

Numerical methods for biological flows laden with deformable capsules and solid particles

by

Damien P. Huet

B.Sc., Lycée Pierre Corneille & ENSTA Paris, 2016

M.Sc., ENSTA Paris, 2019

M.Sc., KTH Royal Institute of Technology, 2019

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

The Faculty of Graduate and Postdoctoral Studies

(Mathematics)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

May 2023

© Damien P. Huet 2023

The following individuals certify that they have read, and recommend to the Faculty of Graduate and Postdoctoral Studies for acceptance, the thesis entitled:

Numerical methods for biological flows laden with deformable capsules and solid particles

submitted by **Damien P. Huet** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Mathematics**.

Examining Committee:

Anthony Wachs, Mathematics and Chemical & Biological Engineering
Supervisor

Pierre-Yves Lagrée, Mechanical Engineering, Sorbonne Université
External examiner

Chen Greif, Computer Science
University examiner

Gwynn Elfring, Mechanical Engineering
University examiner

Brian Wetton, Mathematics
Supervisory committee member

Ian Frigaard, Mathematics and Mechanical Engineering
Supervisory committee member

Abstract

Biological fluids such as blood accomplish many vital tasks in the human body, including carrying oxygen and nutrients to tissues, regulating internal temperature and pH, or transporting white blood cells to infected areas. A better understanding of these fluids can provide insight into many pathologies such as the formation of aneurysms and the effect of sickle cell disease on the flow of red blood cells, as well as help design efficient diagnosis tools on microfluidic devices. Such fluids are composed of a continuous viscous phase and suspended bodies, including rigid particles and deformable membranes enclosing an inner fluid, referred to as capsules. In this thesis, we develop numerical tools aiming to simulate cell-resolved biological fluids such as blood. In a first part, we focus on the dispersed solid phase, a field known as granular mechanics. In this context, we implement a contact force able to accurately model static assemblies of granular media. After extensive validation, we use this contact model in a purely granular setting to study avalanches of entangled particles. Our numerical results are compared to experiments and show very good qualitative and quantitative agreement. Moreover, we present a variety of novel avalanching behaviors, as well as an intermittent regime in which reproducibility is lost. After analyzing the microstructure of granular assemblies in this regime, we conclude that it likely arises from mesoscale clusters of particles. In a second part, we concentrate on flowing biological capsules. We develop an adaptive front-tracking method which enables simulations of capsules in very large geometries for a wide range of Reynolds number. We validate our solver extensively and we show excellent qualitative and quantitative agreement with the literature. We then study the dynamics of capsules flowing through a sharp corner, a commonly encountered geometry in microfluidic devices. We analyze the trajectory, normalized velocity and area variations of the capsules and we show that in our case of strong confinement, the capsules interact weakly unless they are located very close to each other. Finally, we present and implement a fully Eulerian alternative method to simulate flowing capsules, and we highlight its advantages and limits.

Lay Summary

Computer simulations are an indispensable tool to understand and predict the behavior of many complex natural and industrial processes. In particular, complex flows such as rockfalls or flowing blood through microcirculation rely heavily on numerical tools as they are described by equations mathematicians cannot solve by hand. In this thesis, we develop improved numerical methods in order to simulate such complex flows. We first focus on granular flows composed of entangled particles. We study their complex intrication behavior as a step forward in the modeling of complex granular media, which are very common in industrial processes. Then we develop an efficient method to simulate highly deformable biological cells such as red blood cells. This method allows us to study the behavior of deformable cells in large and complex geometries, and therefore constitutes a valuable tool for biologists developing lab-on-chip devices used for instance to provide cheap and fast diagnoses.

Preface

The work presented in this document was realized as part of my PhD, that I conducted at the University of British Columbia, Point Grey campus, in the research group of Prof. Anthony Wachs. The research presented in this thesis has led to several publications and submissions in peer-reviewed academic journals, as well as the open-source release of software components, which are listed below:

- Damien P. Huet, Maziyar Jalaal, Rick van Beek, Devaraj van der Meer, and Anthony Wachs. Granular avalanches of entangled rigid particles. *Phys. Rev. Fluids*, 6:104304, Oct 2021. [90]

Chapters 2 and 3 in this thesis are based this paper, except Sections 2.5 and 2.6.1. I was the lead investigator, responsible for code development and validation, running, post-processing and analyzing numerical simulations, manuscript preparation and editing. Maziyar Jalaal, Rick van Beek and Devaraj van der Meer conducted experiments. Anthony Wachs was involved in the original idea, supervision and manuscript review and edits.

- Anthony Wachs, Markus Uhlmann, Jos Derksen, and Damien P. Huet. 7 - modeling of short-range interactions between both spherical and non-spherical rigid particles. In Shankar Subramaniam and S. Balachandar, editors, *Modeling Approaches and Computational Methods for Particle-Laden Turbulent Flows*, Computation and Analysis of Turbulent Flows, pages 217–264. Academic Press, 2023. [203]

Sections 2.5 and 2.6.1 in this thesis are based on Sections 7.5.4 and 7.7.1 of this chapter. I was involved in the original preparation of Sections 7.5.4, 7.7.1 and 7.7.2, and I contributed to reviewing and editing the full manuscript. Anthony Wachs was the lead author of this chapter, and wrote the original draft of all other sections as well as reviewed and edited the full manuscript. Markus Uhlmann and Jos Derksen contributed to reviewing and editing the drafted manuscript.

- Damien P. Huet and Anthony Wachs. A cartesian-octree adaptive front-tracking solver for immersed biological capsules in large complex domains. *arXiv preprint arXiv:2211.15814*, 2022 [91]. Submitted for review to Journal of Computational Physics (2022).

Chapter 4 in this thesis is based on this paper. I was the lead investigator, responsible for the original idea, code development, code validation, as well as running, post-processing and analyzing numerical simulations and preparing the original manuscript. Anthony Wachs was involved in supervision and manuscript review and edits.

- Damien P. Huet, Antoine Morente, Guodong Gai and Anthony Wachs. Motion and deformation of capsules flowing through a corner in the inertial and non-inertial regimes. 2023. Submitted for review to Physical Review Fluids (2023).

Chapter 5 in this thesis is based on this paper. I was the lead investigator, and I was involved in the original idea, code development, simulation setup, post-processing and analyzing results and manuscript preparation and editing. Antoine Morente and Guodong Gai were involved in running simulations, post-processing and analyzing results, and manuscript preparation. Anthony Wachs was involved in manuscript review and edits.

- Open-source release of the adaptive Front-Tracking Method, Damien P. Huet, 2022. URL: <http://basilisk.fr/sandbox/huet>. [80]

The documented code and post-processing routines allowing to reproduce simulations and results from Chapters 4 and 5 are publicly available in the above repository.

Table of Contents

Abstract	iii
Lay Summary	iv
Preface	v
Table of Contents	vii
List of Tables	xii
List of Figures	xiii
List of Algorithms	xxiv
Acknowledgements	xxv
1 Introduction	1
1.1 Numerical modelling of the solid phase	4
1.2 Numerical modelling of immersed deformable capsules	7
I Dispersed solid phase	10
2 Accurate static contact law for the Discrete Element Method (DEM)	11
2.1 Introduction	11
2.2 General structure of the Discrete Element Method	13
2.3 Contact model	15
2.3.1 Contact model in translation	16
2.3.2 Contact model in rotation	17
2.4 Contact model parameters	18
2.4.1 Limitations of the intuitive and physical approach	19
2.4.2 Procedure to find contact parameters in translation	20

Table of Contents

2.4.3	Contact parameters in rotation	22
2.5	Time intergration schemes	23
2.5.1	Integration of the translational motion	23
2.5.2	Integration of the rotational motion	25
2.6	Validation of the numerical method	26
2.6.1	Single sphere bouncing on a horizontal plane	26
2.6.2	Quantifying the residual velocity	27
2.6.3	Hopper discharge	28
2.7	Conclusion	30
3	Granular avalanches of entangled rigid particles	32
3.1	Introduction	32
3.2	Experimental and numerical setup	36
3.2.1	Experimental setup	36
3.2.2	Particles	37
3.2.3	Numerical setup	37
3.3	Experimental and numerical results	40
3.3.1	Qualitative Analysis	40
3.3.2	Macroscopic measurements	43
3.3.3	Microstructure analysis	49
3.4	Discussion	54
3.4.1	Analyzing the intermittent regime with DEM simula- tions	54
3.4.2	On the importance of the contact model with memory	56
3.4.3	On the choice of contact model parameters for non- spherical bodies	57
3.4.4	Analogy with other stable granular assemblies	59
3.5	Conclusion	60
II	Immersed elastic capsules	62
4	A Cartesian-octree adaptive front-tracking solver for im- mersed elastic capsules	63
4.1	Introduction	63
4.2	Governing equations	67
4.2.1	Fluid motion	67
4.2.2	Membrane mechanics	67
4.3	Numerical method	69

Table of Contents

4.3.1	Adaptive Finite Volume solver for the Navier-Stokes equations	69
4.3.2	Second-order treatment of solid embedded boundaries	70
4.3.3	Front-Tracking Method (FTM)	72
4.3.4	Computation of the membrane forces	74
4.4	Validation Cases	77
4.4.1	Elastic and bending forces of an isolated membrane	77
4.4.2	Initially spherical capsule in an unbounded shear flow	79
4.4.3	Initially spherical capsule flowing through a constricted channel	83
4.4.4	Red blood cell in an unbounded shear flow	85
4.4.5	Capsules interception in a shear flow	88
4.5	Results	92
4.5.1	Capsule flowing through a narrow constriction	92
4.5.2	Capsule-laden flows in large complex channel geometries	95
4.6	Conclusion and perspectives	99
5	Motion and deformation of capsules through a corner in the inertial and non-inertial regimes	103
5.1	Introduction	103
5.2	Governing equations and problem statement	107
5.3	Numerical method and validations	109
5.4	Motion and deformation of a single capsule	112
5.4.1	Influence of the Reynolds and Capillary numbers	113
5.4.2	Maximum deformation of the capsule	114
5.4.3	Evolution of the capsule shape	120
5.4.4	Discussion on the Stokes regime	124
5.5	System of two capsules	125
5.5.1	Qualitative analysis: trajectory and capsule shape	125
5.5.2	Quantitative analysis: velocity and membrane area	127
5.5.3	Time evolution of the interspacing distance	130
5.6	Train of ten capsules	132
5.7	Conclusion	137
6	Towards a fully Eulerian adaptive solver for immersed elastic capsules	140
6.1	Introduction	140
6.2	Governing equations	142
6.2.1	Eulerian framework	142

Table of Contents

6.2.2	Membrane mechanics	144
6.2.3	Full system of equations	148
6.3	Numerical method and implementation strategy	149
6.3.1	Definition of the membrane region $\tilde{\Gamma}$	149
6.3.2	Invariance along the normal direction	149
6.3.3	Advection equations	150
6.3.4	Face-centered acceleration term	151
6.4	Validation cases	153
6.4.1	Accuracy of normal vectors	153
6.4.2	Stretching of a flat membrane	154
6.4.3	Capsule deformation in a simple shear flow	162
6.5	Discussion	165
7	Conclusion	168
7.1	Granular dynamics	168
7.2	Immersed elastic capsules	172
7.3	Perspectives	179
	Bibliography	183

Appendices

A	On the contact detection of solid bodies of arbitrary shapes	207
A.1	Broad-phase detection	207
A.2	Narrow-phase detection	209
A.2.1	Prerequisites	209
A.2.2	The GJK distance algorithm	210
A.2.3	Application to DEM simulations	212
B	Granular avalanches of entangled rigid particles: supplementary material	213
B.1	Non-dimensional displacement of the center of mass of the whole granular assembly	213
B.2	Probability density functions of the scaled contact force magnitude in various regimes	216
B.3	Raw data for the analysis of the intermittent regime	218
C	Adaptive immersed boundary stencils	219
C.1	Method	219
C.2	Proof of concept and discussion	222

Table of Contents

D	Motion and deformation of capsules through a corner in the inertial and non-inertial regimes: supplementary material	226
D.1	Time evolution of capsule areas in the case of two interacting capsules	226
D.2	Train of capsules at large initial interspacings	227

List of Tables

- 2.1 Particle properties and contact model parameters of the Hopper discharge simulations. 29
- 3.1 Particle properties and contact model parameters in the numerical simulations of non-convex particles. 38
- 5.1 Maximum area \mathcal{A}_{max} of the leading capsule at different Ca , Re and d_0 130
- 5.2 Maximum area \mathcal{A}_{max} of the trailing capsule at different Ca , Re and d_0 130
- B.1 Average data for the intermittent cases about to collapse (top) and remain stable (bottom). The symbol σ stands for the standard deviation of the data. 218

List of Figures

1.1	A red blood cell (left), an activated platelet (center) and a white blood cell (right) as seen using electron microscopy imaging [10].	2
1.2	Diagram listing the types of interactions in a typical blood sample. The highlighted rows in the table show the interactions actively investigated in the present thesis. For readability, the volume fraction of the blood composition is not respected.	3
2.1	Top-level view of a soft-body DEM algorithm	14
2.2	In soft-body DEM, a contact is modelled with an overlap – Please note that the overlap is intentionally magnified here for visual purposes.	15
2.3	The linear spring-dashpot contact model as a mechanical analog. The dashed line represents the tangential contact plane.	16
2.4	Illustration of the error stemming from the latency in the detection of the beginning and the end of the contact. Adapted from Kruggel-Emden et al. [112].	24
2.5	Error on the penetration depth δ_n of a sphere bouncing on a horizontal plane for the first order Forward Euler scheme (solid line), the second-order Adams-Bashforth scheme (dotted line) and the third order Adams-Bashforth scheme (dash-dotted line). For each scheme, the bouncing sphere is either (i) already in contact with the horizontal plane at $t = 0$ (“x” markers), or (ii) away from the horizontal plane at $t = 0$ (“+” markers).	26
2.6	Time evolution of the average velocity for a system of 750 spheres settling in a box. Dotted line: no memory term in the contact model. Solid line: memory term in the contact model.	28

List of Figures

2.7	Variation of the repose angle θ with the static friction coefficient μ_c . Solid line: this study. Dashed line: adapted from Yan et al. [211]	29
3.1	Schematics of the experimental setup: (a) the dam break apparatus and (b) the shape of the particles.	36
3.2	(a) Example of a post-processing output for the run-out distance L_∞ of a simulation, from a top view of the avalanche. The solid line corresponds to the intensity of a given pixel column (the maximum intensity corresponds to the top of the image, the minimum intensity to the bottom). The dashed line indicates the run-out distance: in this case, $L_\infty = 32$ cm. (b) Example of a post-processing output for the final height H_∞ of a simulation. The solid line indicates the average final height and the dotted lines show the first and third quartiles. For visual purposes, an example with a particularly wide interquartile range (IQR) is shown (a typical IQR is half the size of that shown on this figure).	39
3.3	Snapshots of a top-driven collapse with an initial aspect ratio $H_0/L_0 = 6$. Left: experiments. Right: simulation. Time increment between each frame: 0.145s. Videos of this collapse are available online at https://youtu.be/yzlzn_XrkJA	41
3.4	Snapshots of a buckling collapse with an initial aspect ratio of $H_0/L_0 \approx 7$. Left: experiment. Right: simulation. Time increment between each frame: 0.12s. Videos of this collapse are available online at https://youtu.be/DBmOVcR9PrU	41
3.5	Snapshots of a simulated collapsing assembly (top) and of a simulated stable assembly (bottom). The initial dimensions of both assemblies are identical: only the initial microstructure differs between these two intermittent cases. The frames are taken at the following times: $t = 0$ s, $t = 0.30$ s, $t = 0.60$ s, $t = 0.90$ s, $t = 1.20$ s and $t = 2.00$ s.	42
3.6	Flow maps from (a) experiments and (b) simulations. Each cross represents an experiment or a simulation. Two flow regimes are identified: repeatable and intermittent; and two collapse behaviors are reported: top-driven (observed in all configurations) and buckling (observed only for narrow columns).	43

List of Figures

3.7	Non-dimensional runout distance $(L_\infty - L_0)/L_0$ as a function of the initial aspect ratio H_0/L_0 . Error bars are depicted with vertical segments.	44
3.8	Non-dimensional height H_0/H_∞ as a function of the initial aspect ratio H_0/L_0 . Error bars are depicted with vertical segments.	45
3.9	(a) Comparison of the average non-dimensional translational particle velocity versus non-dimensional time for systems of spherical and cross-shaped particles in systems of initial aspect ratio $a = H_0/L_0 = 3.2$. (b-c) Snapshots of the collapse of a small (b) and large (c) system of crosses at $t = 3\tau_c$, coloured by the norm of their translational velocity from dark blue (smallest) to red (largest). Small system: 3200 spheres or 1000 crosses; Large system: 15000 spheres or 3000 crosses.	47
3.10	Comparison of the average non-dimensional translational particle velocity versus non-dimensional time for a staggered collapse (solid line) and a regular collapse (dashed line) in the case of an initial aspect ratio $a = H_0/L_0 = 1.75$. The only difference between the two collapses is the random initial packing.	48
3.11	Visualization of the contact network in a granular column composed of spheres and crosses. Each segment links the centers of mass of two contacting particles, and its color and width vary according to the magnitude of the total contact force magnitude F_c divided by the average contact force magnitude $\langle F_c \rangle$. In (a) all contacts are shown, while in (b) only the contacts associated with a force magnitude greater than five times the average force magnitude are plotted.	49
3.12	Probability density function (PDF) of the contact force magnitude F_c divided by the average contact force magnitude $\langle F_c \rangle$ just before the gate is opened, in the cases of assemblies of crosses (black) and spheres (blue).	50
3.13	Definition of the branch vector projected onto the vertical xz -plane and of the angle θ it forms with the horizontal direction.	52

List of Figures

3.14	Average fabric anisotropy of various granular assemblies prior to the opening of the gate. (a) Comparison of the fabric anisotropy of spheres (averaged over 30 simulations) to crosses (averaged over 60 simulations) and coins (only one simulation shown) – the curve for coins has been scaled down for readability. (b) Comparison of the fabric anisotropy of crosses about to collapse in a top-driven fashion (averaged over 50 simulations) to crosses about to collapse in a buckling fashion (averaged over 6 simulations). (c) Comparison of the fabric anisotropy of crosses in the intermittent regime about to collapse (averaged over 18 simulations) to remaining stable (averaged over 26 simulations).	53
3.15	Visualization of the force network of two intermittent simulations. Each segment links a contact point to the center of mass of a particle, and is colored according to the magnitude of the normal component of the contact force. (a) The opening of the right gate leads to a granular collapse. (b) The opening of the right gate leads to a stable assembly.	55
3.16	Snapshots of simulations with and without memory in the contact model. Top: with memory. Bottom: without memory. The frames are taken at the following times: $t = 0\text{s}$, $t = 0.15\text{s}$, $t = 0.30\text{s}$, $t = 0.45\text{s}$, $t = 0.60\text{s}$, $t = 0.75\text{s}$, $t = 0.90\text{s}$, $t = 1.05\text{s}$ and $t = 3.00\text{s}$	56
3.17	Variation of the flow behavior with respect to the tangential parameters. Each dot corresponds to a simulation with specific tangential parameters (k_t, γ_t) . Region 2 correctly reproduces experiments. Region 1 leads to avalanches that collapse like convex particles, i.e., the base of the pile drives the motion. Region 3 leads to fully entangled systems that do not collapse.	58
3.18	Surface plots of the tangential parameters k_t and $\gamma_t = \frac{\eta t}{2m}$ for e_n and e_t varying from 0 to 1.	58

List of Figures

4.1	A two-dimensional example of an interpolation stencil estimating the boundary flux Φ^b with second-order accuracy. The gray area denotes the solid, the red line denotes the solid boundary, and the arrows denote the five face fluxes that are computed in the cut cell of interest i, j . The circle dots denote the Eulerian cell centers while the square dots show the locations of the data points Φ from which the boundary flux $F_{i,j}^f$ is interpolated. The interpolation line normal to the solid boundary is represented by a dashed line. In case of Dirichlet boundary conditions, Φ^b is known but the data points Φ_{i+1}^f and Φ_{i+2}^f are themselves interpolated with second-order accuracy along the dotted lines using the centers of the Eulerian grid cells.	71
4.2	Stress-strain response of an isolated flat membrane for the neo-Hookean and Skalak elastic laws. The results from this study are compared to exact expressions derived in [20]. . . .	78
4.3	Schematic of a biconcave capsule centered at the origin, and definition of the polar angle θ	78
4.4	Comparison of the computed mean and Gaussian curvatures (top left), Laplace-Beltrami operator of the mean curvature (top right) and nodal bending force density (bottom) to their analytical expressions. All quantities are plotted against the polar angle θ defined in figure 4.3.	80
4.5	Taylor deformation parameter as a function of the non-dimensional time of an initially spherical Neo-Hookean capsule in an unbounded shear flow. The solid line corresponds to 32 Eulerian cells per initial diameter and a Lagrangian discretization using 1280 triangles, while the dashed line corresponds to 64 Eulerian cells per initial diameter and a Lagrangian discretization using 5120 triangles.	81
4.6	A zoomed in snapshot of a buckling membrane at $Ca = 0.0125$ with a Lagrangian discretization comprising 5120 triangles. This behavior is arising due to the absence of bending stresses, and the buckling wavelength is dependant on the Lagrangian discretization. The color field represents the x -component of the velocity.	82
4.7	Taylor deformation parameter of an isolated capsule undergoing elastic and bending stresses in a shear flow. The capillary number is $Ca = 0.05$ and the non-dimensional bending coefficient is $\tilde{E}_b = 0.0375$	84

