model can handle different viscosity ratios as shown in the previous section. A particle of radius $a = 0.15$ is suspended at the initial position $\mathbf{X}_0 = (0, 1.8)$ and is sedimenting downward as a results of a constant external force $\mathbf{F} = (0, -10)$ acting on the particle. The upper and lower walls are stationary and the flow is assumed to be periodic in x -direction. The material parameters used in governing equations (4.2), (4.4) and (4.8) are $\alpha = 1$, $\beta = 1$, $\varepsilon = 0.0001$, $M = 0.01$, $\rho = 100$. Note that the interfacial thickness $\xi = \sqrt{\varepsilon/\alpha} = 0.01$.

The problem is solved using a mesh with 100×210 elements since it provides an accurate

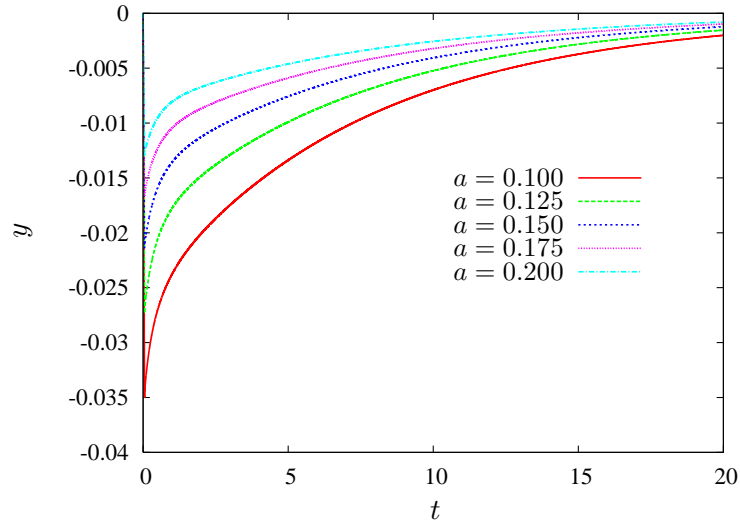


Figure 4.16: Effect of the particle size on the position of the particle as a function of time $t \leq 20$.

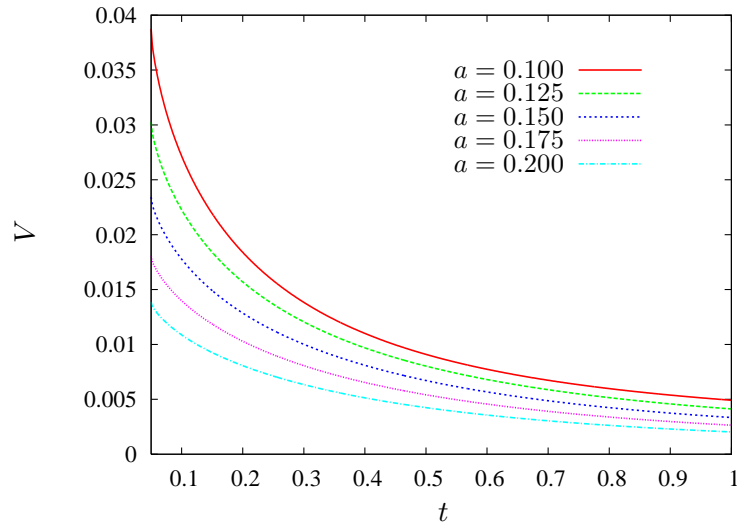


Figure 4.17: Effect of the particle size on the velocity of the particle as a function of time $t \leq 1$.

solution with manageable computational costs. The time step $\Delta t = 0.001$ is used for the simulation. Fig. 4.21 shows the evolution of the structure of multi-layered morphology caused by the sedimenting particle. As the particle passes through the multiple layers of fluids, a fluid layer break up into several drops, then they merge with other layers of the fluid. Fig. 4.22 shows the translational velocity of the particle in y -direction as a function of time. The proposed method can provide fully resolved velocity fields for the particle and fluids, associated with the evolution of the inherently complex morphology.

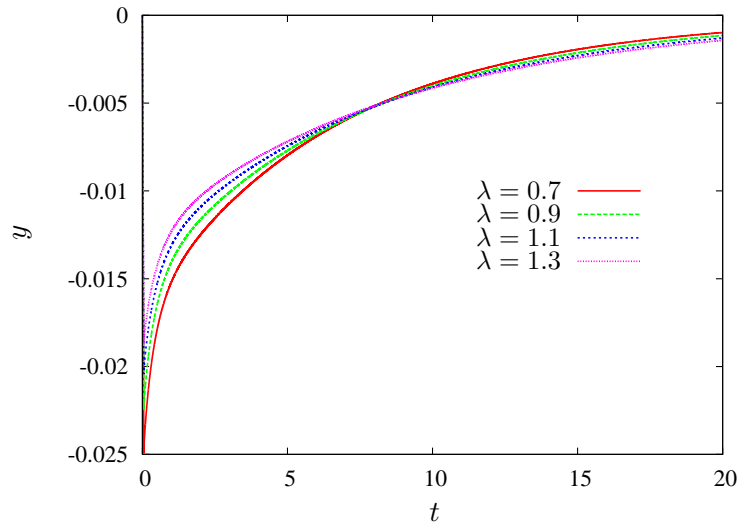


Figure 4.18: Effect of the viscosity ratio on the position of the particle as a function of time $t \leq 20$.

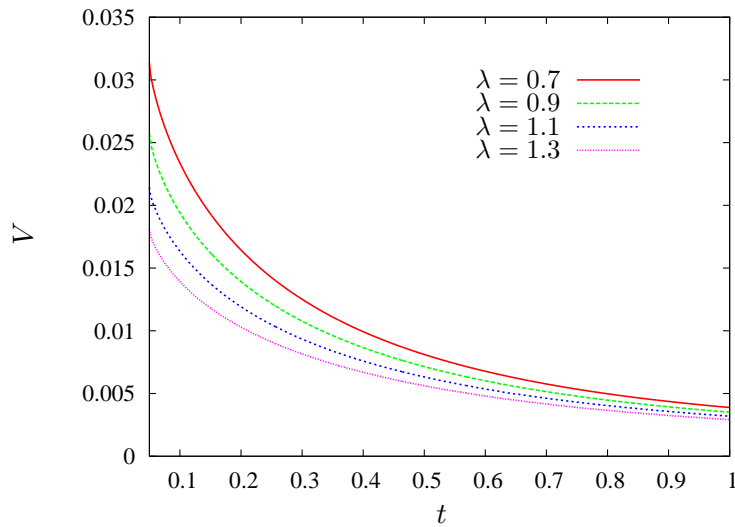


Figure 4.19: Effect of the viscosity ratio on the velocity of the particle as a function of time $t \leq 1$.

4.6 Conclusions

We present a combined model of Cahn-Hilliard theory and extended finite element method for the dynamics of particles suspended in two-phase flows. In the diffuse interface model of Cahn-Hilliard, the interface is considered to have a small but finite thickness. Interface profile and thickness are determined by governing equations that couple thermodynamic and hydrodynamic forces.

For the direct numerical simulation of flows with suspended particles, we use the extended

finite element method, which decouples the fluid and particle domain while using a computational mesh including both fluids and particles. To cope with the movement of particles, a temporary ALE scheme is used to define a mapping of field variables at previous time levels onto the computational mesh at the current time level. The no-slip boundary condition at the particle surface is imposed by using a constraint implemented with Lagrangian multipliers. By combining the diffuse interface model and extended finite element method, the particle motion at a fluid-fluid interface can be simulated on a fixed Eulerian mesh without any need of remeshing.

We present the motion of a single particle at an interface between two fluids. Initially, the fluids and particle are stationary. The initial equilibrium state is disturbed by applying a constant force on the particle for a certain time duration. Then the external force on the particle is released, and the particle moves freely to its equilibrium position under the action of surface tension. As long as the particle stays in between the two fluids during external disturbance, it always came back to its initial equilibrium position. We investigated the effect of interfacial thickness, surface tension, particle size and viscosity ratio of the two fluids on the particle movement towards its equilibrium position. As the interfacial thickness increases, the surface tension increases, the particle size decreases, or the viscosity ratio decreases, the particle moves more rapidly towards its equilibrium position after releasing external force. We also demonstrated the wide applicability of the method and determined the motion of a sedimenting particle passing through multiple layers of fluids and the corresponding morphology change of the fluids. The proposed method is general and is applicable to more complex problems, such as multiple particles in phase separating fluids and structure formation of particles at a fluid-fluid interface. Also the method can be easily extended to three-dimensional simulations without any loss of generality, only requiring heavier computational load. Future work will be focused on these problems.

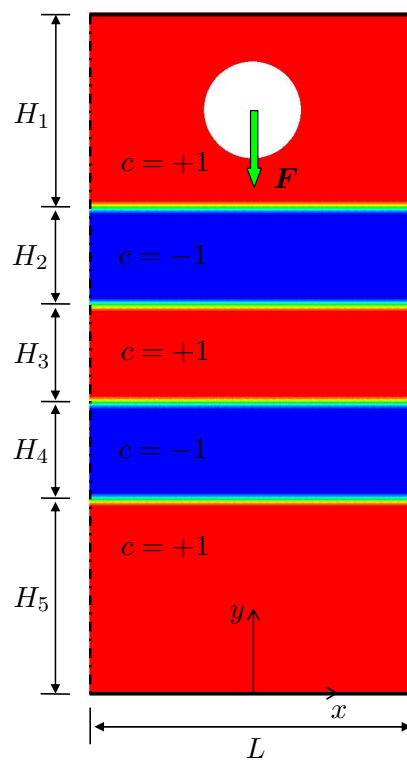


Figure 4.20: Geometry for a particle in multiple layers of fluids.

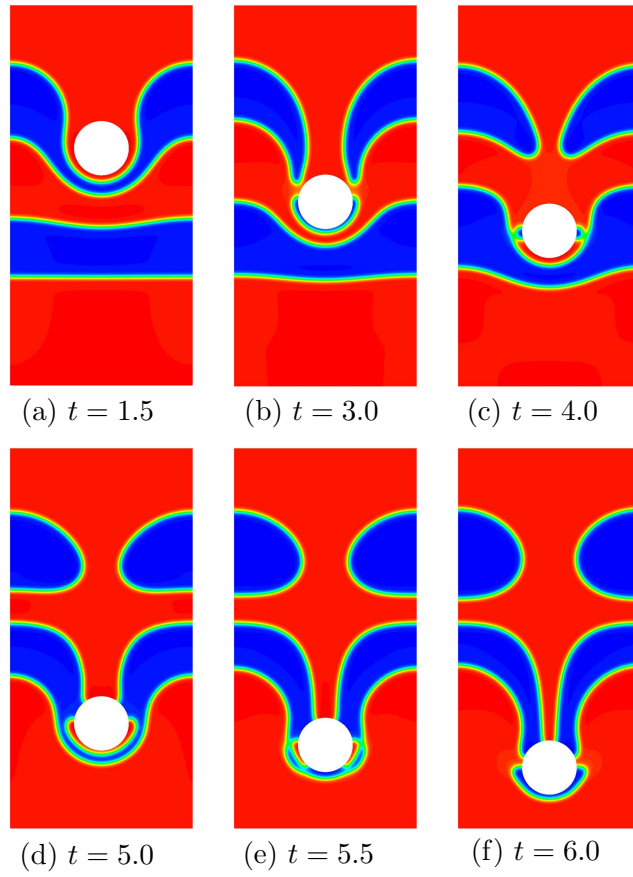


Figure 4.21: Snapshots of a particle passing through multiple layers of fluids.

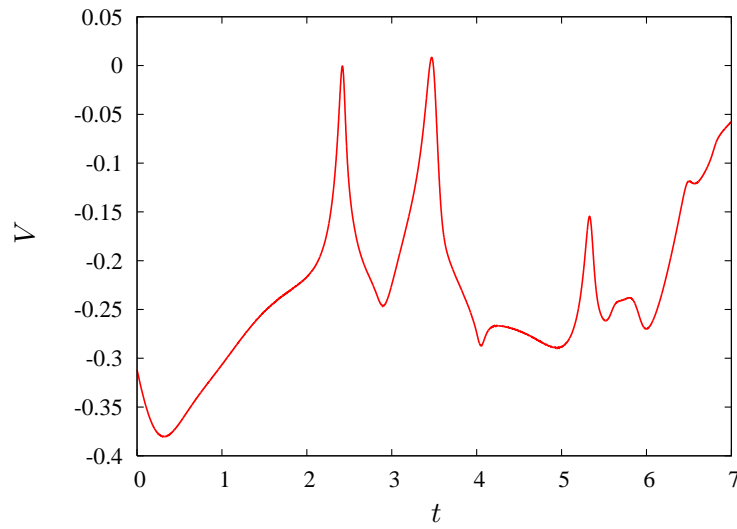


Figure 4.22: Translational velocity of the particle passing through multiple layers of fluids as a function of time.

Chapter 5

Alignment of particles in confined shear flow

In this chapter, the alignment of particles in a confined shear flow of a viscoelastic fluid is quantitatively analyzed using an extended finite element method (XFEM) with a temporary arbitrary Lagrangian-Eulerian (ALE) scheme. We have improved some numerical algorithms from Chapter 3. The no-slip boundary condition on the particle surface is realized by using a newly proposed weak boundary condition, which is equivalent to adding a positive definite stabilizing term in the momentum balance. The subdivided elements at the interface are exactly integrated using higher order Gauss integration rules, which eliminates the need of mesh optimization to avoid very small integration areas. The temporary ALE scheme is extended to be used with a predictor-corrector method for the update of positions of the particles. Furthermore, a decoupled semi-implicit polymer stress formulation is incorporated, which enables simulations of viscoelastic fluid flows without a Newtonian solvent contribution.

We applied the proposed method to the string formation of particles in a confined shear flow of a viscoelastic fluid. Once particles form a string-like structure, the final state is independent of the initial particle positions and the histories to reach the steady-state. For a certain fluid

The content of this Chapter is mainly based on:

Young Joon Choi and Martien A. Hulsen. Alignment of particles in a confined shear flow of a viscoelastic fluid. *Journal of Non-Newtonian Fluid Mechanics*, under review.

The two-particle interactions, which are based on:

Young Joon Choi, Martien A. Hulsen, and Han E.H. Meijer. An extended finite element method for the simulation of particulate viscoelastic flows. *Journal of Non-Newtonian Fluid Mechanics*, 165:607-624, 2010.

are also presented in this chapter for a consistent presentation of the topic. Note that the remaining parts of the paper are presented in Chapter 3.

rheology, the maximum obtainable length of a string of particles is limited. As the Weissenberg number increases, particles can form longer strings. Moderate wall confinement promotes the alignment of particles, however, too strong confinement hinders the alignment by enhancing repulsive interaction between particles. The steady-state angular velocities of particles are compared with respect to the length of strings. If particles can form sufficiently long strings, the steady-state angular velocities of the two end-particles do not change significantly, and those of the non end-particles increase, as the string length increases. In a given string, the angular velocity of the two end-particles is faster than those of the particles in between. We have also presented the steady-state interparticle distance between two neighboring particles in a string. As the string length increases, the interparticle distance increases.

5.1 Introduction

If particles are added to polymeric fluids, microstructures of multiple particles can be induced by flow, which generally affects apparent properties of the suspension such as viscosity, normal stress, etc. In particular, Michele et al. [51] reported alignment of particles in a confined oscillatory shear flow of a viscoelastic fluid, only if the ratio of the normal stress difference to the shear stress exceeds a certain critical value. They also observed that the rotation of particles almost ceased if the particles are lined up. The phenomenon is usually called alignment, string formation or chaining of particles. In this thesis, these terms are used interchangeably. Later, Lyon et al. [52] confirmed the observation of Michele et al. [51], and reported a quantitative reduction in the shear stress owing to the string formation, which shows a relevant link between the rheological properties of suspensions and the formation of microstructures. Won and Kim [53] reported that shear-thinning plays an important role in the formation of string-like structures, even though the driving force for the movement across the main flow direction is the first normal stress difference. Particles are not aligned in non-shear-thinning viscoelastic liquids under shear flow. Also, Scirocco et al. [54] observed no alignment in highly elastic Boger fluids, which supports that shear-thinning is an essential factor in string formation. Moreover, they demonstrated that the string formation occurs in the bulk of the liquid rather than at the walls. Walls seemed to hinder alignment of particles. Contrary to [54], Pasquino et al. [55] observed alignment only at the walls, after having migrated from the bulk to the walls. The alignment may depend on the details of the rheology of the suspending fluid and the size of the particles relative to the gap thickness. Hwang and Hulsén [56] qualitatively analyzed the alignment of particles in a sliding bi-periodic frame using a direct numerical simulation. Motivated by these experimental and numerical results, we are aiming to investigate the alignment of particles quantitatively, although only two-dimensional simulations are presented in the thesis.

In recent years, extended finite element methods (XFEM) are widely used to model diverse kinds of moving interfaces [94]. The current authors already presented the applications of XFEM to problems of moving particles [130], and with the presence of a free surface [137]. For more information of the XFEM, see the references therein. We apply similar ideas to the problem of string formation of particles with improved numerical algorithms. The no-slip boundary condition on the particle surface is realized by using a newly proposed weak boundary condition, extended from a weak Dirichlet boundary condition [138, 139]. The method is equivalent to adding a positive definite stabilizing term in the momentum balance with additional unknowns for the rigid body motion of particles. The proposed weak boundary condition circumvents the LBB condition associated with the traditional Lagrangian multiplier technique, and seems well suited for fast iterative solvers.

The content of this chapter is as follows. In Section 5.2, we give a brief review of the governing equations for the motion of incompressible viscoelastic fluids and freely suspended rigid particles. In Section 5.3, we present a numerical algorithm of the extended finite element method and a temporary arbitrary Lagrangian-Eulerian (ALE) scheme to cope with the movement of particles. In Section 5.4, a convergence analysis of the temporary ALE scheme

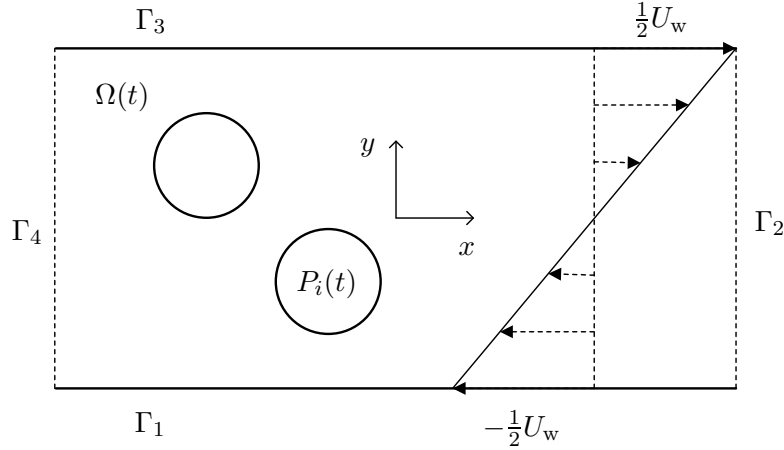


Figure 5.1: Schematic description of the problem.

is presented. In Section 5.5, the proposed method is validated by comparing with known results in the literature. In Section 5.6, numerical results for two-particle interactions in a confined shear flow of a viscoelastic fluid are presented, based on [130]. In Section 5.7, results for the alignment of multiple particles are presented, using the numerical scheme presented in Section 5.3. Conclusions follow in Section 5.8.

5.2 Governing equations

We consider the motion of rigid particles suspended in an incompressible viscoelastic fluid confined between two parallel plates as shown in Fig. 5.1. Let $\Omega(t)$ be the fluid domain and $P_i(t)$ ($i = 1, \dots, N$) be the domain of the i -th particle at time t with the number of particles N . The collective particle region at a certain time t is denoted by $P(t) = \bigcup_{i=1}^N P_i(t)$. The mesh region $\Omega_m = \Omega(t) \cup P(t)$ includes both the fluid and the particles. Note that Ω_m is independent of time.[†] Boundaries are denoted by $\Gamma = \partial\Omega$, $\Gamma_m = \partial\Omega_m$ and $\partial P_i(t)$. We will neglect inertia and body forces for both the fluid and the particles, and the flow is assumed to be periodic in x -direction.

The equations of motion for an incompressible fluid are given by

$$\nabla \cdot \boldsymbol{\sigma} = \mathbf{0} \quad \text{in } \Omega(t), \quad (5.1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega(t), \quad (5.2)$$

where $\boldsymbol{\sigma}$ is the stress tensor and \mathbf{u} is the velocity vector. Eqs. (5.1) and (5.2) are the balance equations of momentum and mass, respectively. For a viscoelastic fluid, the total stress $\boldsymbol{\sigma}$

[†] In this chapter, we have changed notation for a simpler presentation: Ω_m represents the mesh region and $\Omega(t)$ is the fluid domain, contrary to the previous chapters where Ω was used for the mesh region including both the fluid and particles, and $\Omega \setminus P(t)$ was the fluid domain.

can be split into a pressure part, a Newtonian viscous stress and a polymer stress:

$$\boldsymbol{\sigma} = -p\mathbf{I} + 2\eta_s\mathbf{D} + \boldsymbol{\tau}(\mathbf{c}), \quad (5.3)$$

where p is the pressure, \mathbf{I} is the unity tensor, η_s is the Newtonian solvent viscosity, $\mathbf{D} = (\nabla\mathbf{u} + (\nabla\mathbf{u})^T)/2$ is the rate-of-deformation tensor and $\boldsymbol{\tau}(\mathbf{c})$ is the polymer stress which can be represented as a function of the conformation tensor \mathbf{c} . We will use the Giesekus constitutive model for the conformation tensor \mathbf{c} :

$$\lambda \overset{\nabla}{\mathbf{c}} + \mathbf{c} - \mathbf{I} + \alpha(\mathbf{c} - \mathbf{I})^2 = \mathbf{0}, \quad (5.4)$$

where λ is the relaxation time and α is the mobility parameter. The triangle (∇) denotes the upper-convected time derivative, defined as:

$$\overset{\nabla}{\mathbf{c}} = \frac{\partial \mathbf{c}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{c} - (\nabla \mathbf{u})^T \cdot \mathbf{c} - \mathbf{c} \cdot \nabla \mathbf{u}. \quad (5.5)$$

In the Giesekus model, the polymer stress can be written as:

$$\boldsymbol{\tau}(\mathbf{c}) = \frac{\eta_p}{\lambda}(\mathbf{c} - \mathbf{I}), \quad (5.6)$$

where η_p is the polymer viscosity. Furthermore, the equation for the conformation tensor \mathbf{c} , Eq. (5.4), can be transformed to an equivalent log-conformation equation for $\mathbf{s} = \log \mathbf{c}$:

$$\frac{\partial \mathbf{s}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{s} - \mathbf{g}((\nabla \mathbf{u})^T, \mathbf{s}) = \mathbf{0}. \quad (5.7)$$

Once \mathbf{s} has been solved, the conformation tensor \mathbf{c} can be computed from $\mathbf{c} = \exp(\mathbf{s})$. A detailed explanation of the log-conformation representation (LCR) and exact expression for the function $\mathbf{g}((\nabla \mathbf{u})^T, \mathbf{s})$ can be found in [96]. Solving the equation for \mathbf{s} instead of the equation for \mathbf{c} leads to major stability improvements for high Weissenberg number problems. For all numerical simulations in this chapter, we use the log-conformation representation. Hence, we will present the weak formulation, and the boundary and initial conditions only in the LCR.

The boundary and initial conditions are given by:

$$\mathbf{u} = \left(\frac{1}{2}U_w, 0\right) \quad \text{on } \Gamma_1, \quad (5.8)$$

$$\mathbf{u} = \left(-\frac{1}{2}U_w, 0\right) \quad \text{on } \Gamma_3, \quad (5.9)$$

$$\mathbf{u} = \mathbf{U}_i + \boldsymbol{\omega}_i \times (\mathbf{x} - \mathbf{X}_i) \quad \text{on } \partial P_i(t) \quad (i = 1, \dots, N), \quad (5.10)$$

$$(\mathbf{u})_{\Gamma_2} = (\mathbf{u})_{\Gamma_4}, \quad (\mathbf{t})_{\Gamma_2} = (-\mathbf{t})_{\Gamma_4}, \quad (\mathbf{s})_{\Gamma_2} = (\mathbf{s})_{\Gamma_4}, \quad (5.11)$$

$$\mathbf{s}(\mathbf{x}, 0) = \mathbf{s}_0(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega(0). \quad (5.12)$$

Eq. (5.10) represents the no-slip boundary condition on the particle surface, where \mathbf{U}_i is the translational velocity, $\boldsymbol{\omega}_i$ is the rotational velocity and \mathbf{X}_i is the position of the center of

the particle. Eq. (5.11) represents the periodic boundary conditions in x -direction, where $\mathbf{t} = \boldsymbol{\sigma} \cdot \mathbf{n}$ is the traction vector with the outwardly directed unit normal vector \mathbf{n} on the fluid. The periodic boundary conditions are implemented using Lagrangian multipliers in a classical way and will be omitted in a weak form for simplicity of the presentation. Since the flow is assumed to be periodic, the inflow boundary condition for the conformation is absent. In our simulations, we use a stress-free state as initial condition over the whole fluid domain, i.e. $\mathbf{s}_0(\mathbf{x}) = \mathbf{0}, \forall \mathbf{x} \in \Omega(0)$. Since we neglect inertia, an initial condition for the velocity is not necessary.

For the unknown rigid body motion $(\mathbf{U}_i, \boldsymbol{\omega}_i)$ of the particles, we need balance equations for forces and torques on particle surfaces. In the absence of inertia, and external forces and torques on the particles, the net hydrodynamic force \mathbf{F}_i and torque \mathbf{T}_i acting on the particle $P_i(t)$ are zero:

$$\mathbf{F}_i = \int_{\partial P_i(t)} \boldsymbol{\sigma} \cdot \hat{\mathbf{n}} \, ds = \mathbf{0}, \quad (5.13)$$

$$\mathbf{T}_i = \int_{\partial P_i(t)} (\mathbf{x} - \mathbf{X}_i) \times (\boldsymbol{\sigma} \cdot \hat{\mathbf{n}}) \, ds = \mathbf{0}, \quad (5.14)$$

where $\hat{\mathbf{n}}$ is the outwardly directed unit normal vector on the particle surface $\partial P_i(t)$. The particle positions \mathbf{X}_i and angular orientations $\boldsymbol{\Theta}_i$ are obtained from the following kinematic equations:

$$\frac{d\mathbf{X}_i}{dt} = \mathbf{U}_i, \quad \mathbf{X}_i(t=0) = \mathbf{X}_{i,0}, \quad (5.15)$$

$$\frac{d\boldsymbol{\Theta}_i}{dt} = \boldsymbol{\omega}_i, \quad \boldsymbol{\Theta}_i(t=0) = \boldsymbol{\Theta}_{i,0}. \quad (5.16)$$

5.3 Numerical methods

5.3.1 Mixed DEVSS-G/SUPG finite element formulation

In deriving the weak form of the governing equations for a fluid-particle system, we follow the combined equation of motion approach [64], in which the fluid and the particle equation of motion are combined in one equation. For the discretization of the momentum equation, we employ the DEVSS-G method [100, 101].

The no-slip boundary condition on the particle surface (Eq. (5.10)) is imposed by using a weak boundary condition, extended from a weak Dirichlet boundary condition [138, 139]. The formulation is similar to the weak Dirichlet boundary condition [139], but the velocity on the particle surface contains unknowns \mathbf{U}_i and $\boldsymbol{\omega}_i$ in Eq. (5.10). By introducing the variations \mathbf{V}_i and $\boldsymbol{\chi}_i$ of \mathbf{U}_i and $\boldsymbol{\omega}_i$, respectively, we include the force and torque free conditions (Eqs. (5.13)-(5.14)) in the momentum balance. The weak formulation of the governing equations can be stated as follows: Find $\mathbf{u}, p, \mathbf{s}, \mathbf{G}, \mathbf{E}, \mathbf{U}_i$ and $\boldsymbol{\omega}_i$ such that

$$\begin{aligned}
& - (\nabla \cdot \mathbf{v}, p) + ((\nabla \mathbf{v})^T, 2\eta_s \mathbf{D} + \theta(\nabla \mathbf{u} - \mathbf{G}^T) + \boldsymbol{\tau}) \\
& \quad - \sum_{i=1}^N (\mathbf{v} - (\mathbf{V}_i + \boldsymbol{\chi}_i \times (\mathbf{x} - \mathbf{X}_i)), (-p\mathbf{I} + 2\eta_s \mathbf{D} + \boldsymbol{\tau}) \cdot \mathbf{n})_{\partial P_i(t)} \\
& \quad + \sum_{i=1}^N (\mathbf{v} - (\mathbf{V}_i + \boldsymbol{\chi}_i \times (\mathbf{x} - \mathbf{X}_i)), 2\kappa(\mathbf{D}(\mathbf{u}) - \mathbf{E}) \cdot \mathbf{n})_{\partial P_i(t)} = 0, \quad (5.17)
\end{aligned}$$

$$(q, \nabla \cdot \mathbf{u}) = 0, \quad (5.18)$$

$$(\mathbf{H}, -\nabla \mathbf{u} + \mathbf{G}^T) = 0, \quad (5.19)$$

$$\sum_{i=1}^N (\mathbf{K} \cdot \mathbf{n}, \mathbf{u} - (\mathbf{U}_i + \boldsymbol{\omega}_i \times (\mathbf{x} - \mathbf{X}_i)))_{\partial P_i(t)} + (\mathbf{K}, -\mathbf{D}(\mathbf{u}) + \mathbf{E}) = 0, \quad (5.20)$$

for all admissible test functions \mathbf{v} , q , \mathbf{d} , \mathbf{H} , \mathbf{K} , \mathbf{V}_i and $\boldsymbol{\chi}_i$ where (\cdot, \cdot) and $(\cdot, \cdot)_{\partial P_i(t)}$ are proper inner products on the fluid domain $\Omega(t)$ and on the interface $\partial P_i(t)$, respectively. \mathbf{n} is the outwardly directed unit normal vector on the fluid. Some remarks:

- In Eq. (5.17), the boundary integration term $-(\mathbf{v}, (-p\mathbf{I} + 2\eta_s \mathbf{D} + \boldsymbol{\tau}) \cdot \mathbf{n})_{\partial P_i(t)}$ comes from the open boundary formulation [140] on the particle surface $\partial P_i(t)$.
- The term $(\mathbf{V}_i + \boldsymbol{\chi}_i \times (\mathbf{x} - \mathbf{X}_i), (-p\mathbf{I} + 2\eta_s \mathbf{D} + \boldsymbol{\tau}) \cdot \mathbf{n})_{\partial P_i(t)}$ leads to the force and torque free conditions (Eqs. (5.13)-(5.14)).
- The second boundary integration term $(\dots, 2\kappa(\mathbf{D}(\mathbf{u}) - \mathbf{E}) \cdot \mathbf{n})_{\partial P_i(t)}$ stabilizes the method.
- Eq. (5.20) imposes the no-slip boundary condition (Eq. (5.10)) on the particle surface.