List of Figures

4.8	Snapshots of the capsule as it flows through the constriction. Left: Park & Dimitrakopoulos [145]; Right: this study.	86
4.9	Non-dimensional lengths of the capsules in the three directions x , y and z with respect to the non-dimensional x -position of the center of the capsule. Results are compared to [145].	87
4.10	Tumbling motion of an isolated RBC in a shear flow. Top: this study; bottom: Yazdani & Bagchi [214].	88
4.11	Schematic of the two capsules in in the shear flow, prior to the interception. The horizontal and vertical gaps Δx_1 and Δx_2 are defined, and the red arrows represent the velocities of the centers of the capsules.	89
4.12	Snapshots of the interception of two neo-Hookean capsules in a shear flow. Left: Boundary Integral results of Lac et al. [115]. Right: This study. The color field corresponds to the vertical component of the velocity (rescaled for each snapshot).	89
4.13	Non-dimensional vertical gap $\Delta x_2/2a$ against the non-dimension horizontal gap $\Delta x_1/2a$ between the centers of the two capsules. The results from this study are compared to Lac et al. [115].	90
4.14	Non-dimensional vertical gap $\Delta x_2/2a$ against the non-dimensional time $\dot{\gamma}t$ for Reynolds number of 3, 10 and 50. The results of this study are compared to Doddi & Bagchi [44].	91
4.15	Qualitative comparison of a capsule flowing through a constricted channel: when the constriction size is equal to the initial diameter of the capsule (left); and when the constriction size is equal to half of the initial diameter of the capsule, with a finest Eulerian grid resolution of 100 cells per initial diameter (right). Color field: x -component of the velocity field.	93
4.16	Non-dimensional x -, y - and z -lengths of the capsule $l_x/2a$, $l_y/2a$ and $l_z/2a$ with respect to the non-dimensional x -position of the center of the capsule x_c/H_c as it flows though a constriction size equal to: (i) the initial diameter of the capsule (solid line); and (ii) half of the initial diameter of the capsule (dashed line). The dotted line corresponds to a finest Eulerian resolution of 100 Eulerian grid cells per initial diameter (shown for $x_c/H_c = a/2$ only).	94

List of Figures

4.17	Adaptive mesh around a helical geometry and its connecting pipes: (a) full computational box around the whole geometry; (b) three-dimensional trajectory of a capsule at $Re = 10, 50, 100$ (for visual clarity the helix is shrunk in the vertical direction); (c) adaptive mesh and velocity field in the vertical plane. In (c), the color field corresponds to the x component of the velocity, where blue is into the page and orange is out of the page. The capsule is about to cross the vertical plane for the third time, hence the additional small cells inside the third circular cross-section from the top. . . .	97
4.18	Trajectory of a solitary capsule in the helix at $Re = 10, 50, 100$: (a) normalized distance from the helix centerline; (b) radial migration in a plane orthogonal to the helix centerline. N^* corresponds to the number of revolutions around the vertical axis of the helix.	98
5.1	(a) Schematic of the geometry of the fluid domain. The channel has a square cross-section of side length $3\tilde{a}$. (b) Visualization of the full channel and the computational grid over the symmetry plane of the channel.	108
5.2	(a) Centroid velocity of a capsule at $Ca = 0.35$ and $Re = 50$ for two grid resolutions: 32 grid cells per initial diameter (red dotted line) and 64 grid cells per initial diameter (red solid line). The blue curve denotes the deviation in the centroid velocities for these two grid resolutions. (b) Corresponding shape and grid resolutions of the capsule and the flow field: blue means zero velocity and red means large velocity. . . .	111
5.3	Centroid velocity (a) and normalized area (b) of a capsule flowing through a corner from three distinct normalized release distances $D_c = 15, 30$ and 60 , at $Re = 50$ and $Ca = 0.35$	112
5.4	Temporal evolution of the capsule centroid velocity V at fixed Reynolds numbers.	115
5.5	Temporal evolution of the capsule centroid velocity V at fixed Capillary numbers.	116
5.6	Temporal evolution of the capsule surface area \mathcal{A} at fixed Reynolds numbers.	117
5.7	Temporal evolution of the capsule surface area \mathcal{A} at fixed Capillary numbers.	118
5.8	Maximum area \mathcal{A}_{max} as a function of Re and Ca for a single capsule passing through the corner.	118

List of Figures

5.9	Maximum (minimum) velocity V_{max} (V_{min}) as a function of Re and Ca for a single capsule passing through the corner.	119
5.10	Left: Maximum and average tensions in the capsule at $Ca = 0.35$ and $Re = 1, 25$ and 50 . Right: Capsule shape colored by $\tilde{\sigma}_2$ when $\tilde{\sigma}_{2,max}$ reaches its maximum.	121
5.11	Sequence of Capsule outlines for different Ca and Re . The time between each frame is $t = 1.5$	122
5.12	Outlines of a single capsule passing a corner with (a) maximal area \mathcal{A} and (b) maximal velocity v_{max} at various Re and Ca	123
5.13	(a) Trajectory of the two capsules at different Ca and Re . (b) Outlines of the leading and trailing capsules at $Ca = 0.35, Re = 50, d_0 = 0.25$, with comparison to a single capsule.	126
5.14	Temporal evolution of the velocity V and the area \mathcal{A} of capsules at $Ca = 0.35, Re = 50$ with $d_0 = 0.25$: a comparison of the leading, trailing and a single capsule.	127
5.15	Temporal evolution of V of the leading and trailing capsules at different Ca, Re and d_0	129
5.16	Temporal evolution of d for different initial interspacing distance d_0 and Reynolds number Re	132
5.17	Effects of the initial interspacing distance d_0 on the evolution of the capsules velocities V at $Ca = 0.35, Re = 50$	132
5.18	Time evolution of the reduced areas and velocities of ten capsules at $Re = 50$ and $d_0 = 1/8$ for $Ca = 0.15, 0.25$ and 0.35	133
5.19	Temporal evolution of d for a train of 10 capsules at $Re = 50$ and $d_0 = 0.125$ for (a) $Ca = 0.15$, (b) $Ca = 0.25$ and (c) $Ca = 0.35$	134
5.20	\mathcal{A}_{max} as a function of the capsule number.	134
5.21	(a) V_{max} and (b) V_{min} as a function of the capsule number.	135
6.1	Schematic of the Eulerian description of a capsule: the red line denotes the location of the membrane, located in cells where $0 < c < 1$; while the shaded area denotes the location of the membrane neighborhood corresponding to $\tilde{\Gamma} = 1$	144
6.2	Numerical scheme to obtain a face-centered differentiation of a cell-centered tensorial quantity. (a) Differentiation along the normal of the considered face (simple finite differencing). (b) Differentiation along a tangential direction to the considered face (involves vertex-averaging before differentiating).	152

List of Figures

6.3	L_2 -error and L_∞ -error on the normal vectors for a circular membrane, in the whole membrane region $\tilde{\Gamma}$ (top) and restricted to the interfacial cells (bottom). Two methods to compute the normal vectors in the interfacial cells are compared: (i) from the normalized gradient of the mollified VOF function, and (ii) from the height functions method presented in [153].	155
6.4	Elongational flow field leading to a constant membrane stretch.	156
6.5	Strains and stresses in a flat horizontal membrane located at $y = 0$ and subject to a constant stretch.	157
6.6	Elongational flow field leading to membrane stretch varying linearly in the x -direction.	158
6.7	(a) Surface Jacobien J around a horizontal flat membrane (not shown) subject increasing linearly along the x -direction. (b) Corresponding strains, stresses and elastic acceleration and comparison with their analytical solutions, sampled at $x = 0.25$	160
6.8	(a) Surface Jacobien J around a rotated flat membrane subject to a stretch increasing linearly along the membrane. The membrane is represented in grey. (b) Corresponding strains, stresses and elastic acceleration and comparison with their analytical solutions, sampled at $x = 0.25$	161
6.9	A neo-Hookean capsule represented by the fully Eulerian method in a shear flow. The color field represents the magnitude of the x -component of the velocity.	163
6.10	Convergence analysis of the fully Eulerian method in space (a) and time (b) for the case of an initially spherical neo-Hookean capsule in a shear flow in three dimensions. The Capillary number is 0.2.	164
7.1	Qualitative view of sheared suspensions of 100 initially spherical capsules: (a) initial configuration at a volume fraction $\phi = 5\%$, (b) configuration at $\phi = 5\%$ after 5 reference time units, (c) zoom on interacting capsules located in the lower right corner of (b), and (d) visualization of the adaptive grid at time $t = 0$ and at $\phi = 25\%$. The open capsules are a visualization artifact due to periodic boundary conditions. . .	178
7.2	(a) Initial pre-collision configuration. (b) Collision of the top capsule with the channel walls. (c) Post-collision configuration.	180

List of Figures

A.1	Two stages of broad-phase detection: space-discretization in cells and bounding boxes. Particles with the same color will undergo the bounding boxes broad-phase detection, and particles which bounded-boxes overlap will launch narrow-phase detection.	208
A.2	An example of a support mapping $s_{\mathcal{C}}(\boldsymbol{v})$ for the convex \mathcal{C} in the direction \boldsymbol{v}	209
A.3	Minkowski sum and difference of two convex bodies.	210
A.4	The first iterations of the GJK distance algorithm.	211
A.5	When two particles are colliding, the access to the contact information is performed by a second call to the GJK algorithm on slightly shrunk versions of the particles. In this schematic (not to scale), the dotted lines represent the original shape of the particles and the solid lines correspond to their shrunk version.	212
B.1	Non-dimensional final x-displacement of the center of mass of the granular assembly (diamond symbols) and non-dimensional run-out distance (all other symbols) as a function of the initial aspect ratio.	214
B.2	Schematics of the avalanche profiles of the granular assembly for low (a) and high (b) aspect ratios, under the assumption that the slope of the avalanched profile is constant for a given aspect ratio.	215
B.3	Probability density functions (PDFs) of the contact forces F_c just before the gate is opened, in the cases of (a-b) top-driven regime (black) with buckling regime (blue); and (c-d) intermittent regime leading to stable (black) and collapsed (blue) behaviours.	217
C.1	Tree structure of a quadtree grid. The filled circles represent leaf cells while the open circles represent parent cells.	220

List of Figures

C.2	Examples of adaptive IBM stencils configurations. (a) The cell containing \mathbf{x}_0 is at the largest level of the stencil. (b) The cell containing \mathbf{x}_0 is at the lowest level of the stencil. (c) The cell containing \mathbf{x}_0 is neither at the largest nor the lowest level of the stencil. (d) Two successful increases of the stencil size are necessary to obtain a consistent stencil. The dashed line represents the stencil size based on the level of the cell containing \mathbf{x}_0 , while the dotted line represents the effective stencil size. The red dot corresponds to the location \mathbf{x}_0 of the center of the regularized Dirac distribution.	221
C.3	Visualization of the body force term in the x -direction (left) and the IBM stencils defined by a noisy field (right) in the case of a two-dimensional capsule in a shear flow where a forced grid refinement is imposed in parts of the computational domain.	224
C.4	Taylor deformation parameter of a two-dimensional capsule in a shear flow where a forced grid refinement is imposed in parts of the computational domain. Three cases are considered: (i) the membrane is at a constant fine level of refinement (solid black line), (ii) the membrane is at a constant coarse level of refinement (dashed black line), (iii) the membrane spans two levels of refinement (red line).	225
D.1	Temporal evolution of the reduced area \mathcal{A} of the leading and the trailing capsules at various interspacings d_0 and various Capillary and Reynolds numbers Ca and Re	227
D.2	Time evolution of the reduced areas and velocities of ten capsules at $Re = 50$ and $d_0 = 1$ for $Ca = 0.15, 0.25$ and 0.35 . . .	228

List of Algorithms

4.1	Pseudocode for the Finite Element Method	76
A.1	The GJK distance algorithm	211
C.1	Recursive averaging of velocities on leaf cells	222
C.2	Recursive spreading of node force onto leaf cells	223

Acknowledgements

I thank my supervisor Prof. Anthony Wachs for welcoming me in his group, for giving me the opportunity to conduct this research and to present it at multiple occasions, as well as for patiently proof-reading this thesis. I also thank the members of my supervisory committee, Prof. Brian Wetton and Prof. Ian Frigaard, as well as my university and external examiners, Prof. Gwynn Elfring, Prof. Chen Greif and Prof. Pierre-Yves Lagrée, for agreeing to evaluate this dissertation.

Without a doubt, this PhD was a long and challenging journey, but the people around me made it worthwhile. I am immensely grateful to have been surrounded by incredible group members throughout these four and a half years: Can, Mohammad, Christine, Arun, Michael, Arman, Arthur, Aashish, Zihao, Layal, Anirudh, Guodong, Antoine, João, Alireza and Anke. Your support, skills, open mind and kindness helped me tremendously both in my PhD and in my personal journey throughout these years. Some special thanks should be addressed to Can, Arthur, Guodong and Antoine: thank you for sharing your research experience and wisdom, and for helping me strive towards the highest standards of research integrity and transparency. To Aashish and Guodong, with whom I spent most of my time in the lab: thank you for your patience through my endless complaints, and for your willingness to actively listen, challenge my assumptions, and provide support in whatever form I needed. It has been a true honor to share my graduate life with such kind and talented individuals.

Of course my PhD would not have been possible without the unfailing support of my family. To my parents, my sisters and their partners, thank you for understanding my urge to see the world beyond France's borders, first in Sweden and now in Canada. Thank you for always supporting me and making me feel included despite the distance. To my family in Canada, thank you for welcoming me as one of yours and for always encouraging me, celebrating the milestones along the way. To my wife, thank you for your patience and your support, both of which proved to be infinite. You inspired me to grow from these challenging years. Thank you from the bottom of my heart.

Chapter 1

Introduction

Blood is the main carrier of material within the human body, therefore accomplishing many vital tasks. To name a few, the vascular network is responsible for delivering oxygen and nutrients to tissues, transporting humoral and cellular immune response agents to infected areas, stabilizing the temperature and pH of the body, as well as carrying hormones to all components of the endocrine system, thereby controlling the body's metabolism, growth, mood, emotions, sexual function, sleep and blood pressure [74]. As a result, any factor altering properties of the vascular network can have striking consequences: for instance, the formation of an abnormal buldge in the wall of a blood vessel, called an aneurysm, can burst and result in internal bleeding, often leading to death [178]. In oncology, the vascular network is also the main vector of migration of cancerous cells – those cells are referred to as Circulating Tumor Cells (CTCs) –, which can lead to metastases [5, 161, 162]. Because blood is critical in such a vast range of bodily functions, efforts to better model and characterize blood flow can have dramatic impacts in a wide range of medicinal applications. For example, a finer understanding of cell-wall adhesion in blood vessels can help design targeted drug delivery applications, which could dramatically reduce the side effects of heavy cancer treatments such as chemotherapy [11, 113]. Another field of hemodynamics – the study of blood flow – under active research is that of lab-on-chip devices. These microfluidic devices have a carefully engineered geometry in order to accomplish various tasks, such as skimming blood plasma and sorting cells based on size or deformability; and they have already enabled fast and inexpensive cell-segregation and cell-characterization procedures [22, 52, 55, 96, 192].

Blood is a complex fluid: its microscopic structure is composed of the blood plasma (57%¹) laden with Red Blood Cells (RBCs) (about 42%) and other suspended cells such as white blood cells and platelets (about 1%) [74]. While a wide variety of proteins are suspended in the blood plasma, it is not a strong assumption to model it as a Newtonian fluid of viscosity

¹The percentages indicate volume fractions.

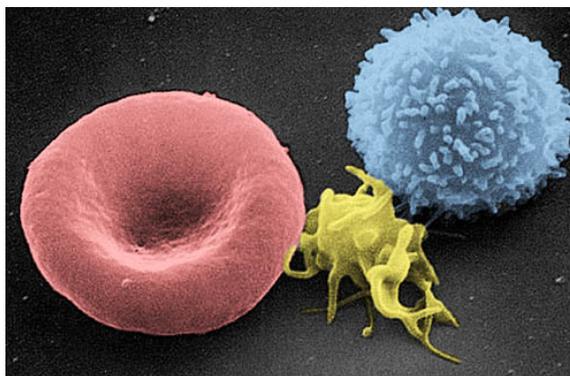


Figure 1.1: A red blood cell (left), an activated platelet (center) and a white blood cell (right) as seen using electron microscopy imaging [10].

$\mu_p = 1.2\text{mPa} \cdot \text{s}$ at 37°C , about 50% greater than that of water at the same temperature [74, 107]. RBCs are highly deformable cells and can be modelled as capsules, i.e. a deformable, elastic membrane enclosing an inner fluid [19]. In the case of RBCs, the inner fluid is the hemoglobin-rich cytoplasm: RBCs are red because the iron present in the hemoglobin molecule is oxidized when the blood is rich in oxygen. At rest, a typical RBC adopts a biconcave shape with a largest diameter of about $7.8\mu\text{m}$ [160]. White blood cells and platelets are stiffer and can be modelled as solid particles: a sphere in the case of white blood cells and an oblate spheroid in the case of inactivated platelets. The diameter of a typical white blood cell is approximately 80% of the greatest diameter of an RBC, and that of a typical inactivated platelet is approximately 20% of the greatest diameter of an RBC. A size and shape comparison of an RBC, a white blood cell and a platelet (activated) is shown in figure 1.1.

While blood flow research was historically driven by experiments, the rapidly rising computing power enabled the first mesoscale three-dimensional simulations of blood flow in 2010 [219]. Undeniably, the major advantage of numerical simulations of biological flows over experiments is the ability to produce cheap, on-demand, and reproducible studies. Indeed, blood is difficult to study experimentally for a number of practical reasons. First, blood can contain viruses and diseases and experiment in hemodynamics must follow a strict biohazard procedure. Moreover, a pure blood sample must usually be used within a few hours from its collection from a donor due to several factors ranging from coagulation to the natural degradation of cells [210], thus adding to the logistic constraints of hemodynamics exper-

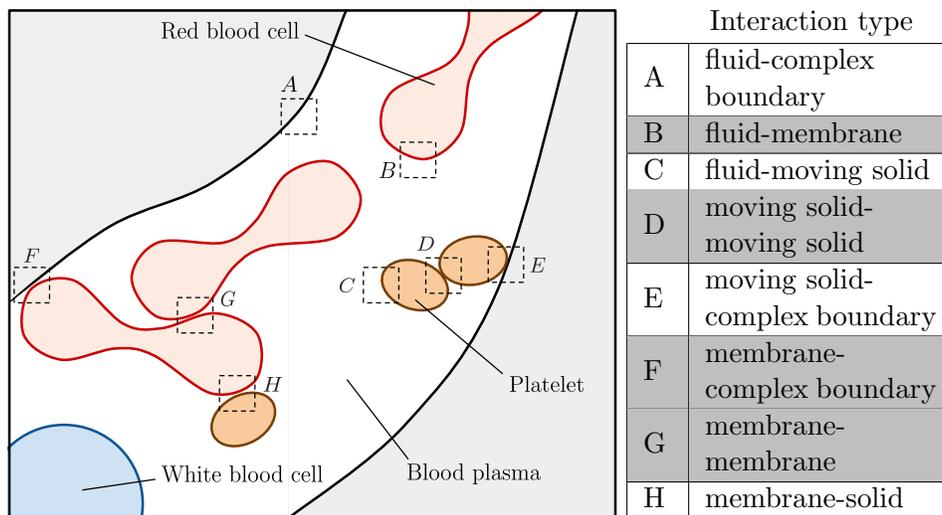


Figure 1.2: Diagram listing the types of interactions in a typical blood sample. The highlighted rows in the table show the interactions actively investigated in the present thesis. For readability, the volume fraction of the blood composition is not respected.

iments. Second, blood composition varies from donor to donor: for instance, the hematocrit – i.e. the volume fraction of RBCs – we indicate above is an average value. In reality, healthy hematocrit levels vary from 41% to 50% in men, and from 36% to 44% in women [135]. Carrying reproducible experiments with such variations of the fluid properties from donor to donor can become a challenging task. As a result, the cost of conducting an experimental study on human blood is in most cases orders of magnitude greater than the cost of renting computing power on a cloud system to carry out the same study numerically. Moreover, numerical simulations allow to investigate quantities that are not accessible to the experimentalist, because the complete three-dimensional flow and force fields are known. This is useful for example if the strains and stresses in the cell membranes are of interest, e.g. to characterize the influence of a channel geometry on the breakability of the cells. For all these reasons, numerical simulations have become increasingly popular in the study of biological fluid flows, especially in applications involving iterative designs such as microfluidic devices, in which the shape of the microchannels is optimized to accomplish a given task.

The goal of the present thesis is to develop numerical tools capable of simulating biological fluids such as blood at the cellular scale. At this scale, all combinations of interactions between fluid, deformable membranes, solid

particles and complex channel geometries need to be considered, as represented in figure 1.2. In part I of this thesis, including Chapters 2 and 3, we focus on the dispersed solid phase and employ the Discrete Element Method (DEM). In the context of figure 1.2, we model interaction D in the very general context of particles of arbitrary shapes. In part II of this thesis, including Chapters 4, 5 and 6, we focus on the simulations of deformable cells and employ a Front-Tracking Method (FTM) on adaptive grids. The FTM models interaction B of figure 1.2, while our adaptive mesh captures the small lubrication layers around deformable membranes, thus treating interactions F and G. The interactions listed in figure 1.2 that are not treated in this thesis are either: (i) well known in the literature and already implemented in our flow solver – namely interactions A and C; or (ii) under current investigation by a group member [139] – namely interactions E and H. Chapter 7 contains a discussion related to parts I and II as well as concluding remarks. In the following two sections, we present a general overview of the context, motivations and significance associated with the two themes of this thesis, while a considerably more in-depth description of the conducted research will be presented in the main body of this document.

Finally, the research conducted in this thesis adheres to high standards of reproducibility and open-source development practices: in part I we take great care to describe extensively our numerical setup, and in particular we list all of the contact model parameters that our solver requires. Moreover our DEM solver will soon be released open-source for everyone to browse and use. In part II, we contribute to an existing open-source software and release our code on the software’s website together with an extensive documentation. Additionally, the input files and post-processing scripts to run every simulation and reproduce every figure presented in part II are also released openly [80].

1.1 Numerical modelling of the solid phase

In the first part of this thesis, we focus on numerical simulations of a dispersed solid phase. While our initial motivation is to study flowing and colliding white blood cells and platelets, which are approximately rigid bodies, our problem can be generalized to all materials made of dispersed solid particles. Consequently we abstract our original problem and we choose to focus on the more general field of granular mechanics. Surprisingly, granular media are far from being well understood despite being the second-most

handled material in industry². While major efforts were undertaken in the past two centuries to describe the behavior of granular media, we still cannot predict with certainty if a cup of rice poured into a funnel will jam or see its flow rate undisturbed; nor can we predict the value of the angle of repose of a pile of particles which shape deviates from the sphere. Still, several approaches can be used to describe granular media and gain insight into their macroscopic properties. If the system of interest is very large, comprising over 10^6 particles, a continuum approach can be used. In this approach, experimental data is gathered and analyzed at a macroscopic level, with the goal to model the relation between the strain applied to the granular material and the resulting stress. In the early 2000s, GDR MiDi, a group of researchers, conducted and analyzed a wide range of experiments and postulated the $\mu(I)$ -rheology phenomenological law, which expresses the viscosity μ of the granular fluid as a function of an inertial number I relating the time scale of microscopic rearrangements and the time scale of macroscopic deformations [56, 117]. This $\mu(I)$ -rheology law allows to formulate a stress-strain relation for the granular material and to find an approximate solution using Navier-Stokes solvers, effectively treating the granular material as a non-Newtonian fluid. Despite being a giant leap forward in the modelling of granular flows, the $\mu(I)$ -rheology approach is limited to very big granular systems composed of monodispersed spheres. If another type of granular material is of interest, costly and time-consuming experiments would need to be performed for this specific type of material before a new $\mu(I)$ -rheological law can be developed and applied only to very big systems. As a result, concomitant to the advances of the continuum approach, numerical methods were developed in order to address smaller systems of particles for which an averaging approximation is not feasible. The most versatile of these numerical methods is the Discrete Elements Method (DEM), in which the dynamics of each individual particle is resolved using Newton's second law of motion. High-performance DEM codes can represent at most 10^8 spherical particles, making this approach irrelevant for very large granular systems featuring hundreds of billions of particles, but suitable for the biological applications we are interested in, which could comprise at most a few hundreds to a few thousands platelets and white blood cells. Moreover the collisions between individual particles need to be resolved in time: because the contact duration is very short for most materials, the time steps employed in DEM are very small and these simulations are limited to small physical time scales. This restriction is compatible with the simulations of cell-resolved biological

²The most handled material in industry is water [102].

flows, as the time scale of interest is at most of the order of seconds. At the core of the DEM is the contact model, which dictates the force exerted on two colliding particles. In most DEM programs, a simple contact model valid for fast collisions is used. However, whenever long-duration contacts occur, a better contact model with a so-called “memory” term that keeps track of the cumulative tangential displacement at the contact point is necessary in order to accurately simulate frictional static granular assemblies. Moreover, the vast majority of DEM software can only represent particles of spherical or spheroidal shapes, while in practice most granular materials are made of particles of arbitrary shape, often non-convex and featuring sharp edges. In the first part of this thesis, we implement an accurate contact law in a DEM software, and we proceed to using this accurate numerical method to analyze granular systems composed of non-convex, entangled particles.

In Chapter 2, we focus on the implementation and the validation of the new contact model in the open-source software Grains3D. We begin by presenting an overview of the DEM, of the more accurate contact model with memory that we implement, and of the various integration schemes that are relevant in the context of the DEM. We note that in DEM contact models, some model parameters are difficult to estimate. As such, we review the literature on this topic and we provide a procedure to set contact model parameters in a rigorous fashion. Then, we proceed with the validation of our DEM solver through a series of test cases. The first validation case is a simple sphere bouncing on a horizontal plane. It aims to demonstrate a degenerescence to first order in the accuracy of any DEM time integration scheme, which is rarely discussed. The second validation case is a measure of the average velocity of a pile of spherical particles confined in a box. This test demonstrates that the memory contact model allows the granular assembly to reach a true static case, in which the translational and angular velocities of the particles approach the machine epsilon. Our third validation case is a quantitative validation of macroscopic predictions. In this case, we consider a hopper discharge of particles forming a conic pile on a flat surface, and we compare the repose angle of the pile to that obtained in the literature. Finally, concluding remarks and perspectives are formulated.

With a validated DEM solver able to represent static and dynamic assemblies of particles of arbitrary shapes, we proceed to Chapter 3, where granular avalanches of entangled particles are analyzed. Our goal is to show the dramatic effect of the particle shape on the overall dynamics of a granular assembly. As such, we choose cross-shaped particles in order to enhance the entanglement of our granular medium, and we consider a dam-break setup with a varying aspect ratio, in which a confined granular column is

released by removing one of the lateral walls. We compare our simulations to experiments conducted by collaborators at the University of Twente, and we show excellent qualitative and quantitative agreements between the numerical data and the experimental data. Qualitatively, very interesting and novel collapsing dynamics are observed, resulting from the entanglement of the particles. Such behaviors include the top-driven collapse, the buckling collapse, and the metastable regime in which the granular column collapses or remains stable in a non-deterministic way. Quantitatively, the runout distance and the height of the granular column are measured and show surprisingly similar trends to that observed in the case of dam-break setups of spherical particles. Our simulation data allows us to access quantities that are not measurable experimentally, and we proceed to analyze the force contact network, the fabric anisotropy and the probability density functions of contact forces, in an attempt to better characterize the metastable regime. We found that none of these measures allows to fully determine whether a metastable assembly will collapse or remain stable after the gate is removed, suggesting that the macroscopic outcome of the entangled granular medium we consider is determined at the mesoscale. Unfortunately, we also found that when the particle shape significantly deviates from that of a sphere, the rigorous procedure presented in Chapter 2 to set the DEM contact model parameters is not valid. As such, we had to rely on trial and error to set our contact model parameters, which is undoubtedly a major limitation of the DEM.

1.2 Numerical modelling of immersed deformable capsules

The second part of this thesis is devoted to the numerical simulations of flows laden with deformable capsules. Such flows display a complex behavior: for instance, experiments show that human blood displays a non-Newtonian behavior when observed at the macroscale; even though the fluids inside and outside the RBCs are both Newtonian at the microscale. The non-Newtonian, shear-thinning, macroscopic behavior of blood is the result of the anisotropy of the RBCs, which tend to adopt random orientations at low shear rates while aligning their longest axis with the flow streamlines at high shear rates, thus resulting in a nonlinear relation between the sample viscosity and the applied shear rate. In light of this observation, if the physical scale of the problem is much larger than the scale of a capsule, modelling blood – or more generally, capsule-laden flows – using a single-phase,

non-Newtonian fluid is a valid approach that has been applied successfully in the case of blood flow simulations [98, 138]. If the problem of interest involves features of size less than two orders of magnitude greater than that of a capsule, such as microchannels in lab-on-chip devices, then the effect of the capsules cannot be averaged and a non-Newtonian single-phase fluid description is not appropriate. In this case, the two-phase description of blood as a capsule-laden fluid has to be adopted, as sketched in figure 1.2. In non-inertial regimes, found in small capillary vessels or in some non-inertial microfluidic devices, the Navier-Stokes equations describing the inner and outer fluids become the linear Stokes equations, and a Boundary Integral Method (BIM) can be used to simulate such flows. In this method, only the boundaries of the fluid domains need to be discretized – i.e. the capsule membrane and the walls of the blood vessel or of the microchannel –, making this approach very computationally efficient even for large, three-dimensional problems featuring thousands of capsules. However, these fast BIM solvers cannot be applied to inertial regimes: when higher velocities are considered, the full nonlinear Navier-Stokes equations need to be solved, and the whole 3D space must be discretized. This approach has been successfully employed to simulate inertial and non-inertial capsule laden-flows, both in microfluidic-relevant geometries [15, 129] and in geometries resembling networks of capillary vessels [7, 16]. In both cases, the relevant size of the problem is much greater than the size of a given capsule, while the dynamics of the flow is determined by the boundary layers along the capsule membranes and the walls. As a result, very fine grid discretizations have to be employed in order to capture the physics that happens in the boundary layers. Because of ease of programming and of designing numerical methods, most Navier-Stokes flow solvers utilize a constant (and often uniform) Cartesian grid in the whole simulation domain, resulting in a fine discretization everywhere – even in areas far away from capsules and walls. This is a sub-optimal usage of computational resources. In part II of this thesis, we implement an adaptive capsule-laden flow solver suitable for inertial and non-inertial regimes, and able to coarsen the grid discretization in regions where high computational accuracy is not needed, enabling our solver to be applied to much larger geometries than its constant grid-size counterparts.

In Chapter 4, we focus on the implementation of our adaptive front-tracking solver. After reviewing the literature on numerical methods for capsule-laden flows, we present the equations governing this problem and we describe the numerical method and the implementation approach that we follow. We then proceed to the validation of our solver: we first validate the finite element method and paraboloid-fitting method computing the elas-

tic and bending stresses on the capsule’s membrane. Then, we consider the full capsule and fluid problem and we compare the results produced by our solver to the literature in a series of validation cases of increasing difficulty, both in inertial and non-inertial regimes. The two final cases we present are demonstration cases that would not be feasible without our adaptive grid: the first case features extreme deformations of a capsule as it squeezes through a narrow constriction, while the second case shows the inertial migration of a single capsule in a very large, three-dimensional helical channel geometry. Finally, we conclude with a discussion on performance, limitations and prospects for this first-of-its-kind adaptive solver for capsule-laden flows.

In Chapter 5, we use our newly implemented adaptive front-tracking solver to study the dynamics of capsules through a sharp corner: a very common geometry in microfluidic devices. After validating our setup for this specific case, we analyze the trajectory, normalized velocity and area deformations of the capsules passing through the corner for channel Reynolds numbers varying from 10^{-2} to 50. This study is performed for a solitary capsule passing through the corner, as well as for a pair of capsules and for a train of ten capsules. We show that, surprisingly, the dynamics of one capsule is not significantly altered by following or preceding capsules, even in the presence of inertia.

Finally, we present in Chapter 6 our initial attempt to implement an adaptive capsule-laden flow solver. Prior to implementing our front-tracking solver described in Chapter 4, where the capsule membrane is discretized using an unstructured mesh (a triangulation), we attempted to compute the membrane stresses by relying solely on the Cartesian grid: an approach referred to as the fully Eulerian method. After a review of the relevant literature, highlighting the promises of this method, we describe the essence of the fully Eulerian method and its associated equations to be solved on the Cartesian grid. We then present our implementation strategy and a series of validation cases. We finish by discussing our hypotheses as to why our implementation of this method was not successful, as well as some inherent limitations associated to fully Eulerian methods.

Part I

Dispersed solid phase

Chapter 2

Accurate static contact law for the Discrete Element Method (DEM)

2.1 Introduction

Granular materials are still far from being well understood despite their ubiquitous presence in natural and industrial processes. Yet, in many applications it is critical to have reliable tools to provide both qualitative and quantitative information about granular materials. The information sought can include flow conditions, jamming or buckling effects and stability of a granular assembly; as well as the flow rate of a hopper discharge, the run-out distance of a landslide or the added mass on walls from the Janssen effect [8]. Historically, granular systems have been studied by conducting experiments of simplified systems, e.g. a dam break collapse or a hopper discharge, with the aim to derive empirical laws or to challenge theoretical models. A short review of the experimental work on dam break collapses is given in Section 3.1. However, conducting experiments is a long and costly process, and in most cases only global quantities such as the angle of repose or the average porosity are accessible to the experimentalist – a notable exception is the velocity field of two-dimensional flows which can be accessed via Particle Image Velocimetry (PIV) and Particle Tracking Velocimetry (PTV) techniques [2, 41]. In the past three decades, the increase of computing power allowed the emergence of computational methods for the simulation of granular systems. Such numerical simulations alleviate some of the shortcomings of the experimental approach: a numerical study can be conducted for a fraction of the cost of its experimental counterpart, and it allows fine analyses of both global and local quantities. Numerical simulations also don't suffer from limitations of the experimental apparatus itself: for instance, it is easy to numerically simulate a dam break collapse with no friction between the opening gate and the granular medium, or with

2.1. Introduction

no vibration from the mechanism lifting the gate, thus providing simulation data free of imperfections from the experimental apparatus. Of course, numerical simulations have their own drawbacks, which typically depend on the specific simulation method considered.

There are many ways to numerically study granular systems, and the choice of a given method depends on the specific application. For systems with a very large amount of particles such as powders or large industrial applications with over a billion particles, a continuum approach must be used. In this class of problems, if the granular system is composed of spherical particles, the $\mu(I)$ -rheology [56, 117] has proved to be a reliable rheological model able to accurately reproduce transient flows and steady-state quantities from experiments and analytical solutions. When smaller granular systems are considered, or when the particle shape deviates from the sphere, the Discrete Element Method (DEM) is preferred in one of its two variations. First, the smooth DEM can be used with a large amount of particle – up to 10^8 – but it imposes strict restrictions on the time step as it is controlled by the time scale of the collision time, which is orders of magnitude lower than that of the time scale of interest. Second, the stiff DEM does not show such strict time step restriction as it relies on an implicit numerical method, but it is only appropriate for a relatively small number of particles – up to a few thousands – since a linear system of size N is solved, with N being the number of particles. In the present work, we are interested in short-lived systems with a reasonably large amount of arbitrary-shaped particles ranging from a few thousands to a few millions, implying the use of the smooth DEM.

The goal of this chapter is to present the smooth DEM with a strong emphasis on the contact model and the time integration of the equations of motion. We propose to implement an accurate contact model in the already existing code Grains3D [168, 169, 202] and we show that in most applications, a so-called “memory” contact model is necessary to accurately reproduce the long-term behavior of the system. This chapter is organized as follows. In Section 2.2, we describe the framework of the smooth DEM. The accurate contact model we implement and the choice of the model parameters are presented in Section 2.3. We then give a short review of the numerical integration of the equations of motion in translation and in rotation in Section 2.5. In Section 2.6, we demonstrate the performance of the various integration schemes and we validate our implementation of the memory contact model on systems composed of spherical particles. We discuss our findings and conclude in Section 2.7.

2.2 General structure of the Discrete Element Method

The numerical simulations in this study are performed using the standard soft-body Discrete Element Method (DEM) as introduced by Cundall and Strack [40]. In DEM, each particle is tracked individually, and the position and velocity of particle i are determined by explicitly integrating Newton's second law of motion:

$$\frac{d\mathbf{v}_i}{dt} = \frac{\mathbf{F}_{i,ext}}{m_i} \quad (2.1)$$

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{v}_i \quad (2.2)$$

$$\frac{d\mathbf{J}_i \cdot \boldsymbol{\omega}_i}{dt} = \mathbf{M}_{i,ext} \quad (2.3)$$

$$\frac{d\boldsymbol{\theta}_i}{dt} = \boldsymbol{\omega}_i \quad (2.4)$$

where t is the time, \mathbf{x}_i the position of the center of mass of particle i , m_i its mass, \mathbf{v}_i its velocity, \mathbf{J}_i its inertia tensor, $\boldsymbol{\theta}_i$ its angular position, $\boldsymbol{\omega}_i$ its angular velocity, and $\mathbf{F}_{i,ext}$ and $\mathbf{M}_{i,ext}$ are the sums of external forces and torques exerted on the particle, respectively.

A change of momentum for a given particle is due to the action of body forces, such as gravity, as well as inter-particle or particle-wall contact forces. Figure 2.1 shows a top-level description of a typical DEM algorithm, that can be summarized in three steps to be performed at each time step and for each particle: (i) the contact detection, (ii) the computation of forces due to inter-particle or particle-wall contacts and (iii) the numerical integration of the equations of motion Eq. (2.1)-Eq. (2.4).

Step (i) is a purely computational problem and its complexity depends on the shape of the particles at play. In the case of spherical particles, the contact is trivially assessed from the position of the centers of mass of the two potentially contacting spheres, leading to a very fast detection (the same applies to a sphere-infinite plane contact). Several approaches are possible in order to handle the detection of contacts between non-spherical particles. The method of glued spheres involves approximating any shape by a cluster of N overlapping spheres. Two glued-spheres particles are in contact if at least two of their sub-spheres overlap. As N increases, the approximation of the shape improves but the contact detection becomes more computationally expensive as it scales with N^2 . Several studies [101,

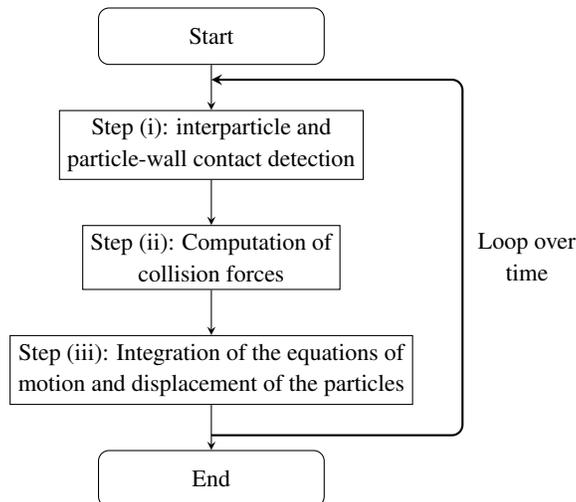


Figure 2.1: Top-level view of a soft-body DEM algorithm

[112, 169] have mentioned that the dynamics is severely perturbed by the artificial surface roughness, and Kruggel-Emden [112] shows that having N of the order of 10^2 is still not enough to obtain realistic trajectories for a single quasi-sphere bouncing on a flat surface. This makes the method of glued spheres out of reach, or at least extremely inefficient with respect to the following methods. If particles are convex polyhedra, the Common Plane algorithm as used by Cundall [39] and later improved by Nezami et al. [143] has shown remarkable performance and stability. At the price of rounded edges, particles of arbitrary shapes can also be described by super-quadric functions. In that case, the contact detection between two super-quadric shapes is an optimization problem the complexity of which depends on the level of angularity of the shapes [149]. For applications where angular and non-polytope particles need to be described, the most versatile contact detection algorithm has been proposed by Gilbert, Johnson and Keerthi [62]. It involves solving iteratively small linear systems of size 4 (in 3 dimensions) until a reasonable approximation of the contact point has been found. Our DEM solver Grains3D uses the Van den Bergen implementation of this so-called GJK algorithm [197]. Since the GJK algorithm is limited to convex bodies, non-convex particles are handled by decomposing their shape into a set of elementary convex particles on which the GJK algorithm can be performed successively, as described in [169]. A brief overview of the GJK algorithm is presented in Appendix A.

Step (ii) will be discussed in detail in the next two sections, and a short

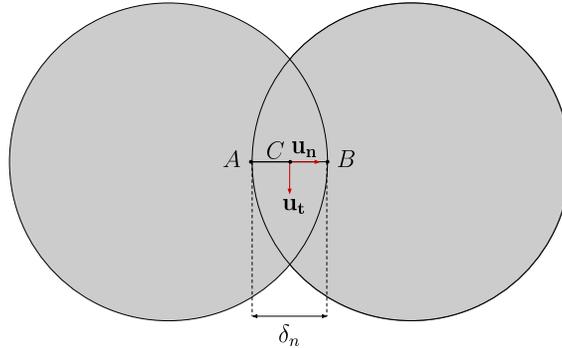


Figure 2.2: In soft-body DEM, a contact is modelled with an overlap – Please note that the overlap is intentionally magnified here for visual purposes.

review of step (iii) is presented in Section 2.5.

2.3 Contact model

In soft-body DEM, the computation of contact forces is critical as it determines the mechanical response of the system. Contact forces are computed using geometrical features of the contact region, defined in figure 2.2. As shown in figure 2.2, the contact is modelled as an overlap of the particle shapes. We denote by \mathbf{u}_n and \mathbf{u}_t the unitary normal and tangential vectors, respectively, at the contact point, by C the point of application of the force and by δ_n the penetration depth – also called overlapping distance. Note that the point C here is chosen as the midpoint of the segment \overline{AB} formed by the two points that define the greatest penetration depth δ_n . While this choice is natural for spherical particles, the point of application of the force is less straightforward for particles of complex shape, especially when the physical force is not applied at a single point but is distributed over a line or a surface. Seelen et al. [181] discuss ways to impose a physically meaningful contact point for most line- and surface-distributed force configurations. In this study, we consider the occurrence of face-face and face-edge contacts to be minimal with respect to the rest of the contact configurations and model face-face and face-edge contacts with a single contact point similarly to the way we model the other contact configurations. DEM that represent particles as triangulated surfaces [73, 215] or by a Discrete Function Representation (DFR) [126, 209] are naturally able to distribute the force over the multiple contact points of one particle that by construction coincide with nodes of a triangulated face that belong to the other particle and vice-versa.