The κ parameter is chosen equal to the zero-shear-rate viscosity $\kappa = \eta_0 = \eta_s + \eta_p$, which seems to produce optimal convergence [139]. Furthermore, the DEVSS-G parameter θ is chosen to the polymer viscosity, $\theta = \eta_p$.

If we eliminate \mathbf{E} altogether, we obtain the following *primal* formulation:

$$\begin{aligned}
& - (\nabla \cdot \mathbf{v}, p) + ((\nabla \mathbf{v})^T, 2\eta_s \mathbf{D} + \theta(\nabla \mathbf{u} - \mathbf{G}^T) + \boldsymbol{\tau}) \\
& \quad - (\mathbf{v} - (\mathbf{V}_i + \boldsymbol{\chi}_i \times (\mathbf{x} - \mathbf{X}_i)), (-p\mathbf{I} + 2\eta_s \mathbf{D} + \boldsymbol{\tau}) \cdot \mathbf{n})_{\partial P_i(t)} \\
& \quad + \kappa \sum_{i=1}^N L_i (\mathbf{v} - (\mathbf{V}_i + \boldsymbol{\chi}_i \times (\mathbf{x} - \mathbf{X}_i)), \mathbf{u} - (\mathbf{U}_i + \boldsymbol{\omega}_i \times (\mathbf{x} - \mathbf{X}_i))) = 0, \quad (5.21)
\end{aligned}$$

$$(q, \nabla \cdot \mathbf{u}) = 0, \quad (5.22)$$

$$(\mathbf{H}, -\nabla \mathbf{u} + \mathbf{G}^T) = 0. \quad (5.23)$$

In Eq. (5.21), $L_i(\mathbf{v}, \mathbf{u})$ is defined as:

$$L_i(\mathbf{v}, \mathbf{u}) = M_{km}^{-1}(\mathbf{v}\mathbf{n}, \phi_k)_{\partial P_i(t)} : (\phi_m, \mathbf{n}\mathbf{u} + \mathbf{u}\mathbf{n})_{\partial P_i(t)} \quad (5.24)$$

where ϕ_k are the shape functions for the interpolation of \mathbf{E} and $M_{km} = (\phi_k, \phi_m)$ with the summation convention for the indices k and m running over all nodes in the mesh. Note that if we use discontinuous shape functions for the interpolation of \mathbf{E} , M_{km} can be computed on element level of elements intersected by the particle surface ∂P_i , which will be explained again in the next section 5.3.2. The method is equivalent to adding a positive definite stabilizing term in the momentum balance. Since the mass matrix M_{km} is positive definite, the matrix multiplication with M_{km}^{-1} can easily be computed with a Cholesky decomposition. Contrary to Lagrangian multipliers with additional unknowns $(\mathbf{U}_i, \boldsymbol{\omega}_i)$ [130], the diagonals of $(\mathbf{U}_i, \boldsymbol{\omega}_i)$ are filled with positive values, which helps the use of iterative solvers. The detailed derivation of the primal form is quite similar to the weak Dirichlet boundary condition and can be found in [139].

For moving boundary problems, if the mesh moves independently from the material motion, it is usually called an arbitrary Lagrangian-Eulerian (ALE) formulation [61]. The ALE formulation helps to minimize mesh distortion while still being able to track material boundaries. The relationship between time derivatives of a quantity f in material and spatial domains is given by

$$\dot{f} = \frac{\delta f}{\delta t} + (\mathbf{u} - \mathbf{u}_m) \cdot \nabla f, \quad (5.25)$$

where \dot{f} is the material derivative, $\delta f/\delta t$ is the grid time derivative and \mathbf{u}_m is the mesh velocity. We incorporate an ALE scheme to cope with the movement of the particles, which will be explained in Sec. 5.3.3.

We decouple the momentum and mass balance equations from the constitutive equation as proposed by D'Avino and Hulsen [141], in which the stress tensor $\boldsymbol{\tau}$ is replaced by a time-discretized but space-continuous form of the constitutive equation:

$$\begin{aligned} \boldsymbol{\tau}(\mathbf{c}^{n+1}) = G\Delta t & \left(-(\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{c}^n + (\nabla \mathbf{u}^{n+1})^T \cdot \mathbf{c}^n + \mathbf{c}^n \cdot \nabla \mathbf{u}^{n+1} \right) \\ & + G\mathbf{h}(\mathbf{c}^n, \Delta t) - G\mathbf{I}, \end{aligned} \quad (5.26)$$

where Δt is the computational time step and

$$\mathbf{h}(\mathbf{c}^n, \Delta t) = \mathbf{c}^n - \frac{\Delta t}{\lambda} (\mathbf{c}^n - \mathbf{I} + \alpha(\mathbf{c}^n - \mathbf{I})^2) \quad (5.27)$$

for the Giesekus model. We use L^2 -projection of $\mathbf{c}^n = \exp(\mathbf{s}^n)$ in Eq. (5.26) for a better numerical stability:

$$(\mathbf{e}, \mathbf{c}^n) = (\mathbf{e}, \exp(\mathbf{s}^n)), \quad (5.28)$$

where \mathbf{e} is a test function. By substituting Eq. (5.26) into Eq. (5.17), we get a Stokes-like problem for $(\mathbf{u}^{n+1}, p^{n+1}, \mathbf{G}^{n+1})$ depending on \mathbf{c}^n . Then the value of \mathbf{c}^{n+1} can be found by solving the constitutive equation with known values of $(\mathbf{u}^{n+1}, p^{n+1}, \mathbf{G}^{n+1})$.

For the time discretization of the constitutive equation, we use a second-order time-integration scheme, based on a Gear scheme, proposed by D'Avino and Hulsen [141]. We incorporate the log-conformation representation (LCR) [96] and the SUPG stabilization technique [102], as well:

$$\begin{aligned} (\mathbf{d} + \tau(\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{d}, \frac{\frac{3}{2}\mathbf{s}^{n+1} - 2\mathbf{s}^n + \frac{1}{2}\mathbf{s}^{n-1}}{\Delta t} \\ + (\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{s}^{n+1} - \mathbf{g}(\mathbf{G}^{n+1}, \hat{\mathbf{s}}^{n+1})) = 0, \end{aligned} \quad (5.29)$$

where $\mathbf{s} = \log \mathbf{c}$, $\hat{\mathbf{s}}^{n+1} = 2\mathbf{s}^n - \mathbf{s}^{n-1}$ and \mathbf{d} is a test function for the log-conformation \mathbf{s} . We follow the SUPG parameter τ given by Choi et al. [130].

5.3.2 A brief description of XFEM

The authors already introduced XFEM for the problem of moving particles [130], and with the presence of a free surface [137] in viscoelastic fluids. Here we briefly review XFEM applied to moving particle problems. The fluid domain $\Omega(t)$ is fully included in a larger mesh region Ω_m which is independent in time. The discretization of the velocity, pressure, log-conformation and velocity gradient projections is given by:[†]

$$\mathbf{u}_h(\mathbf{x}) = \sum_k \varphi_k(\mathbf{x}) \mathbf{u}_k, \quad \mathbf{x} \in \Omega(t), \quad (5.30)$$

$$p_h(\mathbf{x}) = \sum_k \psi_k(\mathbf{x}) p_k, \quad \mathbf{x} \in \Omega(t), \quad (5.31)$$

$$\mathbf{s}_h(\mathbf{x}) = \sum_k \psi_k(\mathbf{x}) \mathbf{s}_k, \quad \mathbf{x} \in \Omega(t), \quad (5.32)$$

$$\mathbf{G}_h(\mathbf{x}) = \sum_k \psi_k(\mathbf{x}) \mathbf{G}_k, \quad \mathbf{x} \in \Omega(t), \quad (5.33)$$

$$\mathbf{E}_h(\mathbf{x}) = \sum_k \phi_k(\mathbf{x}) \mathbf{E}_k, \quad \mathbf{x} \in \Omega(t). \quad (5.34)$$

We use a bi-quadratic Q_2 interpolation (φ_k) for the velocity \mathbf{u} ; and a bi-linear Q_1 interpolation (ψ_k) for the pressure p , log-conformation tensor \mathbf{s} , velocity gradient projection \mathbf{G} , and the projected conformation \mathbf{c} in Eq. (5.28). The shape function ϕ_k is taken Q_2 interpolation on element level and is discontinuous across element boundaries, whereas all others are continuous. Due to the discontinuous shape functions ϕ_k , the symmetric tensor \mathbf{E} can be computed on element level of elements intersected by the particle surface ∂P_i , and is eliminated from the global system vector. Equivalently, M_{km}^{-1} can be computed on element level of intersected

[†] In this chapter, we have dropped the Heaviside function in the approximation of the field variables, contrary to the previous chapters, for a simpler presentation. Note that the field variables are non-existent outside of the fluid domain. For two-sided discontinuities, the Heaviside function is necessary in the approximation, as introduced in Chapter 2. However, we will not deal with two-sided discontinuity problems in this thesis.

Table 5.1: Subdomain integration rules.

	quadrilateral subdomains	triangular subdomains
$A_{\text{int}}/A_{\text{elem}} > 10^{-2}$	3×3	6-point
$10^{-6} < A_{\text{int}}/A_{\text{elem}} \leq 10^{-2}$	3×3	16-point
$A_{\text{int}}/A_{\text{elem}} \leq 10^{-6}$	The element is discarded.	

elements if the primal form (Eq. (5.21)) is used. Hence, in practice, the computation of M_{km}^{-1} with a Cholesky decomposition is not costly.

Note that the shape functions are only defined in the nodes inside the fluid domain Ω and associated to the elements intersected by the interface $\partial P_i(t)$. However, they are not evaluated outside the domain Ω for intersected elements. Since the weak form is defined on Ω only, the integration should be performed only on part of an element for an intersected element. For the integration on the part of an element, we use a quadtree subdivision and further triangulation of the smallest quads near the interface, similar to [130, 137, 139], using a signed-distance function as a levelset function to find the surface of particles. In simulations, we use five level subdivisions; a 3×3 Gauss integration rule for undivided elements and quadrilateral subdomains, and a 6-point (exact up to 4th-order polynomials) or 16-point (exact up to 8th-order polynomials) Gauss integration rule [142] for triangular subdomains depending on the ratio of the integration area (A_{int}) over the element area (A_{elem}). The integration rules used are summarized in Table 5.1. If the integration area of an element is very small compared to the element area ($A_{\text{int}}/A_{\text{elem}} \leq 10^{-6}$), the element is discarded, i.e. the element is treated as if it is fully outside of the fluid domain Ω .

5.3.3 Temporary ALE scheme for moving particle problems

The current authors proposed a temporary ALE scheme to a moving free surface problem of extrudate swell [137]. Here we apply the same idea to a problem of moving particles. Let's suppose that a particle moves from $\hat{\mathbf{X}}_i^n$ at the previous time level t^n to $\hat{\mathbf{X}}_i^{n+1}$ at the current time level t^{n+1} by assuming that $\hat{\mathbf{X}}_i$ denotes the known position of the particle. Field variables at the previous time level, such as \mathbf{u}^n and \mathbf{s}^n , can become undefined near the boundary of the particle since there was no fluid flow at time level t^n . To handle this problem, we incorporate a temporary ALE scheme which maps field variables at the previous time level onto the current time level [130, 137].

We solve a mesh displacement $\mathbf{d}_m(\mathbf{x}, t^n)$ such that mesh nodes near a particle follow the motion of the particle, whereas, mesh nodes near the walls are stationary:

$$\nabla^2 \mathbf{d}_m = \mathbf{0} \quad \text{in } \Omega_m, \quad (5.35)$$

$$\mathbf{d}_m = \mathbf{0} \quad \text{on } \Gamma_1 \text{ and } \Gamma_3, \quad (5.36)$$

$$(\mathbf{d}_m)_{\Gamma_2} = (\mathbf{d}_m)_{\Gamma_4}, \quad (5.37)$$

$$\mathbf{d}_m = \hat{\mathbf{X}}_i^{n+1} - \hat{\mathbf{X}}_i^n \quad \text{on } \partial P_i(t^n). \quad (5.38)$$

Note that above equations are solved on the whole computational domain Ω_m , using a similar technique to the fictitious domain method [64], and Eq. (5.38) can be realized either by using a constraint implemented with Lagrangian multipliers or by using a weak Dirichlet boundary condition [139]. We have also tested solving Eq. (5.35) only on the fluid domain Ω using XFEM accompanied with a weak Dirichlet boundary condition [139] for Eq. (5.38). We do not recognize any significant difference between these methods in view of accuracy and stability of the flow problem. However, the weak Dirichlet boundary condition can be preferable for large three-dimensional simulations since it circumvents the LBB condition associated with the Lagrangian multiplier technique, and seems to be well suited for fast iterative solvers [139].

Then, the ALE mesh $\mathbf{x}_{\text{ALE}}^{n+1}$ is constructed by the following mesh advection equations:

$$\mathbf{x}_{\text{ALE}}^{n+1} = \mathbf{x}_m + \mathbf{d}_m(\mathbf{x}_m, t^n) \quad (5.39)$$

Note that the mesh displacement field $\mathbf{d}_m(\mathbf{x}_m, t^n)$ is defined on the Eulerian mesh \mathbf{x}_m at time level t^n with respect to the particle position $\hat{\mathbf{X}}_i^n$. In the ALE mesh, nodes near the particle follow the motion of the particle from $\hat{\mathbf{X}}_i^n$ to $\hat{\mathbf{X}}_i^{n+1}$ by the imposition of Eq. (5.38). Note that the particle moves on a fixed Eulerian mesh \mathbf{x}_m . Similarly, the ALE mesh $\mathbf{x}_{\text{ALE}}^n$ is constructed by:

$$\mathbf{x}_{\text{ALE}}^n = \mathbf{x}_m + \mathbf{d}_m(\mathbf{x}_m, t^{n-1}) \quad (5.40)$$

with the displacement field at $t = t^{n-1}$. The construction of the ALE meshes $\mathbf{x}_{\text{ALE}}^n$ and $\mathbf{x}_{\text{ALE}}^{n+1}$ is shown in Fig. 5.2. By using an ALE technique and a semi-implicit Gear scheme, the time discretization of the log-conformation equation (Eq. (5.7)) can be written as:

$$\frac{3}{2} \frac{\mathbf{s}^{n+1}}{\Delta t} + (\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{s}^{n+1} = \frac{2\mathbf{s}^n - \frac{1}{2}\mathbf{s}^{n-1}}{\Delta t} + 2\mathbf{g}((\nabla \mathbf{u}^{n+1})^T, \mathbf{s}^n) - \mathbf{g}((\nabla \mathbf{u}^{n+1})^T, \mathbf{s}^{n-1}). \quad (5.41)$$

In Eq. (5.41), the unknown at the current time level, \mathbf{s}^{n+1} , is computed on the Eulerian mesh \mathbf{x}_m , whereas, field variables at previous time levels, such as \mathbf{s}^n and \mathbf{s}^{n-1} , are mapped along with the ALE meshes (Fig. 5.3):

$$\mathbf{s}^n = \mathbf{s}(\Phi_n^{-1}(\mathbf{x}), t^n), \quad (5.42)$$

$$\mathbf{s}^{n-1} = \mathbf{s}(\Phi_{n-1}^{-1} \circ \Phi_n^{-1}(\mathbf{x}), t^{n-1}). \quad (5.43)$$

Hence, the inverse mapping Φ_n^{-1} can be easily implemented in the finite element context: find the reference coordinates of \mathbf{x} in the ALE mesh $\mathbf{x}_{\text{ALE}}^{n+1}$, then interpolate the values of \mathbf{s} at the same reference coordinates in the previous computational mesh \mathbf{x}_m . The inverse mapping Φ_{n-1}^{-1} can be done by repeating the procedure once more (see Fig. 5.2). The mesh velocity at the current time level is defined by a second-order backwards differencing (BDF2):

$$\mathbf{u}_m^{n+1}(\mathbf{x}) = \frac{\frac{3}{2}\mathbf{x} - 2\mathbf{x}^n + \frac{1}{2}\mathbf{x}^{n-1}}{\Delta t} \quad (5.44)$$

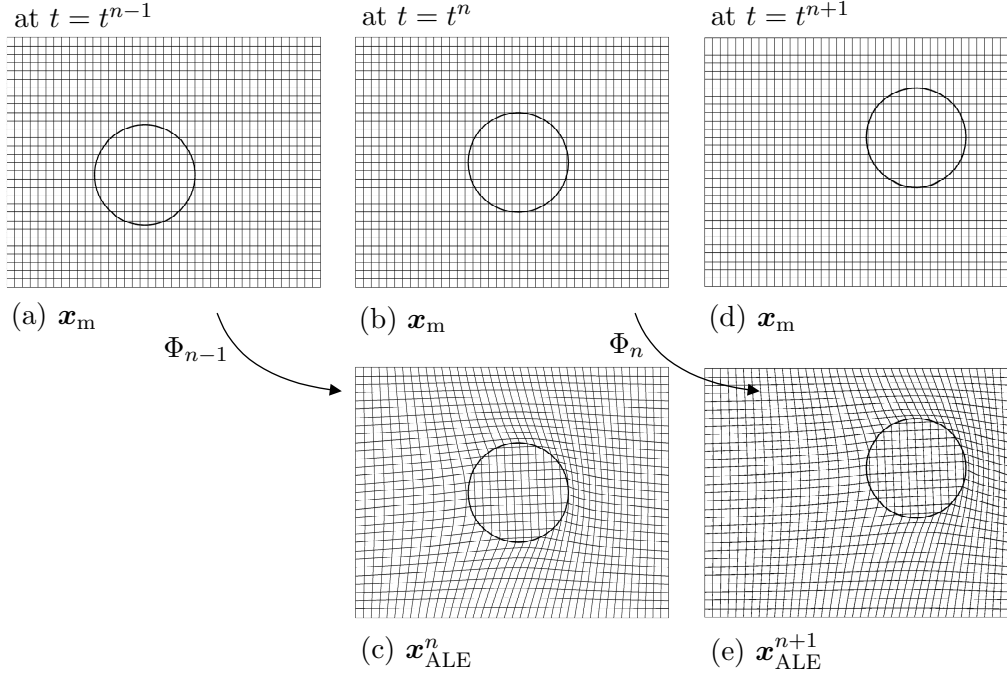
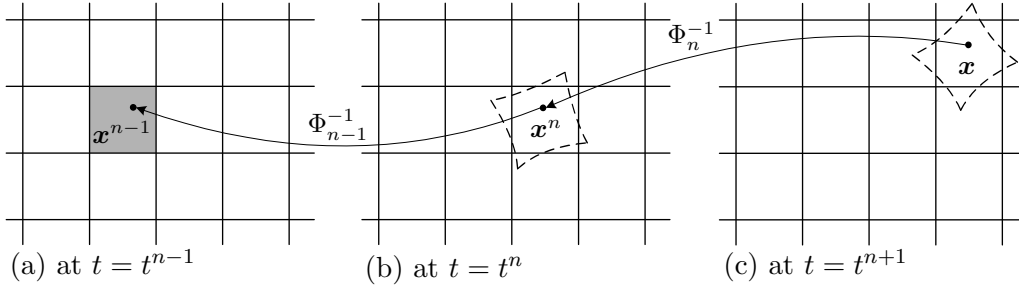


Figure 5.2: Second-order temporary ALE scheme.

Figure 5.3: Mapping using a second-order temporary ALE scheme. The dashed element at $t = t^n$ and $t = t^{n+1}$ represents the element in the ALE meshes \mathbf{x}_{ALE}^n and \mathbf{x}_{ALE}^{n+1} , respectively.

at each coordinate \mathbf{x} . At the next time level $t = t^{n+2}$, we delete \mathbf{x}_{ALE}^n , and construct a new temporary ALE mesh \mathbf{x}_{ALE}^{n+2} . Note that \mathbf{x}_{ALE}^{n+1} is kept unchanged.

It is worthy to compare differences in the temporary ALE schemes between [130] and the current work. In [130], the ALE mesh \mathbf{x}_{ALE}^{n+1} is constructed from \mathbf{x}_{ALE}^n , not from \mathbf{x}_m as in the current work. Hence, at the next time step $t = t^{n+2}$, both ALE meshes \mathbf{x}_{ALE}^{n+1} and \mathbf{x}_{ALE}^{n+2} were (re)-constructed in [130]. In this work, we need to construct \mathbf{x}_{ALE}^{n+2} only, simply keeping \mathbf{x}_{ALE}^{n+1} . The difference in the construction of ALE meshes also affects the order of finding reference coordinates. Let's consider the coordinates \mathbf{x} at $t = t^{n+1}$ in Fig. 5.3 (c). First we find the reference coordinates of \mathbf{x} in the ALE mesh \mathbf{x}_{ALE}^{n+1} for the inverse mapping Φ_n^{-1} . In this work, the Φ_n^{-1} maps the reference coordinates of \mathbf{x} in \mathbf{x}_{ALE}^{n+1} to \mathbf{x}^n on the background Eulerian mesh \mathbf{x}_m . Then we find the reference coordinates of \mathbf{x}^n in the ALE mesh \mathbf{x}_{ALE}^n , which is represented by dashed lines in Fig. 5.3 (b). The inverse mapping Φ_{n-1}^{-1} is performed in the same way.

By contrast, in [130], the Φ_n^{-1} is equivalent to following the same reference coordinates of the dashed element in Fig. 5.3 (b), then we interpolate the physical values defined in the background Eulerian mesh at the coordinates \boldsymbol{x}^n . Note also that, in the current work, the ALE meshes are constructed with respect to the position of particles, contrary to the velocity of particles in [130]. Since we use a predictor-corrector method (see Sec. 5.3.4) using the position as a primary variable in Eq. (5.45), it is a natural choice to construct an ALE mesh using a displacement field with respect to the given positions. The new temporary ALE scheme can be easily applied to the problem in [130] as well.

Since the temporary ALE meshes are independent from the current computational mesh, we can freely apply local mesh refinements on the current computational mesh without any need of further considerations on the ALE meshes, as already pointed out in [130]. In this chapter, for the local mesh refinements along the interface, we use a grid deformation method [117], in which mesh nodes are redistributed close to the interface while preserving the mesh topology.

5.3.4 Time integration

We use a semi-implicit stress formulation for the time integration of the evolution equation of the log-conformation tensor \boldsymbol{s} (Eq. (5.29)) decoupled from the momentum balance [141]. Initially, the viscoelastic polymer stress is set to zero over the whole fluid domain. We apply the following procedure at every time step.

Step 1. Predict the particle configuration using a linear extrapolation:

$$\hat{\boldsymbol{X}}_i^{n+1} = 2\boldsymbol{X}_i^n - \boldsymbol{X}_i^{n-1}, \quad (5.45)$$

$$\hat{\boldsymbol{\Theta}}_i^{n+1} = 2\boldsymbol{\Theta}_i^n - \boldsymbol{\Theta}_i^{n-1}. \quad (5.46)$$

For circular particles, the update of angular rotations is not necessary. At the first time step, we use

$$\hat{\boldsymbol{X}}_i^{n+1} = \boldsymbol{X}_i^n. \quad (5.47)$$

Step 2. Construct a temporary ALE mesh using Eqs. (5.39)-(5.40) for the interpolation of field variables at previous time levels. Note that the ALE meshes are constructed regarding predicted particle positions, such as $\hat{\boldsymbol{X}}_i^{n-1}$, $\hat{\boldsymbol{X}}_i^n$ and $\hat{\boldsymbol{X}}_i^{n+1}$, since field variables are calculated on these particle configurations. At the first time step, we use a first-order temporary ALE scheme, similar to [137].

Step 3. Compute a deformed computational mesh for local mesh refinements, as in [130]. A deformed computational mesh is derived from an initial regular mesh, in accordance with the predicted positions of the particles $\hat{\boldsymbol{X}}_i^{n+1}$.

Step 4. Compute \boldsymbol{u}^{n+1} , p^{n+1} , \boldsymbol{G}^{n+1} , \boldsymbol{U}_i^{n+1} and $\boldsymbol{\omega}_i^{n+1}$ from the momentum balance, continuity equation and gradient projection equation:

$$\begin{aligned}
& - (\nabla \cdot \mathbf{v}, p^{n+1}) + ((\nabla \mathbf{v})^T, 2\eta_s \mathbf{D}(\mathbf{u}^{n+1}) + \theta(\nabla \mathbf{u}^{n+1} - (\mathbf{G}^T)^{n+1}) + \boldsymbol{\tau}^{n+1}) \\
& - (\mathbf{v} - (\mathbf{V}_i + \boldsymbol{\chi}_i \times (\mathbf{x} - \mathbf{X}_i^{n+1})), (-p^{n+1} \mathbf{I} + 2\eta_s \mathbf{D}(\mathbf{u}^{n+1}) + \boldsymbol{\tau}^{n+1}) \cdot \mathbf{n})_{\partial P_i(t^{n+1})} \\
& + \kappa \sum_{i=1}^N L_i (\mathbf{v} - (\mathbf{V}_i + \boldsymbol{\chi}_i \times (\mathbf{x} - \mathbf{X}_i^{n+1})), \mathbf{u}^{n+1} - (\mathbf{U}_i^{n+1} + \boldsymbol{\omega}_i^{n+1} \times (\mathbf{x} - \mathbf{X}_i^{n+1}))) = 0,
\end{aligned} \tag{5.48}$$

$$(q, \nabla \cdot \mathbf{u}^{n+1}) = 0, \tag{5.49}$$

$$(\mathbf{H}, -\nabla \mathbf{u}^{n+1} + (\mathbf{G}^T)^{n+1}) = 0, \tag{5.50}$$

where

$$\begin{aligned}
\boldsymbol{\tau}^{n+1} = G\Delta t & (- (\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{c}^n + (\nabla \mathbf{u}^{n+1})^T \cdot \mathbf{c}^n + \mathbf{c}^n \cdot \nabla \mathbf{u}^{n+1}) \\
& + G\mathbf{h}(\mathbf{c}^n, \Delta t) - G\mathbf{I}. \tag{5.51}
\end{aligned}$$

Note that we use L^2 -projection of $\mathbf{c}^n = \exp(\mathbf{s}^n)$ given by Eq. (5.28), which is solved on the fluid domain Ω by using XFEM.

Step 5. Solve the log-conformation tensor \mathbf{s}^{n+1} by replacing the evolution equation of the log-conformation tensor (Eq. (5.7)) using a second-order semi-implicit Gear scheme:

$$\begin{aligned}
(\mathbf{d} + \tau(\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{d}, \frac{3}{2} \frac{\mathbf{s}^{n+1}}{\Delta t} + (\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{s}^{n+1}) = \\
(\mathbf{d} + \tau(\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{d}, \frac{2\mathbf{s}^n - \frac{1}{2}\mathbf{s}^{n-1}}{\Delta t} + 2\mathbf{g}(\mathbf{G}^{n+1}, \mathbf{s}^n) - \mathbf{g}(\mathbf{G}^{n+1}, \mathbf{s}^{n-1})). \tag{5.52}
\end{aligned}$$

At the first time step, we use a first-order Euler scheme:

$$\begin{aligned}
(\mathbf{d} + \tau(\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{d}, \frac{\mathbf{s}^{n+1}}{\Delta t} + (\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{s}^{n+1}) = \\
(\mathbf{d} + \tau(\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{d}, \frac{\mathbf{s}^n}{\Delta t} + \mathbf{g}(\mathbf{G}^{n+1}, \mathbf{s}^n)). \tag{5.53}
\end{aligned}$$

Step 6. Correct the particle configuration using a trapezoidal rule:

$$\mathbf{X}_i^{n+1} = \mathbf{X}_i^n + \frac{1}{2}(\mathbf{U}_i^{n+1} + \mathbf{U}_i^n)\Delta t, \tag{5.54}$$

$$\boldsymbol{\Theta}_i^{n+1} = \boldsymbol{\Theta}_i^n + \frac{1}{2}(\boldsymbol{\omega}_i^{n+1} + \boldsymbol{\omega}_i^n)\Delta t. \tag{5.55}$$

For circular particles, the update of angular rotations is not necessary. At the first time step, we use

$$\mathbf{X}_i^{n+1} = \mathbf{X}_i^n + \mathbf{U}_i^{n+1}\Delta t. \tag{5.56}$$

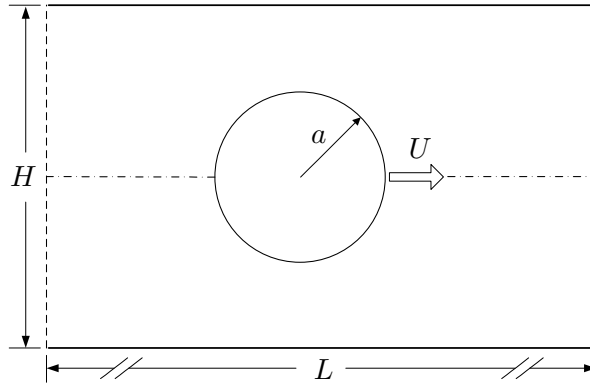


Figure 5.4: Geometry for a translating cylinder in a channel.

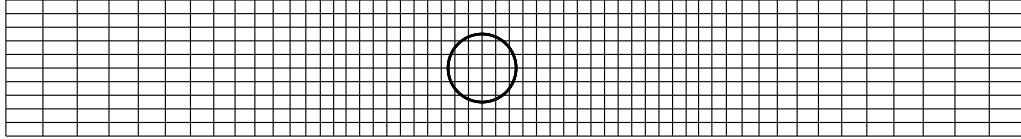


Figure 5.5: Base mesh for a translating cylinder in a channel. The actual mesh used in simulations is about 7 times finer in each direction than the mesh shown here.

5.4 Convergence analysis of the temporary ALE scheme

In order to validate the second-order temporary ALE scheme introduced in Sec. 5.3.3, we devise a test problem of a translating cylinder in a long straight channel (see Fig. 5.4): the length of the channel $L = 30$, the height of the channel $H = 4$, the radius of the cylinder $a = 1$, and the translational speed of the cylinder $U = 1$. We assume no-slip boundary conditions on the cylinder and on the channel walls, and also assume the flow to be periodic in x -direction. We use the Giesekus model with a material parameter set: $\eta_s = 0$, $\eta_p = 1$, $\lambda = 0.5$ and $\alpha = 0.1$. The base mesh with 550 elements is shown in Fig. 5.5 for a better understanding. In the region $-5 \leq x \leq 5$, with the origin at the channel center, we use regular square elements, and the element size increases gradually to the channel exits. In Fig. 5.5, the center coordinate of the cylinder $\mathbf{X} = (-1, 0)$. Note that the cylinder moves only within the regularly refined center region. In actual simulations, we use a mesh with 26250 elements, which is about 7 times finer in each direction than the base mesh shown in Fig. 5.5.

We monitor the value of $K = \int_{P(t)} (\boldsymbol{\tau} \cdot \mathbf{n}) \cdot \mathbf{e}_x ds$ on the cylinder surface $P(t)$ for $0 \leq t \leq \lambda$. Note that K represents the drag component on the cylinder caused by the polymer stress $\boldsymbol{\tau}$. The relative L^2 -error is defined as

$$\epsilon_{L^2} = \frac{\|K_h - \bar{K}\|_{L^2}}{\|\bar{K}\|_{L^2}} = \frac{(\int_0^\lambda |K_h - \bar{K}|^2 dt)^{1/2}}{(\int_0^\lambda \bar{K}^2 dt)^{1/2}}, \quad (5.57)$$

where the subscript ‘ h ’ indicates the discretized field variables. For the reference value \bar{K} , we

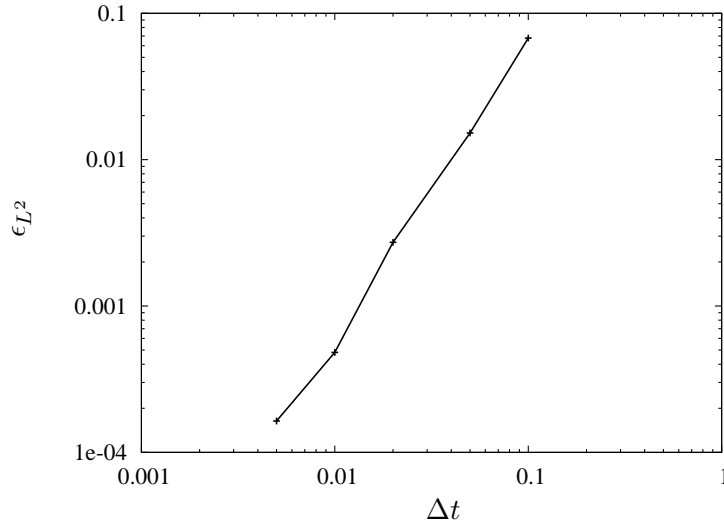


Figure 5.6: Convergence of K in relative L^2 -norm with respect to Δt .

compute the problem with $\Delta t = 10^{-4}$. The relative L^2 -errors with various Δt 's are shown in Fig. 5.6, which shows that the convergence rate is second-order with respect to Δt .

Note that we have also chosen the reference value \bar{K} with $\Delta t = 2 \times 10^{-4}$ and 5×10^{-5} , and it does not affect the relative L^2 -errors in Fig. 5.6 significantly. Hence, $\Delta t = 10^{-4}$ is small enough as a reference for the convergence check in the range of Δt 's used in Fig. 5.6.

5.5 Validation

We will validate the proposed method by comparing with the results in [39] and [130]. The problem is the particle migration in shear flow confined between two walls, which was originally solved by D'Avino et al. [39] by using boundary-fitted meshes and a conventional ALE scheme. The dimensionless parameters governing the problem are the Weissenberg number $Wi = \lambda\dot{\gamma} = \lambda U_w/H$, the mobility parameter α of the Giesekus model, the viscosity ratio η_s/η_p and the confinement ratio of a particle $\epsilon = 2a/H$, where a is the radius of the particle and H is the channel height. In this problem, we choose $Wi = 1$, $\alpha = 0.2$, $\eta_s/\eta_p = 0.1$ and $\epsilon = 0.1$. We will investigate the migrational velocity V and angular velocity ω of the particle for different initial particle positions Y_0 .

In these simulations, we use a grid deformation method for the local mesh refinements around the particle surface as described in [130]. Our mesh has a similar resolution near the particle surface as the mesh M4 in [39]. Fig. 5.7 and Fig. 5.8 show the non-dimensional migrational velocity $V^* = V/H\dot{\gamma}$ and angular velocity $\omega^* = \omega/\dot{\gamma}$, respectively, as a function of non-dimensional time $t^* = t\dot{\gamma}$ for different initial positions $Y_0^* = Y_0/H = \{0.1, 0.2, 0.3, 0.35, 0.4\}$. Note that we define a clockwise rotation as positive in Fig. 5.8. The results obtained by using the proposed methods (solid red lines) are compared with those of [130] (dotted blue lines), where particle motions are realized by using constraints implemented with Lagrangian mul-

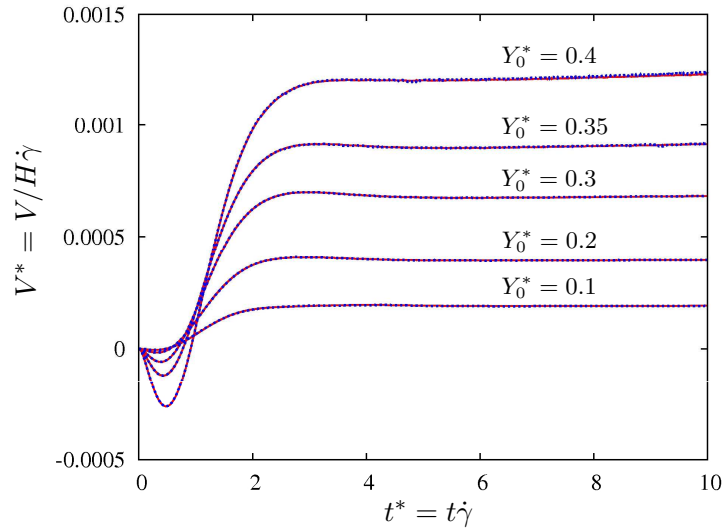


Figure 5.7: Migration velocity of a particle as a function of time for different initial positions.

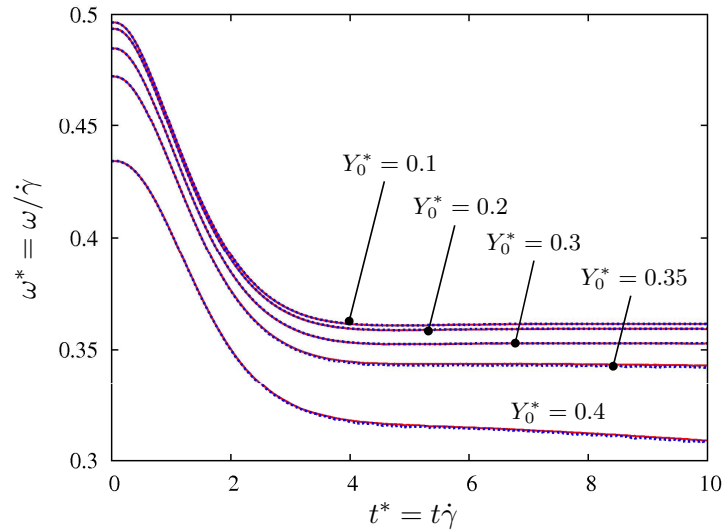


Figure 5.8: Angular velocity of a particle as a function of time for different initial positions.

multipliers and explicit polymer stress formulations are used in the momentum balance. Both results show good agreement, which validates the proposed temporary ALE scheme combined with the semi-implicit polymer stress formulation and weak boundary conditions on the particle surface.

5.6 Interaction of two particles

5.6.1 Problem description

We consider interactions of two particles in shear flow confined between two walls. The problem is schematically shown in Fig. 5.9. In a long straight channel, two particles are freely

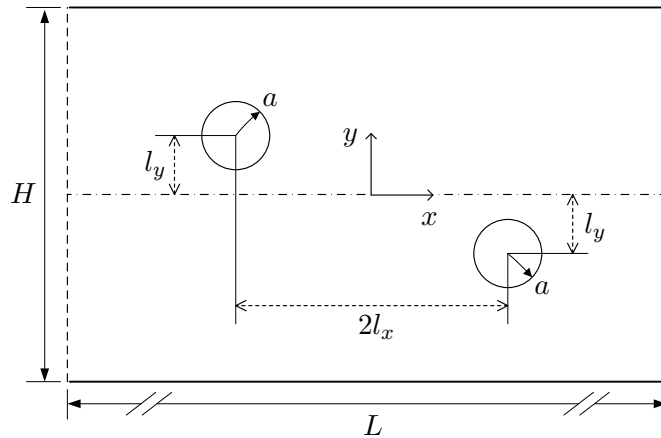


Figure 5.9: Geometry for two-particle interactions in confined shear flow.

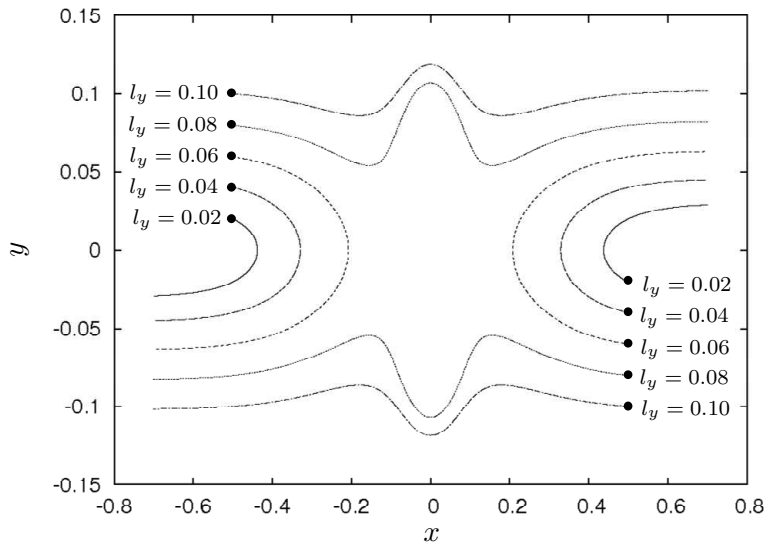


Figure 5.10: Trajectories of two particles in Stokes flow for varying l_y .

suspended, and their initial positions are $(-l_x, l_y)$ and $(l_x, -l_y)$. The top and bottom walls move at constant velocities $\frac{1}{2}U_w$ and $-\frac{1}{2}U_w$, respectively. We fix $H = 1$, $L = 4$, $a = 0.1$ and $l_x = 0.5$. We assume the flow to be periodic in x -direction. We change l_y and investigate the corresponding motion of the particles.

5.6.2 Particle motion in Stokes flow

First, we will show the particle motion in Stokes flow. Fig. 5.10 shows the trajectories of two particles for varying l_y . If initial separations of particles are small, such as $l_y = 0.02, 0.04$, and 0.06 , two particles repel each other and the trajectories are reversed. The motion of the particles and the corresponding meshes are shown in Fig. 5.11 for the case of $l_y = 0.06$. We can clearly see the U-turns of the two particles. To distinguish the two

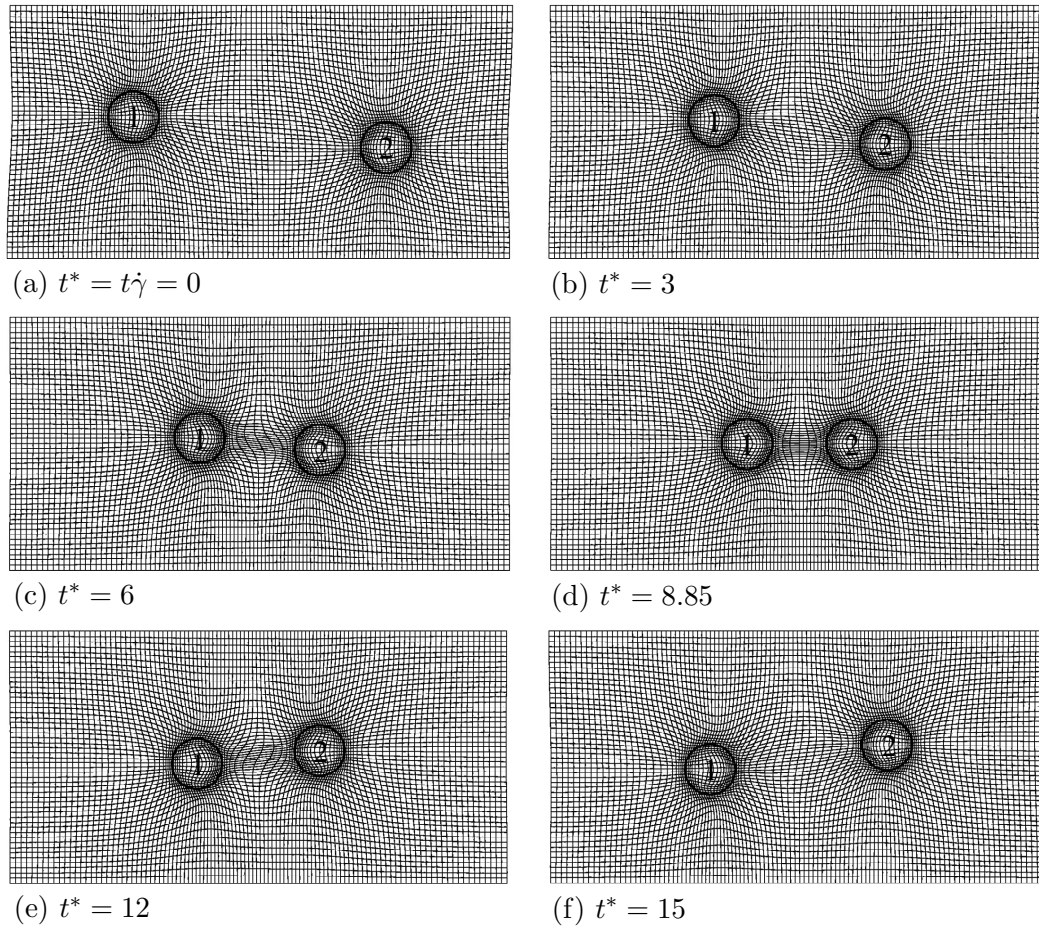


Figure 5.11: Snapshots showing the motion of the particles and the corresponding meshes in Stokes flow for $l_y = 0.06$.

particles, the initial top-left particle is labeled as “1” and the initial bottom-right particle is labeled as “2”. We only show the region near the two particles, and the actual length of the channel used in the computation is longer than the one shown in Fig. 5.11. If initial separations of particles are large, such as $l_y = 0.08$ and 0.10 , two particles keep going in their original directions and pass each other as shown in Fig. 5.10. Fig. 5.12 shows the passing behavior of the two particles for the case of $l_y = 0.08$.

In comparison, Hwang et al. [73] simulated the motion of two particles in an unbounded sliding bi-periodic frame. In their simulations, the particles always pass each other regardless of initial separations in y -direction. Hence, the reversing particle trajectories are likely caused by additional effects due to the presence of confining walls.

5.6.3 Particle motion in a viscoelastic fluid

Now let’s consider the two-particle interactions in shear flow of a viscoelastic fluid. We use the Giesekus constitutive model with dimensionless parameters - the Weissenberg number

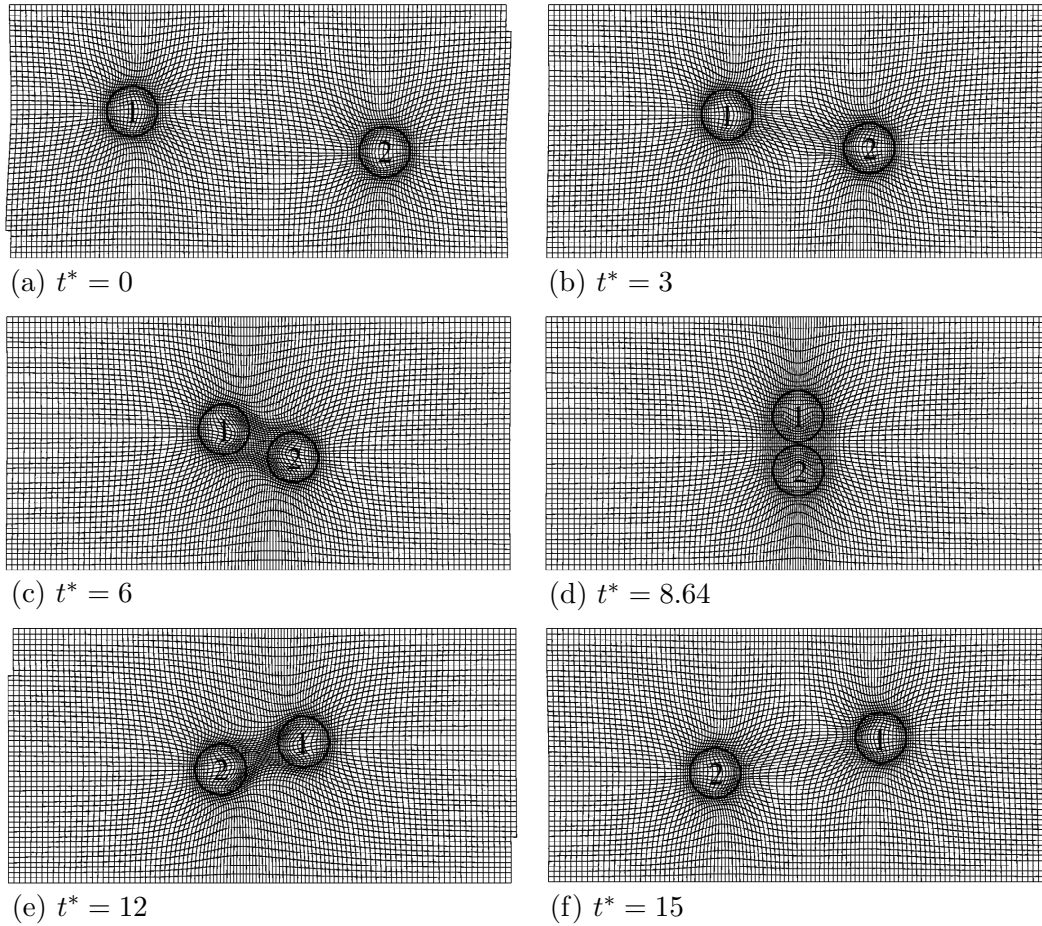


Figure 5.12: Snapshots showing the motion of the particles and the corresponding meshes in Stokes flow for $l_y = 0.08$.

$Wi = \lambda\dot{\gamma} = \lambda U_w/H = 1$, the viscosity ratio $\beta = \eta_s/(\eta_s + \eta_p) = 0.1$, and the mobility parameter $\alpha = 0.01$. Fig. 5.13 shows the trajectories of the two particles for varying l_y in a viscoelastic fluid. For small initial separations, such as $l_y = 0.02$ and 0.04 , two particles reverse their trajectories; for large initial separations $l_y = 0.10$, two particles pass each other, similarly to the Stokes flow problems in the previous section. For intermediate initial separations, such as $l_y = 0.06$ and 0.08 , we observe a third class of motion in which two particles come together, then they rotate at the fixed final positions. The final positions of the two particles are represented by \times 's in Fig. 5.13. This kind of interaction is shown in Fig. 5.14, and the translational and the rotational velocities of the initial top-left particle is shown in Fig. 5.15, for the case of $l_y = 0.08$. Due to the symmetry of the problem, the evolution of the angular velocity of the other particle is the same as the first one. The x -directional and the y -directional translational velocities U and V , respectively, eventually become zero, hence the particles move to final positions. Then the particles rotate with a non-zero terminal angular velocity. A more interesting observation is that the final particle positions are the same for the cases of $l_y = 0.06$ and $l_y = 0.08$ as shown in Fig. 5.13. Hence, their terminal angular

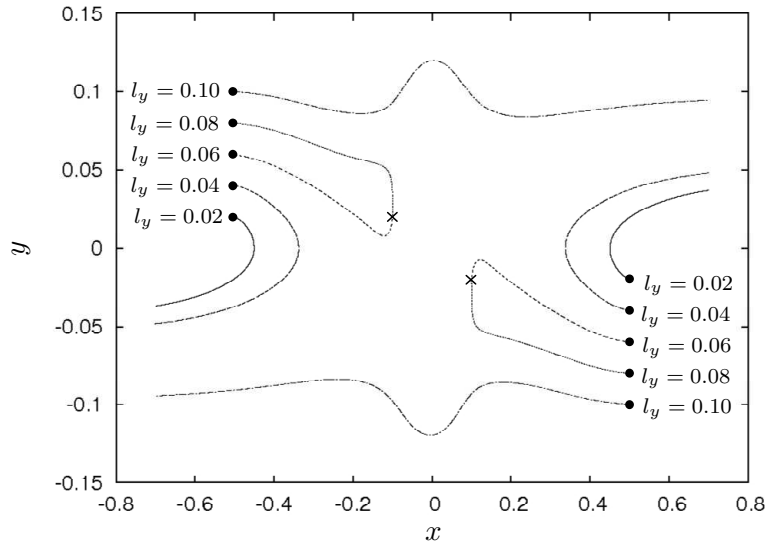


Figure 5.13: Trajectories of two particles in a viscoelastic fluid for varying l_y .

velocities are all the same even though the histories and the times required to reach the terminal angular velocity are different.

5.7 Interaction of multiple particles

The alignment of two particles in confined shear flow of a Giesekus fluid is presented in the previous section, which is based on [130], by using an explicit polymer stress formulation and a Lagrangian multiplier technique for the imposition of the no-slip boundary condition. Here we present alignment of multiple particles using the proposed weak boundary condition with a semi-implicit polymer stress formulation. We will start with three particles in confined shear flow. The geometry for three-particle interactions is shown in Fig. 5.16. The height of the channel is H , the length of the channel is L and the radius of particles is a . We only consider particles of the same size. The initial positions of three particles are $(-l_x, l_y)$, $(0, 0)$ and $(l_x, -l_y)$. Due to point symmetry about the origin, the center particle does not translate and only rotates at the initial position. We assume the flow to be periodic in x -direction. We use the Giesekus constitutive model for a viscoelastic fluid, which can be characterized with three dimensionless parameters - the Weissenberg number $Wi = \lambda\dot{\gamma} = \lambda U_w/H$, the mobility parameter α , and the viscosity ratio $\beta = \eta_s/\eta_0$ with the zero-shear-rate viscosity $\eta_0 = \eta_s + \eta_p$.

5.7.1 Convergence check

First we will present the results of mesh convergence. We use a domain with $H = 1$ and $L = 4$, and the radius of particles $a = 0.1$. The initial positions of three particles are $(-0.5, 0.1)$, $(0, 0)$ and $(0.5, -0.1)$, i.e. $l_x = 0.5$, $l_y = 0.1$. We use three meshes - M1 (200×50 elements), M2 (320×80 elements) and M3 (440×110 elements), with the grid deformation method [117] for mesh refinements around particle surface [130], similar to Figs. 5.11, 5.12

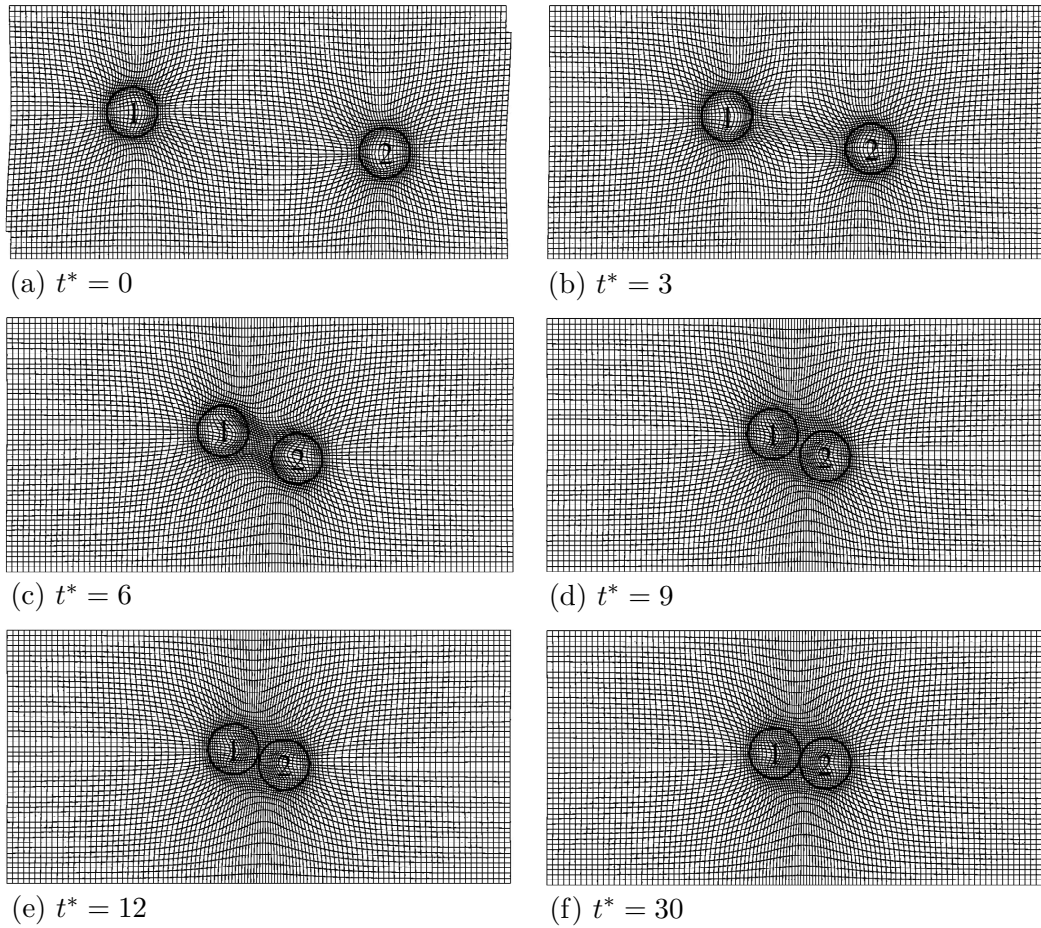


Figure 5.14: Snapshots showing the motion of the particles and the corresponding meshes in a viscoelastic fluid for $l_y = 0.08$.

and 5.14. Fig. 5.17 shows the angular velocity of the center particle $\omega^* = \omega/\dot{\gamma}$ as a function of time $t^* = t\dot{\gamma}$, where $\dot{\gamma} = U_w/H$ is the shear-rate. Note that we define a clockwise rotation as positive. The result of the mesh M1 shows slight wiggles in time and deviates a little from the results of M2 and M3, which are almost indistinguishable in Fig. 5.17. We also observed similar convergence behavior in the translational velocity, and for the other two particles as well. For all our simulations for $H = 1$, we use the mesh M3. For different channel heights, we keep the element size the same as M3, and increase or decrease the number of elements proportional to the channel height.

5.7.2 Alignment of three particles

We use the same domain as in the previous section, i.e. $H = 1$, $L = 4$ and $a = 0.1$. The alignment of two particles for the given geometry is already shown in Section 5.6 for $Wi = 1$, $\alpha = 0.01$, $\beta = 0.1$. Hence we begin with these parameters for three-particle problems. Fig. 5.18 shows the trajectories of the top-left particle for different initial positions. Black

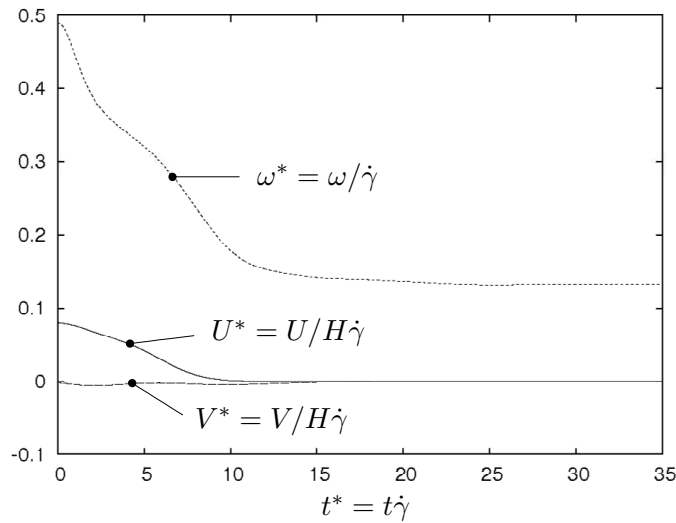


Figure 5.15: Velocities of the top-left particle in a viscoelastic fluid for $l_y = 0.08$.

dots represent initial positions and solid lines are trajectories. Note that the trajectories of the lower-right particle are point symmetric about the center of the channel, and the center particle only rotates at fixed position. As shown in Fig. 5.18, we never observed alignment of the three particles for the given fluid rheology, contrary to two-particle problems in Section 5.6. The motion of the particles and the trace of the polymer stress, $\text{tr}(\boldsymbol{\tau}) = \tau_{xx} + \tau_{yy}$, are shown in Fig. 5.19 for the case of $l_x = 0.5$, $l_y = 0.08$. Temporarily, the three particles form a string-like structure around $t^* = 10 \sim 20$, however eventually they repel under shear flow. An interesting observation is that once the particles repel, the trajectories are converging to a single line regardless of the initial positions (see Fig. 5.18). However we do not have an intuitive explanation for this observation yet. Note that, in Fig. 5.18, we only show the repelling trajectories whereas the particles can pass and keep moving in their original directions if the initial l_y is large, similar to two-particle problems shown in Section 5.6.

To make alignment or string formation of the three particles, we try to increase the Weissenberg number while keeping other parameters the same. Fig. 5.20 shows the aligning trajectories of the top-left particle for $\text{Wi} = 2$ with different initial positions. As two-particle problems in Section 5.6, the final position is independent of initial positions if the three particles form a string-like structure. The aligning motion of the three particles for the initial positions with $l_x = 0.5$, $l_y = 0.1$ is shown in Fig. 5.21. The translational and angular velocities of the top-left particle in time are shown in Fig. 5.22: the x - and y -directional translational velocities, $U^* = U/H\dot{\gamma}$ and $V^* = V/H\dot{\gamma}$ respectively, eventually become zero and the angular velocity $\omega^* = \omega/\dot{\gamma}$ reaches a non-zero terminal value. Note that the angular velocity of the center particle for the given condition is already shown in Fig. 5.17 for a mesh convergence check, which also verifies reaching a steady-state. At the steady-state, the gap between particles is about 15% of the particle radius for the specific problem. Note that we do not include any heuristic repulsive forces to avoid particle overlapping, neither attractive forces on the particles. The alignment of the particles is manifested by hydrodynamic interactions only.

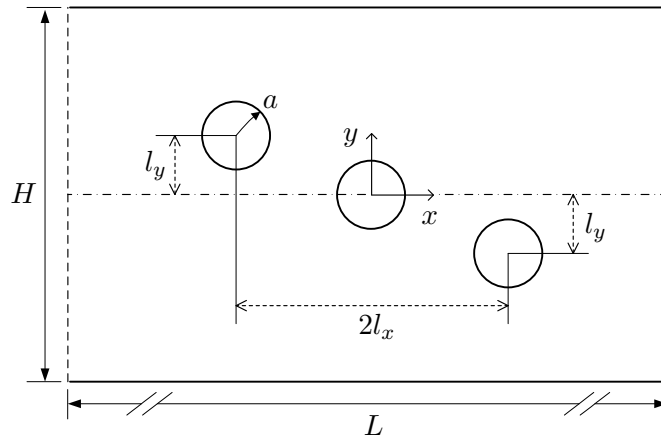


Figure 5.16: Geometry for three-particle interactions in confined shear flow.