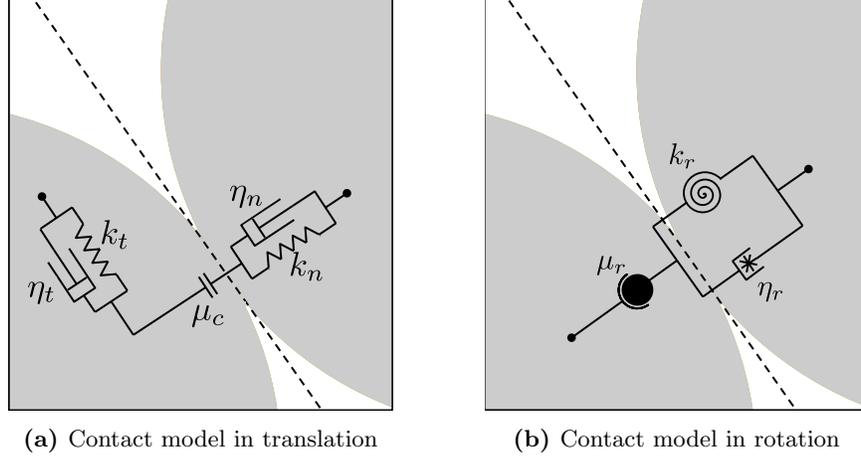


Figure 2.3: The linear spring-dashpot contact model as a mechanical analog. The dashed line represents the tangential contact plane.

Once the point of application of the contact force is defined, two contact models are typically used to compute the contact forces: the Hertz dashpot model and the linear spring-dashpot model. While the former has the advantage of being analytically derived for the contact of two spheres and other simple shapes – such as a sphere-plane configuration – there is no analytical generalization to arbitrary shapes, therefore excluding its use in this study. As a result, we adopt the latter spring-dashpot model – its schematic view is presented in figure 2.3 – and we elaborate below on its formulation and implementation in translation both in the normal direction and in the tangential direction. The formulation of the contact model in rotation is given in Appendix 2.3.2.

2.3.1 Contact model in translation

The normal contact force \mathbf{F}_n is expressed as follows:

$$\mathbf{F}_n = \max \left(k_n \delta_n + \eta_n \frac{d\delta_n}{dt} , 0 \right) \mathbf{u}_n \quad (2.5)$$

where k_n and η_n are the normal stiffness coefficient and the normal damping coefficient, respectively. The normal force represents the viscoelastic response to the contact, and is intrinsically repulsive. Note that at the end of the contact, the term $k_n \delta_n$ could become smaller in absolute value than

2.3. Contact model

the negative term $\eta_n d\delta_n/dt$. Therefore, the role of the $\max(\cdot, 0)$ condition is to prevent \mathbf{F}_n from becoming an attractive force.

The tangential friction force \mathbf{F}_t is also modeled as a linear spring-dashpot, though slightly modified to allow for the Coulomb friction saturation:

$$\mathbf{F}_t = \min(\| -k_t \boldsymbol{\delta}_t - \eta_t \mathbf{v}_t \|, \mu_c \|\mathbf{F}_n\|) \mathbf{u}_{t,\text{cumul}}, \quad (2.6)$$

$$\mathbf{u}_{t,\text{cumul}} = -\frac{k_t \boldsymbol{\delta}_t + \eta_t \mathbf{v}_t}{\|k_t \boldsymbol{\delta}_t + \eta_t \mathbf{v}_t\|} \quad (2.7)$$

where $\mathbf{u}_{t,\text{cumul}}$ is the cumulative tangential unit vector, \mathbf{v}_t is the tangential relative velocity, k_t and η_t are the tangential stiffness coefficient and the tangential damping coefficient, respectively, and $\boldsymbol{\delta}_t$ is the cumulative tangential displacement. The reason $\mathbf{u}_{t,\text{cumul}}$ is introduced instead of simply having \mathbf{F}_t in the direction of $-\mathbf{v}_t$ is to keep a meaningful tangential direction even when the system reaches a static state, i.e., when $\|\mathbf{v}_t\|$ approaches the machine epsilon. The term $\boldsymbol{\delta}_t$ is often referred to as *history-* or *memory-*term because it stores information about the contact at previous instants. It is expressed as:

$$\boldsymbol{\delta}_t = \begin{cases} q_{rot} \boldsymbol{\delta}_t^{t-\Delta t} q_{rot}^{-1} + \int_t^{t+\Delta t} \mathbf{v}_t(s) ds & \text{if } \|F_t\| \leq \mu_c \|\mathbf{F}_n\| \\ \frac{-\mu_c \|\mathbf{F}_n\| \mathbf{u}_{t,\text{cumul}} - \eta_t \mathbf{v}_t}{k_t} & \text{if } \|F_t\| > \mu_c \|\mathbf{F}_n\| \end{cases} \quad (2.8)$$

In Eq. (2.8), $\boldsymbol{\delta}_t^{t-\Delta t}$ is the cumulative tangential displacement at the previous instant (if the contact existed), q_{rot} is the rotation quaternion from the tangential plane at the previous instant to the current tangential plane, and μ_c is the Coulomb sliding coefficient. If we neglect the rotation of the tangential plane, $\boldsymbol{\delta}_t$ becomes $\int_{t_0}^t \mathbf{v}_t(s) ds$ and can be seen as the length of the contact path since the contact start time t_0 . This is at least true until $\|F_t\|$ reaches its saturation value $\mu_c \|\mathbf{F}_n\|$. If the tangential friction desaturates, starting a new path integral $\int_0^{\Delta t} \mathbf{v}_t(s) ds$ would result in a sudden and unphysical discontinuity in \mathbf{F}_t . Defining $\boldsymbol{\delta}_t$ as in Eq. (2.8) ensures that the tangential friction will remain continuous through desaturation [36, 133].

2.3.2 Contact model in rotation

The friction torque \mathbf{M}_r is modeled in a similar fashion as the tangential force, and is a three-dimensional adaptation of the two-dimensional work of Ai et al. [4]:

$$\mathbf{M}_r = \mathbf{M}_k + \mathbf{M}_d \quad (2.9)$$

2.4. Contact model parameters

with \mathbf{M}_k and \mathbf{M}_d defined incrementally as follows:

$$\mathbf{M}_k^{t+\Delta t} = \begin{cases} q_{rot} \mathbf{M}_k^t q_{rot}^{-1} - k_r \Delta \boldsymbol{\theta} & \text{if } \|\mathbf{M}_k^{t+\Delta t}\| \leq M_{max} \\ M_{max} \mathbf{u}_{r,cumul} & \text{otherwise} \end{cases} \quad (2.10)$$

$$\mathbf{M}_d = \begin{cases} -\eta_r \frac{d\boldsymbol{\theta}}{dt} & \text{if } \|\mathbf{M}_k^{t+\Delta t}\| < M_{max} \\ 0 & \text{if } \|\mathbf{M}_k^{t+\Delta t}\| = M_{max} \end{cases} \quad (2.11)$$

$$\text{with } \mathbf{u}_{r,cumul} = \frac{\mathbf{M}_k^t - k_r \Delta \boldsymbol{\theta}}{\|\mathbf{M}_k^t - k_r \Delta \boldsymbol{\theta}\|} \quad (2.12)$$

$$\text{and } M_{max} = \mu_r R_{ij} \|\mathbf{F}_n\| \quad (2.13)$$

In the above equations, k_r and η_r are the rolling stiffness coefficient and the rolling damping coefficient, respectively, $\Delta \boldsymbol{\theta} = \Delta t d\boldsymbol{\theta}/dt$ is the rotation increment, μ_r is a Coulomb-like static rolling friction coefficient, M_{max} is a saturation torque, $\mathbf{u}_{r,cumul}$ is the cumulative rolling direction unit vector and $R_{ij} = R_i R_j / (R_i + R_j)$ is the reduced radius. Note that in accordance with Ai et al. [4], the saturation condition in Eq. (2.10) and Eq. (2.11) only takes into account the spring-like component \mathbf{M}_k of the total torque.

2.4 Contact model parameters

The above contact model requires to set five parameters: the stiffness and damping coefficients k_i and η_i in both translational directions, as well as the static friction coefficient μ_c . If a rotational friction is considered, three additional coefficients need to be determined, as introduced in Section 2.3.2. Most of these coefficients do not have a clear physical meaning, and assigning meaningful values to them is not a straightforward task. As a result, there is no consensus in the literature on a well defined and universal methodology to determine the magnitude of contact model parameters, and most of these values have generally been established for spherical particles only. We present below our procedure that is designed by carefully reviewing the literature and assembling parts of previously existing procedures which require the least amount of hypothesis or sloppy formulations.

2.4.1 Limitations of the intuitive and physical approach

From Hooke's law and Hertz theory, and following Dziugys and Peters [47] (Eq. (43) in their study), one can relate material properties to some contact model parameters such as the normal stiffness k_n :

$$k_n = \frac{4}{3} \frac{E_i E_j}{E_i(1 - \sigma_j^2) + E_j(1 - \sigma_i^2)} R_{ij} \quad (2.14)$$

or, for two particles of the same material:

$$k_n = \frac{2E_i R_{ij}}{3(1 - \sigma^2)} \quad (2.15)$$

where E_i and σ_i are the Young's modulus and the Poisson ratio, respectively, of material i .

Then, it is not difficult to find the expression of the normal restitution coefficient e_n [202]:

$$e_n = \exp\left(-\gamma_n \frac{\pi}{\sqrt{\omega_0^2 - \gamma_n^2}}\right) \quad (2.16)$$

where $\gamma_n = \eta_n/(2m_{ij})$ and $\omega_0^2 = k_n/m_{ij}$ is the resonance frequency of the system, with $m_{ij} = m_i m_j / (m_i + m_j)$ the reduced mass of the two contacting bodies.

The time step Δt is set according to the contact time T_c . As summarized by Kruggel-Emden [112], the number of recommended time steps per contact varies strongly from author to author and is often set regardless of the integration scheme. From the analysis of the accuracy of dozens of integration schemes, Kruggel-Emden [112] finds that having 20 time steps per contact is typically sufficient to get a resolved enough simulation. In a simple two-sphere contact case, the contact time is expressed as:

$$T_c = \frac{\pi}{\sqrt{\omega_0^2 - \gamma_n^2}} = \sqrt{\frac{m_{ij}(\pi^2 + \ln(e_n)^2)}{k_n}} \quad (2.17)$$

For steel, typically $E = 210$ GPA, $\sigma = 0.3$, $e_n = 0.75$ and the density ρ is 7.85 kg/m^3 . Considering two colliding steel balls of radius 1mm, we find $k_n \approx 7.8 \times 10^7 \text{ N/m}$. This means that the contact time T_c is close to $5 \times 10^{-8} \text{ s}$, imposing $\Delta t \leq 2.5 \times 10^{-9} \text{ s}$. For computational reasons, such a small time step is practically unachievable.

2.4.2 Procedure to find contact parameters in translation

At this stage, performing reliable DEM simulations can seem out of reach. However, it has been shown by Yan et al. [211] and Cleary [33] that for rigid particles a physical value for the normal stiffness k_n does not play a significant role in the overall dynamics of the system, and that decreasing its value by orders of magnitudes – depending on the system – can be safely performed. In fact, it is not the actual stiffness that drives the behavior of the granular medium, and decreasing its value primarily increases the contact time T_c , allowing for larger time steps Δt . While the artificially lengthened contact time T_c is kept much smaller than the time scale of the system, the dynamics of the granular medium is unchanged. For a given force exerted on a particle, the stiffness also controls the extent of overlap between the contacting particles that is of major importance for obvious geometric reasons. On the basis of this approach, contact model parameters are set using the following procedure:

Step 1: Determine values for δ_{max} , e_n , e_t and μ_c .

Based on empirical observations, Yan et al. [211] and Cleary [33] concluded that an appropriate particle overlap should remain below 0.5% of the particle radius. The restitution coefficient e_n lies by definition in the range 0 to 1, and its value is easy to measure experimentally, at least for spheres. The sliding friction coefficient is also straightforward to measure experimentally and typically ranges from 0 to 1.5. The tangential restitution coefficient e_t is a less straightforward parameter to measure, and can be set with a calibration procedure.

Step 2: Compute k_n from an estimated maximum impact velocity v_{max} .

Solving analytically for the overlap δ_n in a simple two-sphere gravity-less contact case and computing its maximum value δ_{max} leads to

$$\delta_{max} = \frac{v_{max}}{\omega} e^{-\gamma_n T_{max}} \sin(\omega T_{max}) \quad (2.18)$$

with

$$\omega = \sqrt{\omega_0^2 - \gamma_n^2} = \sqrt{\frac{k_n}{m_{ij}} - \gamma_n^2} \quad (2.19)$$

2.4. Contact model parameters

and

$$T_{max} = \frac{1}{\omega} \arctan(\zeta), \quad \zeta = \frac{\omega}{\gamma_n} = -\frac{\pi}{\ln(e_n)}. \quad (2.20)$$

From Eq. (2.18), we can solve for k_n :

$$k_n = m_{ij} \left(1 + \frac{1}{\zeta^2}\right) \left(\frac{v_{max}}{\delta_{max}} e^{-\frac{1}{\zeta} \arctan(\zeta)} \sin(\arctan(\zeta))\right)^2 \quad (2.21)$$

Step 3: Compute the contact time T_c and the normal damping coefficient.

T_c is computed from Eq. (2.17). Then, using Eq. (2.16), the normal damping coefficient η_n is given by:

$$\eta_n = -2m_{ij} \frac{\omega_0 \ln(e_n)}{\sqrt{\pi^2 + (\ln(e_n))^2}} \quad (2.22)$$

Step 4: Compute the tangential parameters k_t and η_t .

Following Pournin [155] (Eq. 22 in his study), the tangential model parameters can be linked to the normal parameters in the following way:

$$k_t = \frac{k_n}{1 + m_{ij} \left(\frac{R_i^2}{I_i} + \frac{R_j^2}{I_j}\right)} \frac{\pi^2 + \ln(e_t)^2}{\pi^2 + \ln(e_n)^2} \quad (2.23)$$

$$\eta_t = \frac{\eta_n}{1 + m_{ij} \left(\frac{R_i^2}{I_i} + \frac{R_j^2}{I_j}\right)} \frac{\ln(e_t)}{\ln(e_n)} \quad (2.24)$$

where I_i and I_j are the moments of inertia about the rolling axes of the particles.

At this point we remind the reader that the above procedure to find translational parameters has been derived for spherical particles, and that an extension to non-spherical particles is not straightforward *a priori*. In the case of ellipsoids, Seelen [181] has modified this expression by replacing the radii R_1 and R_2 by the distance between the contact point and the centre of gravity of the two ellipsoids. We discuss this extension to non-spherical bodies in Section 3.4.3.

2.4.3 Contact parameters in rotation

In rotation, we set the contact parameters based on the approach of Jiang et al. [99], as it is developed from a credible micro-mechanical model. Since the contact between two rigid bodies never occurs over a point but rather over a surface, Jiang et al. [99] consider an infinite distribution of spring-dashpots over this surface and derive a relation between the contact parameters in translation and in rotation. The friction torque between two spheres of identical radius R is expressed as follows:

$$\mathbf{M}_r = \frac{(\delta_r R)^2 (k_n \theta + \eta_n \dot{\theta})}{12} \mathbf{u}_{r,\text{cumul}} \quad (2.25)$$

$$\text{and } \|\mathbf{M}_r\| \leq M_{\max} = \frac{1}{6} \delta_r R \|\mathbf{F}_n\| \quad (2.26)$$

with θ the relative rotation between the two rigid bodies and δ_r a quantity related to the surface area of contact, the so-called *shape parameter*, to be calibrated with experiments. Both rolling friction models from Ai et al. [4] and Jiang et al. [99] introduce a saturation torque M_{\max} given by Eq. (2.13) and Eq. (2.26), allowing a relationship between the shape parameter δ_r and the static rolling friction coefficient μ_r :

$$\delta_r = 6\mu_r \quad (2.27)$$

From Eq. (2.25) and Eq. (2.27), the rolling contact parameters can be related to the normal contact parameters through the static rolling friction coefficient:

$$\begin{cases} k_r = 3k_n \mu_r^2 R^2 \\ \eta_r = 3\eta_n \mu_r^2 R^2 \end{cases} \quad (2.28)$$

Note that the parameter μ_r still needs to be calibrated with experiments, but its physical meaning is clearer than the physical meaning of the shape parameter δ_r :

$$\mu_r = \tan \beta, \quad (2.29)$$

where β is the maximum slope angle at which the rolling friction torque balances the torque due to gravity [4].

2.5 Time intergration schemes

2.5.1 Integration of the translational motion

In order to numerically integrate Eq. (2.1) and Eq. (2.2), numerical schemes commonly used in DEM can be categorized in two families: (i) One-step schemes and (ii) Multi-step schemes [112].

One-step schemes includes the simple Forward-Euler scheme, that computes the position $\mathbf{x}(t + \Delta t)$ and velocity $\mathbf{v}(t + \Delta t)$ at the next time step from the position $\mathbf{x}(t)$, velocity $\mathbf{v}(t)$ and acceleration $\mathbf{a}(t)$ at the current time-step:

$$\begin{cases} \mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \Delta t \mathbf{v}(t) \\ \mathbf{v}(t + \Delta t) = \mathbf{v}(t) + \Delta t \mathbf{a}(t) \end{cases} \quad (2.30)$$

The Forward-Euler scheme is simply a Taylor series where the terms of order two and above have been truncated. This scheme is of order one both for the postion $\mathbf{x}(t)$ and the velocity $\mathbf{v}(t)$. To achieve higher orders of accuracy, one could truncate the Taylor series to keep higher order terms, but in this case the derivatives of the acceleration must be approximated carefully to preserve the desired accuracy. The one-step schemes family also includes the classical second and fourth order Runge-Kutta schemes, however they require several force evaluations per time step which prohibits their usage in the smooth DEM. The most commonly used scheme is the second order Leapfrog, which gets its name from the half time step offset of the velocity with respect to the position:

$$\begin{cases} \mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \Delta t \mathbf{v}(t + \frac{\Delta t}{2}) \\ \mathbf{v}(t + \frac{\Delta t}{2}) = \mathbf{v}(t - \frac{\Delta t}{2}) + \Delta t \mathbf{a}(t) \end{cases} \quad (2.31)$$

In addition to second order accuracy, requiring only one force evaluation per time step, being explicit and stable, the Leapfrog scheme is also symplectic. Its minor drawback is to compute the velocity at $t + \Delta t/2$, which necessitates (i) the computation of $\mathbf{v}(\Delta t/2)$ from the initial condition $\mathbf{v}(t = 0)$ with a one-step scheme, e.g., the Forward-Euler method; and (ii) to interpolate the velocity whenever $\mathbf{v}(t)$ is needed: $\mathbf{v}(t) = (\mathbf{v}(t + \Delta t/2) + \mathbf{v}(t - \Delta t/2))/2$.

Members of the second family of multi-step schemes allow to reach higher accuracies with synchronized velocities and positions, while still requiring only one evaluation of the forces per time step. This comes at the expense of storing information from previous time steps. One example is the second

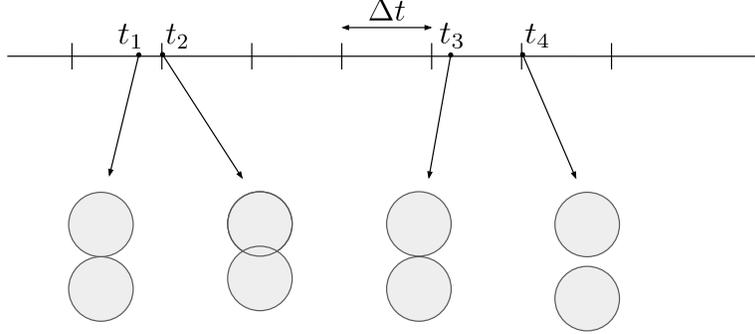


Figure 2.4: Illustration of the error stemming from the latency in the detection of the beginning and the end of the contact. Adapted from Kruggel-Emden et al. [112].

order Adams-Bashforth scheme, expressed as follows:

$$\begin{cases} \mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \frac{\Delta t}{2} (3\mathbf{v}(t) - \mathbf{v}(t - \Delta t)) \\ \mathbf{v}(t + \Delta t) = \mathbf{v}(t) + \frac{\Delta t}{2} (3\mathbf{a}(t) - \mathbf{a}(t - \Delta t)) \end{cases} \quad (2.32)$$

Higher order Adams-Bashforth schemes are straightforward to implement at the cost of accessing information further in the past [112]. More advanced high order schemes can be constructed, such as predictor-correctors, but their usage in DEM remains limited. As such, they will not be discussed here and the interested reader is referred to [47, 112].

In addition to the theoretical accuracy of the integration schemes, extra errors are introduced at the beginning and the end of the contact. As discussed by Kruggel-Emden et al. in [112] and sketched in figure 2.4, the exact contact time t_1 happens in general between two time steps, i.e., the contact is detected at the next time step at time t_2 . During the latency time $t_2 - t_1$ in the detection of the contact, the contacting particles experience an unconstrained motion instead of a motion impeded by repulsive contact forces. Symmetrically, the end of the contact is detected at time t_4 instead of time t_3 , and the contacting particles experience fictitious contact forces during the time interval $t_4 - t_3$. The resulting errors are often overlooked, but they lead to a degenerescence to first order on the accuracy of the positions and the velocities of the contacting particles, as verified in Section 2.6.1. This means that the order of accuracy of the integration scheme is not an important factor in applications involving fast binary contacts. On the other hand, if the contact duration is very long this order degenerescence from the detection of the contact time is less relevant, and the use of higher

order integration schemes is justified.

After reviewing a wide range of integration schemes, Kruggel-Emden et al.[112] conclude that in the smooth DEM, the use of lower order schemes with reasonably small time steps – they advise about 20 time steps per contact – is more efficient than higher order schemes with larger time steps. This conclusion ties in with the popular choice of the Leapfrog scheme, which is as computationally cheap as the Forward-Euler method while being more stable, second order accurate, and symplectic. This Leapfrog scheme is implemented in our solver Grains3D in order to integrate the equations of motion in translation.