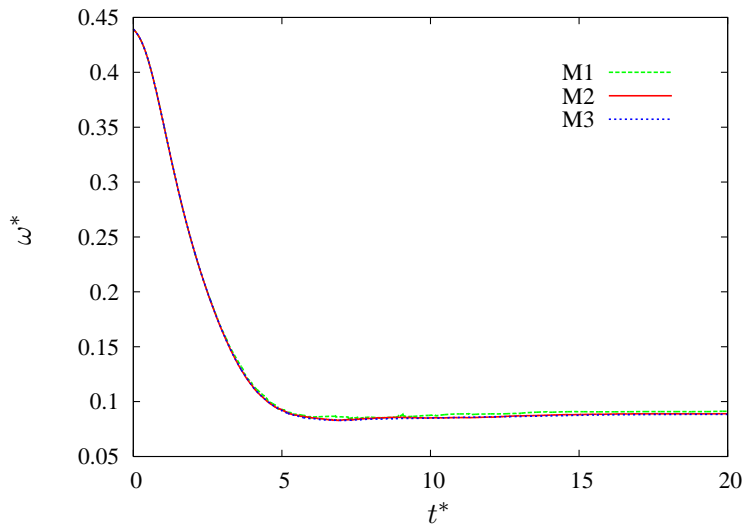


Figure 5.17: Angular velocity of the center particle as a function of time using different meshes.

In order to judge whether particles form a stable string-like structure, we monitor the angular velocities of the particles. If $|\omega^*(t^{n+1}) - \omega^*(t^n)| < 10^{-6}$ for every particle, we regard that the steady-state is obtained. Note that the angular velocity never reaches a steady-state value if particles repel, even though they look like a string temporarily as Fig. 5.19 around $t^* = 10 \sim 20$.

Michele et al. [51] experimentally observed that the angular velocities of spherical particles almost vanish if they form a string-like structure in a viscoelastic fluid. Hwang and Hulsén [56] also observed a significant reduction in average angular velocity of particles as they form a string by a direct numerical simulation. Motivated by these observations, we present the steady-state angular velocities of particles quantitatively for different lengths of strings, despite of two-dimensional simulations. Note that, if a rigid particle is sheared in an unconfined Stokes flow, the angular velocity of the particle is half of the shear-rate, i.e.

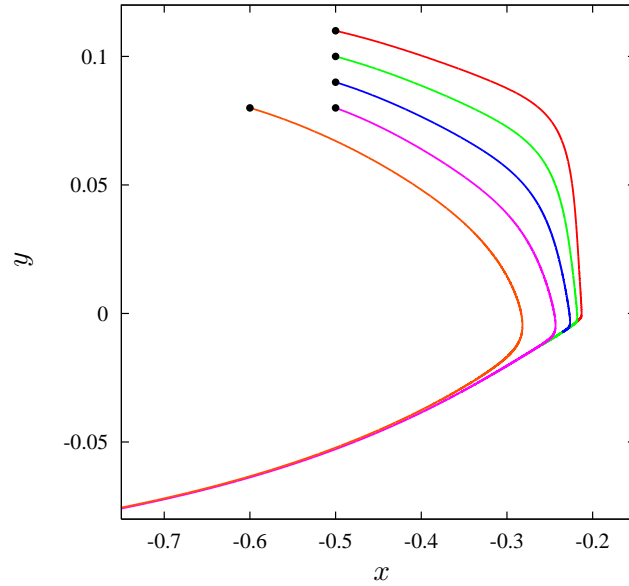


Figure 5.18: Repelling trajectories of the top-left particle for different initial positions for $Wi = 1, \alpha = 0.01, \beta = 0.1$. Black dots represent initial positions of the top-left particle.

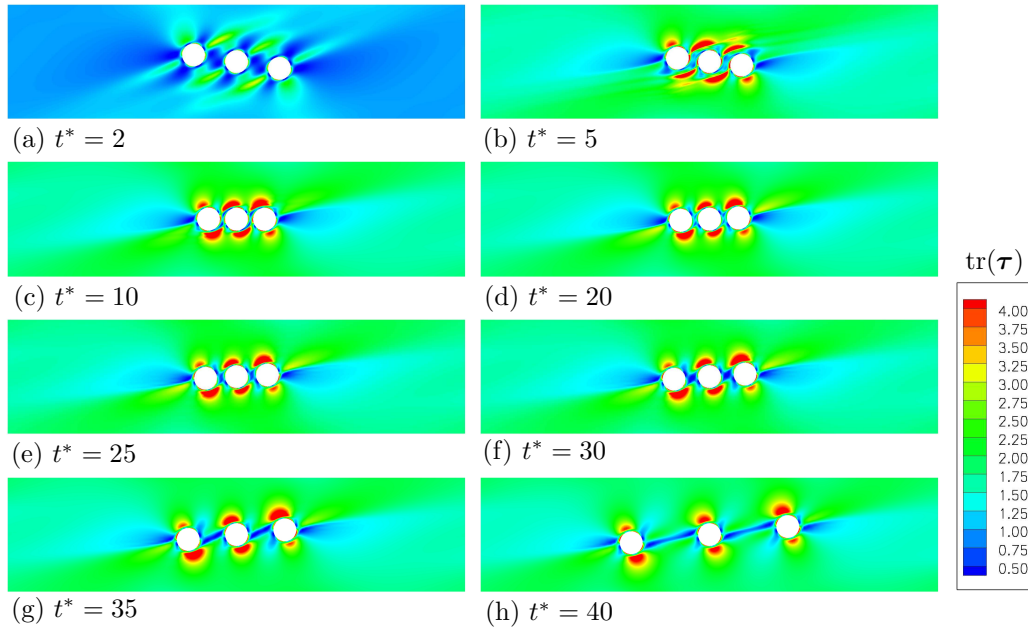


Figure 5.19: Snapshots of repelling motion of the particles for $Wi = 1, \alpha = 0.01, \beta = 0.1, l_x = 0.5, l_y = 0.08$. Contours are the trace of the polymer stress, $\text{tr}(\boldsymbol{\tau})$.

$\omega^* = \omega/\dot{\gamma} = 0.5$ [3, 4]. Fig. 5.23 compares the steady-state angular velocities of aligned particles for $Wi = 2, \alpha = 0.01, \beta = 0.1$. Note that the steady-state angular velocities are independent of the initial particle positions. If three particles form a string-like structure, the center particle rotates slower than the two end-particles, as shown in Fig. 5.23 (c). Moreover,

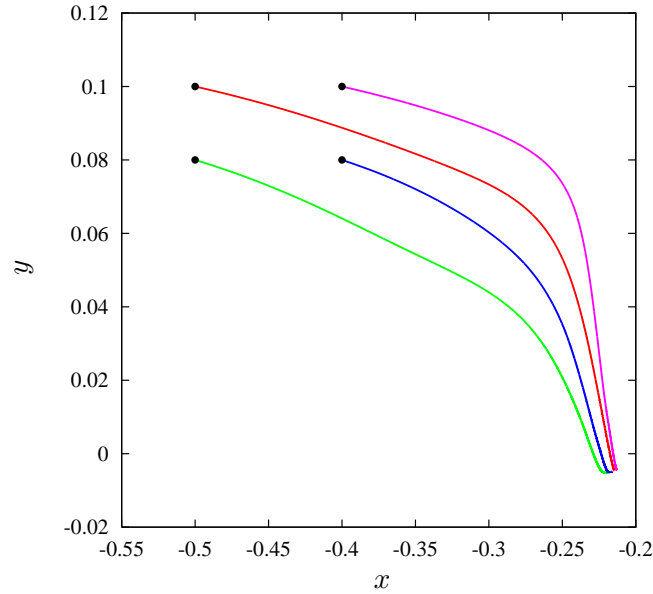


Figure 5.20: Trajectories of the top-left particle for different initial positions for $Wi = 2$, $\alpha = 0.01$, $\beta = 0.1$. Black dots represent initial positions of the top-left particle. Note that the particle moves to a final position independent from initial positions.

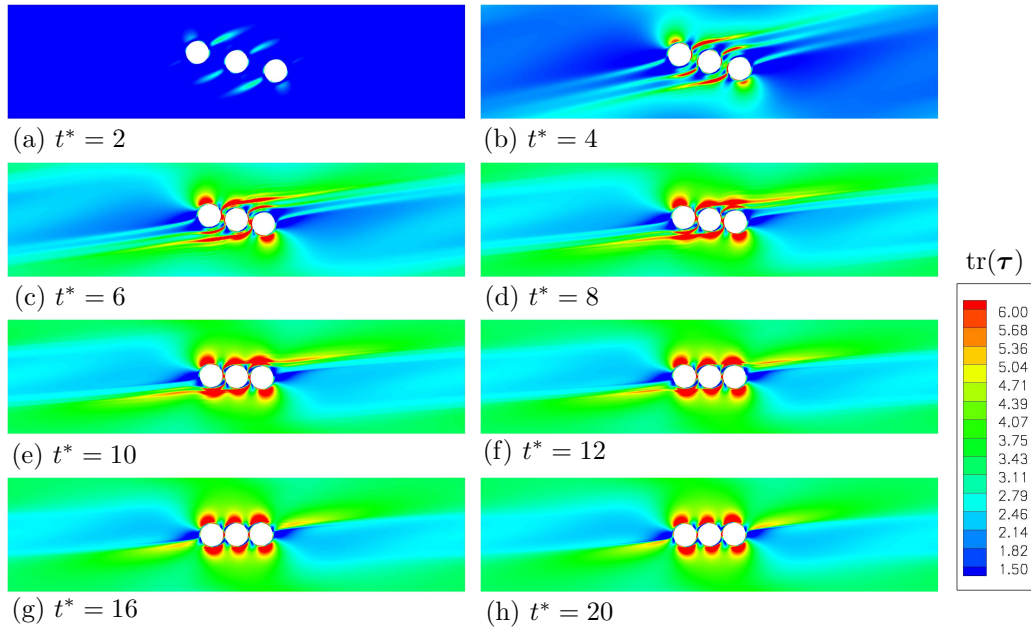


Figure 5.21: Snapshots of alignment of the particles for $Wi = 2$, $\alpha = 0.01$, $\beta = 0.1$, $l_x = 0.5$, $l_y = 0.1$. Contours are the trace of the polymer stress, $\text{tr}(\boldsymbol{\tau})$.

the angular velocity of the center particle is slower than that of the two-particle string.

In Fig. 5.23, we have also presented the interparticle distance $d^* = d/a$ at the steady-state, where d is the gap between two neighboring particles and a is the radius of the particles. Note

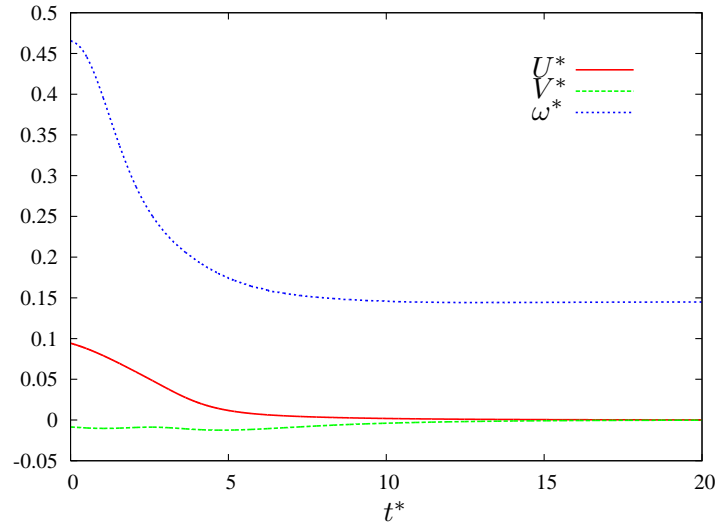


Figure 5.22: Velocities of the top-left particle as a function of time for $Wi = 2, \alpha = 0.01, \beta = 0.1, l_x = 0.5, l_y = 0.1$.

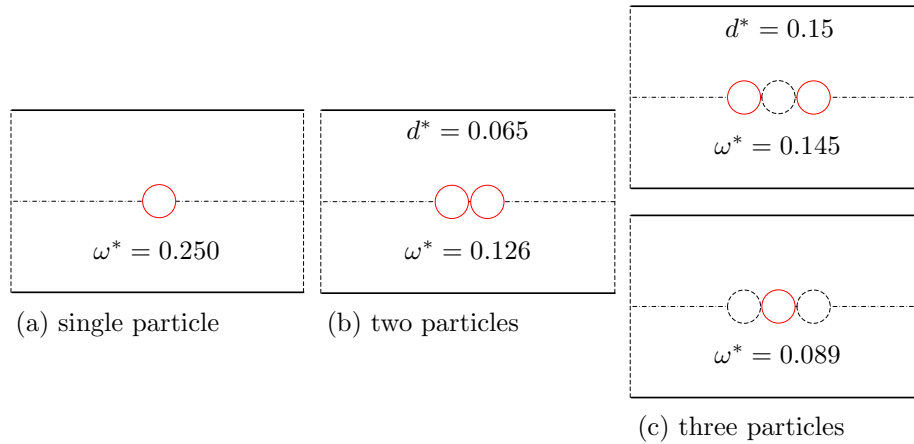


Figure 5.23: The steady-state angular velocities of particles for different lengths of strings for $Wi = 2, \alpha = 0.01, \beta = 0.1, H = 1, a = 0.1$. The value of angular velocity corresponds to the particles shown by solid red lines. The interparticle distance d^* at the steady-state is also presented.

that $d^* = 0.065$ and $d^* = 0.15$ in Fig. 5.23 (b) and Fig. 5.23 (c) means that the interparticle distance is 6.5% and 15%, respectively, of the particle radius. The interparticle distance of the three-particle string is larger than that of the two-particle string for the given problem. Note that the diagrams in Fig. 5.23 (b) and Fig. 5.23 (c) are schematic and do not represent the actual interparticle distances. They indicate the corresponding particles for the steady-state angular velocity shown.

We have tried to make strings of four particles and longer to compare the steady-state angular velocities, however we never observed alignment of four particles up to $Wi = 3$ while fixing other parameters ($H = 1, a = 0.1, \alpha = 0.01, \beta = 0.1$), as summarized in Table. 5.2,

Table 5.2: String formation of particles for $H = 1, a = 0.1, \alpha = 0.01, \beta = 0.1$

	number of particles in a string				
	2	3	4	5	6
Wi=1	+	-	-	-	-
Wi=2	+	+	-	-	-
Wi=3	+	+	-	-	-

where ‘+’ means that we observe string formation and ‘-’ means that we never observe string formation for the given condition. This observation suggests that the maximum obtainable length of a string of particles is limited for a certain Weissenberg number. We will discuss the alignment of multiple particles and the corresponding steady-state angular velocities further in Sec. 5.7.5.

5.7.3 Wall confinement

In this section, we investigate the effect of wall confinement on the string formation of three particles. Here we fix the channel length $L = 4$, the particle radius $a = 0.1$ and the initial particle positions to $(-0.5, 0.1)$, $(0, 0)$ and $(0.5, -0.1)$, i.e. $l_x = 0.5$, $l_y = 0.1$, then will vary the channel height H (see Fig 5.16). The confinement ratio is defined as $\delta = H/2a$, which means that the channel height is δ -times larger than the diameter of the particles. For example, the confinement ratios $\delta = 2, 4$, and 10 are shown in Fig. 5.24 with the initial particle positions to have a clear picture of the confinement. The minimum confinement ratio for the given initial positions is 2 (Fig. 5.24 (a)), in which the walls touch the end-particles at the given initial configuration. The wall confinement is stronger at lower confinement ratios.

For $\delta \geq 25$, we use a mesh which is regularly refined around the particles and the element size increases gradually to the walls (Fig. 5.25), to minimize computational costs in a large domain. Note that the element size near the particles is the same as the mesh M3; and for $\delta < 25$, we use a regular mesh over the whole domain as explained in Sec. 5.7.1. Note also that we use the grid deformation method [117] for mesh refinements around particle surface similar to [130].

The motion of the particles for various confinement ratios and Weissenberg numbers is characterized in Fig. 5.26. Note that fixed $\alpha = 0.01$ and $\beta = 0.1$ are used for all simulations in Fig. 5.26. If the particles are suspended in a Newtonian fluid (Wi=0), they pass each other and keep moving in their original directions for confinement ratio $\delta > 3$. If $2 < \delta \leq 3$, the particles can not pass due to the presence of the walls (see Fig. 5.27), and they are stuck or jammed in the middle of the channel if the Weissenberg number is not too high. Once the particles are jammed, our simulations can not continue since the particles overlap each other.

In a highly confined geometry of $\delta \leq 4$, we do not observe string formation of the particles up to $Wi = 3$. If the confinement ratio is increased to $\delta = 5$, the particles can form a string-like structure for $Wi \geq 1.5$. Hence we can conclude that strong wall confinement enhances repelling motion of particles, and then hinders alignment.

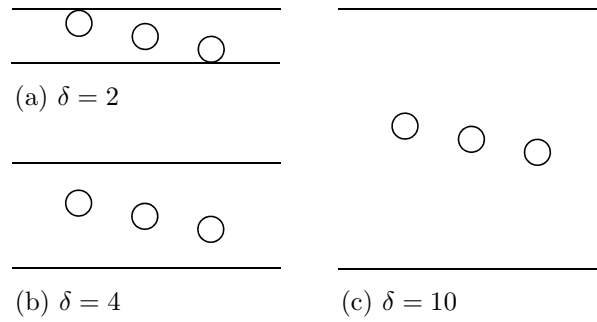


Figure 5.24: The confinement ratio $\delta = H/2a$ with different channel heights. Note that the actual channel length L is two times longer than shown here.

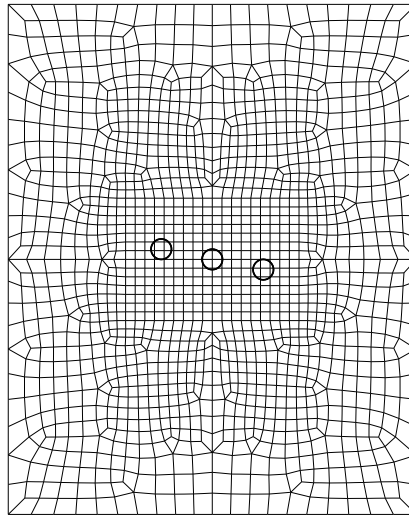


Figure 5.25: A mesh for the confinement ratio $\delta = 25$. Note that the actual mesh used in simulations is much more refined than shown here.

In contrast, if the confinement is weak, such as $\delta > 10$, the region of passing trajectories becomes wider, which means that the particles can more easily pass each other. If $\delta \geq 50$, it can be regarded as an unconfined flow since the particle motion is not affected by the further increase of the confinement ratio as shown in Fig 5.28, where we can still observe the string formation of particles for $Wi \geq 1$ in Fig. 5.26. Hence we can expect that the alignment of particles can occur even in an unconfined or infinite shear flow.

Moderate confinement ($\delta = 6 \sim 10$) seems to help the alignment of the particles, where the particles form a string-like structure for $Wi \geq 0.8$. Note that the transition between different kinds of particle motion is not sharp and somewhat continuous, as also observed by Hwang and Hulsen [56]. We judge the particles to form a string only if the angular velocity of each particle reaches a steady-state value, as explained in Sec. 5.7.2.

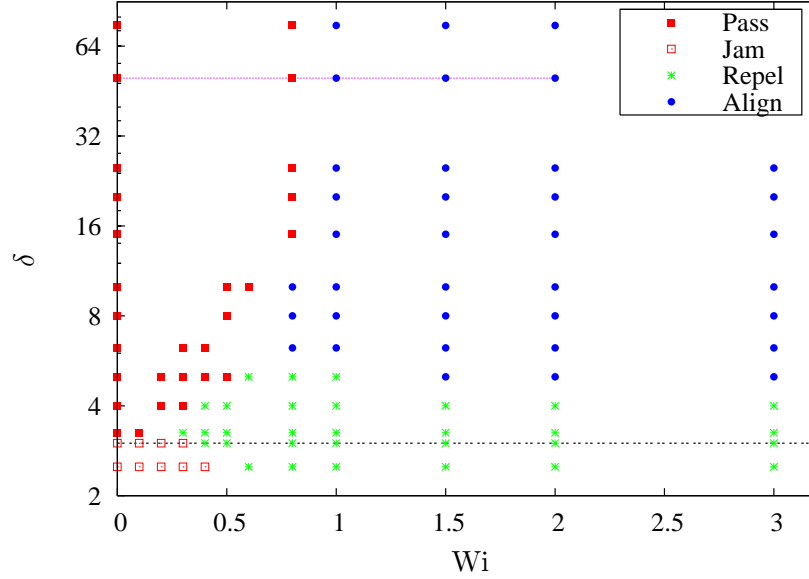


Figure 5.26: The effect of wall confinement on the motion of the three particles with identical initial positions. Below the black dotted line ($\delta = 3$), the particles can not pass; above the purple line ($\delta = 50$), the effect of walls on the movement of the particles is negligible.

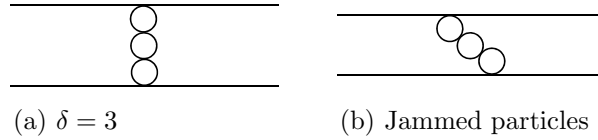


Figure 5.27: (a) The confinement ratio $\delta = 3$ with vertically aligned particles. If $\delta \leq 3$, the particles can not pass due to the presence of the walls. (b) A schematic of jammed particles.

5.7.4 Viscosity ratio

In this section, we investigate the effect of the viscosity ratio $\beta = \eta_s/\eta_0 = \eta_s/(\eta_s + \eta_p)$ on the string formation of three particles. We fix the channel height $H = 1$, the channel length $L = 4$, the particle radius $a = 0.1$, the initial particle positions corresponding to $l_x = 0.5$, $l_y = 0.1$, the Weissenberg number $Wi = 1$ and the mobility parameter $\alpha = 0$, then will vary the viscosity ratio β . Note that for $\alpha = 0$, the Giesekus model is identical to the Oldroyd-B model. In addition, we define the elasticity parameter S as the ratio of the first normal stress difference (N_1) to the shear stress (σ_{12}):

$$S = \frac{1}{2} \frac{N_1}{\sigma_{12}}. \quad (5.58)$$

With the factor $1/2$, the S parameter is related to the Weissenberg number by $S = (1 - \beta)Wi$ for the Oldroyd-B model ($\alpha = 0$).

For $\beta = 0$ and 0.1 ($S = 1$ and 0.9 , respectively), the three particles form a string; for $\beta = 0.3$ and 0.4 ($S = 0.7$ and 0.6 , respectively), the particles show repelling motion; for

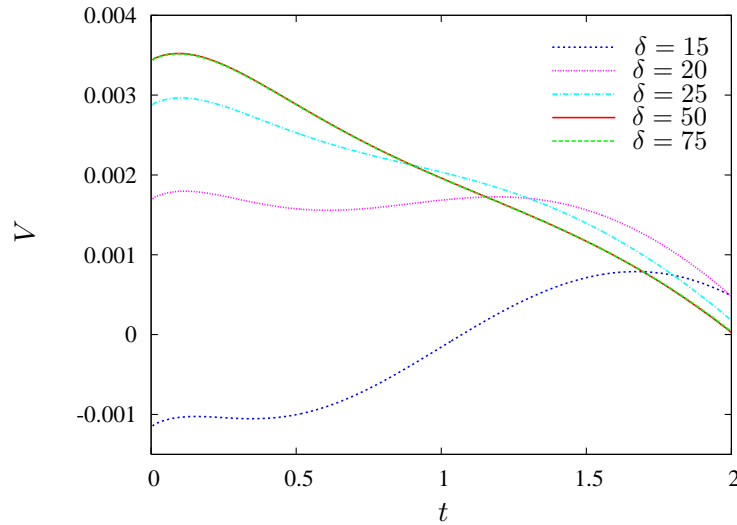


Figure 5.28: The y -directional translational velocity V of the top-left particle for various confinement ratios as a function of time ($Wi=2$). If $\delta \geq 50$, the motion of the particle is not affected by the confinement. We have also observed similar convergence behavior in each velocity component of every particle for all Weissenberg numbers simulated. Hence, if $\delta \geq 50$, it can be regarded as an unconfined flow.

$\beta \geq 0.7$ ($S \leq 0.3$), the particles show passing motion. Hence we can conclude that decreasing the viscosity ratio β , or increasing the elasticity parameter S , enhances the string formation of particles. This observation is quite intuitive: as the viscosity ratio decreases, the fluid becomes more elastic which strengthens the alignment of particles.

The observation of the motion of three particles with varying viscosity ratios agrees quite well with the results of Hwang and Hulsen [56], in which they qualitatively analyzed structural transitions from random particle configuration to string formation of particles in a sliding bi-periodic frame. For a given $Wi = 1$, particles formed a string for $\beta = 0.1$, and it appeared random for $\beta \geq 0.7$. Around $\beta = 0.3 \sim 0.5$, they observed clustering of particles. Hence, the passing motion in this work corresponds to random configuration in [56], which also implies that the fluid elasticity is too low to form a microstructure of particles. The repelling motion corresponds to the clustering in [56]. In this regime, the fluid elasticity pushes particles together, i.e. particles are attracted. However, the degree of elasticity is not strong enough to keep the particles in place, and they can not make a stable string.

5.7.5 Alignment of multiple particles

In Sec. 5.7.2, the alignment of three particles and the corresponding steady-state angular velocities of the particles have been presented. Based on the results of Sec. 5.7.2, we are aiming to form longer lengths of strings by changing some parameters used for obtaining Table 5.2. For the number of particles larger than 3, the initial positions of the particles are extended in a similar way as three-particle problems shown in Fig. 5.16. The distance, in each axial direction, between the center of two neighboring particles is l_x and l_y . For a given

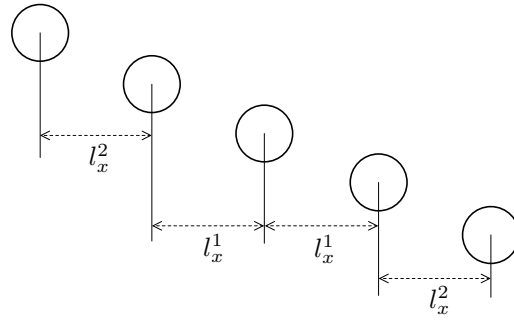


Figure 5.29: A schematic of non-equally spaced initial positions of five particles with $l_x^1 \neq l_x^2$.

fluid rheology, we perform extensive numerical simulations by varying the initial positions in a range of $2.5a \leq l_x \leq 7a$ and $0.25a \leq l_y \leq 1.3a$. We have also tested non-equally spaced initial positions, for example, as schematically shown in Fig. 5.29 with $l_x^1 \neq l_x^2$. Note that initial positions of particles are always point symmetric about the origin. If particles form a string, their final states are independent of initial positions, as already observed for two-particle problems in [130] and three-particle problems in Sec. 5.7.2.

First, we reduce the particle radius $a = 0.08$ to increase the confinement ratio ($\delta = 6.25$), motivated by Sec. 5.7.3. The results are summarized in Table 5.3, where we obtain strings of length up to 4 for $Wi = 3$. The values of S are also given in Table 5.3, and these values are the same in Table 5.2 for each corresponding Weissenberg number since we use the same

Table 5.3: String formation of particles for $H = 1, a = 0.08, \alpha = 0.01, \beta = 0.1$

	number of particles in a string				
	2	3	4	5	6
Wi=1 ($S = 0.88$)	+	+	-	-	-
Wi=2 ($S = 1.67$)	+	+	-	-	-
Wi=3 ($S = 2.34$)	+	+	+	-	-

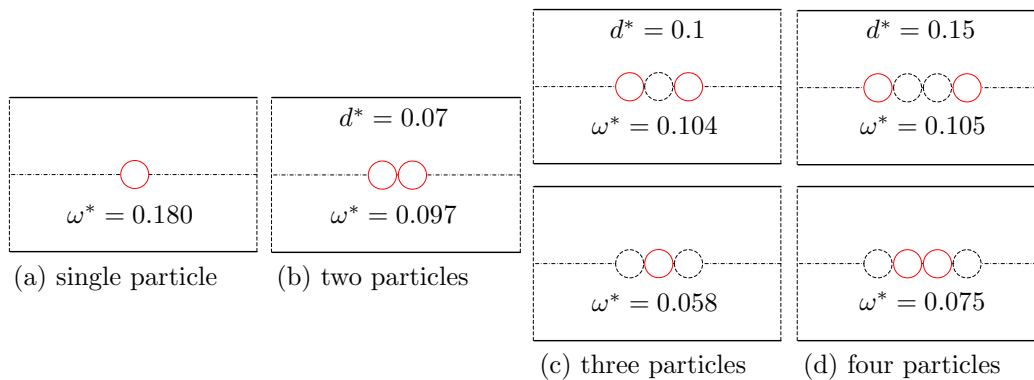


Figure 5.30: The steady-state angular velocities of particles for different lengths of strings for $Wi = 3, \alpha = 0.01, \beta = 0.1, H = 1, a = 0.08$. The averaged interparticle distance d^* at the steady-state is also presented.

fluid rheology. Note that here we have only changed the particle radius, or the corresponding confinement ratio from $\delta = 5$ to $\delta = 6.25$. The steady-state angular velocities for the case of $Wi = 3$ in Table 5.3 are shown in Fig. 5.30. As the length of a string increases up to the length of 3, the steady-state angular velocities decreases, with respect to the center particle for the three-particle problem, similar to the case of Fig. 5.23. However, if the length of a string is further increased to 4, the steady-state angular velocity increases, especially for non end-particles. We have also presented the steady-state interparticle distance d^* in Fig. 5.30. In Fig. 5.30 (d), we do not discriminate between the outermost and innermost interparticle distances, contrary to the steady-state angular velocities. The given interparticle distance is the averaged interparticle distance. In Fig. 5.30, the interparticle distance increases as the length of a string increases.

Now we will change the value of the mobility parameter α of the Giesekus fluid model to obtain far longer strings of particles, while keeping other parameters the same as in Table 5.2. Note that the confinement ratio $\delta = 5$. Table 5.4 is obtained by setting $\alpha = 0$, which is identical to the Oldroyd-B model. In order to obtain the maximum string length for the case of $Wi = 3$ in Table 5.4, we increase the length of the channel L , while keeping the other parameters the same, to minimize the interaction between the real and imaginary particles across the periodic unit. Fig. 5.31 shows a mesh for the length $L = 7$ with seven particles as an example: the mesh is refined in the center region where the particles move, and the element size increases gradually to the channel exits. Note that in actual simulations, the mesh size in the refined center region is the same as that of mesh M3. We have tested $L = 5, 6,$ and 7 for the six-particle problem in Table 5.4, and do not recognize any significant difference in the motion of the particles compared with that of the particles in $L = 4$. It verifies that the string of the six particles for $Wi = 3$ is a stable isolated string even in the channel length $L = 4$. Then we repeat the same procedure by increasing the number of particles in various channel lengths until we do not observe the string formation any more. We observe the string formation up to length 8 for $Wi = 3$. Note that in the original channel length $L = 4$,

Table 5.4: String formation of particles for $H = 1, a = 0.1, \alpha = 0, \beta = 0.1$

	number of particles in a string									
	2	3	4	5	6	7	8	9	11	
Wi=1 ($S = 0.9$)	+	+	-	-	-	-	-	-	-	
Wi=2 ($S = 1.8$)	+	+	+	+	-	-	-	-	-	
Wi=3 ($S = 2.7$)	+	+	+	+	+	+	+	-	-	

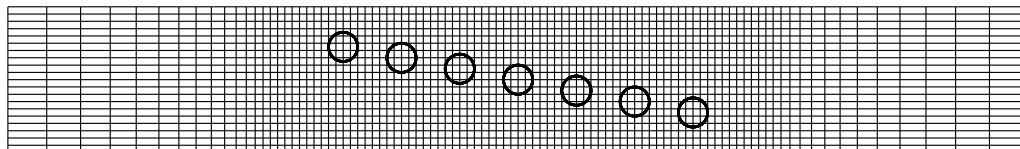


Figure 5.31: A mesh for the length $L = 7$ with 7 particles. Note that the actual mesh used in simulations is much more refined than shown here.

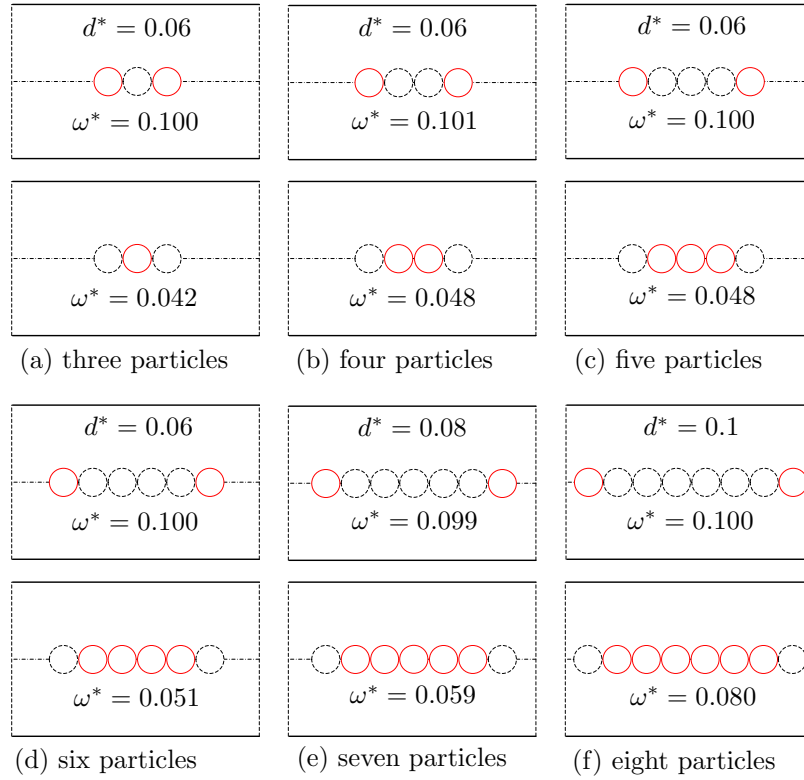


Figure 5.32: The steady-state angular velocities of particles for different lengths of strings for $Wi = 3, \alpha = 0, \beta = 0.1, H = 1, a = 0.1$. The averaged interparticle distance d^* at the steady-state is also presented.

9 particles seem to form a string. However, if we increase the channel length $L \geq 5$, we do not observe string formation of 9 particles. For 11 particles, they do not form a string even in the channel length $L = 4$, nor in a longer channel. Hence we can conclude that the maximum length of strings for the given fluid rheology is 8 for $Wi = 3$.

In this section, we have presented the formation of stable isolated strings, and the maximum obtainable string length increases with increasing the Weissenberg number. By contrast, in the work of Hwang and Hulslen [56], the length of string is unclear since the particles spread out over the entire length of the periodic unit. It is also worthy to note on the kissing-tumbling-tumbling- (\dots) phenomenon of two particles in a sliding bi-periodic frame observed by Hwang et al. [74]. We simulate the problem in a confined geometry using the same dimensions and the fluid rheology as presented in [74]. The only difference is that we use the periodicity only in x -direction, contrary to the bi-periodicity in [74]. The particles are confined between two parallel walls in our simulations. We also observe similar kissing-tumbling-tumbling- (\dots) motion of the two-particles. However, if we increase the length of the channel, while keeping other parameters the same, the particles approach and tumble to each other, then just pass and keep moving in their original direction. We never observe the kissing-tumbling-tumbling- (\dots) motion in a longer channel. Hence we regard that the tumbling-tumbling- (\dots) motion is most likely caused by strong interaction between the real and imaginary particles across

the periodic unit since the length of the domain is too short.

If Table 5.4 is compared to Table 5.2, decreasing the mobility parameter α enhances the alignment of particles. The results are somewhat contradictory to the experimental results of Won & Kim [53] and Scirocco et al. [54], in which they never observed alignment of particles in viscoelastic fluids with constant viscosity. However, the use of the Oldroyd-B model, which has a constant viscosity, promotes the alignment compared to the Giesekus model with non-zero mobility parameter α , which is shear-thinning. Hwang and Hulsen [56] also reported that decreasing α enhances the string formation for constant β , i.e. $\alpha = 0$ gives the most probable alignment, which is in good agreement with our observation. The alignment of particles may depend on the details of the rheology of the suspending fluid and the constitutive model used. It requires further investigation.

Fig. 5.32 compares the angular velocities of particles for $Wi = 3$ in Table 5.4 for string lengths longer than 3. Still the angular velocity of the center particle of length 3 is the lowest, although the difference becomes smaller for the case shown here. If the length of a string is longer than 3, the steady-state angular velocities of the two end-particles do not change significantly as the string length increases. By contrast, the steady-state angular velocities of the non end-particles increase as the length increases. Note that the angular velocity of the two end-particles is faster than those of the particles in between, and the angular velocities of the non end-particles are the same in a given string. We have also presented the steady-state interparticle distance d^* in Fig. 5.32. Note that d^* in Fig. 5.32 is the averaged interparticle distance at the steady-state in a given string. The interparticle distance increases as the length of a string increases, especially in the region where the angular velocities of the non end-particles increase.

5.8 Conclusions

Three different regimes of two-particle motions in a confined shear flow of a viscoelastic fluid are characterized according to initial separations of particles. For small initial separations, two particles repel each other and their trajectories are reversed; for large initial separations, two particles keep going in their original directions and pass each other, similarly to the particle motions in Stokes flow problems. For intermediate initial separations, two particles come together and reach final fixed positions, then they rotate with a non-zero terminal angular velocity. In the intermediate regime, the final positions of particles are independent of their initial positions.

Motivated by the two-particle interaction problem, the string formation of multiple particles is quantitatively analyzed by using a weak formulation for imposing the no-slip boundary condition on the particle surface, which is equivalent to adding a positive definite stabilizing term in the momentum balance. Once particles form a string-like structure, the final state is independent of the initial particle positions and the histories to reach the steady-state. The maximum obtainable length of a string of particles is limited for a certain fluid rheology. As the Weissenberg number increases, particles can form longer strings. For the three-particle problem presented, moderate wall confinement helps the string formation of particles, how-

ever, too strong confinement hinders the alignment by enhancing repulsive interaction between particles. We have compared the steady-state angular velocities of particles with respect to the length of strings. In a given string, the angular velocity of the two end-particles is faster than those of the particles in between. The angular velocities of the non end-particles are the same in a string. If the length of a string is longer than 3, the steady-state angular velocities of the two end-particles do not change significantly, and those of the non end-particles increase, as the string length increases. We have also presented the steady-state interparticle distance between two neighboring particles in a string. As the string length increases, the interparticle distance increases.

The method proposed is quite general and can be extended to three-dimensional simulations without any loss of generality. Future work will be focused on three-dimensional simulations of the motion of multiple particles in a viscoelastic fluid.

Chapter 6

Particle migration in extrusion flow

In this chapter, the simulation of the extrusion of a particle filled viscoelastic fluid is carried out by using a combined XFEM scheme both for the particle and the free surface. The main advantage of the proposed method is that the movement of the free surface and particles can be simulated on a fixed Eulerian mesh without any need of remeshing.

The swell ratios of an upper-convected Maxwell fluid for various Deborah numbers are presented to validate the method with the presence of the free surface. Note that the movement of particles is already verified in Chapters 3 and 5. The swell ratio of an upper-convected Maxwell fluid is compared with those of the moving boundary-fitted mesh problems of the conventional ALE technique. The proposed XFEM combined with the temporary ALE scheme can provide similar accuracy to the boundary-fitted mesh problems for low Deborah numbers. For high Deborah numbers, the method seems to be more stable for the extrusion problem.

The migration of a particle near the free surface of an extrusion flow is also presented. The presence of the particle disturbs the stress distribution and the free surface profile significantly. We found that the particle moves away from the free surface as the Weissenberg number increases. As the particle size increases, the particle moves towards the free surface.

The content of this Chapter is based on:

Young Joon Choi and Martien A. Hulsen. Migration of a particle near the free surface of an extrusion flow. in preparation.

Young Joon Choi and Martien A. Hulsen. Simulation of extrudate swell using an extended finite element method. *Korea-Australia Rheology Journal*, in press.

6.1 Introduction

Extrusion is a process in which a material is pushed or drawn through a die of a desired cross section. If a viscoelastic material is drawn through a die, the diameter or thickness of the extrudate can increase significantly compared with that of the die, which is usually referred to as extrudate swell. The mechanism of extrudate swell is well described using the idea of the elastic recovery [143]. In the process, polymer melts and concentrated solutions exhibit flow instabilities at a critical shear rate [144]. The development of flow instabilities and the associated surface irregularities during extrusion can be reduced or eliminated upon the incorporation of rigid particles in the melts, especially as the concentration of the particles is increased [145]. Also polymer melts usually contain particle additives to improve mechanical, but also non-mechanical product properties, such as color, heat resistance, electric conductivity etc. It is therefore interesting to investigate the migration of suspended particles near the free surface of an extrusion process.

For a numerical prediction of extrudate swell, an arbitrary Lagrangian-Eulerian (ALE) technique is commonly used [146, 147], in which elements of a mesh are only moved normal to the flow direction, independent from the material motion. If the elements of the mesh become too distorted, a generation of a new mesh is required and the solution should be projected onto the new mesh. However, the generation of the boundary-fitted mesh is still a challenging task, if the shape of the free surface is complex, especially in three-dimensional simulations. There are also many kinds of unsteady flows which may need frequent remeshing such as mold-filling, fiber spinning, film blowing etc.

Alternatively, we propose a method which can simulate moving boundary problems without any need of remeshing using an extended finite element method (XFEM). In this method, the time dependent fluid domain is fully immersed in a larger time independent mesh region, while the degrees of freedom outside of the fluid domain are discarded. One of the most advantageous aspects of the XFEM is that we can capture stress jumps although an interface is not aligned with boundaries of elements. The current authors already presented an application of the XFEM to the problem of moving particles in a viscoelastic fluid [130]. The current work is an extension of our previous study [130] to the problem associated with the presence of a free surface. For more information of the XFEM, see the references in [130], and a review paper by Belytschko et al. [94].

The content of the chapter is as follows. In Section 6.2, we give a brief review of the governing equations for the motion of incompressible viscoelastic fluids. In Section 6.3, we present a numerical algorithm of the extended finite element method and a temporary arbitrary Lagrangian-Eulerian scheme. In Section 6.4, numerical results for extrudate swell are presented to validate the proposed method for the movement of free surface. In Section 6.5, migration of a particle near the free surface of an extrusion flow is presented. Conclusions follow in Section 6.6.

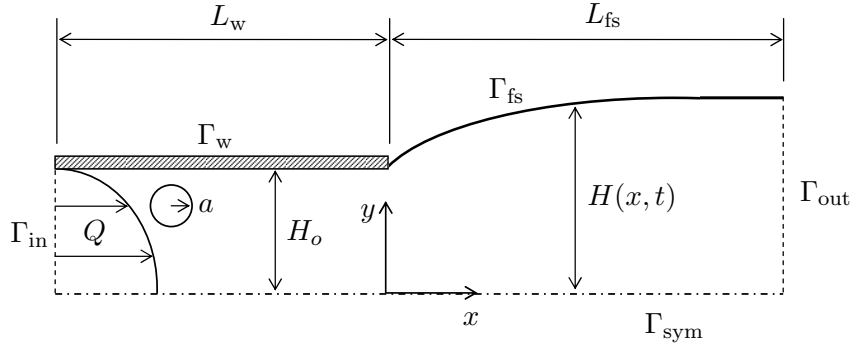


Figure 6.1: Schematic description of a particle suspended in an extrusion flow.

6.2 Governing equations

We consider the motion of a rigid particle suspended in an extrusion of a viscoelastic fluid as shown in Fig. 6.1. Let $\Omega(t)$ be the domain of the fluid and $P(t)$ be the domain of the particle at time t . Boundaries are denoted by $\Gamma(t) = \partial\Omega(t)$ and $\Gamma_p(t) = \partial P(t)$. The fluid is assumed to be incompressible, and inertia and body forces both for the fluid and the particle are neglected.

The equations of motion for the fluid are given by the following momentum and mass balance, respectively:

$$\nabla p - \nabla \cdot \boldsymbol{\tau} = \mathbf{0} \quad \text{in } \Omega(t), \quad (6.1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega(t), \quad (6.2)$$

where p is the pressure, $\boldsymbol{\tau}$ is the polymer stress tensor and \mathbf{u} is the velocity vector. Note that we neglect a Newtonian solvent contribution to the stress, which is common for polymer melts or concentrated polymer solutions. If we assume that Ω is two-dimensional, the free surface can be represented by a one-dimensional height function $H(x, t)$ [146]. The evolution of the flow domain can be determined by the kinematic equation [146]:

$$\frac{\partial H}{\partial t} + u_x \frac{\partial H}{\partial x} = u_y, \quad (6.3)$$

where u_x and u_y are the velocity components at the free surface. For a constitutive model of a viscoelastic fluid, we use the Giesekus model:

$$\lambda \overset{\nabla}{\mathbf{c}} + \mathbf{c} - \mathbf{I} + \alpha(\mathbf{c} - \mathbf{I})^2 = \mathbf{0}, \quad (6.4)$$

where \mathbf{c} is the conformation tensor, \mathbf{I} is the unity tensor, λ is the relaxation time and α is the mobility parameter. The triangle (∇) denotes the upper-convected time derivative, defined as:

$$\overset{\nabla}{\mathbf{c}} = \frac{\partial \mathbf{c}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{c} - (\nabla \mathbf{u})^T \cdot \mathbf{c} - \mathbf{c} \cdot \nabla \mathbf{u}. \quad (6.5)$$

In the Giesekus model, the polymer stress can be written as:

$$\boldsymbol{\tau}(\mathbf{c}) = \frac{\eta_p}{\lambda}(\mathbf{c} - \mathbf{I}), \quad (6.6)$$

where η_p is the polymer viscosity. Furthermore, the equation for the conformation tensor \mathbf{c} , Eq. (6.4), can be transformed to an equivalent log-conformation equation for $\mathbf{s} = \log \mathbf{c}$ to achieve major stability improvements:

$$\frac{\partial \mathbf{s}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{s} - \mathbf{g}((\nabla \mathbf{u})^T, \mathbf{s}) = \mathbf{0}. \quad (6.7)$$

Once \mathbf{s} has been solved, the conformation tensor \mathbf{c} can be computed from $\mathbf{c} = \exp(\mathbf{s})$. A detailed explanation of the log-conformation representation (LCR) and exact expression for the function $\mathbf{g}((\nabla \mathbf{u})^T, \mathbf{s})$ can be found in [96]. For all numerical simulations in this chapter, we use the log-conformation representation. Hence, we will present the weak formulation, and the boundary and initial conditions only in the LCR.

The boundary and initial conditions are given by:

$$\mathbf{u} = \mathbf{0} \quad \text{on } \Gamma_w, \quad (6.8)$$

$$\mathbf{u} = \mathbf{U} + \boldsymbol{\omega} \times (\mathbf{x} - \mathbf{X}) \quad \text{on } \Gamma_p(t), \quad (6.9)$$

$$\mathbf{t} = (-p\mathbf{I} + \boldsymbol{\tau}) \cdot \mathbf{n} = \mathbf{0} \quad \text{on } \Gamma_{fs}, \quad (6.10)$$

$$\mathbf{t} = (-p\mathbf{I} + \boldsymbol{\tau}) \cdot \mathbf{n} = \mathbf{0} \quad \text{on } \Gamma_{out}, \quad (6.11)$$

$$u_y = 0, \quad t_x = (-p\mathbf{I} + \boldsymbol{\tau}) \cdot \mathbf{e}_x = 0 \quad \text{on } \Gamma_{sym}, \quad (6.12)$$

$$\text{open boundary condition for } (\mathbf{u}, p, \mathbf{s}) \quad \text{on } \Gamma_{in}, \quad (6.13)$$

$$\int_{\Gamma_{in}} \mathbf{u} \cdot \mathbf{n} \, ds = -Q \quad \text{on } \Gamma_{in}, \quad (6.14)$$

$$\mathbf{s}(\mathbf{x}, t = 0) = \mathbf{s}_0(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega(0), \quad (6.15)$$

$$H(x, t = 0) = H_0, \quad \forall x \geq 0, \quad (6.16)$$

$$H(x = 0, t) = H_0, \quad \forall t \geq 0, \quad (6.17)$$

where Γ_w , Γ_{fs} , Γ_{out} , Γ_{in} and Γ_{sym} represent the boundary of die wall, free surface, outflow, inflow and symmetry, respectively. \mathbf{n} is the outwardly directed unit normal vector on the fluid and \mathbf{e}_x is unit vector in x -direction. Eq. (6.9) represents the no-slip boundary condition on the particle surface, where \mathbf{U} is the translational velocity, $\boldsymbol{\omega}$ is the rotational velocity and \mathbf{X} is the position of the center of the particle. In Eq. (6.13), the open boundary condition is used both for the momentum balance [140] and the inlet stress condition of the constitutive equation [148]. In Eq. (6.14), the flow rate Q is imposed on the inflow boundary by using a Lagrangian multiplier technique. In our simulations, we use a stress-free state as initial condition over the whole fluid domain, i.e. $\mathbf{s}_0(\mathbf{x}) = \mathbf{0}, \forall \mathbf{x} \in \Omega(0)$. Since we neglect inertia, an initial condition for the velocity is not necessary.

For the unknown rigid body motion $(\mathbf{U}, \boldsymbol{\omega})$ of the particle, we need balance equations for forces and torques on the particle surface. In the absence of inertia, and external forces and

torques on the particle, the net hydrodynamic force \mathbf{F} and torque \mathbf{T} acting on the particle $P(t)$ are zero:

$$\mathbf{F} = \int_{\partial P(t)} \boldsymbol{\sigma} \cdot \hat{\mathbf{n}} \, ds = \mathbf{0}, \quad (6.18)$$

$$\mathbf{T} = \int_{\partial P(t)} (\mathbf{x} - \mathbf{X}) \times (\boldsymbol{\sigma} \cdot \hat{\mathbf{n}}) \, ds = \mathbf{0}, \quad (6.19)$$

where $\hat{\mathbf{n}}$ is the outwardly directed unit normal vector on the particle surface Γ_p . The particle positions \mathbf{X} and angular orientations $\boldsymbol{\Theta}$ are obtained from the following kinematic equations:

$$\frac{d\mathbf{X}}{dt} = \mathbf{U}, \quad \mathbf{X}(t=0) = \mathbf{X}_0, \quad (6.20)$$

$$\frac{d\boldsymbol{\Theta}}{dt} = \boldsymbol{\omega}, \quad \boldsymbol{\Theta}(t=0) = \boldsymbol{\Theta}_0. \quad (6.21)$$

6.3 Numerical methods

6.3.1 Weak form

In deriving the weak form of the governing equations for a fluid-particle system, we follow the combined equation of motion approach [64], in which the fluid and the particle equations of motion are combined in one equation. For the discretization of the momentum equation, we employ the DEVSS-G method [100, 101]. The no-slip boundary condition on the particle surface (Eq. (6.9)) is imposed using a weak boundary condition, by adding a positive definite stabilizing term in the momentum balance [149]. By introducing the variations \mathbf{V} and $\boldsymbol{\chi}$ of \mathbf{U} and $\boldsymbol{\omega}$, respectively, we include the force and torque free conditions (Eqs. (6.18)-(6.19)) in the momentum balance. The weak formulation of the governing equations can be stated as follows:

$$\begin{aligned} & - (\nabla \cdot \mathbf{v}, p) + ((\nabla \mathbf{v})^T, \theta(\nabla \mathbf{u} - \mathbf{G}^T) + \boldsymbol{\tau}) - (\mathbf{v}, (-p\mathbf{I} + \boldsymbol{\tau}) \cdot \mathbf{n})_{\Gamma_{\text{in}}} + (\mathbf{v} \cdot \mathbf{n}, \lambda)_{\Gamma_{\text{in}}} \\ & \quad - (\mathbf{v} - (\mathbf{V} + \boldsymbol{\chi} \times (\mathbf{x} - \mathbf{X})), (-p\mathbf{I} + \boldsymbol{\tau}) \cdot \mathbf{n})_{\Gamma_p} \\ & \quad + \kappa M_{km}^{-1} ((\mathbf{v} - (\mathbf{V} + \boldsymbol{\chi} \times (\mathbf{x} - \mathbf{X}))) \mathbf{n}, \phi_k)_{\Gamma_p} : \\ & (\phi_m, \mathbf{n}(\mathbf{u} - (\mathbf{U} + \boldsymbol{\omega} \times (\mathbf{x} - \mathbf{X}))) + (\mathbf{u} - (\mathbf{U} + \boldsymbol{\omega} \times (\mathbf{x} - \mathbf{X}))) \mathbf{n})_{\Gamma_p} = 0, \end{aligned} \quad (6.22)$$

$$(q, \nabla \cdot \mathbf{u}) = 0, \quad (6.23)$$

$$(\mathbf{H}, -\nabla \mathbf{u} + \mathbf{G}^T) = 0, \quad (6.24)$$

$$(\mu, \mathbf{u} \cdot \mathbf{n})_{\Gamma_{\text{in}}} = -Q, \quad (6.25)$$

where \mathbf{v} , q , \mathbf{H} and μ are test functions for the velocity \mathbf{u} , pressure p , projected velocity gradient \mathbf{G} and Lagrangian multiplier λ , respectively. (\cdot, \cdot) , $(\cdot, \cdot)_{\Gamma_{\text{in}}}$ and $(\cdot, \cdot)_{\Gamma_p}$ are proper inner products on the fluid domain $\Omega(t)$, on the inflow boundary Γ_{in} and on the particle surface Γ_p , respectively. \mathbf{n} is the outwardly directed unit normal vector on the fluid. In

Eq. (6.22), the DEVSS-G parameter θ is chosen equal to the polymer viscosity, $\theta = \eta_p$; and the open boundary condition is used at the inflow boundary [140, 148]. $M_{km} = (\phi_k, \phi_m)$ with the summation convention for the indices k and m running over all nodes in the mesh, and the shape functions ϕ_k will be taken equal order with the velocity shape functions, but discontinuous across element boundaries [139, 149]. Due to the discontinuous shape functions ϕ_k , M_{km}^{-1} can be computed on element level of intersected elements in Eq. (6.22). Since the mass matrix M_{km} is positive definite, the matrix multiplication with M_{km}^{-1} can easily be computed with a Cholesky decomposition. The κ parameter is chosen to the zero-shear-rate viscosity $\kappa = \eta_0 = \eta_p$, since the Newtonian viscosity is neglected, which seems to produce optimal convergence [139].

For moving boundary problems, if the mesh moves independently from the material motion, it is usually called an arbitrary Lagrangian-Eulerian (ALE) formulation [61]. The ALE formulation helps to minimize mesh distortion while still being able to track material boundaries. The relationship between time derivatives of a quantity f in material and spatial domains is given by

$$\dot{f} = \frac{\delta f}{\delta t} + (\mathbf{u} - \mathbf{u}_m) \cdot \nabla f, \quad (6.26)$$

where \dot{f} is the material derivative, $\delta f/\delta t$ is the grid time derivative and \mathbf{u}_m is the mesh velocity. We incorporate an ALE scheme to cope with the movement of the free surface, which will be explained in Sec 6.3.3.

We decouple the momentum and mass balance equations from the constitutive equation as proposed by D'Avino and Hulsen [141], in which the stress tensor $\boldsymbol{\tau}$ is replaced by a time-discretized but space-continuous form of the constitutive equation:

$$\boldsymbol{\tau}(\mathbf{c}^{n+1}) = G\Delta t \left(-(\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{c}^n + (\nabla \mathbf{u}^{n+1})^T \cdot \mathbf{c}^n + \mathbf{c}^n \cdot \nabla \mathbf{u}^{n+1} \right) + G\mathbf{h}(\mathbf{c}^n, \Delta t) - G\mathbf{I}, \quad (6.27)$$

where Δt is the computational time step and

$$\mathbf{h}(\mathbf{c}^n, \Delta t) = \mathbf{c}^n - \frac{\Delta t}{\lambda} (\mathbf{c}^n - \mathbf{I} + \alpha(\mathbf{c}^n - \mathbf{I})^2) \quad (6.28)$$

for the Giesekus model. We use L^2 -projection of $\mathbf{c}^n = \exp(\mathbf{s}^n)$ in Eq. (6.27) for a better numerical stability:

$$(\mathbf{e}, \mathbf{c}^n) = (\mathbf{e}, \exp(\mathbf{s}^n)), \quad (6.29)$$

where \mathbf{e} is a test function. By substituting Eq. (6.27) into Eq. (6.22), we get a Stokes-like problem for $(\mathbf{u}^{n+1}, p^{n+1}, \mathbf{G}^{n+1})$ depending on \mathbf{c}^n . Then the value of \mathbf{c}^{n+1} can be found by solving the constitutive equation with known values of $(\mathbf{u}^{n+1}, p^{n+1}, \mathbf{G}^{n+1})$.

For the time discretization of the constitutive equation, we use a second-order time-integration scheme, based on a Gear scheme, proposed by D'Avino and Hulsen [141]. We incorporate the log-conformation representation (LCR) [96] and the SUPG stabilization tech-

nique [102], as well:

$$\begin{aligned} (\mathbf{d} + \tau(\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{d}, \frac{\frac{3}{2}\mathbf{s}^{n+1} - 2\mathbf{s}^n + \frac{1}{2}\mathbf{s}^{n-1}}{\Delta t} \\ + (\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{s}^{n+1} - \mathbf{g}(\mathbf{G}^{n+1}, \hat{\mathbf{s}}^{n+1})) = 0, \end{aligned} \quad (6.30)$$

where $\mathbf{s} = \log \mathbf{c}$, $\hat{\mathbf{s}}^{n+1} = 2\mathbf{s}^n - \mathbf{s}^{n-1}$ and \mathbf{d} is a test function for the log-conformation \mathbf{s} . We follow the SUPG parameter τ given by Choi et al. [130].

For the discretization of the evolution equation of the height function (Eq. (6.3)), a second-order Gear scheme and the SUPG stabilization technique [102] are used:

$$\left(K + \hat{\tau} u_x \frac{\partial K}{\partial x}, \frac{3}{2} \frac{H^{n+1}}{\Delta t} + u_x^{n+1} \frac{\partial H^{n+1}}{\partial x} \right)_l = \left(K + \hat{\tau} u_x \frac{\partial K}{\partial x}, 2 \frac{H^n}{\Delta t} - \frac{1}{2} \frac{H^{n-1}}{\Delta t} + u_y^{n+1} \right)_l, \quad (6.31)$$

where K is the test function for the height function H and $H^k = H(x, t = t^k)$ for $k = n-1, n, \text{ or } n+1$. Note that $(\cdot, \cdot)_l$ is an inner product for a one-dimensional convection problem. The SUPG parameter $\hat{\tau}$ is given by $\hat{\tau} = h/2U$, where h is a size of the line element and U is a characteristic velocity magnitude.

6.3.2 A brief description of XFEM

The authors already introduced XFEM for the problem of moving particles in viscoelastic fluids [130, 149]. Here we apply XFEM to the problem of a moving boundary due to the existence of free surface in a similar way. The fluid domain $\Omega(t)$ and the particle domain $P(t)$ is fully included in a larger mesh region Ω_m which is independent in time. Boundaries of Ω_m are denoted by $\Gamma_m = \partial\Omega_m$. The discretization of the velocity, pressure, log-conformation and velocity gradient projection is given by

$$\mathbf{u}_h(\mathbf{x}) = \sum_k \varphi_k(\mathbf{x}) \mathbf{u}_k, \quad \mathbf{x} \in \Omega(t), \quad (6.32)$$

$$p_h(\mathbf{x}) = \sum_k \psi_k(\mathbf{x}) p_k, \quad \mathbf{x} \in \Omega(t), \quad (6.33)$$

$$\mathbf{s}_h(\mathbf{x}) = \sum_k \psi_k(\mathbf{x}) \mathbf{s}_k, \quad \mathbf{x} \in \Omega(t), \quad (6.34)$$

$$\mathbf{G}_h(\mathbf{x}) = \sum_k \psi_k(\mathbf{x}) \mathbf{G}_k, \quad \mathbf{x} \in \Omega(t). \quad (6.35)$$

We use a bi-quadratic Q_2 interpolation (φ_k) for the velocity \mathbf{u} ; and a bi-linear Q_1 interpolation (ψ_k) for the pressure p , log-conformation tensor \mathbf{s} , velocity gradient projection \mathbf{G} , and the projected conformation \mathbf{c} in Eq. (6.29). Note that the shape functions are only defined in the nodes inside the fluid domain Ω and associated to the elements intersected by the boundary Γ_{fs} . However, they are not evaluated outside the domain Ω for elements intersected by the boundary Γ_{fs} . Since the weak form is defined on Ω only, the integration should be performed only on part of an element for an intersected element. For the integration on the part of

Table 6.1: Subdomain integration rules.

	quadrilateral subdomains	triangular subdomains
$A_{\text{int}}/A_{\text{elem}} > 10^{-2}$	3×3	6-point
$10^{-6} < A_{\text{int}}/A_{\text{elem}} \leq 10^{-2}$	3×3	16-point
$A_{\text{int}}/A_{\text{elem}} \leq 10^{-6}$	The element is discarded.	

an element, we use a quadtree subdivision and further triangulation of the smallest quads near the free surface, similar to [130], using a levelset function $\zeta(x, y) = H(x, t) - y$ to find the free surface. In simulations, we use five level subdivisions; a 3×3 Gauss integration rule for undivided elements and quadrilateral subdomains, and a 6-point or 16-point Gauss integration rule [142] for triangular subdomains depending on the ratio of the integration area (A_{int}) over the element area (A_{elem}). The integration rules used are summarized in Table 6.1. If the integration area of an element is very small compared to the element area ($A_{\text{int}}/A_{\text{elem}} \leq 10^{-6}$), the element is discarded, i.e. the element is treated as if it is fully outside of the fluid domain Ω .