2.5.2 Integration of the rotational motion

In the case of spherical particles, the numerical integration of the angular equations can be performed using the same numerical schemes as presented above, as Eq. (2.3) and Eq. (2.4) are similar to Eq. (2.1) and Eq. (2.2). However, when other particle shapes are considered the inertia tensor \mathbf{J}_i becomes time-dependent which complexifies the time integration of Eq. (2.3). Since the inertia tensor is constant in a body-fixed frame of reference, i.e., a frame of reference attached to the considered particle, denoted with the superscript b in the following, Eq. (2.3) can be simplified to isolate the term $d\omega_i^b/dt$:

$$\frac{d\mathbf{J}_i\boldsymbol{\omega}_i}{dt} = \mathbf{T}_i \Leftrightarrow \frac{d\mathbf{J}_i^b\boldsymbol{\omega}_i^b}{dt} + \boldsymbol{\omega}_i^b \times \mathbf{J}_i^b\boldsymbol{\omega}_i^b = \mathbf{J}_i^b\frac{d\boldsymbol{\omega}_i^b}{dt} + \boldsymbol{\omega}_i^b \times \mathbf{J}_i^b\boldsymbol{\omega}_i^b = \mathbf{T}_i^b \quad (2.33)$$

$$\Leftrightarrow \frac{d\boldsymbol{\omega}_i^b}{dt} = \left(\mathbf{J}_i^b\right)^{-1} \left(\mathbf{T}_i^b - \boldsymbol{\omega}_i^b \times \mathbf{J}_i^b\boldsymbol{\omega}_i^b\right) \quad (2.34)$$

While Eq. (2.34) is conceptually simpler to integrate in time than Eq. (2.3), the rotation of $\boldsymbol{\omega}_i$ and \mathbf{T}_i from the space-fixed to the body-fixed frame of reference adds a significant computational overload. Moreover, if a second order scheme is sought, particular attention is needed in the computation of $\boldsymbol{\omega}_i^b$ and \mathbf{T}_i^b , since they may require the angular position that is not yet known at the current time step. In such cases, predictor-corrector schemes such as the PCDM method [180, 218] are good candidates. For the sake of simplicity a first order scheme is implemented in our solver Grains3D to integrate the equations of motion in rotation.

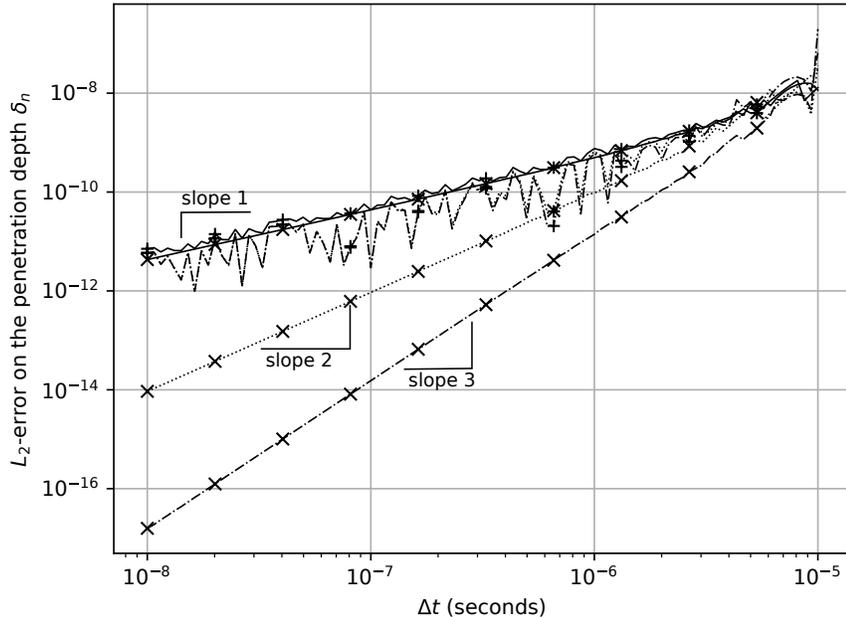


Figure 2.5: Error on the penetration depth δ_n of a sphere bouncing on a horizontal plane for the first order Forward Euler scheme (solid line), the second-order Adams-Bashforth scheme (dotted line) and the third order Adams-Bashforth scheme (dash-dotted line). For each scheme, the bouncing sphere is either (i) already in contact with the horizontal plane at $t = 0$ (“x” markers), or (ii) away from the horizontal plane at $t = 0$ (“+” markers).

2.6 Validation of the numerical method

The code Grains3D used in this study has successfully reproduced numerous experimental results in the past in the case of a contact model without memory [46, 63, 64, 169, 202]. Here, we only seek validation of the memory contact model presented in Section 2.3.1 and Section 2.3.2. We perform this validation for systems of spherical particles that are the most common systems studied in the literature. We also illustrate the order degenerescence due to the contact detection latency discussed in Section 2.5.1.

2.6.1 Single sphere bouncing on a horizontal plane

To illustrate the order degenerescence discussed in Section 2.5.1, we consider a sphere of mass m and initial velocity v_0 bouncing on a horizontal plane. Two

cases are considered: (i) the time $t = 0$ coincides with the beginning of the contact between the sphere and the horizontal plane, or (ii) the sphere is not yet in contact with the horizontal plane at $t = 0$. We show the results of this simple test case using the first order Forward Euler scheme, and the second and third order Adams-Bashforth schemes. The test is performed without gravity and only the normal response of the contact model is important since the tangential velocity is zero at all times. The analytical solution for the penetration depth δ_n is the solution of a simple damped harmonic oscillator equation:

$$\frac{\partial^2 \delta_n}{\partial t^2} + 2\gamma_n \frac{\partial \delta_n}{\partial t} + \omega_0^2 \delta_n = 0 \quad (2.35)$$

$$\text{with } \gamma_n = \frac{\eta_n}{2m} \quad \text{and} \quad \omega_0^2 = \frac{k_n}{m}. \quad (2.36)$$

Since in practice $\gamma_n < \omega_0$ the solution to the above equation is:

$$\delta_n(t) = \frac{v_0}{\omega} e^{-\gamma_n t} \sin \omega t, \quad \text{with } \omega^2 = \omega_0^2 - \gamma_n^2. \quad (2.37)$$

Note that Eq. (2.37) is only valid when the contact exists, i.e., for $t < T_C$ with $T_C = \frac{\pi}{\omega}$ the contact duration.

Figure 2.5 shows the the L_2 -error of the penetration depth δ_n in each case of the aforementioned bouncing sphere configuration. When the contact already exists at $t = 0$, the L_2 -error of each considered integration scheme scales with the expected order of accuracy. However, if the sphere is not yet in contact with the horizontal plane at $t = 0$, every scheme sees its order of accuracy reduced to first order. As discussed in Section 2.5.1, this order degenerescence is a consequence of the time of the beginning of the contact being resolved at first order only.

2.6.2 Quantifying the residual velocity

The main motivation to introduce the memory terms $k_t \delta_t$ and $k_r \Delta \theta$ from Eq. (2.6) and Eq. (2.10) is to allow the system to reach a static state. Without these terms, the translational and rotational velocities never reach values close to the machine epsilon. Figure 2.6 shows how this contact model with history dramatically improves the static state of a system of 750 spheres settling in a box. Each line corresponds to one simulation with random initial positions of the spheres. We can see that for some of the randomly generated initial conditions it may take longer to reach a static state, but the static state is eventually reached. For a simulation time sufficiently large,

2.6. Validation of the numerical method

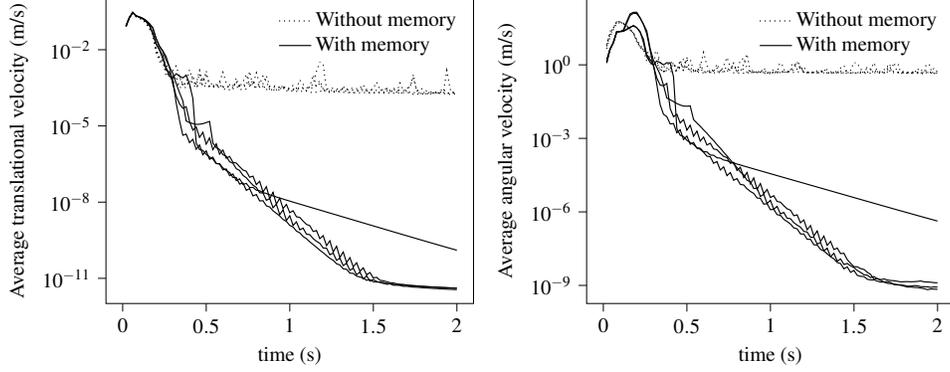


Figure 2.6: Time evolution of the average velocity for a system of 750 spheres settling in a box. Dotted line: no memory term in the contact model. Solid line: memory term in the contact model.

the residual velocity of the particles is orders of magnitude lower when the contact model includes the history terms: lower than 10^{-10} m/s versus larger than 10^{-4} m/s in translation, and lower than 10^{-8} rad/s versus larger than 10^{-1} rad/s in rotation. These results show that the contact model with history is at least able to yield true static states where the residual velocity of the particles is as close as possible to the machine epsilon. In the next validation case, we challenge this model further and show that it reproduces physical results for the classical test case of the hopper discharge.

2.6.3 Hopper discharge

In this test case, we consider a hopper discharge performed several times with various translational static friction coefficients μ_c . As μ_c increases, we measure the repose angle of the pile. The details of the simulation parameters are presented in table 2.1. The particle density ρ , particle radius R and normal restitution coefficient e_n are chosen to match those of Yan et al. [211], while the rest of the coefficients are determined using the procedure described in Section 2.4.2.

Our simulation results are shown in figure 2.7, together with the results from Yan et al. The agreement is satisfactory: both curves follow the same increasing trend and show the same saturation behavior, with a repose angle of about 36° for $\mu_c = 0.5$. One can notice that for μ_c in the range $[0.05, 0.20]$ the slope is steeper in our simulations than in the numerical study of Yan et al. This discrepancy may be explained by the use of a different contact model (Yan et al. use the Hertz-Mindlin model) or by a slightly different

2.6. Validation of the numerical method

Name	Symbol	Value	Unit
Particle density	ρ	2.5×10^3	kg/m ³
Particle radius	R	10^{-3}	m
Normal stiffness (giving $\delta_{\max} = 0.005 \times R$)	k_n	$\approx 7.34 \times 10^5$	N/m
Normal restitution coefficient	e_n	0.45	n/a
Tangential stiffness	k_t	$\approx 2.26 \times 10^5$	N/m
Tangential damping factor	γ_t	$\approx 4.0 \times 10^4$	s ⁻¹
Static friction coefficient particle-particle	$\mu_{c,\text{part-part}}$	0.05 – 0.65	n/a
Static friction coefficient particle-wall	$\mu_{c,\text{part-wall}}$	0.45	n/a
Rolling friction coefficient particle-particle	$\mu_{r,\text{part-part}}$	0.45	n/a
Rolling friction coefficient particle-wall	$\mu_{r,\text{part-wall}}$	0.1	n/a
Time step	Δt	5.0×10^{-7}	s

Table 2.1: Particle properties and contact model parameters of the Hopper discharge simulations.

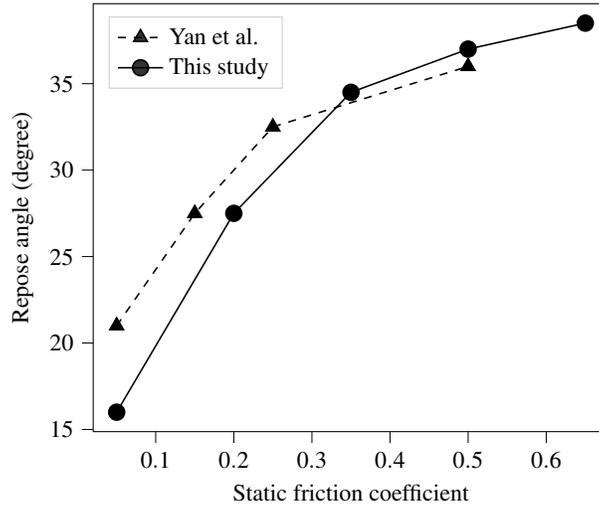


Figure 2.7: Variation of the repose angle θ with the static friction coefficient μ_c . Solid line: this study. Dashed line: adapted from Yan et al. [211]

setup, for instance with regards to the opening of the gate.

2.7 Conclusion

In this chapter, we have described the smooth Discrete Element Method (DEM) with the aim to simulate granular flows composed particles of arbitrary shapes. Our description was focused on the contact model with a so-called “memory” term, as well as on the numerical integration of the equations of motion, both in translation and in rotation. With regards to the contact model, we have implemented a tangential and rotational friction terms which allows the granular assembly to reach a static state where the velocity of each particle approaches the machine epsilon. In hand with the formulation of the memory contact model, we reviewed the procedure to give relevant values to the numerous model parameters, some of which have no clear physical meaning. Our resulting procedure is derived for spherical particles: its validity for non-spherical shapes is therefore not guaranteed, although previous studies using a similar procedure obtained satisfactory results with ellipsoidal particles [181]. Two cases were considered to validate the implementation of our memory contact model: (i) the average translational and rotational velocities of 750 settling spheres were analyzed and shown to approach the machine epsilon, performing over eight orders of magnitude better than the contact model without the memory term; and (ii) the repose angle of a granular assembly composed of spheres discharged from a hopper was plotted for a range of Coulomb friction coefficients against a reference value from the literature, and satisfactory agreement was found. Then, with regards to the numerical integration of the equations of motion, we have demonstrated the order degenerescence to first order of the smooth DEM, as was first mentionned by Kruggel-Emden et al. [112]. This degenerescence of the order of accuracy of the computed dynamics is due to a latency in the detection of the contacts, which is inherent to the design of the smooth DEM. In our minimal example of a sphere impacting a flat surface, we show that despite numerical integrators of second or even third order of accuracy, the error on the position of the centroid of the particle with respect to its analytical solution is only first order.

Implementing the aforementioned memory contact model in the granular solver Grains3D makes it possible to accurately simulate granular systems of complex shaped particles in a wide range of static and dynamic regimes. In the next chapter, we propose to study numerically and experimentally dam break collapses of cross-shaped particles: as will be discussed in Sec-

2.7. Conclusion

tion 3.4.2, accurately modelling static contacts in such systems is critical to properly simulate the base of the granular column and in turn, the whole granular assembly.

Chapter 3

Granular avalanches of entangled rigid particles

3.1 Introduction

In the past two decades there has been a growing interest in the study of granular systems, in most cases composed of spherical particles. In the case of rectangular and axisymmetric dam break configurations, Lajeunesse et al. [118] have provided experimental results and a force-balance model that correctly reproduces the scaling of the run-out distance as a function of the initial aspect ratio. Balmforth et al. [14] also conducted dam break experiments and Kerswell [106] developed a depth-average model to explain their observations. In the meantime, the smooth DEM gained popularity with the increase of computing power. Most often, the particle shape is simplified to spheres, ellipsoids or even composite particles made of overlapping spheres. However, numerous studies [33, 34, 183] show that the particle blockiness, particle surface roughness and particle aspect ratio have a major impact on the dynamics and the packing properties of the considered systems. Non-convex particles are even less represented in experimental and numerical studies, even though most natural processes involve arbitrary shapes, including non-convex ones. It is reasonable to expect that the flow properties and steady-state quantities are significantly different in the case of convex and non-convex particle shapes, as Rémond et al. [176], Gravish et al. [66] and more recently Landauer et al. [119] and Wang et al. [205] have observed with packings of non-convex particles.

When studying granular materials, dam break collapses on a horizontal flat surface have been popular due to the relative simplicity of this setup and the rich physics that granular media exhibit in this configuration. Already in 2005, Staron and Hinch [185] use a two-dimensional discrete element method to study the collapse and spreading of disk-shaped particles onto a horizontal plane. They show that for initial aspect ratios $a \gtrsim 2.5$, the column has a period of free-fall, whereas columns with $a \lesssim 2.5$ have not. This observation

explains the existence of two distinct scaling laws for the run-out distance versus the initial aspect ratio, as previously reported by Lajeunesse et al. [118] and Lube et al. [130, 131]. It also highlights that the run-out distance is not only dependent on effective flow properties such as basal friction, but is also strongly affected by the fall dynamics. Moving to three-dimensional simulations, Lacaze et al. [116] propose a comparison between DEM simulations and experimental results in the case of a planar channel – of width less than two particle diameters. The response they obtain exhibits a flowing top layer and their DEM simulations are in good qualitative and quantitative agreement with their experiments. Furthermore, they already highlight the difficulty of assigning values to friction parameters in their DEM simulations and the need to rely on calibration experiments. Girolami et al. [63] later performed fully three-dimensional avalanches of spherical particles with DEM simulations and compare their numerical results to the experiments of Lajeunesse et al. [118]. The quantitative scaling with the initial height-to-width aspect ratio and the qualitative flow structure and scaled free surface show very good agreement with Lajeunesse’s experimental data. Girolami et al. [64] later use DEM simulations to study unchanneled dam break flows, and show that the run-out distance versus initial aspect ratio scaling power lies in between that of previously reported channelized and axisymmetric flows. With regards to numerical simulations of avalanches of non-spherical particles, Lim et al. [122] study two-dimensional collapses of spheres, convex particles and non-convex grit-like particles. Although not compared to experimental data, their results reveal that the run-out distance and angle of repose are greater for non-convex particles than for convex particles and spheres. Finally, Rakotonirina et al. [168] consider granular slumpings of a realistic number of particles – over two millions – in channels of periodic and finite width. Claiming their icosahedral particles have similar shapes to those used in the experiments of Lube et al. [130] and Balmforth et al. [14], they quantitatively compare the run-out distance and its scaling law versus the initial aspect ratio to those experiments. They obtain a very good agreement with the data reported by Balmforth et al., and an overestimated run-out distance when compared to the experimental results of Lube et al., highlighting the relatively wide range of reported data.

The past decade has seen rapid developments in the field of DEM of non-spherical particles, and we shall recall briefly here the main methods and recent contributions relevant to the simulations of non-spherical particles. When the shape of grains deviates from discs or spheres, the first challenge is to describe the shape. While it is possible to mesh the surface of each particle, the computational cost of the contact detection algorithm

limits this approach to a small number of particles [73, 215], and other approaches are preferred for large granular systems. Ellipses and super-quadric shapes were among the first non-spherical shapes to be considered in DEM simulations [124, 175, 209]. Super-quadric shapes still receive considerable interest as they offer a unified framework to represent a wide range of shapes [126, 127, 149]. However, with greater blockiness the contact detection algorithm of super-quadric shapes become more and more costly. As a result, in the case of particles that have faces and edges, a polyhedral representation as in [168, 169, 181, 202] is better suited than high order, i.e., “sharp”, super-quadrics. In terms of physics, we note that most of the studies regarding non-spherical particles, and especially regarding non-convex particles, focus on random packings – see, e.g., [66, 119, 132, 176, 181, 205] – resulting in a gap between our understanding of static and dynamic assemblies of non-spherical particles. With regards to static assemblies of non-spherical particles, Seelen et al. [181] present a unified framework to treat any convex shape including cuboids, ellipsoids and more complex polyhedra that are close to sand grains and woodchips. They investigate the packing fraction as a function of the distance to the wall, and report a strong influence of the aspect ratio of ellipsoids and cylinders. Packings of non-convex particles have been studied by Ludewig et al. [132], Gravish et al. [66], Rakotonirina et al. [169] and Landauer et al. [119], among others. Ludewig et al. consider the compaction of non-convex particles made of composite overlapping spheres with varying sphericity. They show that decreasing the sphericity of the shapes – which in the case of their study results in increasing their concavity – leads to more interlocking of the granular assembly, measured in terms of the mobility of the particles. Furthermore, they show that the greater the concavity of the grains is, the greater the stability of the whole assembly is despite a lower packing fraction. More recently, Landauer et al. [119] analyze the bulk and tapping densities as well as the angle of repose of granular assemblies of various shapes, including the non-convex stellated octahedron, tetrapod and quadropod shapes. Their work belongs to the few DEM studies [65, 66, 169, 201] that validate their simulations of non-convex particles with experiments, and as such they highlight the difficulty to find contact parameters and conclude that calibration is necessary for every shape. We face here the same difficulty with assigning values to our friction model parameters. We discuss this issue in Section 3.4.3. On the dynamic behavior of non-spherical particles, Rakotonirina et al. [169] consider rotating drums filled with particles of spherical, convex and non-convex shapes and report a wide range of flow regimes depending on the angular velocity of the rotating drum – the non-convex shape exhibiting

3.1. Introduction

the greatest complexity and variety of regimes. Govender et al. [65] show with experiments of 3D printed non-convex particles and GPU-based DEM simulations that in the case of a hopper discharge the dynamic behavior of the assembly is strongly impacted by the concavity of the grains, since the non-convexity gives rise to jamming effects. Interestingly, the authors of [65] are able to observe significant differences in the flow behavior with Schönhardt polyhedra compared to triangular prisms while a Schönhardt polyhedron only slightly differs from its convex counterpart, the triangular prism. However, due to hardware limitations, the force model used in their studies does not include a tangential memory friction term which we shall see in Section 3.4.2 is crucial to reproduce static states.

In this chapter, we investigate experimentally and numerically the behavior of granular materials beyond the well-known spherical or convex shapes, as a step forward in the pursuit of modeling realistic granular media. We choose non-convex particles with a large aspect ratio that are very prone to entanglement and we perform dam break experiments and simulations. We report novel flow behaviors typical of particle entanglement. Our traditional CPU-based architecture allows us to include a memory friction term in order to accurately reproduce stable configurations of particles of any shape, including of non-convex shape. We show that for the sole dam break setup, considering non-convex particles opens the Pandora box to a new range of granular behaviors that were not reported yet in the literature, such as the top-driven collapse or the intermittent regime – a regime where reproducibility is lost and where the flow outcome is determined by the random initial particle configuration. We perform DEM simulations of our experimental setup using the code Grains3D introduced in Chapter 2, and we show that our numerical simulations accurately reproduce both qualitative and quantitative results. Then, we explore the information provided by the microstructure and the force network, we highlight the importance of the memory term in the tangential friction model, and we discuss the choice of contact parameters.

The rest of this chapter is organized as follows. In Section 3.2 we present the flow configuration and describe the experimental equipment used to record the granular collapses. We analyze experimental and numerical results in Section 3.3 in terms of dynamics of the collapse, flow map, final height, run-out distance, time evolution, force network and fabric anisotropy. We then discuss in Section 3.4 the key elements of the DEM that allow for a quantitative match of the numerical results with the experimental data. Finally, we give concluding remarks in Section 3.5.

3.2. Experimental and numerical setup

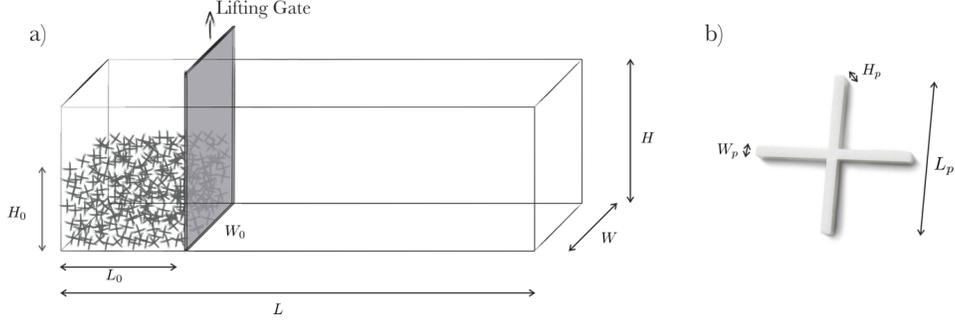


Figure 3.1: Schematics of the experimental setup: (a) the dam break apparatus and (b) the shape of the particles.

3.2 Experimental and numerical setup

3.2.1 Experimental setup

The tests are performed in a dam break setup sketched in figure 3.1a. The box is made of acrylic and its dimension is $L \times H \times W = 64 \times 30 \times 20$ cm. The setup features a sliding gate (made of PVC). Slots are milled on the side walls to guide the vertical motion of the gate, and therefore, its location can be easily adjusted to $L = 4, 8, 12, 20$ and 30 cm – or $L \approx 1.5L_p, 3.0L_p, 4.4L_p, 7.4L_p$ and $11.1L_p$ respectively, where $L_p = 27$ mm is the characteristic length of the particles shown in figure 3.1b. A pneumatic cylinder was used to control the release of the gate. The gate releases with an (almost constant) acceleration of $(43 \pm 3 m/s^2)$, reaching the speed of $\sim 3 \pm 0.1 m/s$ in 0.07 s (calculated by image processing). The frames holding the pneumatic cylinder were placed on a separate optical table. Hence, the vibration of the gate coming to a halt does not affect the experiments.

Four light sources are placed at different angles to illuminate the particles. A high-speed camera (FASTCAM Mini UX100) is used in combination with a Carl-Zeiss Makro-Planar T* 50 mm f/2 ZF.2 lens. Videos are captured at 250 frames per second, and the imaging setup resulted in a pixel size of ~ 0.43 mm.

In each experiment, the initial granular pile has dimensions of $L_0 \times H_0 \times W_0$, where L_0 is the length, H_0 is the height, and W_0 is the width. All these parameters are changed in the experiments resulting in 46 different experimental conditions, each repeated up to 8 times.

3.2.2 Particles

Cross-shaped plastic tile spacers (from Faithfull) are used as particles (see figure 3.1b). The particles have a length of $L_p = 27\text{mm}$, with arm width of $W_p = 2\text{mm}$, and thickness of $H_p = 3\text{mm}$. Hence, the volume of each particle is 0.312 cm^3 . The mass density of the particles is measured to be 770 kg/m^3 .

We followed Balmforth and Kreswell [14] to characterize the friction properties of the particles. We found the “bed” friction angle (between acrylic and particle) to be $\delta = 32^\circ \pm 5^\circ$ and the “internal” friction angle (between particles) to be $\theta = 24^\circ \pm 2^\circ$. Compared to glass particles (*c.f.* Balmforth and Kreswell), the particles used in the present study have a larger bed angle of friction but a similar internal friction angle.

3.2.3 Numerical setup

To numerically reproduce the experiments, we use the contact model with history described in Section 2.3 – with no rolling friction as our particles are neither spheres, cylinders nor ellipsoids. Our contact model is validated for systems of spherical particles, as this model has not yet been applied to non ellipsoidal particles in the literature. For each simulation, we proceed as follows: first, we generate the same geometry as in the experimental apparatus. Then, the particles are poured in the container with random initial positions until the system reaches a static state. At $t = 0$, the gate is removed. The number of particles considered ranges from 1300 to 7000, and the values assigned to the contact model parameters are listed in table 3.1. The normal stiffness k_n was computed to yield a maximum overlap distance δ_{\max} of about 1% of the radius of the sphere of equal volume – using an impact velocity corresponding to a free fall from a 30cm height, i.e., $\sqrt{2 \cdot 9.81 \cdot 0.3} \approx 2.4\text{m/s}$. This leads to $\delta_{\max} \approx 5 \times 10^{-5}\text{ m}$. In practice, this maximum overlap is very rarely reached, and the average overlap is orders of magnitude lower – typically between 0.1% and 0.001% of the radius of the sphere of equal volume. Please note that the contact parameters in the tangential direction are not set following the procedure described in Section 2.4.2. We discuss the reasons why this procedure is not followed in Section 3.4.3.

To ensure each case is treated in a systematic way during the post-processing step, we determine quantities such as the final height and the final length using image-processing routines. In particular, a quantity that is not obvious to define is the final length of the granular assembly, also

3.2. Experimental and numerical setup

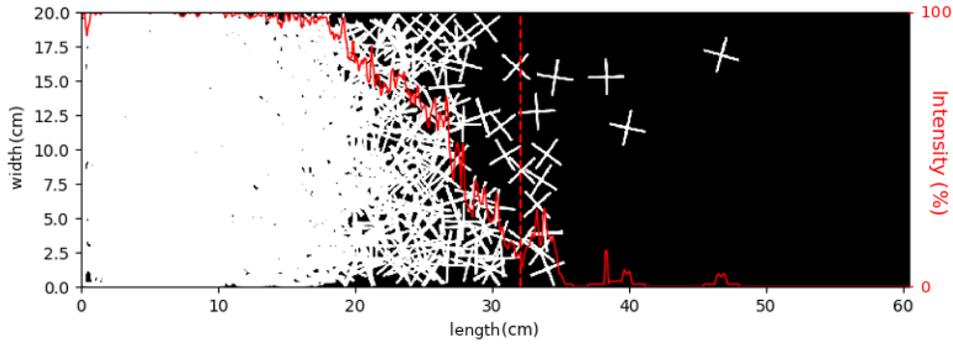
Name	Symbol	Value	Unit
Particle density	ρ	770	kg/m ³
Normal stiffness	k_n	4.5×10^5	N/m
Normal restitution coefficient	e_n	0.85	n/a
Tangential stiffness	k_t	3×10^2	N/m
Tangential damping factor	γ_t	10^3	s ⁻¹
Static friction coefficient particle-particle	$\mu_{c,\text{part-part}}$	0.46	n/a
Static friction coefficient particle-wall	$\mu_{c,\text{part-wall}}$	0.62	n/a
Time step	Δt	2.5×10^{-6}	s

Table 3.1: Particle properties and contact model parameters in the numerical simulations of non-convex particles.

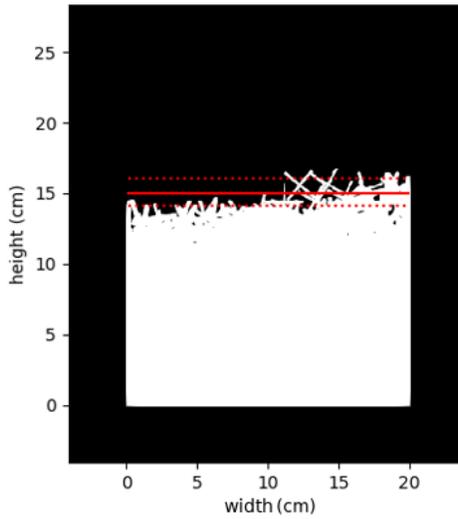
called run-out distance, L_∞ . We follow the same approach as Rakotonirina et al. [169] and define the run-out distance as follows: using a top view of the final configuration, the run-out distance L_∞ is the *leftmost* column of pixels with a fraction of white pixels – referred to as intensity of the pixel column – that is less than 10%. Figure 3.2a shows a typical output of this post-processing routine. The solid line corresponds to the pixel column intensity and the dashed line corresponds to the run-out distance L_∞ . Given the uncertainty of this measurement, we provide error bars in figure 3.7 with a minimum value corresponding to the run-out distance based on a 20% intensity threshold and a maximum value corresponding to the run-out distance based on a 1% intensity threshold. Similarly, the initial and final heights of the granular assembly are measured using an image processing routine that computes the average height of the pile along the width of the channel, shown in figure 3.2b. The solid line corresponds to the average height and the dotted lines indicate the first and third quartiles, used to compute the error bars in figure 3.8.

Finally, in order to quantitatively compare flow properties of non-convex particles and spheres, we carry out numerical simulations of dam break avalanches of spheres using the same numerical setup and contact parameters as Girolami et al. [63]. In particular we compute cases 1, 2 and 3 – each case is computed ten times corresponding to ten different initial microstructures – from Table 1 of [63] and we successfully reproduce results from figure 4 in [63] which shows a very satisfactory agreement between numerical simulations of spherical particles and the experiments of Lajeunesse

3.2. Experimental and numerical setup



(a)



(b)

Figure 3.2: (a) Example of a post-processing output for the run-out distance L_∞ of a simulation, from a top view of the avalanche. The solid line corresponds to the intensity of a given pixel column (the maximum intensity corresponds to the top of the image, the minimum intensity to the bottom). The dashed line indicates the run-out distance: in this case, $L_\infty = 32$ cm. (b) Example of a post-processing output for the final height H_∞ of a simulation. The solid line indicates the average final height and the dotted lines show the first and third quartiles. For visual purposes, an example with a particularly wide interquartile range (IQR) is shown (a typical IQR is half the size of that shown on this figure).

et al. [118]. In the following, we use these simulations to access data that is not reported in [63], such as the average velocity versus time.

3.3 Experimental and numerical results

3.3.1 Qualitative Analysis

Figures 3.3 and 3.4 present the experimental and numerical results of two novel avalanche behaviors. For those two figures only, in order to visually compare the simulations with the experiments, a gate effect is included in the simulations, i.e., the particles in contact with the gate experience friction when the gate slides upwards at a finite speed. The top-driven collapse, in figure 3.3, is by far the most common avalanching mechanism of our cross-shaped particles and is observed for all aspect ratios. During this collapse, the top part of the column falls first in a wave-like fashion. As particles are entangled, their fall is impeded by their neighbours, except for the top layer of particles which experiences less entanglement. As this top layer can fall freely off the granular column, a new layer of particles can repeat this process. This top-driven collapse is not observed for spherical or even convex-shaped particles [14, 63, 118]: it is intrinsically linked to the entanglement of the cross-shaped particles. Sometimes, an initial microstructure leads to another novel avalanche behavior presented in figure 3.4. Unlike the top-driven collapse, the avalanche is triggered below the top layer of particles, around the mid height of the column. It results in the column bending forward during the first stage of the avalanche, before fully collapsing. This buckling behavior is found to be far less frequent than the top-driven collapse, and is only observed for tall narrow columns as shown in figure 3.6.

It is observed experimentally and numerically that sometimes even the top-layer of particles is too entangled to avalanche, and the granular column remains stable after the opening of the gate. These entangled configurations are typical of non-convex particles and pseudo-stable arches have been reported in the case of a hopper discharge of Schönhardt polyhedra by Govender et al. [65]. We find experimentally and numerically that for particular initial dimensions of the column, whether an avalanche occurs or the system remains stable is determined by the random initial configuration of the particles. We denote by *intermittent* this behavior where reproducibility is lost, and show an example in figure 3.5. The intermittence is another novelty of these entangled flows, as a granular dam break performed under the same conditions produces the same macroscopic outcome when convex particles

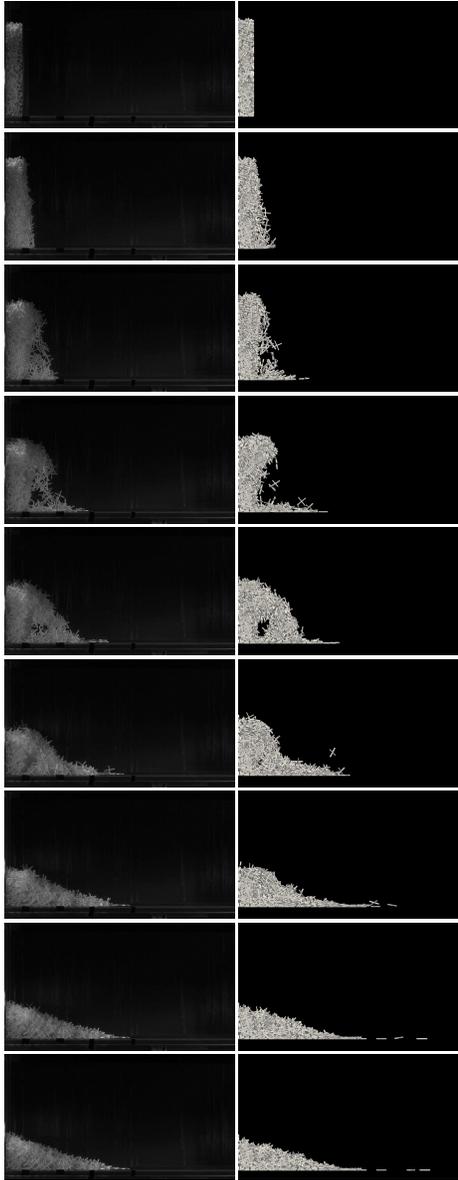


Figure 3.3: Snapshots of a top-driven collapse with an initial aspect ratio $H_0/L_0 = 6$. Left: experiments. Right: simulation. Time increment between each frame: 0.145s. Videos of this collapse are available online at https://youtu.be/yzlzn_XrkJA.

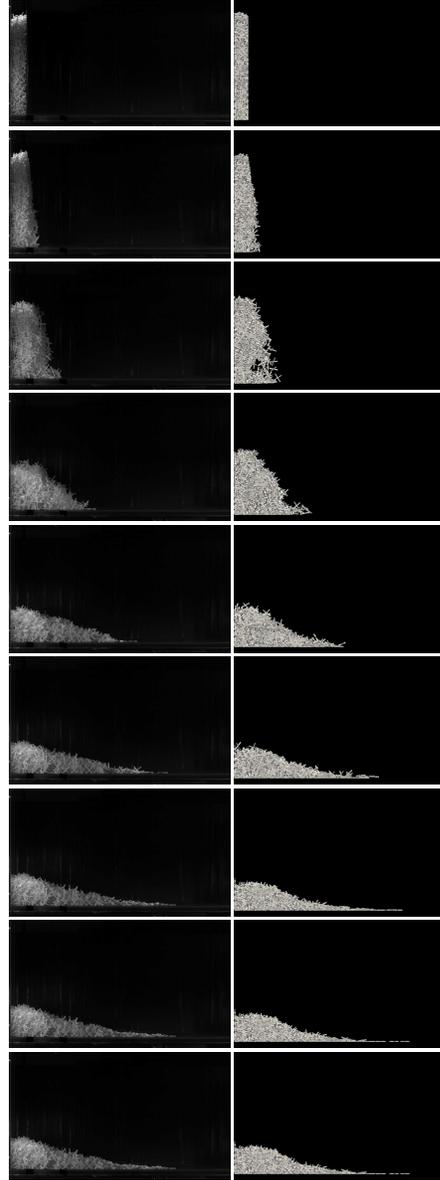


Figure 3.4: Snapshots of a buckling collapse with an initial aspect ratio of $H_0/L_0 \approx 7$. Left: experiment. Right: simulation. Time increment between each frame: 0.12s. Videos of this collapse are available online at <https://youtu.be/DBmOVcR9PrU>.

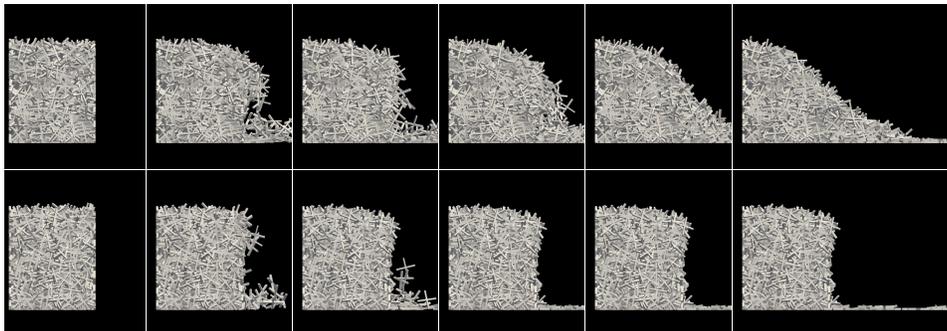


Figure 3.5: Snapshots of a simulated collapsing assembly (top) and of a simulated stable assembly (bottom). The initial dimensions of both assemblies are identical: only the initial microstructure differs between these two intermittent cases. The frames are taken at the following times: $t = 0\text{s}$, $t = 0.30\text{s}$, $t = 0.60\text{s}$, $t = 0.90\text{s}$, $t = 1.20\text{s}$ and $t = 2.00\text{s}$.

are considered.

Figure 3.6 shows the experimental and numerical flow regime maps with respect to the initial length L_0 and initial height H_0 . Each case was reproduced three to five times in order to assess its intermittence. In this figure, the transitions between flow regimes are not sharp and the purpose of these flow regime maps is to report qualitative trends rather than to provide accurate flow transitions. In our numerical simulations, the transition from the repeatable regimes to the intermittent regimes differs from the experiments at high aspect ratios. Indeed, it is found numerically that tall columns of initial height H_0 greater than 20 cm always result in an avalanche, while experimentally some cases were observed to be intermittent – for instance $(L_0, H_0) = (12, 25)$. This can be explained by a slightly inaccurate choice of parameters in table 3.1, and a discussion on the choice of tangential friction parameters is provided in Section 3.4.3. Nevertheless, we note that the numerical flow regime map agrees reasonably well with the experimental map, in particular in the region where buckling collapses occur as well as on the transition between the repeatable regimes and the intermittent regimes at low aspect ratios.

Aside from the intermittent regimes, we also recover the collapsed regime and the fractured regime that have already been reported for systems of spherical particles [14, 118]. In the collapsed case, an avalanche occurs over the whole length of the granular assembly, resulting in a final height H_∞ lower than the initial height H_0 . In the fractured case, the initial length L_0 of the granular assembly is large enough such that some particles located to the left of a fractured line are not disturbed by the avalanche. In a fractured

3.3. Experimental and numerical results

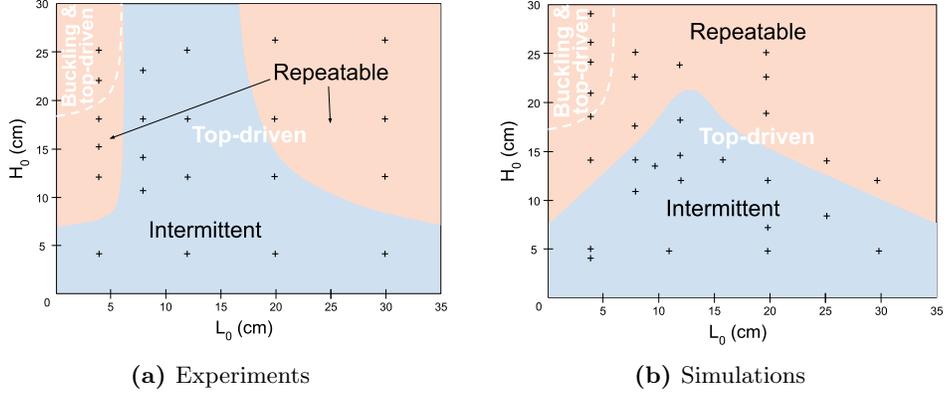


Figure 3.6: Flow maps from (a) experiments and (b) simulations. Each cross represents an experiment or a simulation. Two flow regimes are identified: repeatable and intermittent; and two collapse behaviors are reported: top-driven (observed in all configurations) and buckling (observed only for narrow columns).

case, the final height H_∞ of the granular assembly equals its initial height H_0 .