6.3.3 Temporary ALE scheme for moving boundary problems

Let's suppose that the free surface moves from \hat{H}^n at the previous time level t^n to \hat{H}^{n+1} at the current time level t^{n+1} by assuming that \hat{H} denotes the known height of the free surface. Field variables at the previous time level, such as \mathbf{u}^n and \mathbf{s}^n , can become undefined near the free surface since there was no fluid flow at time level t^n . To handle this problem, we devise a temporary ALE scheme which maps field variables at the previous time level onto the current time level, similar to [130].

We solve a mesh displacement \mathbf{d}_m such that mesh nodes near the free surface follow the movement of the free surface, whereas, mesh nodes far away from the free surface are stationary:

$$\nabla^2 \mathbf{d}_m = \mathbf{0} \quad \text{in } \Omega_m, \quad (6.36)$$

$$\mathbf{d}_m = \mathbf{0} \quad \text{on } \Gamma_m \setminus \Gamma_{\text{out}}, \quad (6.37)$$

$$\mathbf{n} \cdot \nabla \mathbf{d}_m = \mathbf{0} \quad \text{on } \Gamma_{\text{out}}, \quad (6.38)$$

$$\mathbf{d}_m = (\hat{H}^{n+1} - \hat{H}^n) \mathbf{e}_y \quad \text{on } \Gamma_{\text{fs}}(t^n), \quad (6.39)$$

$$\mathbf{d}_m = \hat{\mathbf{X}}^{n+1} - \hat{\mathbf{X}}^n \quad \text{on } \Gamma_p(t^n), \quad (6.40)$$

where \mathbf{e}_y is the unit vector in y -direction at the free surface. Note that above equations are solved on the whole computational mesh Ω_m , not on the fluid domain Ω , using a similar technique to the fictitious domain method [64]. Eqs. (6.39) and (6.40) can be realized either by using a constraint implemented with distributed Lagrangian multipliers or by using a weak Dirichlet boundary condition [139].

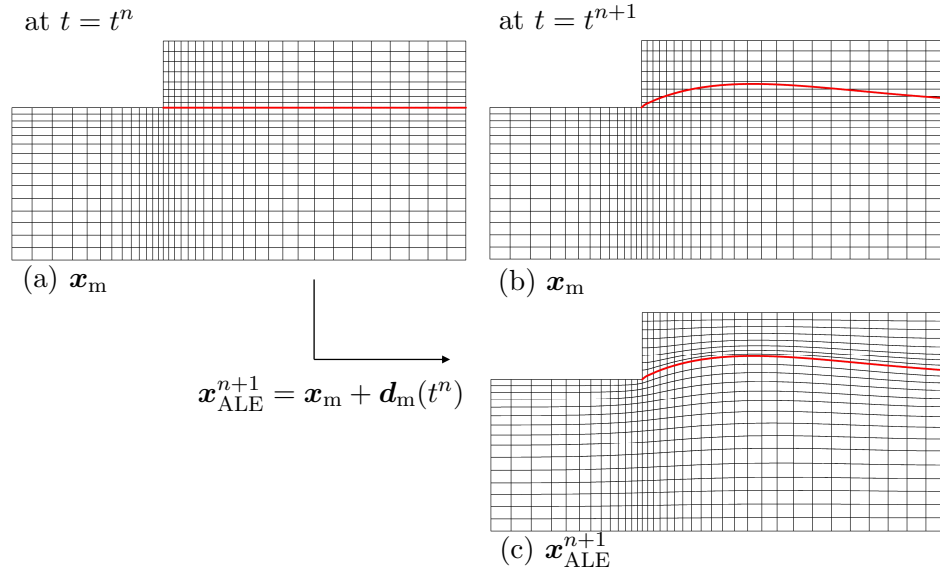
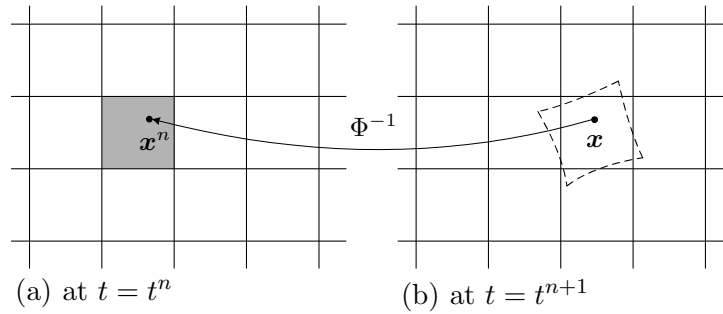


Figure 6.2: First-order temporary ALE scheme.

Figure 6.3: Mapping using a first-order ALE scheme. The dashed element at $t = t^{n+1}$ represents the advection of the shaded element at $t = t^n$.

Basic idea: first-order ALE scheme

An ALE mesh at the current time level $\mathbf{x}_{\text{ALE}}^{n+1}$ is constructed by the following mesh advection equation:

$$\mathbf{x}_{\text{ALE}}^{n+1} = \mathbf{x}_m + \mathbf{d}_m(\mathbf{x}_m, t^n) \quad (6.41)$$

The construction of the ALE mesh $\mathbf{x}_{\text{ALE}}^{n+1}$ is shown in Fig. 6.2. Note that the free surface moves on a fixed Eulerian mesh \mathbf{x}_m .

By using an ALE technique and a semi-implicit Euler scheme, the time discretization of the log-conformation equation (Eq. (6.7)) can be written as:

$$\frac{\mathbf{s}^{n+1}}{\Delta t} + (\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{s}^{n+1} = \frac{\mathbf{s}^n}{\Delta t} + \mathbf{g}((\nabla \mathbf{u}^{n+1})^T, \mathbf{s}^n). \quad (6.42)$$

In Eq. (6.42), the unknown at the current time level, \mathbf{s}^{n+1} , is computed on an Eulerian mesh, whereas, field variables at the previous time level, such as \mathbf{s}^n , are mapped along with the ALE

mesh $\mathbf{x}_{\text{ALE}}^{n+1}$. The mapping of field variables at the previous time level is shown in Fig. 6.3. Let's consider the log-conformation tensor $\mathbf{s}^n = \mathbf{s}(\mathbf{x}^n, t^n)$, for example:

$$\mathbf{s}^n = \mathbf{s}(\mathbf{x}^n(\mathbf{x}), t^n) \quad (6.43)$$

$$= \mathbf{s}(\Phi^{-1}(\mathbf{x}), t^n) \quad (6.44)$$

The mapping Φ is the advection of the ALE mesh given by Eq.(6.41). Hence, the inverse mapping Φ^{-1} can be easily implemented in the finite element context: find the reference coordinates of \mathbf{x} in the ALE mesh $\mathbf{x}_{\text{ALE}}^{n+1}$, then interpolate the values of \mathbf{s} at the same reference coordinates in the computational mesh \mathbf{x}_m . The mesh velocity at the current time level is defined as:

$$\mathbf{u}_m^{n+1}(\mathbf{x}) = \frac{\mathbf{x} - \mathbf{x}^n}{\Delta t} \quad (6.45)$$

at each coordinate \mathbf{x} .

Extension to a second-order scheme

The idea introduced in the previous section can easily be extended to higher-order schemes. In this section, we describe a second-order scheme of a temporary ALE technique. The ALE meshes $\mathbf{x}_{\text{ALE}}^n$ and $\mathbf{x}_{\text{ALE}}^{n+1}$ are constructed as

$$\mathbf{x}_{\text{ALE}}^n = \mathbf{x}_m + \mathbf{d}_m(\mathbf{x}_m, t^{n-1}) \quad (6.46)$$

$$\mathbf{x}_{\text{ALE}}^{n+1} = \mathbf{x}_m + \mathbf{d}_m(\mathbf{x}_m, t^n) \quad (6.47)$$

For the time discretization of the log-conformation equation (Eq. (6.7)), we use a semi-implicit Gear scheme:

$$\frac{3}{2} \frac{\mathbf{s}^{n+1}}{\Delta t} + (\mathbf{u}^{n+1} - \mathbf{u}_m^{n+1}) \cdot \nabla \mathbf{s}^{n+1} = \frac{2\mathbf{s}^n - \frac{1}{2}\mathbf{s}^{n-1}}{\Delta t} + 2\mathbf{g}((\nabla \mathbf{u}^{n+1})^T, \mathbf{s}^n) - \mathbf{g}((\nabla \mathbf{u}^{n+1})^T, \mathbf{s}^{n-1}). \quad (6.48)$$

As in the first order scheme, in Eq. (6.48), the unknown at the current time level, \mathbf{s}^{n+1} , is computed on the Eulerian mesh, whereas, field variables at previous time levels, such as \mathbf{s}^n

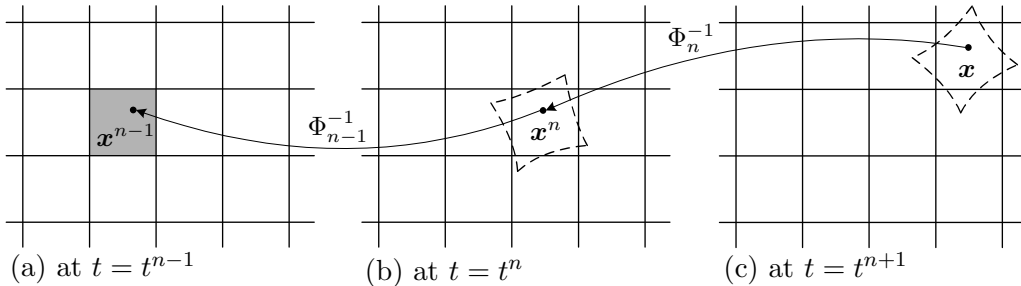


Figure 6.4: Mapping using a second-order ALE scheme.

and \mathbf{s}^{n-1} , are mapped along with the ALE meshes (Fig. 6.4):

$$\mathbf{s}^n = \mathbf{s}(\Phi_n^{-1}(\mathbf{x}), t^n), \quad (6.49)$$

$$\mathbf{s}^{n-1} = \mathbf{s}(\Phi_{n-1}^{-1} \circ \Phi_n^{-1}(\mathbf{x}), t^{n-1}). \quad (6.50)$$

The mesh velocity at the current time level is defined by using second-order backwards differencing (BDF2):

$$\mathbf{u}_m^{n+1}(\mathbf{x}) = \frac{\frac{3}{2}\mathbf{x} - 2\mathbf{x}^n + \frac{1}{2}\mathbf{x}^{n-1}}{\Delta t} \quad (6.51)$$

at each coordinate \mathbf{x} .

At the next time level $t = t^{n+2}$, we delete $\mathbf{x}_{\text{ALE}}^n$, and construct a new temporary ALE mesh $\mathbf{x}_{\text{ALE}}^{n+2}$. Note that $\mathbf{x}_{\text{ALE}}^{n+1}$ is kept unchanged.

In [130], the temporary ALE mesh was constructed with respect to the velocity of particles (in the context of the current chapter, it corresponds to the velocity of the free surface) at previous time levels. By contrast, in this chapter, the ALE meshes are constructed with respect to the position of the free surface and the particle. Since we use a predictor-corrector method (see Sec. 6.3.4) using the position as a primary variable in Eqs. (6.52) and (6.53), it is a natural choice to construct an ALE mesh using a displacement field with respect to the given positions at the previous time level and the current time level. The new scheme is general, and can be easily applied to the problem in [130] as well.

6.3.4 Time integration

We use a semi-implicit stress formulation for the time integration of the evolution equation of the log-conformation tensor \mathbf{s} (Eq. (6.30)) decoupled from the momentum balance [141]. Initially, the viscoelastic polymer stress is set to zero over the whole fluid domain. The initial height of the free surface $H(x, t = 0) = H_0$ is given as an initial condition. We apply the following procedure at every time step.

Step 1. Update the position of the free surface and the particle using a predictor.

$$\hat{H}^{n+1} = 2H^n - H^{n-1}, \quad (6.52)$$

$$\hat{\mathbf{X}}^{n+1} = 2\mathbf{X}^n - \mathbf{X}^{n-1}. \quad (6.53)$$

Note that the update of angular rotations is not necessary since we particle is circular. At the first time step, we use

$$\hat{H}^{n+1} = H^n, \quad (6.54)$$

$$\hat{\mathbf{X}}^{n+1} = \mathbf{X}^n. \quad (6.55)$$

Step 2. Construct a temporary ALE mesh using Eqs. (6.46)-(6.47) for the interpolation of field variables at previous time levels. Note that the ALE meshes are constructed with respect to the predicted free surface, such as \hat{H}^{n-1} , \hat{H}^n and \hat{H}^{n+1} , since field variables are

calculated on these free surface configurations. At the first time step, we use a first-order scheme given in Eq. (6.41).

Step 3. Compute \mathbf{u}^{n+1} , p^{n+1} and \mathbf{G}^{n+1} from the momentum balance, continuity equation and gradient projection equation:

$$\begin{aligned}
& -(\nabla \cdot \mathbf{v}, p^{n+1}) + ((\nabla \mathbf{v})^T, \theta(\nabla \mathbf{u}^{n+1} - (\mathbf{G}^T)^{n+1}) + \boldsymbol{\tau}^{n+1}) \\
& \quad - (\mathbf{v}, (-p^{n+1} \mathbf{I} + \boldsymbol{\tau}^{n+1}) \cdot \mathbf{n})_{\Gamma_{\text{in}}} + (\mathbf{v} \cdot \mathbf{n}, \lambda^{n+1})_{\Gamma_{\text{in}}} \\
& \quad - (\mathbf{v} - (\mathbf{V} + \boldsymbol{\chi} \times (\mathbf{x} - \mathbf{X}^{n+1})), (-p^{n+1} \mathbf{I} + \boldsymbol{\tau}^{n+1}) \cdot \mathbf{n})_{\Gamma_{\text{p}}} \\
& \quad + \kappa M_{km}^{-1}((\mathbf{v} - (\mathbf{V} + \boldsymbol{\chi} \times (\mathbf{x} - \mathbf{X}^{n+1})))\mathbf{n}, \phi_k)_{\Gamma_{\text{p}}} : \\
& \quad (\phi_m, \mathbf{n}(\mathbf{u}^{n+1} - (\mathbf{U}^{n+1} + \boldsymbol{\omega}^{n+1} \times (\mathbf{x} - \mathbf{X}^{n+1})))) \\
& \quad + (\mathbf{u}^{n+1} - (\mathbf{U}^{n+1} + \boldsymbol{\omega}^{n+1} \times (\mathbf{x} - \mathbf{X}^{n+1})))\mathbf{n})_{\Gamma_{\text{p}}} = 0, \quad (6.56)
\end{aligned}$$

$$(q, \nabla \cdot \mathbf{u}^{n+1}) = 0, \quad (6.57)$$

$$(\mathbf{H}, -\nabla \mathbf{u}^{n+1} + (\mathbf{G}^T)^{n+1}) = 0, \quad (6.58)$$

$$(\mu, \mathbf{u}^{n+1} \cdot \mathbf{n})_{\Gamma_{\text{in}}} = -Q, \quad (6.59)$$

where

$$\begin{aligned}
\boldsymbol{\tau}^{n+1} = G\Delta t & (- (\mathbf{u}^{n+1} - \mathbf{u}_{\text{m}}^{n+1}) \cdot \nabla \mathbf{c}^n + (\nabla \mathbf{u}^{n+1})^T \cdot \mathbf{c}^n + \mathbf{c}^n \cdot \nabla \mathbf{u}^{n+1}) \\
& + G\mathbf{h}(\mathbf{c}^n, \Delta t) - G\mathbf{I}. \quad (6.60)
\end{aligned}$$

Note that we use L^2 -projection of $\mathbf{c}^n = \exp(\mathbf{s}^n)$ given by Eq. (6.29), which is solved only in the fluid domain Ω by using XFEM.

Step 4. Solve the log-conformation tensor \mathbf{s}^{n+1} by replacing the evolution equation of the log-conformation tensor (Eq. (6.7)) using a second-order semi-implicit Gear scheme:

$$\begin{aligned}
& (\mathbf{d} + \tau(\mathbf{u}^{n+1} - \mathbf{u}_{\text{m}}^{n+1}) \cdot \nabla \mathbf{d}, \frac{3}{2} \frac{\mathbf{s}^{n+1}}{\Delta t} + (\mathbf{u}^{n+1} - \mathbf{u}_{\text{m}}^{n+1}) \cdot \nabla \mathbf{s}^{n+1}) = \\
& (\mathbf{d} + \tau(\mathbf{u}^{n+1} - \mathbf{u}_{\text{m}}^{n+1}) \cdot \nabla \mathbf{d}, \frac{2\mathbf{s}^n - \frac{1}{2}\mathbf{s}^{n-1}}{\Delta t} + 2\mathbf{g}(\mathbf{G}^{n+1}, \mathbf{s}^n) - \mathbf{g}(\mathbf{G}^{n+1}, \mathbf{s}^{n-1})). \quad (6.61)
\end{aligned}$$

At the first time step, we use a first-order Euler scheme:

$$\begin{aligned}
& (\mathbf{d} + \tau(\mathbf{u}^{n+1} - \mathbf{u}_{\text{m}}^{n+1}) \cdot \nabla \mathbf{d}, \frac{\mathbf{s}^{n+1}}{\Delta t} + (\mathbf{u}^{n+1} - \mathbf{u}_{\text{m}}^{n+1}) \cdot \nabla \mathbf{s}^{n+1}) = \\
& (\mathbf{d} + \tau(\mathbf{u}^{n+1} - \mathbf{u}_{\text{m}}^{n+1}) \cdot \nabla \mathbf{d}, \frac{\mathbf{s}^n}{\Delta t} + \mathbf{g}(\mathbf{G}^{n+1}, \mathbf{s}^n)). \quad (6.62)
\end{aligned}$$

Step 5. Update the free surface position by solving Eq. (6.31) for the given fluid velocity at the predicted free surface. Note that the discretization of the one-dimensional height problem can be chosen independently from the fluid element. We have tested one quadratic P_2 element, one linear P_1 element, and two linear P_1 elements for the discretization of the

height function for one bi-quadratic Q_2 velocity element. We do not detect any significant difference between these interpolations. In simulations, we use two linear P_1 line elements for one Q_2 fluid velocity element.

Step 6. Update the particle position using a corrector:

$$\mathbf{X}^{n+1} = \mathbf{X}^n + \frac{1}{2}(\mathbf{U}^{n+1} + \mathbf{U}^n)\Delta t. \quad (6.63)$$

At the first time step, we use

$$\mathbf{X}^{n+1} = \mathbf{X}^n + \mathbf{U}^{n+1}\Delta t. \quad (6.64)$$

6.4 Extrudate swell

We consider a two-dimensional extrusion of an upper-convected Maxwell fluid, which is identical to the Giesekus model with $\alpha = 0$. Note that the Newtonian viscosity is neglected. A domain with a slit length 2, half-width 1 and the length of the extrudate sheet 3 is used for simulations, as proposed by Crochet and Keunings [150]. We apply flow rate $Q = 1$ at the inflow boundary Γ_{in} , which implies a unit mean velocity. The maximum shear rate on the upstream wall $\dot{\gamma}_w$ is 3 for the given problem. The Deborah number is defined by $\text{De} = \lambda\dot{\gamma}_w$. We calculate the swell ratio H_{max}/H_0 at steady-state for different Deborah numbers. In order to judge whether a steady-state is obtained, we solve the problem until the height of the free surface does not change up to an order of $O(10^{-7})$, i.e. $|H(x, t^{n+1}) - H(x, t^n)| < 10^{-7}$ for all nodes x .

The problem is solved using four different meshes $M1$, $M2$, $M3$ and $M4$, which are summarized in Table 6.2. Fig. 6.5 shows the free surface profile at steady-state for Deborah number $\text{De} = 0.75$ using the mesh $M1$. Note that the free surface is immersed on a fixed Eulerian mesh for XFEM. The swell ratios for different Deborah numbers are given in Table 6.3. The results are also compared with those of a moving boundary-fitted mesh (MBFM) using a conventional ALE scheme, and those of Crochet and Keunings [150]. For the MBFM problem, we use a fine mesh with 9900 elements, which is finer than the mesh $M4$, as a reference. All the values of swell ratios show quite good agreements up to Deborah number $\text{De} = 0.75$. For these cases, the results of $M3$ and $M4$ are exactly the same up to 3 decimal figures. Note that, here we increase the number of elements for mesh refinements while maintaining the element size distribution similar to the mesh $M1$ which is shown in Fig. 6.5. We also obtained similar mesh convergence behavior only by shifting elements towards the die exit using the same number of elements of the mesh $M1$. Crochet and Keunings [150] seem to obtain the correct swell ratios in this way using a mesh with minimal number of triangular elements. Note that the swell ratio is determined mainly by the stress distribution at the die exit and

Table 6.2: Meshes used for the simulation of extrudate swell.

	$M1$	$M2$	$M3$	$M4$
Number of elements	1520	3233	6080	9187

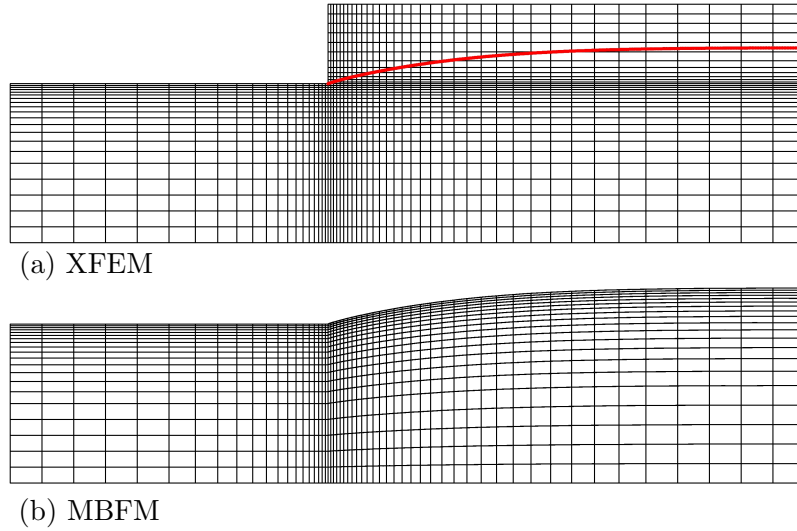


Figure 6.5: Free surface profile using (a) XFEM and (b) MBFM for $De=0.75$.

Table 6.3: Swell ratio for different Deborah numbers

	XFEM				MBFM	Ref. [150]
	$M1$	$M2$	$M3$	$M4$		
De=0.00	1.189	1.188	1.188	1.188	1.188	1.188
De=0.25	1.168	1.168	1.168	1.168	1.169	1.169
De=0.50	1.185	1.187	1.189	1.189	1.191	1.189
De=0.60	1.199	1.201	1.203	1.203	1.207	1.205
De=0.75	1.225	1.227	1.232	1.232	1.237	1.236
De=1.00	1.277	1.283	1.287	1.290	unstable	-
De=1.25	1.343	1.351	1.358	1.362	unstable	-
De=1.50	1.421	1.433	1.442	1.448	unstable	-
De=1.70	1.491	1.508	1.518	1.526	unstable	-

stresses decay on the downstream of the extrudate. Hence the refined elements along the horizontal line at $y = H_0, x > 0$ for XFEM are not really necessary. It is caused by a particular structured mesh used. Using an unstructured mesh, which is only refined at the die exit, might be preferable to get an optimal computing performance.

For $De \geq 1$, our conventional ALE scheme with moving boundary-fitted mesh failed: the solutions become unstable and eventually diverge. Contrary to that, the proposed XFEM gives stable solution up to $De = 1.7$, which shows better numerical stability compared with MBFM. Note that, for $De \geq 1.8$, the simulations using XFEM also become unstable. We think that the superior stability of XFEM over MBFM might be related to the angular mesh resolution at the die exit and skewed elements as the swell ratio increases, as schematically shown in Fig. 6.6. For MBFM, there are only 2 elements at the die exit since we use a particular structured mesh with quadrilateral elements. For XFEM, there is one more element at the die exit, and the number of degrees of freedom increases as the surface height increases.

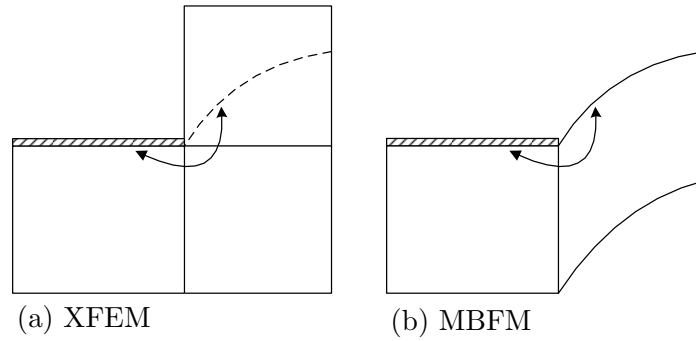


Figure 6.6: Schematic description of the angular mesh resolution at the die exit.

Also, the shape of the elements remains square, which is optimal. Hence we may capture the stress singularity better at the die exit using XFEM. By using adaptive remeshing techniques, the instability issue in MBFM problems can likely be resolved. Using triangular elements might also be helpful. However, this is beyond the scope of this thesis.

Note that, for $De \geq 1.5$, the given geometry of the extrudate length seems a little short; the outflow boundary is still in the developing flow region. To get a precise value of the swell ratio for $De \geq 1.5$, a longer computational domain should be used.

It is worthy to note the computational time as well for a fair comparison between XFEM and MBFM. For the given problem of extrudate swell, a moving boundary-fitted mesh can provide the benefit of minimal computational time because elements can be moved only normal to the flow direction without remeshing if the swell ratio is not too high. In actual simulations, the computational time of MBFM is about 20% less than that of XFEM using the mesh M_4 , since XFEM requires additional computational costs of ALE mesh generation, mapping of field variables, subdomain integration etc. However, boundary-fitted meshes are not always preferable. For complicated flow geometries, caused e.g. by high swell ratio or irregular free surface profiles, a frequent remeshing is needed to avoid mesh distortion, and the solution should be projected from the previous mesh to the current mesh. However the generation of a boundary-fitted mesh is not an easy task, especially in three-dimensional problems. For these cases, the proposed extended finite element method, with a regular mesh which is independent in time, is easier to implement.

6.5 Particle migration in extrusion flow

6.5.1 Problem description

Initially a particle is positioned in the upstream channel of a half-height 1 and length 3, and the length of the extrudate is 4, i.e. $H_0 = 1$, $L_w = 3$, $L_{fs} = 4$ (see Fig. 6.1). We apply flow rate $Q = 1$ at the inflow boundary Γ_{in} , which implies a unit mean velocity ($U_{ave} = 1$). The Weissenberg number is defined by $Wi = \lambda \dot{\gamma}_{ave} = \lambda U_{ave} / H_0$. Note that $De = \lambda \dot{\gamma}_w = 3Wi$ for the upper-convected Maxwell model ($\alpha = 0$), where $\dot{\gamma}_w$ is the maximum shear rate on the upstream wall. Otherwise stated, the radius of the particle $a = 0.1$ and the initial position of

Table 6.4: Meshes used for the simulation.

	M0	M1	M2	M3	M4
Number of elements	2310	8100	18820	35665	50965

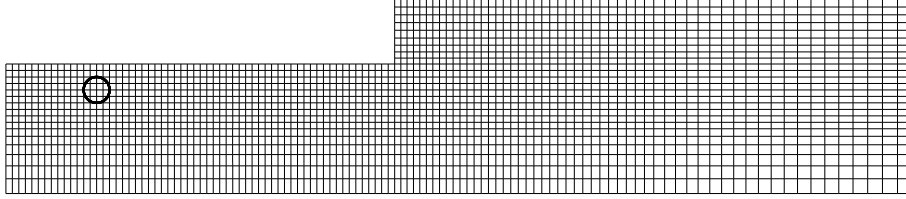


Figure 6.7: The base mesh M0 from which the other meshes are derived.

the center of the particle $\mathbf{X}_0 = (-2.3, 0.8)$. The mobility parameter α of the Giesekus model is fixed to 0.1.

6.5.2 Mesh convergence

We solve the problem using four different meshes M1, M2, M3 and M4, which are summarized in Table 6.4, and the base mesh M0 is shown in Fig. 6.7 for a better understanding of the mesh used. For a channel region close to the wall ($x \leq 0, y \geq 0.5$), where the particle moves, we use square elements of equal size. The element size increases gradually to the boundaries of the domain. If we do not use equal-sized square elements near the particle in the channel region, it may cause a numerical instability problem, especially if the particle is close to the wall. In the extrudate region, it does not really matter since stresses decay on the downstream of the extrudate.

Fig. 6.8 shows the x -directional translational velocity of the particle as a function of time using different meshes. The result of the mesh M1 deviates a little from the results of M3 and M4, which are almost indistinguishable in Fig. 6.8. We also observed similar convergence behavior in the y -directional translational velocity and the angular velocity of the particle. For all our simulations, we use the mesh M3.

6.5.3 Particle motion

Fig. 6.9 shows the movement of the particle and free surface near the die exit for $Wi = 0.2$. The blue line represents the undisturbed free surface, which means the free surface profile without the particle, for a comparison with the free surface in the presence of the particle. The presence of the particle disturbs the free surface profile significantly, and the particle moves above the undisturbed free surface. In Fig. 6.9, the mesh M0 is shown for a better understanding of the movement of the particle and free surface in a fixed Eulerian mesh. Note that the mesh M3 is used for actual simulation. The movement of the particle and free surface is also shown in Fig. 6.10 with the contours of the polymer stress component τ_{xx} .