3.3.2 Macroscopic measurements

Run-out distance and height ratios

The run-out distance L_∞ and the final height H_∞ of the avalanches are two easily accessed quantities that can be used to quantify to what extent avalanches of non-convex particles differ from avalanches of convex particles, as well as to what extent our simulations agree with our experiments. Figure 3.7 shows that the non-dimensional run-out distance $(L_\infty - L_0)/L_0$ as a function of the initial aspect ratio H_0/L_0 follows the same linear trend in the experimental and numerical cases. The error bars of the experimental and numerical data points largely overlap, and the overall agreement between our simulations and our experiments is very satisfactory. This figure also includes experimental data from Lajeunesse et al. [118] and Balmforth et al. [14], who carried out similar experiments with spherical and grit particles, respectively. Figure 3.7 shows that, surprisingly, a very similar power law of the form $K(H_0/L_0)^n$ is followed by spherical and cross-shaped particles, even though the avalanche dynamics is very different. The power coefficient n seems to be approximately equal to 1 for our cross-shaped particles, as found by Lajeunesse et al. [118] and Balmforth et al. [14] for spheres or grit particles. The transition from $n \approx 1$ to $n \approx 2/3$ around $H_0/L_0 = 3$ reported

3.3. Experimental and numerical results

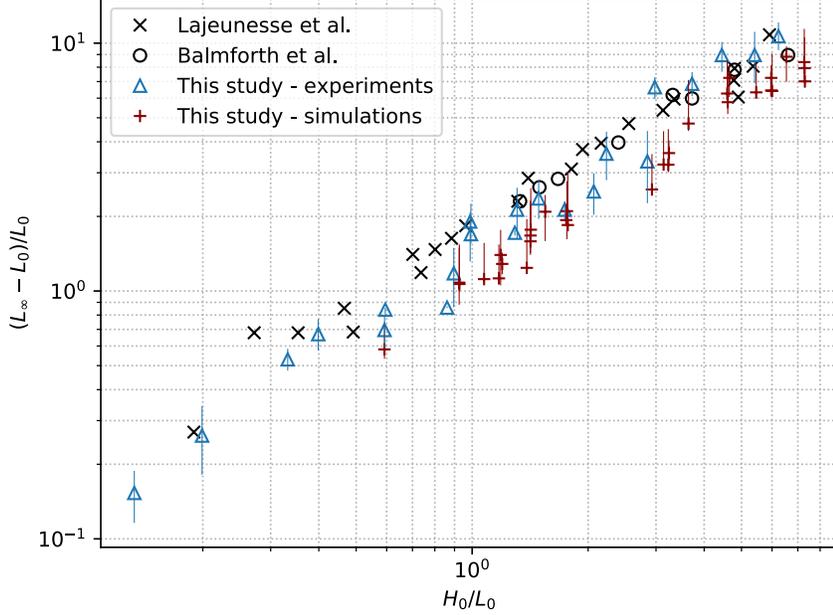


Figure 3.7: Non-dimensional runout distance $(L_\infty - L_0)/L_0$ as a function of the initial aspect ratio H_0/L_0 . Error bars are depicted with vertical segments.

by Lajeunesse et al. [118] for spheres is not very visible for our cross-shaped particles in Figure 3.7. However, the analysis of the center of mass of the assembly reveals very clearly this slope transition around $H_0/L_0 = 3$ and indeed agrees with the measurements of [118] who considered initial aspect ratios of up to 20. The interested reader is referred to Appendix B.1 for more details on this analysis. We also note that the prefactor K is slightly lower for our cross-shaped particles as a signature of the shape and of the corresponding avalanche dynamics. Moreover, for $H_0/L_0 \gtrsim 4$, the run-out distance of all particle shapes, i.e., the spheres of Lajeunesse et al. [118], the grit particles of Balmforth et al. [14] and our cross-shaped particles, almost matches, indicating that in more inertial regimes, the shape matters less.

Likewise, figure 3.8 shows the non-dimensional height ratio H_0/H_∞ as a function of the initial aspect ratio H_0/L_0 for our simulations and our experiments, and also includes data obtained with grit particles by Balmforth et al. [14]. Again, our simulations agree very well with our experiments, and surprisingly again the cross-shaped particles follow the same trend as the

3.3. Experimental and numerical results

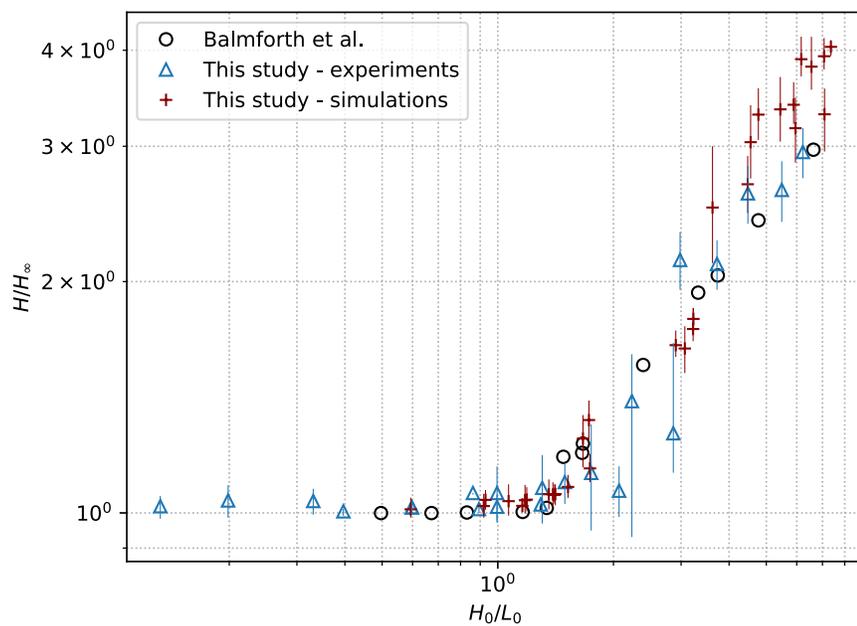


Figure 3.8: Non-dimensional height H_0/H_∞ as a function of the initial aspect ratio H_0/L_0 . Error bars are depicted with vertical segments.

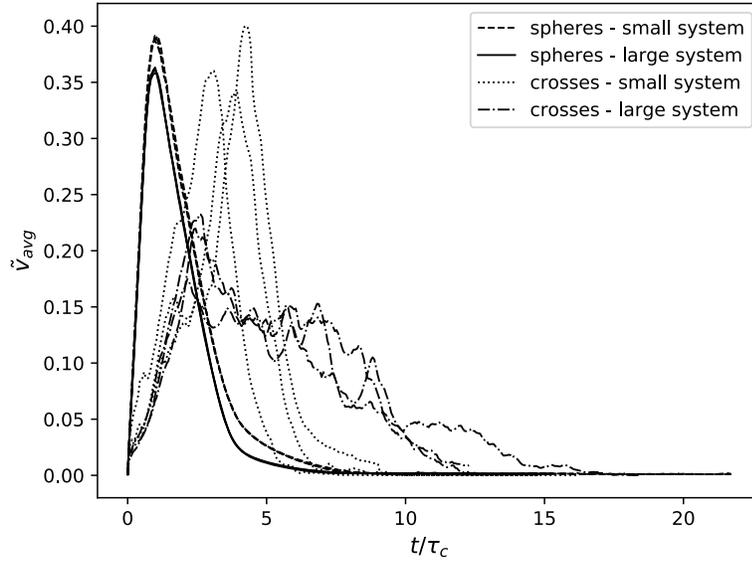
grit particles, despite fundamentally different dynamics.

Characteristic time of collapse

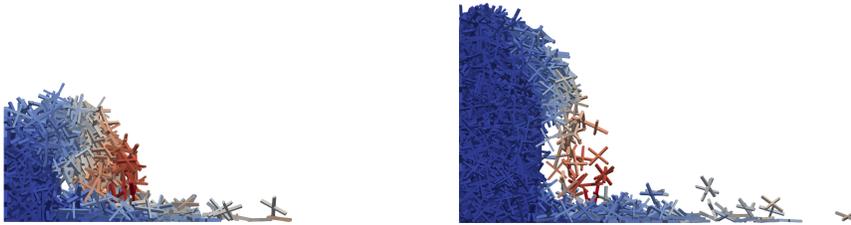
We quantitatively compare the time evolution of avalanches of crosses and spheres. Figure 3.9a shows the average non-dimensional translational particle velocity $\tilde{v}_{\text{avg}} = v_{\text{avg}}/\sqrt{gH_i}$ versus the non-dimensional time $\tau_c = \sqrt{H_i/g}$ for avalanches of spherical particles and cross-shaped particles with an initial aspect ratio of 3.2. Small systems – 3200 spheres or 1000 crosses – and large systems – 15000 spheres or 3000 crosses – are considered, and for each case three simulations with different initial microstructure are plotted in order to show reproducibility. On the one hand, the time evolution of the velocity of small and large systems of spherical particles appear to be very reproducible, as the curves for each case superimpose very well. Moreover, the small and large systems behave similarly, with the average velocity reaching its maximum of 35% to 39% of $v_0 = \sqrt{gH_i}$ at $t = \tau_c$ and dropping below 5% of v_0 at $t \approx 4\tau_c$. On the other hand, the velocity of cross-shaped particles is far less reproducible from one run to another. The small systems of crosses seem to follow a trajectory close to that observed with spheres, but delayed by two to four time units τ_c , with a maximum velocity of 34% to 39% of v_0 . However, the large systems of crosses exhibit a very different trend: for each run the curve is flatter, its maximum is comprised between 18% and 22% of v_0 and occurs at $t \approx 2.5\tau_c$ and the average velocity drops below 5% of v_0 at $t \approx 10\tau_c$. Figure 3.9b and figure 3.9c show snapshots of small and large systems of crosses at $t = 3\tau_c$. Particles are coloured by the norm of their translational velocity. The main difference in the dynamics of the collapse of small and large systems of crosses is that small systems are narrow enough – less than the size of two particles – for the whole column to bend forward during the top driven collapse. For instance, we see in figure 3.9b that the highest point of the column is located far away from the left wall, close to the previous location of the gate. In contrast, large systems – of width greater than the size of three particles – do not bend forward during the top-driven collapse: figure 3.9c shows how the highest point of the granular assembly is close to the left wall. Thus, the rate of particles flowing down the granular column is lower than that of smaller systems where the whole column bends forward. As a result, the maximum average velocity is lower and the collapse time is greater for large systems of crosses than for small systems of crosses.

A more striking time-dependent behavior inherent to entangled particles is the case of staggered collapses, which have been observed experimentally

3.3. Experimental and numerical results



(a)



(b)

(c)

Figure 3.9: (a) Comparison of the average non-dimensional translational particle velocity versus non-dimensional time for systems of spherical and cross-shaped particles in systems of initial aspect ratio $a = H_0/L_0 = 3.2$. (b-c) Snapshots of the collapse of a small (b) and large (c) system of crosses at $t = 3\tau_c$, coloured by the norm of their translational velocity from dark blue (smallest) to red (largest). Small system: 3200 spheres or 1000 crosses; Large system: 15000 spheres or 3000 crosses.

3.3. Experimental and numerical results

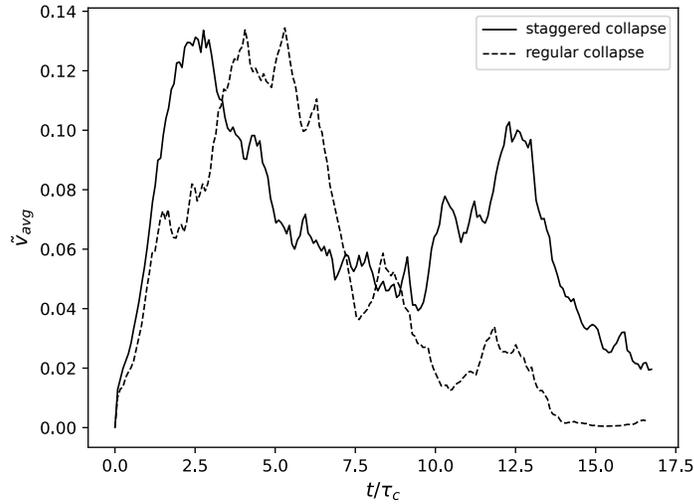


Figure 3.10: Comparison of the average non-dimensional translational particle velocity versus non-dimensional time for a staggered collapse (solid line) and a regular collapse (dashed line) in the case of an initial aspect ratio $a = H_0/L_0 = 1.75$. The only difference between the two collapses is the random initial packing.

and numerically to appear in the case of initial aspect ratios corresponding to the intermittent and fractured regimes. In an unrepeatable way, some initial microstructures lead to collapses that come to nearly a halt for a few τ_c before continuing until fully avalanched, as shown in figure 3.10. It is important to note that the two curves in figure 3.10 correspond to two identical simulations with the exception of different initial microstructures. A partial staggered collapse is often observed for those regimes, i.e., the collapse is three dimensional and one region in the width direction halts its avalanche for up to a few seconds – a very long time for these kind of granular systems – before fully collapsing.

This quantitative analysis in time suggests that the free fall characteristic time τ_c , although relevant in the case of avalanches of spherical particles [14, 118], does not describe collapses of entangled cross-shaped particles well. As such, defining a relevant time scale τ_c for granular collapses of entangled cross-shaped particles appears to be a complex matter and is left for future studies.

3.3.3 Microstructure analysis

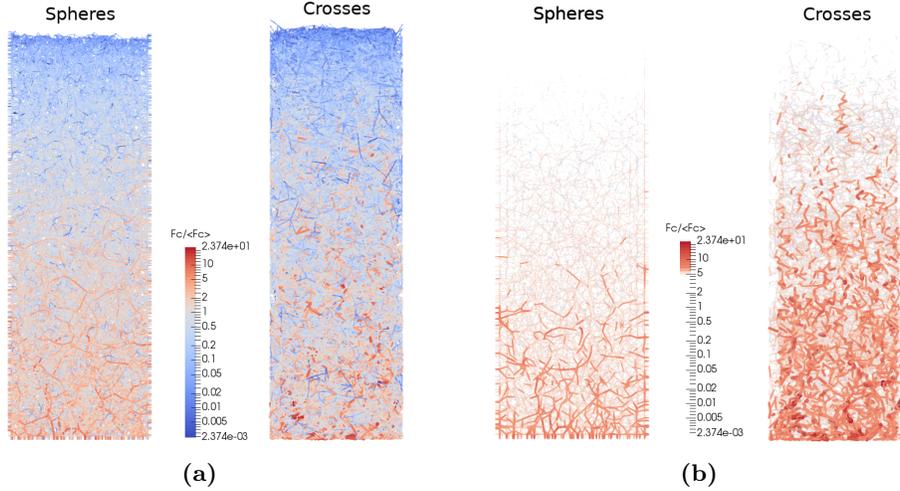


Figure 3.11: Visualization of the contact network in a granular column composed of spheres and crosses. Each segment links the centers of mass of two contacting particles, and its color and width vary according to the magnitude of the total contact force magnitude F_c divided by the average contact force magnitude $\langle F_c \rangle$. In (a) all contacts are shown, while in (b) only the contacts associated with a force magnitude greater than five times the average force magnitude are plotted.

We now extract information about the contact force network from our DEM simulations. This type of information cannot be measured experimentally. Figure 3.11 shows the contact network for a column of 15,000 spheres (41,971 contacts in total) and a column of 12,000 crosses (98,939 contacts in total). The corresponding average coordination number is therefore $2 \cdot 41,971/15,000 \approx 5.6$ for spheres and $2 \cdot 98,939/12,000 \approx 16.5$ for crosses. The much larger average coordination number of crosses compared to that of spheres is expected as the arms of the crosses allow to reach more neighbouring particles located farther away. In figure 3.11, each segment represents a branch vector, i.e., each segment links the centers of mass of two contacting particles, and is colored according to $F_c / \langle F_c \rangle$ where F_c is the contact force magnitude and $\langle F_c \rangle$ is the average contact force magnitude in the whole granular assembly. While it is not convenient to interpret figure 3.11a where all the branch vectors are shown, we plot in figure 3.11b the branch vectors associated to a contact force magnitude that is larger than 5 times the average contact force magnitude. Figure 3.11b clearly highlights that the cross-shaped particle assembly features many more and more ho-

3.3. Experimental and numerical results

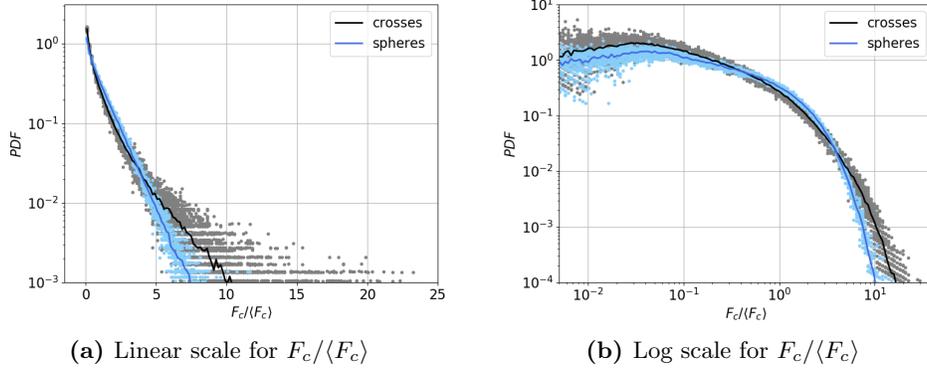


Figure 3.12: Probability density function (PDF) of the contact force magnitude F_c divided by the average contact force magnitude $\langle F_c \rangle$ just before the gate is opened, in the cases of assemblies of crosses (black) and spheres (blue).

more homogeneously distributed large forces than the spherical particle assembly. Indeed, figure 3.11 shows that large contact forces in the spherical particle assembly are located along what is commonly referred to in the literature as a *backbone*, i.e., a sub-network that features a small number of connected large contact forces, also called force chains, and that supports most of the external stress [164]. This prominent backbone sub-network is not visible in the cross-shaped particle assembly.

Probability density functions of contact forces

We analyze the probability density function (PDF) of the contact force magnitude F_c divided by the average contact force magnitude $\langle F_c \rangle$ in our granular assemblies. In figure 3.12, we compare the PDFs of $F_c/\langle F_c \rangle$ just before opening the gate in granular assemblies of crosses to that in granular assemblies of spheres. Each solid line corresponds to the average over N simulations and each dot represents a data point, i.e., a contact force magnitude in one of the N simulations. We take $N_{\text{spheres}} = 30$ and $N_{\text{crosses}} = 60$. In figures 3.12a and 3.12b, the average PDF of $F_c/\langle F_c \rangle$ in the assembly of crosses is considerably broader than that in the assembly of spheres. The largest $F_c/\langle F_c \rangle$ in the assembly of crosses is 32 while it is only 14 in the assembly of spheres. Forces in the range $F_c/\langle F_c \rangle \gtrsim 4$ are more probable in the assembly of crosses than in the assembly of spheres, in accordance with our qualitative observations of figure 3.11b. While the large forces are more homogeneously distributed in the assembly of crosses than in the assembly of spheres (no prominent backbone, see Section 3.3.3), considering the whole force range

3.3. Experimental and numerical results

and the coordination number indicates that assemblies of crosses are more connected but also less homogeneous. Figures 3.12a and 3.12b agree well with previous works published in the literature. Saint-Cyr et al. [177] also reported broader PDFs of $F_c/\langle F_c \rangle$ in the case of non-convex particles compared to the case of disks in their 2-dimensional study of sheared particles with various degrees of concavity. Estrada et al. [51] studied disk-shaped particles and modeled their entanglement by inhibiting rotational motion. Their PDFs of $F_c/\langle F_c \rangle$ (see figures 14 and 15 in their study) are very similar to our PDFs plotted in figures 3.12a and 3.12b. A notable difference with [51], however, is that the force network of their entangled granular assemblies qualitatively shows a well defined backbone while our assemblies of cross-shaped particles do not, as previously discussed in Section 3.3.3. We attribute this discrepancy to two factors: (i) no gravity in [51] versus gravity in our simulations; therefore our simulations cannot show “floating particles”, i.e., particles with no contact with any neighbouring particle, that promote the formation of a backbone and (ii) our high aspect-ratio particles not only reach neighbours located farther away than their disk-shaped counterpart, but their non-convexity allows two particles to contact at multiple points, therefore dividing a branch vector into several weaker branch vectors. To summarize, the PDFs of $F_c/\langle F_c \rangle$ shown in figures 3.12a and 3.12b, the absence of backbone in figure 3.11b together with the observed intermittent regime which highlights the high stability of assemblies of crosses show that assemblies of crosses do not necessitate the presence of a backbone to resist an exerted stress load. In a broader sense, this conclusion suggests that in granular assemblies of entangled particles both the diffuse sub-network of relatively large contact forces as well as the sub-network of small forces play an important role in resisting the exerted stress load [51]. We plot in Appendix B.2 similar comparisons of PDFs of $F_c/\langle F_c \rangle$ for: (i) assemblies of crosses about to collapse in a top-driven fashion versus in a buckling fashion, and (ii) assemblies in the intermittent regime about to collapse versus remaining stable. In both cases, the PDFs of $F_c/\langle F_c \rangle$ superimpose very well, suggesting that the distribution of the contact force magnitude is not relevant in determining the collapse outcome.

Fabric anisotropy

In granular dynamics, fabric anisotropy refers to the preferred directions of branch vectors $\overrightarrow{C_1 C_2}$ connecting the centers of mass of two contacting particles. While the branch vector direction coincides with that of the normal contact force in the case of spheres, it does not in the general case

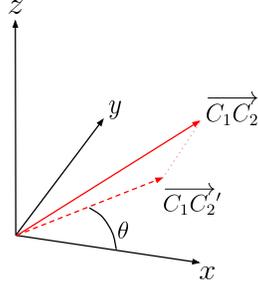


Figure 3.13: Definition of the branch vector projected onto the vertical xz -plane and of the angle θ it forms with the horizontal direction.

of non-spherical particles. In the following, we restrict the analysis to the branch vectors projected onto the xz vertical plane normal to the width of the channel, and we define the angle θ formed by the horizontal direction x and the projected branch vector $\overline{C_1C_2'}$, as shown in figure 3.13. Figure 3.14 compares the average fabric anisotropy of assemblies of spheres, crosses and coin-shaped particles. The collapse behavior of the assemblies of crosses considered in figure 3.14 belongs to various regimes: top-driven, buckling, intermittent-stable and intermittent-collapsed. Undesirable wall effects are minimized as follows: (i) the length L_0 of the initial column is chosen to be greater than 5 times the particle circumscribed diameter (except in figure 3.14b where the aspect ratio is required to be very high in order to observe buckling collapses as shown in figure 3.6) and (ii) contacts with the walls are excluded from the analysis.