To quantify the migration of the particle in an extrusion flow, we define a separation

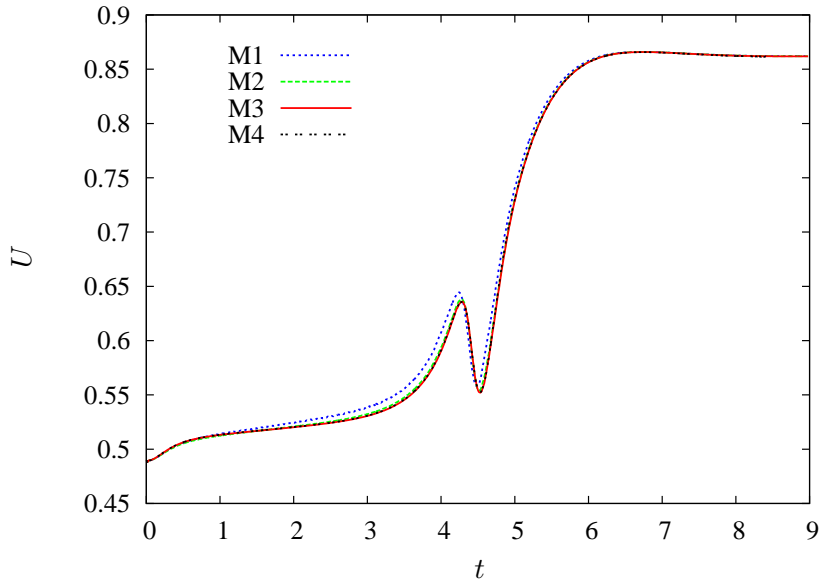


Figure 6.8: The x -directional velocity of the particle as a function of time using different meshes.

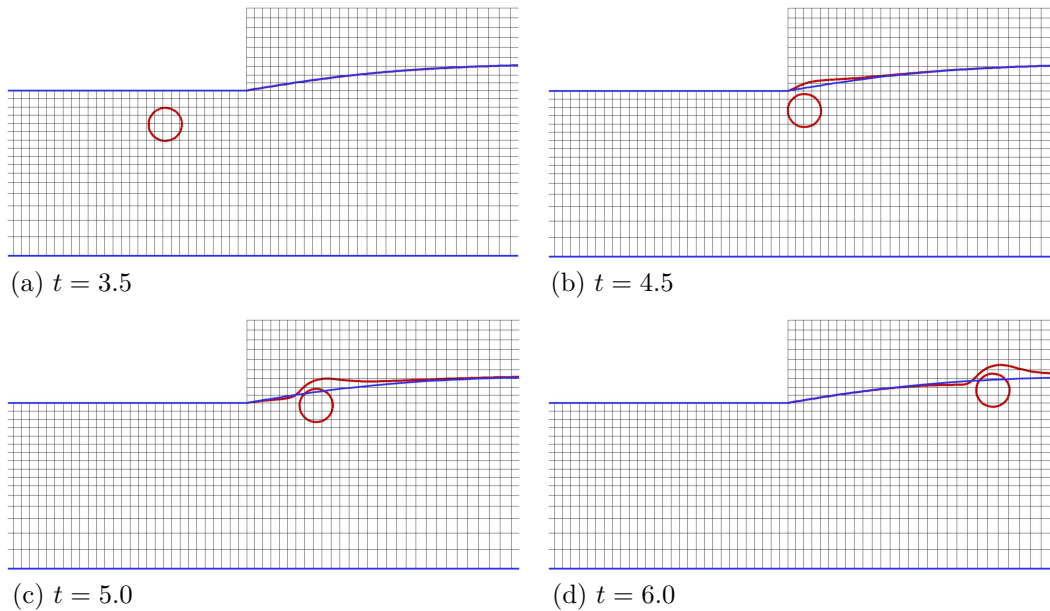


Figure 6.9: Particle migration near the die exit of an extrusion flow for $Wi = 0.2$. The blue line represents the undisturbed free surface, and the red line represents the free surface with the particle. Note that the mesh M0 is shown here for a better understanding. In actual simulation, the mesh M3 is used.

function s which represents the distance between the particle and the undisturbed free surface at the steady-state $\bar{H}(x) = H(x, t \rightarrow \infty)$:

$$s(x) = Y(x) + a - \bar{H}(x), \quad (6.65)$$

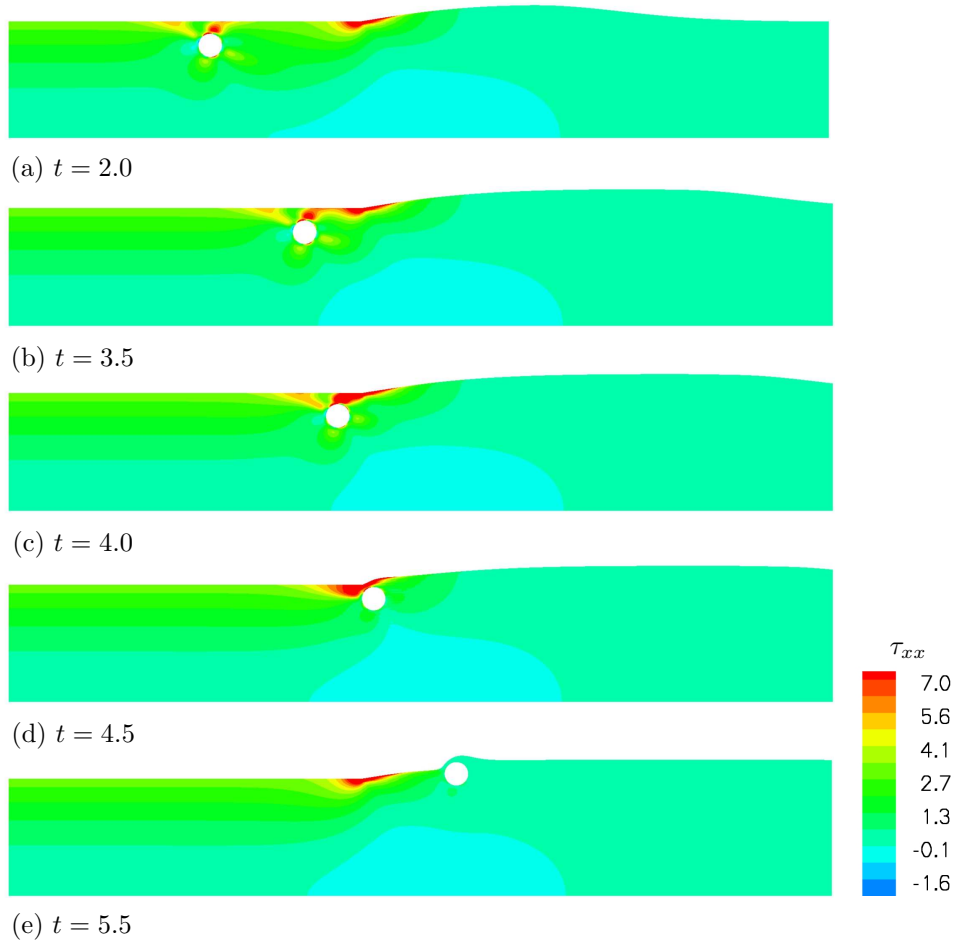
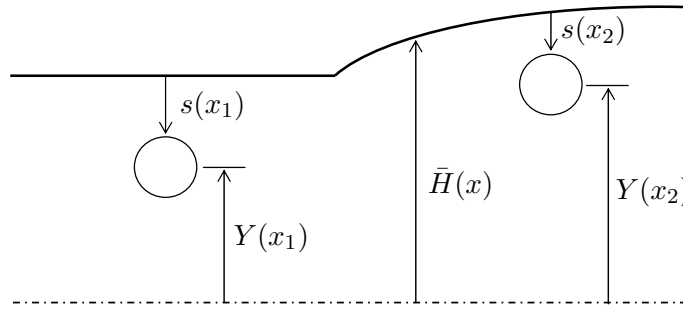
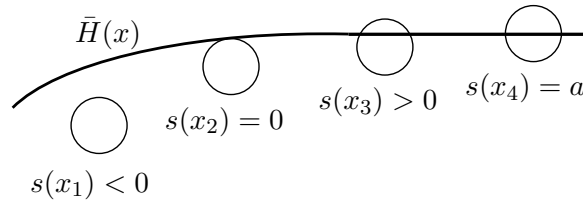


Figure 6.10: Snapshots of the motion of the particle and free surface for $Wi = 0.2$. Contours are the polymer stress component τ_{xx} .

where $Y(x)$ is the vertical position of the center of the particle at x and a is the radius of the particle. The separation function s is schematically shown in Fig. 6.11. If the top of the particle is below $\bar{H}(x)$, s is negative; if the top of the particle is above $\bar{H}(x)$, s is positive; if the top of the particle touches $\bar{H}(x)$, $s = 0$ (Fig. 6.12). Note that we solve an additional problem to obtain the undisturbed surface profile at the steady-state using XFEM as explained in Sec. 6.4 which is based on [137].

Fig. 6.13 shows the trajectory of the center of the particle and the undisturbed free surface at the steady-state for $Wi = 0.2$. The corresponding separation function $s(x)$ is shown in Fig. 6.14. Note that the radius of the particle $a = 0.1$ for the given problem. At the initial position of the particle, $s = -a = -0.1$ which means that the gap between the particle and the wall is the same as the radius of the particle. In the channel region $x \leq 0$, the separation s must be negative since the particle can not move above the wall. In the channel region, the absolute value of s increases, which means that the particle migrates towards the center of the channel. As already shown in Fig. 6.9, the particle moves above the undisturbed free surface after passing the die exit ($x = 0$). Hence the separation s becomes positive, then reaches a

Figure 6.11: Schematic description of the separation function s .Figure 6.12: Typical values of the separation function s .

positive terminal value as the motion of the particle reaches a quasi-steady condition.

6.5.4 Weissenberg number

We investigate the effect of the Weissenberg number on the particle migration. Fig. 6.15 shows the separation of the particle for various Weissenberg numbers. Since the particle migrates towards the center of the channel in the channel region, we also show two Newtonian cases which make a limiting band of viscoelastic cases in the channel. Note that the particle does not migrate in a Newtonian fluid. In the extrudate region ($x > 0$), all viscoelastic cases are outside of the Newtonian band. Moreover, as the Weissenberg number increases, the separation s has smaller positive values, which means that the particle migrates inwards to the centerline of the extrudate. Hence we can conclude that the particle center moves away from the free surface as the Weissenberg number increases.

6.5.5 Particle size

Fig. 6.16 shows the separation of the particle for various particle radii. Note that the initial separation is fixed to $s_0 = -0.1$. The initial y -coordinate of the particle center is $Y_0 = H_0 + s_0 - a = 0.9 - a$. In the channel region, as the particle size increases, the particle migrates further towards the center of the channel. After passing the die exit, a bigger particle moves further above the undisturbed free surface, having a larger positive s -values. Hence the particle center moves towards the free surface as the particle size increases.

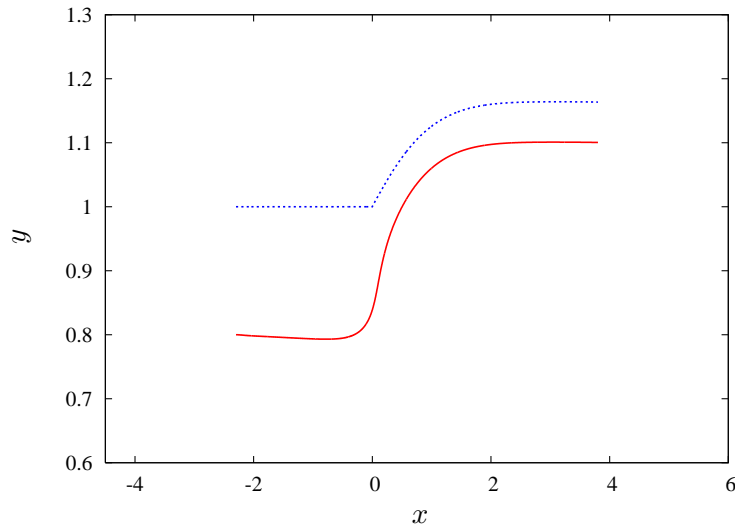


Figure 6.13: Trajectory of the center of the particle for $Wi = 0.2$. The dotted blue line represents the undisturbed free surface at the steady-state $\bar{H}(x)$.

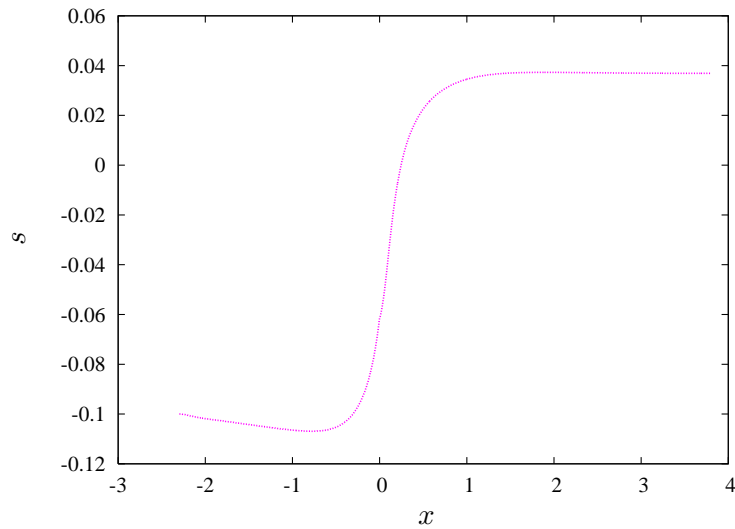


Figure 6.14: The separation of the particle from the undisturbed free surface at the steady-state for $Wi = 0.2$.

6.6 Conclusions

An extended finite element method has been presented for the simulation of the extrusion of particle filled viscoelastic fluids. In this method, the time dependent fluid domain is included in a larger time independent mesh domain. To cope with the movement of the free surface and particle, a temporary ALE scheme is incorporated, which defines a mapping of field variables at previous time levels on the computational mesh at the current time level. The main advantage of the proposed method is that the movement of the free surface and particle can be simulated on a fixed Eulerian mesh without any need of remeshing. The no-

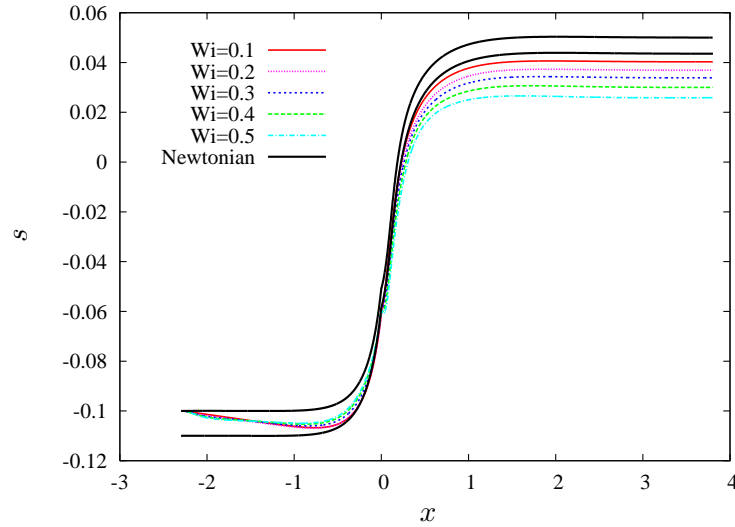


Figure 6.15: The separation of the particle for various Weissenberg numbers. As the Weissenberg number increases, the particle migrates inward from the free surface.

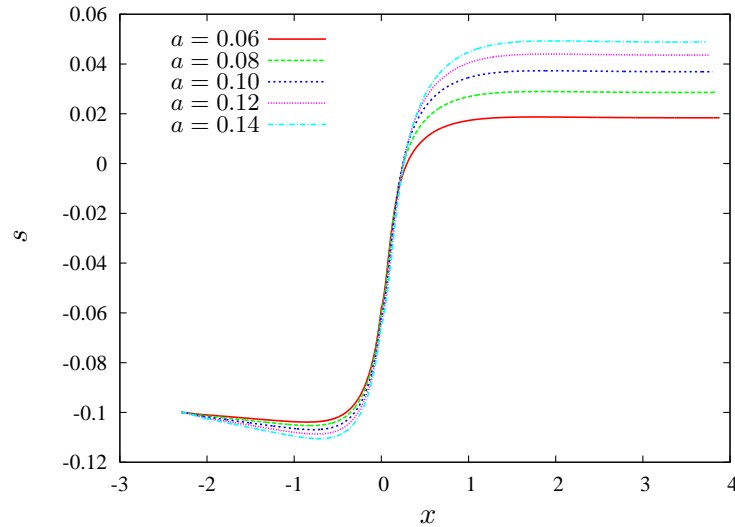


Figure 6.16: The separation of the particle for various particle radii. As the particle size increases, the particle moves further above the undisturbed free surface.

slip boundary condition on the particle surface is realized by using a weak boundary condition on the particle surface, which is equivalent to adding a positive definite stabilizing term in the momentum balance.

To validate the method with the presence of the free surface, the swell ratios of an upper-convected Maxwell fluid for various Deborah numbers are compared with those of the moving boundary-fitted mesh problems of the conventional ALE technique. Our method can provide similar accuracy to the boundary-fitted mesh problems. Moreover, the method gives better numerical stability for higher Deborah numbers.

We have also presented the extrusion of the Giesekus fluid with a freely suspended particle. The presence of the particle disturbs the stress distribution and the free surface profile significantly. To quantify the migration of a particle in extrusion flow, a separation function is proposed, which represents the distance between the particle and the undisturbed free surface at the steady-state. We found that the particle moves away from the free surface as the Weissenberg number increases. As the particle size increases, the particle moves towards the free surface. Future work will be focused on simulations of the motion of multiple particles in extrusion flow.

Chapter 7

Conclusions and prospects

7.1 Conclusions

In the thesis, an extended finite element method (XFEM) has been presented for the direct numerical simulation of particulate complex flows, e.g. viscoelastic suspending media, two-phase flows, and with the presence of a free surface. To cope with the movement of particles and/or free surfaces, a temporary arbitrary Lagrangian-Eulerian (ALE) scheme has been developed to define a mapping of field variables at previous time levels onto the computational mesh at the current time level. The main advantage of the method is that the movement of particles can be simulated on a fixed Eulerian mesh without any need of remeshing.

The proposed method has been applied to various particulate flow problems: the flow of a viscoelastic fluid around a stationary cylinder in Chapter 2, the particle migration in circular Couette flow of a viscoelastic fluid in Chapter 3, the dynamics of particles suspended in two-phase flows in Chapter 4, the alignment of particles in confined shear flow of a viscoelastic fluid in Chapter 5, and the migration of a particle in extrusion flow in Chapter 6.

The accuracy and convergence of the method have been verified by comparing with boundary-fitted mesh problems, using the flow of a viscoelastic fluid around a stationary cylinder as a benchmark problem. The results are also compared with fictitious domain methods. Our method shows similar accuracy compared with boundary-fitted mesh problems, and significant improvement of local accuracy around the rigid body compared with fictitious domain methods.

The particle migration in circular Couette flow of a Giesekus fluid is presented. In particular, the effects of the initial particle position, the Weissenberg number, the mobility parameter of the Giesekus model and the particle size are investigated. The particle migrates to a stabilized radial position near the outer cylinder regardless of its initial position. As the

Weissenberg number increases, the particle migrates more rapidly towards the outer cylinder, and the stabilized radial position is shifted towards the outer cylinder. With increasing mobility parameter of the Giesekus model, the rate of migration decreases. Increasing the mobility parameter can reverse the direction of migration if the initial position of the particle is very close to the inner cylinder. With increasing particle size, the particle migrates more rapidly towards the outer cylinder.

A model for the dynamics of particles suspended in two-phase flows is presented by coupling the Cahn-Hilliard theory with the extended finite element method. To demonstrate and validate the technique, the dynamics of a single particle at a fluid-fluid interface is studied. The initial equilibrium state is disturbed by applying a constant force on the particle for a certain time duration, then the particle moves to its equilibrium position under the action of surface tension. In particular, the effects of interfacial thickness, surface tension, particle size and viscosity ratio of the two fluids are investigated. As the interfacial thickness increases, the surface tension increases, the particle size decreases, or the viscosity ratio decreases, the particle moves more rapidly towards its equilibrium position. The movement of a particle passing through multiple layers of fluids is also presented to demonstrate the wide applicability of the method to problems associated with the complex morphology of the fluids.

The two-particle interactions in confined shear flow of a viscoelastic fluid are presented. There are three different regimes of particle motions according to initial separations of particles. For small initial separations, two particles repel each other and their trajectories are reversed; for large initial separations, two particles keep going in their original directions and pass each other, similarly to the particle motions in Stokes flow problems. For intermediate initial separations, two particles come together and reach final fixed positions, then they rotate with a non-zero terminal angular velocity. In the intermediate regime, the final positions of particles are independent of their initial positions.

Motivated by the two-particle interaction problem, the alignment of multiple particles is analyzed quantitatively. The maximum obtainable length of a string of particles is limited for a certain fluid rheology. As the fluid elasticity increases, particles can form longer strings. Once particles form a string, the final state is independent of the initial particle positions and the histories to reach the steady-state. Moderate wall confinement promotes the alignment of particles, however, too strong confinement hinders the alignment by enhancing repulsive interaction between particles. The steady-state angular velocities of particles are compared with respect to the length of strings. If particles can form sufficiently long strings, the steady-state angular velocities of the two end-particles do not change significantly, and those of the non end-particles increase, as the string length increases. In a given string, the angular velocity of the two end-particles is faster than those of the particles in between. We have also presented the steady-state interparticle distance between two neighboring particles in a string. As the string length increases, the interparticle distance increases.

To validate the proposed method with the presence of a free surface, the swell ratios of an upper-convected Maxwell fluid for various Deborah numbers are compared with those of the moving boundary-fitted mesh problems of the conventional ALE technique. Our method can provide similar accuracy to the boundary-fitted mesh problems for low Deborah numbers.

Moreover, the method gives better numerical stability for higher Deborah numbers.

The simulation of the extrusion of a particle filled viscoelastic fluid is carried out by using a combined XFEM scheme both for the particle and the free surface. The migration of a particle near the free surface of an extrusion flow is investigated. The presence of the particle disturbs the stress distribution and the free surface profile significantly. The particle moves away from the free surface as the Weissenberg number increases. As the particle size increases, the particle moves towards the free surface.

7.2 Prospects

The proposed method is quite general, and can be extended to diverse particulate flow problems, e.g. electrophoresis, the inclusion of inertial effects, and non-circular particle geometries such as elliptic particles. These problems are already taken into account and worked quite well compared with known results in the literature.

To have a realistic prediction of the motion of particles, three-dimensional simulations are necessary. The proposed method can be extended to three-dimensional simulations without any loss of generality, however it requires heavy computational loads. Parallel computation techniques and fast iterative solvers are highly recommended to manage computational costs. The weak boundary condition on the particle surface, introduced in Chapter 5, can be preferable over the traditional Lagrangian multiplier technique since it circumvents the LBB condition and seems to be well suited for fast iterative solvers.

Simulations of concentrated suspensions might be interesting, which can be applied to Pickering emulsions [118], minimizing flow instabilities with suspended particles [145], etc. However, it may require a collision scheme to avoid particle overlapping. Some suggestions for challenging particulate flow simulations are Brownian motion in colloids, deformable particles, particles in phase separating fluids, structure formation of particles at a fluid-fluid interface, and particle-drop interaction.

References

- [1] R.P. Chhabra. *Bubbles, Drops, and Particles in Non-Newtonian Fluids*. Taylor & Francis, 2nd edition, 2007.
- [2] G. B. Jeffery. The motion of ellipsoidal particles immersed in a viscous fluid. *Proceedings of the Royal Society A*, 102:161–179, 1922.
- [3] B.J. Trevelyan and S.G. Mason. Particle motions in sheared suspensions. I. Rotations. *Journal of Colloid Science*, 6:354–367, 1951.
- [4] R.G. Cox, I.Y.Z. Zia, and S.G. Mason. Particle motions in sheared suspensions. XXV. Streamlines around cylinders and spheres. *Journal of Colloid and Interface Science*, 27:7–18, 1968.
- [5] C.R. Robertson and A. Acrivos. Low Reynolds number shear flow past a rotating circular cylinder. Part 1. Momentum transfer. *Journal of Fluid Mechanics*, 40:685–704, 1970.
- [6] C.J. Lin, J.H. Peery, and W.R. Schowalter. Simple shear flow round a rigid sphere: inertial effects and suspension rheology. *Journal of Fluid Mechanics*, 44:1–17, 1970.
- [7] C.A. Kossack and A. Acrivos. Steady simple shear flow past a circular cylinder at moderate Reynolds numbers: a numerical solution. *Journal of Fluid Mechanics*, 66:353–376, 1974.
- [8] G.G. Poe and A. Acrivos. Closed-streamline flows past rotating single cylinders and spheres: inertia effects. *Journal of Fluid Mechanics*, 72:605–623, 1975.
- [9] G. D’Avino, M.A. Hulsen, F. Snijkers, J. Vermant, F. Greco, and P.L. Maffettone. Rotation of a sphere in a viscoelastic liquid subjected to shear flow. Part I: Simulation results. *Journal of Rheology*, 52:1331–1346, 2008.
- [10] F. Snijkers, G. D’Avino, P.L. Maffettone, F. Greco, M.A. Hulsen, and J. Vermant. Rotation of a sphere in a viscoelastic liquid subjected to shear flow. Part II: Experimental results. *Journal of Rheology*, 53:459–480, 2009.
- [11] F. Snijkers, G. D’Avino, P.L. Maffettone, F. Greco, M.A. Hulsen, and J. Vermant. Effect of viscoelasticity on the rotation of a sphere in shear flow. *Journal of Non-Newtonian Fluid Mechanics*, 166:363–372, 2011.

-
- [12] G. D'Avino, G. Cicale, M.A. Hulsen, F. Greco, and P.L. Maffettone. Effects of confinement on the motion of a single sphere in a sheared viscoelastic liquid. *Journal of Non-Newtonian Fluid Mechanics*, 157:101–107, 2009.
- [13] G. Segré and A. Silberberg. Radial particle displacements in Poiseuille flow of suspensions. *Nature*, 189:209–210, 1961.
- [14] G. Segré and A. Silberberg. Behaviour of macroscopic rigid spheres in Poiseuille flow. Part 1. Determination of local concentration by statistical analysis of particle passages through crossed light beams. *Journal of Fluid Mechanics*, 14:115–135, 1962.
- [15] G. Segré and A. Silberberg. Behaviour of macroscopic rigid spheres in Poiseuille flow. Part 2. Experimental results and interpretation. *Journal of Fluid Mechanics*, 14:136–157, 1962.
- [16] R.G. Cox and H. Brenner. The lateral migration of solid particles in Poiseuille flow - I. Theory. *Chemical Engineering Science*, 23:147–173, 1968.
- [17] B.P. Ho and L.G. Leal. Inertial migration of rigid spheres in two-dimensional unidirectional flows. *Journal of Fluid Mechanics*, 65:365–400, 1974.
- [18] P. Vasseur and R.G. Cox. The lateral migration of a spherical particle in two-dimensional shear flows. *Journal of Fluid Mechanics*, 78:385–413, 1976.
- [19] R.G. Cox and S.G. Mason. Suspended particles in fluid flow through tubes. *Annual Review of Fluid Mechanics*, 3:291–316, 1971.
- [20] L.G. Leal. *Laminar Flow and Convective Transport Processes*. Butterworth-Heinemann, 1992.
- [21] J. Feng, H.H. Hu, and D.D. Joseph. Direct simulation of initial value problems for the motion of solid bodies in a Newtonian fluid. Part 2. Couette and Poiseuille flows. *Journal of Fluid Mechanics*, 277:271–301, 1994.
- [22] P. Vasseur and R.G. Cox. The lateral migration of spherical particles sedimenting in a stagnant bounding fluid. *Journal of Fluid Mechanics*, 80:561–591, 1977.
- [23] J. Feng, H.H. Hu, and D.D. Joseph. Direct simulation of initial value problems for the motion of solid bodies in a Newtonian fluid. Part 1. Sedimentation. *Journal of Fluid Mechanics*, 261:95–134, 1994.
- [24] A. Karnis and S.G. Mason. Particle motions in sheared suspensions. XIX. Viscoelastic media. *Transactions of the society of rheology*, 10:571–592, 1966.
- [25] D.J. Highgate and R.W. Whorlow. End effects and particle migration effects in concentric cylinder rheometry. *Rheologica Acta*, 8:142–151, 1969.
- [26] F. Gauthier, H.L. Goldsmith, and S.G. Mason. Particle motions in non-Newtonian media. I. Couette flow. *Rheologica Acta*, 10:344–364, 1971.

-
- [27] F. Gauthier, H.L. Goldsmith, and S.G. Mason. Particle motions in non-Newtonian media. II. Poiseuille flow. *Transactions of the society of rheology*, 15:297–330, 1971.
- [28] M.A. Jefri and A.H. Zahed. Elastic and viscous effects on particle migration in plane-Poiseuille flow. *Journal of Rheology*, 33:691–708, 1989.
- [29] J. Feng and D.D. Joseph. The motion of solid particles suspended in viscoelastic liquids under torsional shear. *Journal of Fluid Mechanics*, 324:199–222, 1996.
- [30] B.M. Lormand and R.J. Phillips. Sphere migration in oscillatory Couette flow of a viscoelastic fluid. *Journal of Rheology*, 48:551–570, 2004.
- [31] F. Snijkers. *Effects of viscoelasticity on particle motions in sheared suspensions*. PhD thesis, Katholieke Universiteit Leuven, Leuven, Belgium, 2009.
- [32] B.P. Ho and L.G. Leal. Migration of rigid spheres in a two-dimensional unidirectional shear flow of a second-order fluid. *Journal of Fluid Mechanics*, 76:783–799, 1976.
- [33] L.G. Leal. The motion of small particles in non-Newtonian fluids. *Journal of Non-Newtonian Fluid Mechanics*, 5:33–78, 1979.
- [34] P. Brunn. The motion of rigid particles in viscoelastic fluids. *Journal of Non-Newtonian Fluid Mechanics*, 7:271–288, 1980.
- [35] L.G. Leal. Particle motions in a viscous fluid. *Annual Review of Fluid Mechanics*, 12:435–476, 1980.
- [36] J. Mewis and N.J. Wagner. Current trends in suspension rheology. *Journal of Non-Newtonian Fluid Mechanics*, 157:147–150, 2009.
- [37] P.Y. Huang, J. Feng, H.H. Hu, and D.D. Joseph. Direct simulation of the motion of solid particles in Couette and Poiseuille flows of viscoelastic fluids. *Journal of Fluid Mechanics*, 343:73–94, 1997.
- [38] J. Feng, P.Y. Huang, and D.D. Joseph. Dynamic simulation of sedimentation of solid particles in an Oldroyd-B fluid. *Journal of Non-Newtonian Fluid Mechanics*, 63:63–88, 1996.
- [39] G. D’Avino, T. Tuccillo, P.L. Maffettone, F. Greco, and M.A. Hulsen. Numerical simulations of particle migration in a viscoelastic fluid subjected to shear flow. *Computers & Fluids*, 39:709–721, 2010.
- [40] G. D’Avino, P.L. Maffettone, F. Greco, and M.A. Hulsen. Viscoelasticity-induced migration of a rigid sphere in confined shear flow. *Journal of Non-Newtonian Fluid Mechanics*, 165:466–474, 2010.
- [41] M.M. Villone, G. D’Avino, M.A. Hulsen, F. Greco, and P.L. Maffettone. Numerical simulations of particle migration in a viscoelastic fluid subjected to Poiseuille flow. *Computers & Fluids*, 42:82–91, 2011.

- [42] A. Karnis, H.L. Goldsmith, and S.G. Mason. The kinetics of flowing suspensions. I. Concentrated suspensions of rigid particles. *Journal of Colloid and Interface Science*, 22:531–553, 1966.
- [43] E.C. Eckstein, D.G. Bailey, and A.H. Shapiro. Self-diffusion of particles in shear flow of a suspension. *Journal of Fluid Mechanics*, 79:191–208, 1977.
- [44] F. Galada-Maria and A. Acrivos. Shear-induced structure in a concentrated suspension of solid spheres. *Journal of Rheology*, 24(6):799–814, 1980.
- [45] D. Leighton and A. Acrivos. The shear-induced migration of particles in concentrated suspensions. *Journal of Fluid Mechanics*, 181:415–439, 1987.
- [46] A.L. Graham, S.A. Altobelli, E. Fukushima, L.A. Mondy, and T.S. Stephens. Note: NMR imaging of shear-induced diffusion and structure in concentrated suspensions undergoing Couette flow. *Journal of Rheology*, 35(1):191–201, 1991.
- [47] J.R. Abbott, N. Tetlow, A.L. Graham, S.A. Altobelli, E. Fukushima, L.A. Mondy, and T.S. Stephens. Experimental observations of particle migration in concentrated suspensions: Couette flow. *Journal of Rheology*, 35(5):773–795, 1991.
- [48] N. Phan-Thien, A.L. Graham, S.A. Altobelli, J.R. Abbott, and L.A. Mondy. Hydrodynamic particle migration in a concentrated suspension undergoing flow between rotating eccentric cylinders. *Industrial & Engineering Chemistry Research*, 34:3187–3194, 1995.
- [49] R.J. Phillips, R.C. Armstrong, R.A. Brown, A.L. Graham, and J.R. Abbott. A constitutive equation for concentrated suspensions that accounts for shear-induced particle migration. *Physics of Fluids A*, 4(1):30–40, 1992.
- [50] T.-W. Pan and R. Glowinski. Direct simulation of the motion of neutrally buoyant circular cylinders in plane Poiseuille flow. *Journal of Computational Physics*, 181:260–279, 2002.
- [51] J. Michele, R. Pötzold, and R. Donis. Alignment and aggregation effects in suspensions of spheres in non-Newtonian media. *Rheologica Acta*, 16:317–321, 1977.
- [52] M.K. Lyon, D.W. Mead, R.E. Elliott, and L.G. Leal. Structure formation in moderately concentrated viscoelastic suspensions in simple shear flow. *Journal of Rheology*, 45:881–890, 2001.
- [53] D. Won and C. Kim. Alignment and aggregation of spherical particles in viscoelastic fluid under shear flow. *Journal of Non-Newtonian Fluid Mechanics*, 117:141–146, 2004.
- [54] R. Scirocco, J. Vermant, and J. Mewis. Effect of the viscoelasticity of the suspending fluid on structure formation in suspensions. *Journal of Non-Newtonian Fluid Mechanics*, 117:183–192, 2004.

-
- [55] R. Pasquino, F. Snijers, N. Grizzuti, and J. Vermant. The effect of particle size and migration on the formation of flow-induced structures in viscoelastic suspensions. *Rheologica Acta*, 49:993–1001, 2010.
- [56] W.R. Hwang and M.A. Hulsen. Structure formation of non-colloidal particles in viscoelastic fluids subjected to simple shear flow. *Macromolecular Materials and Engineering*, 296:321–330, 2011.
- [57] J. Vermant and M.J. Solomon. Flow-induced structure in colloidal suspensions. *Journal of Physics: Condensed Matter*, 17:R187–R216, 2005.
- [58] H.H. Hu, D.D. Joseph, and M.J. Crochet. Direct simulation of fluid particle motions. *Theoretical and Computational Fluid Dynamics*, 3:285–306, 1992.
- [59] H.H. Hu. Direct simulation of flows of solid-liquid mixtures. *International Journal of Multiphase Flow*, 22:335–352, 1996.
- [60] H.H. Hu, N.A. Patankar, and M.Y. Zhu. Direct numerical simulations of fluid-solid systems using the arbitrary Lagrangian-Eulerian technique. *Journal of Computational Physics*, 169:427–462, 2001.
- [61] T.J.R. Hughes, W.K. Liu, and T.K. Zimmermann. Lagrangian-Eulerian finite element formulation for incompressible viscous flows. *Computer Methods in Applied Mechanics and Engineering*, 29:329–349, 1981.
- [62] R. Glowinski, T.-W. Pan, and J. Periaux. A fictitious domain method for external incompressible viscous flow modeled by Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 112:133–148, 1994.
- [63] R. Glowinski, T.-W. Pan, and J. Periaux. Distributed Lagrange multiplier methods for incompressible viscous flow around moving rigid bodies. *Computer Methods in Applied Mechanics and Engineering*, 151:181–194, 1998.
- [64] R. Glowinski, T.-W. Pan, T.I. Hesla, and D.D. Joseph. A distributed Lagrange multiplier/fictitious domain method for particulate flows. *International Journal of Multiphase Flow*, 25:755–794, 1999.
- [65] N.A. Patankar, P.Y. Huang, T. Ko, and D.D. Joseph. Lift-off of a single particle in Newtonian and viscoelastic fluids by direct numerical simulation. *Journal of Fluid Mechanics*, 438:67–100, 2001.
- [66] H.G. Choi and D.D. Joseph. Fluidization by lift of 300 circular particles in plane Poiseuille flow by direct numerical simulation. *Journal of Fluid Mechanics*, 438:101–128, 2001.
- [67] D.D. Joseph and D. Ocando. Slip velocity and lift. *Journal of Fluid Mechanics*, 454:263–286, 2002.

- [68] T. Ko, N.A. Patankar, and D.D. Joseph. Lift and multiple equilibrium positions of a single particle in Newtonian and Oldroyd-B fluids. *Computers & Fluids*, 35:121–146, 2006.
- [69] P. Singh, D.D. Joseph, T.I. Hesla, R. Glowinski, and T.-W. Pan. A distributed Lagrange multiplier/fictitious domain method for viscoelastic particulate flows. *Journal of Non-Newtonian Fluid Mechanics*, 91:165–188, 2000.
- [70] R. Glowinski, T.-W. Pan, T.I. Hesla, D.D. Joseph, and J. Périaux. A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: Application to particulate flow. *Journal of Computational Physics*, 169:363–426, 2001.
- [71] Z. Yu, N. Phan-Thien, Y. Fan, and R.I. Tanner. Viscoelastic mobility problem of a system of particles. *Journal of Non-Newtonian Fluid Mechanics*, 104:87–124, 2002.
- [72] Z. Yu, A. Wachs, and Y. Peysson. Numerical simulation of particle sedimentation in shear-thinning fluids with a fictitious domain method. *Journal of Non-Newtonian Fluid Mechanics*, 136:126–139, 2006.
- [73] W.R. Hwang, M.A. Hulsen, and H.E.H. Meijer. Direct simulation of particle suspensions in sliding bi-periodic frames. *Journal of Computational Physics*, 194:742–772, 2004.
- [74] W.R. Hwang, M.A. Hulsen, and H.E.H. Meijer. Direct simulations of particle suspensions in a viscoelastic fluid in sliding bi-periodic frames. *Journal of Non-Newtonian Fluid Mechanics*, 121:15–33, 2004.
- [75] W.R. Hwang, G.W.M. Peters, M.A. Hulsen, and H.E.H. Meijer. Modeling of flow-induced crystallization of particle-filled polymers. *Macromolecules*, 39:8389–8398, 2006.
- [76] G. D’Avino, P.L. Maffettone, M.A. Hulsen, and G.W.M. Peters. A numerical method for simulating concentrated rigid particle suspensions in an elongational flow using a fixed grid. *Journal of Computational Physics*, 226:688–711, 2007.
- [77] G. D’Avino, P.L. Maffettone, M.A. Hulsen, and G.W.M. Peters. Numerical simulation of planar elongational flow of concentrated rigid particle suspensions in a viscoelastic fluid. *Journal of Non-Newtonian Fluid Mechanics*, 150:65–79, 2008.
- [78] T.G. Kang, M.A. Hulsen, P.D. Anderson, J.M.J. den Toonder, and H.E.H. Meijer. Chaotic mixing induced by a magnetic chain in a rotating magnetic field. *Physical Review E*, 76(6):066303, 2007.
- [79] T.G. Kang, M.A. Hulsen, J.M.P. den Toonder, P.D. Anderson, and H.E.H. Meijer. A direct simulation method for flows with suspended paramagnetic particles. *Journal of Computational Physics*, 227:4441–4458, 2008.
- [80] B.H. Yang, J. Wang, D.D. Joseph, H.H. Hu, T.-W. Pan, and R. Glowinski. Migration of a sphere in tube flow. *Journal of Fluid Mechanics*, 540:109–131, 2005.

-
- [81] J.F. Brady and G. Bossis. Stokesian dynamics. *Annual Review of Fluid Mechanics*, 20:111–157, 1988.
- [82] C.S. Peskin. The fluid dynamics of heart valves: experimental, theoretical, and computational methods. *Annual Review of Fluid Mechanics*, 14:235–259, 1982.
- [83] R. Mittal and G. Iaccarino. Immersed boundary methods. *Annual Review of Fluid Mechanics*, 37:239–261, 2005.
- [84] A.M. Ardekani and R.H. Rangel. Numerical investigation of particle-particle and particle-wall collisions in a viscous fluid. *Journal of Fluid Mechanics*, 596:437–466, 2008.
- [85] A.M. Ardekani, S. Dabiri, and R.H. Rangel. Collision of multi-particle and general shape objects in a viscous fluid. *Journal of Computational Physics*, 227:10094–10107, 2008.
- [86] A.M. Ardekani, S. Dabiri, and R.H. Rangel. Deformation of a droplet in a particulate shear flow. *Physics of Fluids*, 21:093302, 2009.
- [87] A.J.C. Ladd. Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 1. Theoretical foundation. *Journal of Fluid Mechanics*, 271:285–309, 1994.
- [88] A.J.C. Ladd. Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 2. Numerical results. *Journal of Fluid Mechanics*, 271:311–339, 1994.
- [89] I. Babuška and J.M. Melenk. The partition of unity method. *International Journal for Numerical Methods in Engineering*, 40:727–758, 1997.
- [90] T. Belytschko and T. Black. Elastic crack growth in finite elements with minimal remeshing. *International Journal for Numerical Methods in Engineering*, 45:601–620, 1999.
- [91] N. Moës, J. Dolbow, and T. Belytschko. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering*, 46:131–150, 1999.
- [92] G.J. Wagner, N. Moës, W.K. Liu, and T. Belytschko. The extended finite element method for rigid particles in Stokes flow. *International Journal for Numerical Methods in Engineering*, 51:293–313, 2001.
- [93] A. Gerstenberger and W.A. Wall. An eXtended Finite Element Method/Lagrange multiplier based approach for fluid-structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 197:1699–1714, 2008.

- [94] T. Belytschko, R. Gracie, and G. Ventura. A review of extended/generalized finite element methods for material modeling. *Modelling and Simulation in Materials Science and Engineering*, 17:043001, 2009.
- [95] R. Fattal and R. Kupferman. Constitutive laws for the matrix-logarithm of the conformation tensor. *Journal of Non-Newtonian Fluid Mechanics*, 123:281–285, 2004.
- [96] M.A. Hulsen, R. Fattal, and R. Kupferman. Flow of viscoelastic fluids past a cylinder at high Weissenberg number: Stabilized simulations using matrix logarithms. *Journal of Non-Newtonian Fluid Mechanics*, 127:27–39, 2005.
- [97] F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods*. Springer, 1991.
- [98] R.A. Brown, M.J. Szady, P.J. Northy, and R.C. Armstrong. On the numerical stability of mixed finite-element methods for viscoelastic flows governed by differential constitutive equations. *Theoretical and Computational Fluid Dynamics*, 5:77–106, 1993.
- [99] R. Guénette and M. Fortin. A new mixed finite element method for computing viscoelastic flows. *Journal of Non-Newtonian Fluid Mechanics*, 60:27–52, 1995.
- [100] F.P.T. Baaijens. Mixed finite element methods for viscoelastic flow analysis: a review. *Journal of Non-Newtonian Fluid Mechanics*, 79:361–385, 1998.
- [101] A.C.B Bogaerds, A.M. Grillet, G.W.M. Peters, and F.P.T. Baaijens. Stability analysis of polymer shear flows using the extended pom-pom constitutive equations. *Journal of Non-Newtonian Fluid Mechanics*, 108:187–208, 2002.
- [102] A.N. Brooks and T.J.R. Hughes. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 32:199–259, 1982.
- [103] N. Sukumar, D.L. Chopp, N. Moës, and T. Belytschko. Modeling holes and inclusions by level sets in the extended finite-element method. *Computer Methods in Applied Mechanics and Engineering*, 190:6183–6200, 2001.
- [104] A. Hansbo and P. Hansbo. A finite element method for the simulation of strong and weak discontinuities in solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 193:3523–3540, 2004.
- [105] P.M.A. Areias and T. Belytschko. A comment on the article “A finite element method for the simulation of strong and weak discontinuities in solid mechanics” by A. Hansbo and P. Hansbo [Comput. Methods Appl. Mech. Engrg. 193 (2004) 3523-3540] . *Computer Methods in Applied Mechanics and Engineering*, 195:1275–1276, 2006.
- [106] C. Min and F. Gibou. Geometric integration over irregular domains with application to level-set methods. *Journal of Computational Physics*, 226:1432–1443, 2007.

-
- [107] Y. Fan and M.J. Crochet. High-order finite element methods for steady viscoelastic flows. *Journal of Non-Newtonian Fluid Mechanics*, 57:283–311, 1995.
- [108] F. Shakib, T.J.R. Hughes, and Z. Johan. A new finite element formulation for computational fluid dynamics: X. The compressible Euler and Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 89:141–219, 1991.
- [109] W.J. Lunsman, L. Genieser, R.C. Armstrong, and R.A. Brown. Finite element analysis of steady viscoelastic flow around a sphere in a tube: calculations with constant velocity models. *Journal of Non-Newtonian Fluid Mechanics*, 48:63–99, 1993.
- [110] H. Ji and J.E. Dolbow. On strategies for enforcing interfacial constraints and evaluating jump conditions with the extended finite element method. *International Journal for Numerical Methods in Engineering*, 61:2508–2535, 2004.
- [111] N. Moës, E. Béchet, and M. Tourbier. Imposing Dirichlet boundary conditions in the extended finite element method. *International Journal for Numerical Methods in Engineering*, 67:1641–1669, 2006.
- [112] J.E. Dolbow and L.P. Franca. Residual-free bubbles for embedded Dirichlet problems. *Computer Methods in Applied Mechanics and Engineering*, 197:3751–3759, 2008.
- [113] J. Dolbow and I. Harari. An efficient finite element method for embedded interface problems. *International Journal for Numerical Methods in Engineering*, 78:229–252, 2009.
- [114] R. Guénette, A. Fortin, A. Kane, and J.F. Héту. An adaptive remeshing strategy for viscoelastic fluid flow simulations. *Journal of Non-Newtonian Fluid Mechanics*, 153:34–45, 2008.
- [115] A. Zilian and A. Legay. The enriched space-time finite element method (EST) for simultaneous solution of fluid-structure interaction. *International Journal for Numerical Methods in Engineering*, 75:305–334, 2008.
- [116] G. D’Avino and M.A. Hulsen. A comparison between a collocation and weak implementation of the rigid-body motion constraint on a particle surface. *International Journal for Numerical Methods in Fluids*, 64:1014–1040, 2010.
- [117] M. Grajewski, M. Këster, and S. Turek. Mathematical and numerical analysis of a robust and efficient grid deformation method in the finite element context. *SIAM Journal on Scientific Computing*, 31:1539–1557, 2009.
- [118] S.U. Pickering. Emulsions. *Journal of the Chemical Society, Transactions*, 91:2001–2021, 1907.
- [119] J.D. van der Waals. The thermodynamic theory of capillarity under the hypothesis of a continuous density variation. *Journal of Statistical Physics*, 20:197–244, 1979.

-
- [120] J.W. Cahn and J.E. Hilliard. Free energy of a nonuniform system. I. Interfacial energy. *Journal of Chemical Physics*, 28(2):258–267, 1958.
- [121] D.M. Anderson, G.B. McFadden, and A.A. Wheeler. Diffuse-interface methods in fluid mechanics. *Annual Review of Fluid Mechanics*, 30:139–165, 1998.
- [122] M. Prusty, J. Keestra, J.G.P. Goossens, and P.D. Anderson. Experimental and computational study on structure development of PMMA/SAN blends. *Chemical Engineering Science*, 62:1825–1837, 2007.
- [123] S.M. Wise, J.S. Lowengrub, and V. Cristini. An adaptive multigrid algorithm for simulating solid tumor growth using mixture models. *Mathematical and Computer Modelling*, 53(1-2):1–20, 2011.
- [124] V.V. Khatavkar, P.D. Anderson, and H.E.H. Meijer. Capillary spreading of a droplet in the partially wetting regime using a diffuse-interface model. *Journal of Fluid Mechanics*, 572:367–387, 2007.
- [125] V.V. Khatavkar, P.D. Anderson, P.C. Duineveld, and H.E.H. Meijer. Diffuse-interface modelling of droplet impact. *Journal of Fluid Mechanics*, 581:97–127, 2007.
- [126] V.V. Khatavkar, P.D. Anderson, P.C. Duineveld, and H.E.H. Meijer. Diffuse interface modeling of droplet impact on a pre-patterned solid surface. *Macromolecular Rapid Communications*, 26(4):298–303, 2005.
- [127] C. Tufano, G.W.M. Peters, H.E.H. Meijer, and P.D. Anderson. Effects of partial miscibility on drop-wall and drop-drop interactions. *Journal of Rheology*, 54(1):159–183, 2010.
- [128] P.C. Millett and Y.U. Wang. Diffuse-interface field approach to modeling arbitrarily-shaped particles at fluid-fluid interfaces. *Journal of colloid and interface science*, 353(1):46–51, 2011.
- [129] W.R. Hwang, P.D. Anderson, and M.A. Hulsen. Chaotic advection in a cavity flow with rigid particles. *Physics of Fluids*, 17(4):043602, 2005.
- [130] Y.J. Choi, M.A. Hulsen, and H.E.H. Meijer. An extended finite element method for the simulation of particulate viscoelastic flows. *Journal of Non-Newtonian Fluid Mechanics*, 165:607–624, 2010.
- [131] J. Lowengrub and L. Truskinovsky. Quasi-incompressible Cahn-Hilliard fluids. *Proceedings of the Royal Society A*, 454:2617–2654, 1998.
- [132] J.W. Cahn and J.E. Hilliard. Free energy of a nonuniform system. III. Nucleation in a two-component incompressible fluid. *Journal of Chemical Physics*, 31:688–699, 1959.
- [133] J. Lowengrub, J. Goodman, H. Lee, E.K. Longmire, M.J. Shelley, and L. Truskinovsky. Topological transitions in liquid/liquid interfaces. In I. Athanasopoulos, M. Makrakis,

- and J.F. Rodrigues, editors, *Proceedings of the 1997 International Congress on Free Boundary Problems*, Pitman Research Notes. Addison-Wesley Longman, 1998.
- [134] J.W. Cahn. Critical point wetting. *Journal of Chemical Physics*, 66(8):3667–3672, 1977.
- [135] D. Jacqmin. Contact-line dynamics of a diffuse fluid interface. *Journal of Fluid Mechanics*, 402:57–88, 2000.
- [136] P.R. Amestoy and I.S. Duff. Memory management issues in sparse multifrontal methods on multiprocessors. *International Journal of Supercomputer Applications*, 7:64–69, 1989.
- [137] Y.J. Choi and M.A. Hulsen. Simulation of extrudate swell using an extended finite element method. *Korea-Australia Rheology Journal*, in press, 2011.
- [138] A. Gerstenberger and W.A. Wall. An embedded Dirichlet formulation for 3D continua. *International Journal for Numerical Methods in Engineering*, 82:537–563, 2010.
- [139] M.G.H.M. Baltussen, Y.J. Choi, M.A. Hulsen, and P.D. Anderson. Weakly imposed Dirichlet boundary conditions for non-Newtonian fluid flow. *Journal of Non-Newtonian Fluid Mechanics*, 166:993–1003, 2011.
- [140] T.C. Papanastasiou, N. Malamataris, and K. Ellwood. A new outflow boundary condition. *International Journal for Numerical Methods in Fluids*, 14:587–608, 1992.
- [141] G. D’Avino and M.A. Hulsen. Decoupled second-order transient schemes for the flow of viscoelastic fluids without a viscous solvent contribution. *Journal of Non-Newtonian Fluid Mechanics*, 165:1602–1612, 2010.
- [142] L. Zhang, T. Cui, and H. Liu. A set of symmetric quadrature rules on triangles and tetrahedra. *Journal of Computational Mathematics*, 27:89–96, 2009.
- [143] R.I. Tanner. *Engineering Rheology*. Oxford University Press, 1985.
- [144] M.M. Denn. Extrusion instabilities and wall slip. *Annual Review of Fluid Mechanics*, 33:265–287, 2001.
- [145] E. Birinci and D.M. Kalyon. Development of extrudate distortions in poly(dimethyl siloxane) and its suspensions with rigid particles. *Journal of Rheology*, 50:313–326, 2006.
- [146] R. Keunings. An algorithm for the simulation of transient viscoelastic flows with free surfaces. *Journal of Computational Physics*, 62:199–220, 1986.
- [147] V. Ganvir, A. Lele, R. Thakkar, and B.P. Gautham. Prediction of extrudate swell in polymer melt extrusion using an Arbitrary Lagrangian Eulerian (ALE) based finite element method. *Journal of Non-Newtonian Fluid Mechanics*, 156:21–28, 2009.
- [148] T.C. Papanastasiou, V.D. Dimitriadis, L.E. Scriven, C.W. Macosko, and R.L. Sani. On the inlet stress condition and admissibility of solution of fiber-spinning. *Advances in Polymer Technology*, 15(3):237–244, 1996.

- [149] Y.J. Choi and M.A. Hulsen. Alignment of particles in a confined shear flow of a viscoelastic fluid. *Journal of Non-Newtonian Fluid Mechanics*, under review, 2011.
- [150] M.J. Crochet and R. Keunings. Die swell of a Maxwell fluid: Numerical prediction. *Journal of Non-Newtonian Fluid Mechanics*, 7:199–212, 1980.

Samenvatting

Stromingen van met deeltjes gevulde vloeistoffen vinden we in diverse onderzoeksgebieden en industriële processen zoals gefluidiseerde suspensies, elektroforese, filtratie en slurrietransport, en verder in het scheiden van materialen, het verbeteren van de snelheid van mengprocessen en in gevulde polymeren. In veel van deze toepassingen vertoont de vloeistof niet-Newtons gedrag. Vanwege dit inherent niet-lineaire gedrag van de vloeistoffen is een numerieke analyse essentieel om de precieze beweging van de deeltjes in deze niet-Newtonse, viscoelastische vloeistoffen te beschrijven. Het doel van dit proefschrift is dan ook het ontwikkelen van een nieuwe numerieke methode voor het simuleren van complexe viscoelastische vloeistoffen met deeltjes.

Als deeltjes aan een vloeistof worden toegevoegd, verlopen veldgrootheden zoals druk en spanning discontinu over het grensvlak tussen deeltje en vloeistof. Immers, er treedt in het inwendige van het deeltje geen stroming op. De meest intuïtieve methode om dit probleem aan te pakken is het gebruik van een *boundary-fitted* mesh. Het genereren daarvan is echter gecompliceerd en tijdrovend, in het bijzonder in driedimensionale simulaties. Een veelgebruikt alternatief is de *fictitious domain* methode. Het idee hierbij is om ieder deeltje met de omringende vloeistof te vullen om er vervolgens voor te zorgen dat deze binnenvloeistof zich als een star lichaam gaat bewegen. Een nadeel is echter dat discontinuïteiten worden uitgesmeerd en daardoor grootheden als druk en spanning nabij het grensvlak moeilijk te voorspellen zijn. Om dit probleem op te lossen maken we gebruik van de *extended finite element* methode (XFEM) die, net als de *boundary-fitted* mesh methode, discontinuïteiten kan beschrijven. Een belangrijk verschil is echter dat XFEM net als *fictitious domain* gebruik maakt van een relatief eenvoudige mesh. We passen de nieuw ontwikkelde XFEM method toe op diverse stromingsproblemen van met deeltjes gevulde vloeistoffen.

In Hoofdstuk 2 beschrijven we de stroming van een viscoelastische vloeistof rond een stationaire cilinder. Resultaten verkregen met XFEM worden geverifieerd met oplossingen verkregen met een *boundary-fitted* mesh en vergeleken met die van de *fictitious domain* methode. We tonen aan dat XFEM bij het grensvlak een nauwkeurigheid heeft die vergelijkbaar is met die van de *boundary-fitted* mesh methode en significant beter dan die van de *fictitious*

domain methode.

Hoofdstuk 3 onderzoekt de migratie van een deeltje in een (circulaire) Couette stroming van een Giesekus vloeistof. Het deeltje migreert naar een stabiele radiële positie dichtbij de buitencilinder, onafhankelijk van zijn initiële positie. Als het Weissenberg getal wordt verhoogd is de migratie sneller terwijl de stabiele radiële positie in de richting van de buitencilinder verschuift. Bij grotere deeltjes vinden we een snellere migratie.

Hoofdstuk 4 presenteert een model voor de dynamica van deeltjes in tweefasen-stromingen op basis van de Cahn-Hilliard theorie. Deeltjes worden weer beschreven met behulp van de *extended finite element* methode en om te laten zien dat de methode werkt bestuderen we de dynamica van een enkel deeltje op een vloeistof-vloeistof grensvlak met de nadruk op de effecten van grensvladdikte en -spanning en van de deeltjesgrootte en viscositeitsverhouding. Tot slot wordt de beweging van een deeltje door meerdere vloeistoflagen gedemonstreerd.

Hoofdstuk 5 richt zich op het formeren van ketens van deeltjes in afschuifstromingen van viscoelastische vloeistoffen tussen twee vlakken platen (*confined flow*). De maximum haalbare ketenlengte wordt beperkt door het reologisch gedrag en neemt toe met toenemende vloeistofelasticiteit. Zodra deeltjes eenmaal een keten hebben gevormd is hun uiteindelijk toestand onafhankelijk van hun initiële positie. Verkleining van de afstand tussen de platen versterkt de ketenvorming, echter bij zeer kleine afstanden wordt ketenvorming weer tegengegaan doordat de afstotende interacties tussen deeltjes gaan overheersen. Tot slot wordt de stationaire rotatiesnelheid van deeltjes bestudeerd afhankelijk van de ketenlengte.

Hoofdstuk 6 bestudeert *extrudate swell* van een upper-convected Maxwell vloeistof. Gecombineerd met een tijdelijk ALE schema (arbitrary Lagrange-Euler) levert de nieuw voorgestelde XFEM methode voor kleine Deborah getallen een nauwkeurigheid op die vergelijkbaar is met die van de *boundary-fitted* mesh methode. Voor grote Deborah getallen lijkt de nieuwe methode voor dit soort extrusieproblemen zelfs stabielier te zijn dan de *boundary-fitted* mesh methode. Verder wordt de migratie van deeltjes bestudeerd waarin blijkt dat de aanwezigheid van een deeltje de spanningsverdeling in de vloeistof en de vorm van het vrije oppervlak significant verstoort. Bij toenemend Weissenberg getal wordt de afstand van het deeltje tot het vrije oppervlak groter; bij toenemende deeltjesgrootte wordt die afstand juist kleiner.

Hoofdstuk 7 vat de conclusies samen en verder wordt er een aantal aanbevelingen voor verder onderzoek gegeven.

Acknowledgements

I would like to express my sincere gratitude to all the people who helped me to complete my four years PhD study, which was always a great pleasure to me. First of all, I would like to thank my promotor Prof. Han Meijer, and co-promotors Dr. Martien Hulsen and Dr. Patrick Anderson. I truly appreciate your help and advices for the progress of my work. Especially I am deeply indebted to my adviser Martien Hulsen for constant encouragement, invaluable guidance and interesting discussions that have shaped me as a new starter in the research field. I also thank the Dutch Polymer Institute (DPI) for their financial support.

I thank my former supervisor late Prof. T.H. Kwon at POSTECH, and Dr. T.G. Kang and Dr. W.R. Hwang for introducing me to this great group. I also want to give thanks to the committee members, Prof. René de Borst, Prof. Jan Vermant, Prof. Marek Behr and Prof. Hans Kuijpers for reviewing my thesis.

I would like to thank all the members of the Materials Technology (MaTe) group. Especially I would give thanks to Marleen, Yvon and Alice for helping me to stay easily in Eindhoven, and to Leo for supporting computer resources.

I really like the open environment, not only in the group but also in the Netherlands. It's a really interesting experience to live in a totally different cultural atmosphere, almost impossible to explain in a single word - funny, exciting, fuzzy, sometimes even chaotic. I had a lot of fun during my stay in Eindhoven, the Netherlands. Thank you: all friends and colleagues. It's unforgettable in every way and I'll miss you all.

Finally, I wish to express my deepest gratitude to my family; parents, sister and brother for their endless support and encouragement. And now, last but not least, thanks to myself.

최 영 준

Choi, Young Joon
August 16, 2011.

Curriculum Vitae

Young Joon Choi was born on July 2, 1977 in Daegu, South Korea. He received his bachelor's degree, with Magna Cum Laude, in Mechanical Engineering and Mathematics from Pohang University of Science and Technology (POSTECH) in February 2003, and master's degree in Mechanical Engineering from the same university in February 2005. From January 1997 to March 1999, he served in the Army Aviation School, Republic of Korea. After receiving his master's degree, he worked for Samsung Electronics from February 2005 to June 2007. He started his doctoral studies at Eindhoven University of Technology in September 2007, under the supervision of Prof. Han E.H. Meijer and Dr. Martien A. Hulsen. On October 18, 2011, Young Joon Choi will defend his PhD thesis at the Eindhoven University of Technology.