Figure 3.14a reveals that the preferred direction of the branch vectors in the assembly of spheres is the vertical direction, as expected for a random loose packing of spheres under gravity. Conversely, the preferred direction of the branch vectors in the assembly of crosses is the horizontal direction, i.e., orthogonal to the direction of gravity. This result is simply due to the large aspect ratio of crosses and the fact that the stable position of a cross onto a horizontal plane is to lay flat and align its arms with the horizontal direction in order to minimize its gravitational potential energy. Consequently, in the absence of wall effects, the average angular position of crosses has a tendency to get closer to 0° and 180° as the aspect ratio of the cross-shaped particles increases. This can be generalized to other shapes with a large aspect ratio. To support this claim, we also plot in figure 3.14a the fabric anisotropy of coin-shaped particles with a thickness and a circumscribed diameter identical to those of crosses. Since these convex

3.3. Experimental and numerical results

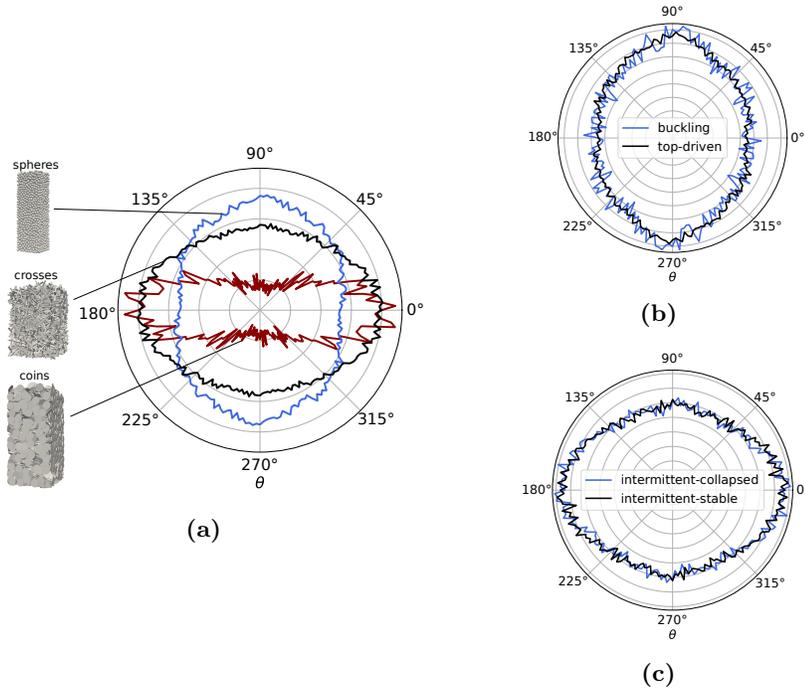


Figure 3.14: Average fabric anisotropy of various granular assemblies prior to the opening of the gate. (a) Comparison of the fabric anisotropy of spheres (averaged over 30 simulations) to crosses (averaged over 60 simulations) and coins (only one simulation shown) – the curve for coins has been scaled down for readability. (b) Comparison of the fabric anisotropy of crosses about to collapse in a top-driven fashion (averaged over 50 simulations) to crosses about to collapse in a buckling fashion (averaged over 6 simulations). (c) Comparison of the fabric anisotropy of crosses in the intermittent regime about to collapse (averaged over 18 simulations) to remaining stable (averaged over 26 simulations).

coin-shaped particles cannot entangle with neighbouring particles the way crosses do, their fabric anisotropy plotted in red in figure 3.14a reveals an even stronger tendency to lay flat and align with the horizontal direction than that of crosses.

Figure 3.14b shows the fabric anisotropy of assemblies of crosses leading to buckling collapse and to top-driven collapse. In this case the preferred direction of the branch vectors is vertical both in the buckling case and in the top-driven case. This result is simply a consequence of the rather small L_0 considered in these cases in order to be able to investigate large initial aspect ratio of the column while keeping the total number of particles in the system at a tractable level. In narrow columns with significant wall effects, crosses are constrained by the lateral walls to align their arms with the vertical direction. Disappointingly, the fabric anisotropy of assemblies of crosses leading to buckling collapse and of assemblies of crosses leading to top-driven collapse almost perfectly overlap. We also compare the fabric anisotropy of assemblies in the intermittent regime about to collapse (named intermittent-collapsed) to remaining stable (named intermittent-stable) in figure 3.14c. This plot does not reveal any marked difference either. The intermittent-collapsed fabric anisotropy and the intermittent-stable fabric anisotropy almost perfectly overlap, as in the comparison between buckling collapse and top-driven collapse.

At that stage, we conclude that the fabric anisotropy and the additional PDFs of $F_c/\langle F_c \rangle$ presented in Appendix B.2 do not seem to be proper descriptors of the microstructure able to reveal salient features that could distinguish the various collapse behaviors: top-driven versus buckling, and intermittent-stable versus intermittent-collapsed. In Section 3.4.1, we attempt to distinguish intermittent-collapsed from intermittent-stable through examining the average coordination number and the Janssen effects on the walls.

3.4 Discussion

3.4.1 Analyzing the intermittent regime with DEM simulations

In this section, we investigate the intermittent regime using our DEM simulations. Figure 3.15 presents a 3-dimensional view of the force network of configurations that (a) collapse and (b) remain stable. No qualitative discrepancy in the contact density nor the force magnitude can be visually noted. We could speculate that in the stable case (b) the largest contact

3.4. Discussion

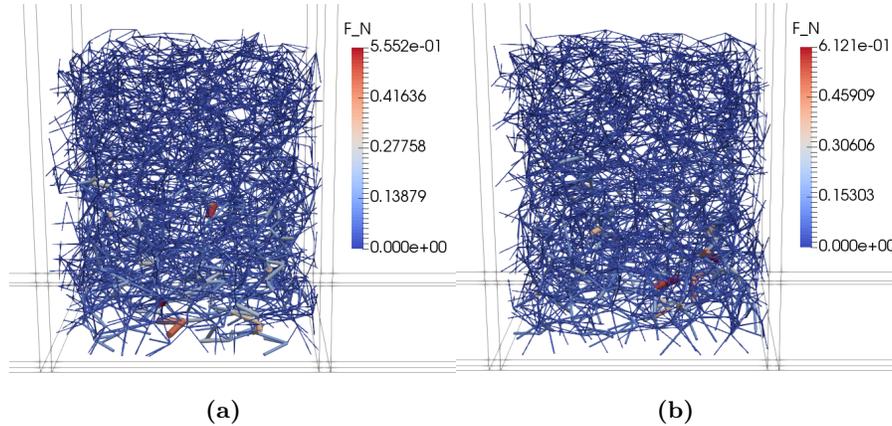


Figure 3.15: Visualization of the force network of two intermittent simulations. Each segment links a contact point to the center of mass of a particle, and is colored according to the magnitude of the normal component of the contact force. (a) The opening of the right gate leads to a granular collapse. (b) The opening of the right gate leads to a stable assembly.

force is located close to the gate, possibly reinforcing the stability of the granular assembly in this area. However, this does not hold for two reasons: (i) this particular stable case also remains stable when the left gate opens; and (ii) some asymmetrical assemblies similar to (b) result in granular collapses. Similarly, we could note the small difference in the maximum force magnitude between cases (a) and (b), but this maximum force magnitude is reversed for other simulations, and cannot be used as a determinant factor to predict a granular collapse.

Using a more systematic approach to investigate intermittence, we compute for each case a number of quantities such as the average coordination number, the Janssen effect on the walls, as well the number of contact points with the bottom wall, with the lateral walls and with the gate. The raw data is presented in appendix B.3. Surprisingly again, no trend is found in any of those average quantities, whether the whole granular assembly is considered or whether the analysis is restricted to subregions such as the region close to the gate and the region close to the top of the gate. We speculate that the stability of the assembly is governed by finer properties of the force network such as local clusters – i.e., subregions of densely entangled particles that are not well connected to the rest of the system – or the presence of force cycles, which tend to increase rigidity of granular assemblies. Investigating granular assemblies from the perspective of network analysis is growing in

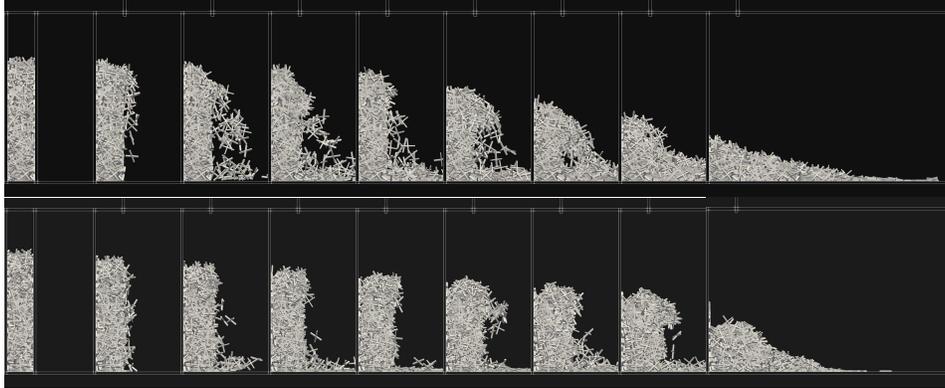


Figure 3.16: Snapshots of simulations with and without memory in the contact model. Top: with memory. Bottom: without memory. The frames are taken at the following times: $t = 0\text{s}$, $t = 0.15\text{s}$, $t = 0.30\text{s}$, $t = 0.45\text{s}$, $t = 0.60\text{s}$, $t = 0.75\text{s}$, $t = 0.90\text{s}$, $t = 1.05\text{s}$ and $t = 3.00\text{s}$.

interest [144], but it remains a difficult task and it is in most cases still restricted to systems of disk-shaped – sometimes spherical – particles. It is far beyond the scope of this thesis to investigate a reliable network analysis for three-dimensional, high-aspect ratios and non-convex particles.

3.4.2 On the importance of the contact model with memory

Since the implementation of the contact model with memory is quite tedious but necessary, it is interesting to compare the results obtained with and without memory. Figure 3.16 shows the qualitative discrepancies of two simulations with memory (top row of images) and without memory (bottom row of images) and with the same initial microstructure. When the memory term is included in the contact model, we can see that the height of the column is decreasing because particles fall off the top of the pile. Conversely, when the memory is not taken into account in the contact model, the height of the column decreases faster while the number of particles released from the top is lower. Indeed, without memory the tangential friction vanishes where the relative velocity approaches zero: as a result the granular column slowly deforms and overpacks at the bottom of the pile, while it should remain static in that region of space. This unphysical behavior was noted by Govender et al. [65]: in their study (figure 16), they comment on the slow deformation of a simulated arch of non-convex particles and note that it does not match experimental data, and that it even leads to the unphysical

discharge of particles. In the light of figure 3.16, we can confidently state that this unphysical deformation is the result of the absence of a memory term in the tangential friction force in their simulations. Figure 3.16 demonstrates that this memory term is of tremendous importance in order to numerically reproduce static configurations or dynamic configurations that include static subsystems.

3.4.3 On the choice of contact model parameters for non-spherical bodies

The tangential contact model parameters listed in table 3.1 are not obtained via the procedure provided in Section 2.4.2. This is because for reasonable values of the normal and tangential energy restitution coefficients e_n and e_t , the avalanche never occurs and the pile of particles remains stable at all times for all aspect ratios. We discuss below why this procedure did not give satisfactory results when applied to our system of cross-shaped particles.

First, we span a wide range of values for the tangential parameters and report the type of behavior that arises. Values of k_t range from 10^1 to 10^6 N/m and values of $\gamma_t = \eta_t/(2m_{ij})$ range from 10^0 to 10^5 s⁻¹. The results are presented in figure 3.17. We can see that only a narrow region of the space formed by possible values of k_t and γ_t leads to a physical output of the DEM simulations. This region corresponds to $k_t \approx 3 \times 10^2$ N/m and $\gamma_t \approx 10^3$ s⁻¹.

Then, we observe from Eq. (2.21–2.24) that once a maximum overlap distance δ_{\max} is set, the only two variables the tangential parameters k_t and η_t are depending on are e_n and e_t . For this analysis we are interested in orders of magnitude only, therefore we assume that the distances between the contact point and the center of mass of the cross-shaped particles are always the same and are equal to the radius of the sphere of the same volume R_s . We also denote by I_s the moment of inertia of the sphere of equal volume. Figure 3.18 shows plots of k_t and γ_t for e_n and e_t living in (0 : 1). This figure shows how k_t tends to zero extremely slowly when $e_n \rightarrow 0$. From Eq. (2.21) and Eq. (2.23) we deduce the behavior of k_t for $e_n \rightarrow 0$ and $e_t \rightarrow 1$:

$$k_t \underset{\substack{e_n \rightarrow 0 \\ e_t \rightarrow 1}}{\sim} \frac{C}{\ln(e_n)^2} \quad \text{with} \quad C = \frac{(\pi v_{\max}/(\delta_{\max} e))^2}{\frac{1}{m_{ij}} + 2 \frac{R_s^2}{I_s}} \quad (3.1)$$

In our case, since $C \approx 2 \times 10^5$ N/m, this implies that e_n must be of the order of 10^{-11} to reach $k_t \approx 3 \times 10^2$ N/m. Such a low value of the energy

3.4. Discussion

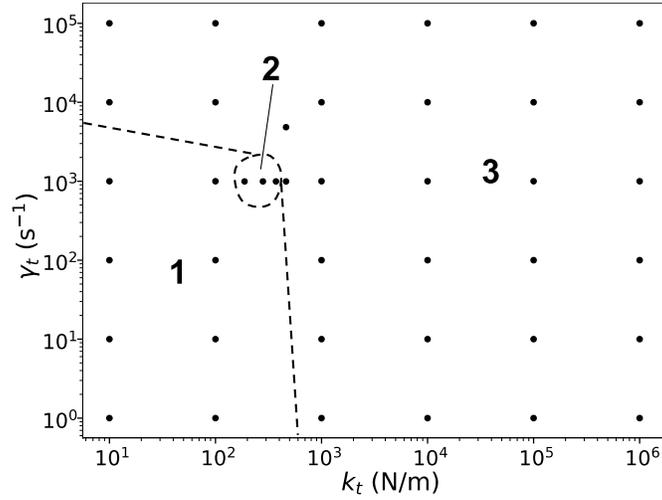


Figure 3.17: Variation of the flow behavior with respect to the tangential parameters. Each dot corresponds to a simulation with specific tangential parameters (k_t, γ_t) . Region 2 correctly reproduces experiments. Region 1 leads to avalanches that collapse like convex particles, i.e., the base of the pile drives the motion. Region 3 leads to fully entangled systems that do not collapse.

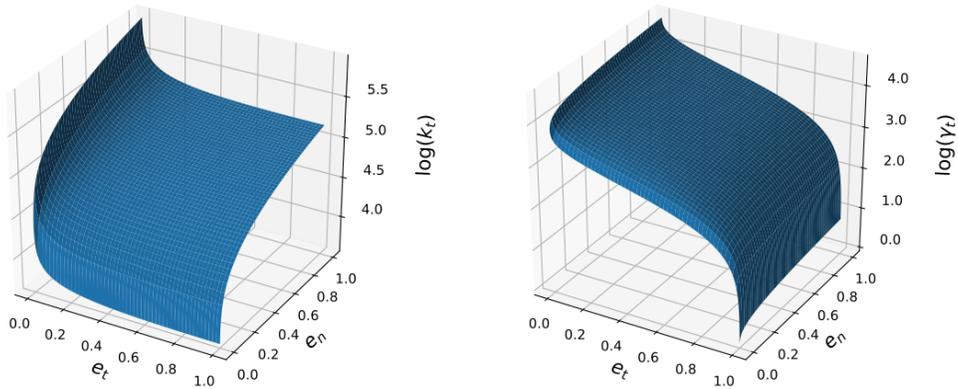


Figure 3.18: Surface plots of the tangential parameters k_t and $\gamma_t = \frac{\eta_t}{2m}$ for e_n and e_t varying from 0 to 1.

restitution coefficient is of course strongly unphysical, since it would imply that a particle impacting a plane bounces back with a velocity 10^{11} times smaller than the impact velocity.

At this point, we recognize that even though this procedure has been successfully applied to spherical shapes and ellipsoids [181], it cannot be used for particle shapes that differ significantly from spheres, such as the non-convex and angular particles considered in this study. To the best of our knowledge, no procedure has been successfully derived to determine methodically the tangential contact parameters of complex-shaped particles. In fact, Landauer et al. [119] face the same difficulties in choosing meaningful friction parameters and conclude that “it is impossible to use DEM simulations as a plug-and-play tool to predict particle packing and flow properties”. As a result, the community keeps relying on calibration studies or on trial and error. Investigating further a reliable, physically-based method or providing calibration guidelines [172] to set contact model parameters in DEM simulations of arbitrary-shaped particles would be of great interest in future works.

3.4.4 Analogy with other stable granular assemblies

Entanglement is not the only mechanism that can lead to stable piles, and such stable granular columns have been reported in the case of cohesive particles [1] and fluid-saturated particles [24]. In [1], Abramian et al. conduct discrete and continuous two-dimensional simulations of the slumping of a column of cohesive particles and observe stable and collapsed regimes. In [24], Bougouin et al. examine the problem of the slumping of a fluid-saturated column of particles. In their study, varying the “column” Bond number, that compares gravity to capillary effects, also leads to stable and collapsed regimes. The authors carry out a dimensional analysis that captures very well the transition between the observed regimes. It is important to stress that in order to observe stable granular assemblies, a stabilizing mechanism at the particle scale is required. While in the case of Bougouin et al. (respectively Abramian et al.) the capillary force (respectively the cohesive force) plays the role of the stabilizing mechanism, in our study the stabilizing effect comes from the inhibition of particle displacement resulting from the entanglement of the particles with their neighbours. In a future work, we might attempt to design a column Bond number for entangled granular media and carry out a dimensional analysis similar to that of [24] in order to capture the transitions between the observed regimes. As this column Bond number would be the ratio of the gravitational force over the

stabilizing force, it would require to include a measure of the aforementioned displacement inhibition due to the presence of entangled neighbours. In a first approximation, this column Bond number could be a function of a mean entanglement number that would depend on the geometric properties of the particles. This entanglement number may be defined following the approach of Gravish et al. [66], i.e., as the number of neighbours crossing a portion of a plane specific to the shape of the considered particles – in the case of our cross-shaped particles, it would likely be a disk of diameter L_p and located in the plane formed by the arms of the crosses. Then, the entanglement number would be averaged over all particles in the column and treated as a particle feature in the dimensional analysis.

3.5 Conclusion

In this chapter, we performed granular dam break experiments and simulations of angular non-convex particles with high aspect ratios. We reported a wide range of flow regimes and dynamic behaviors, such as the intermittent regime and two collapsing dynamics: the top driven collapse and the buckling collapse. Notable differences in the collapses of the considered cross-shaped particles with respect to spheres include a strong three-dimensionality of the avalanche, the absence of repeatability of most behaviors, the presence of staggered collapses and the irrelevance of the time scale of free fall $\tau_c = \sqrt{H_0/g}$. Surprisingly, we showed that despite fundamentally different dynamics, macroscopic quantities of our avalanches of non-convex particles such as the run-out distance and the final height of the pile agree reasonably well with results from the literature obtained with spherical particles. We also showed that our numerical simulations reproduce every flow behavior observed experimentally and we obtained an excellent quantitative agreement between our simulations and our experiments. In particular, we emphasized that the use of a contact model with memory is critical as soon as there is a sub-system in a pseudo static state – in our case, the bottom part of the granular pile. Futures studies could investigate avalanches of very large systems of entangled particles (of the order of 10^6 particles) in order to reduce the uncertainty on the measured run-out distance and final height.

Using our validated simulations, we investigated the contact force magnitude distribution and the fabric anisotropy in the granular assemblies before opening the gate. These data cannot be measured experimentally and is accessible via numerical simulations only. Our analysis of the average

3.5. Conclusion

probability density functions (PDFs) of the scaled contact force magnitude revealed that the distribution of forces in the case of cross-shaped particles is considerably broader than in the case of spheres and that the network of strong forces does not form a backbone in the case of cross-shaped particles. Our attempt to discriminate the top-driven collapse from the buckling collapse, as well as the intermittent-stable regime from the intermittent-collapsed regime by means of PDFs of contact force magnitude and fabric anisotropy was not conclusive, suggesting that these collapse behaviors are governed by other properties of the granular assembly that at this stage we could not identify. Pursuing our investigation of the stability of granular assemblies in the intermittent regime, we found that quantities such as the average coordination number, the average number of contact points with the walls and with the gate as well as the Janssen effect on the walls do not allow either to predict if an intermittent structure is about to collapse. This suggests that whether the structure collapses or remains stable is governed by mesoscale properties of the force network, such as local clustering. Future studies could investigate the mesoscale structures of granular networks of non-spherical particles and their implications on the dynamics of the system, as well as attempt to define an entanglement number as in [66] and investigate its relevance in describing the various observed behaviors. Finally, we discussed the choice of model parameters of our numerical simulations. We found that previously developed procedures used to set the tangential model parameters of spherical and ellipsoidal particles did not lead to physical behaviors when applied to our systems of angular non-convex particles, resulting in a significant amount of time and resources spent to set the model parameters from trial and error. We emphasize the need for a reliable, physically based routine or calibration guidelines to set model parameters in a reproducible way, not only for spherical particles but also for particles of arbitrary shape.

Part II

Immersed elastic capsules

Chapter 4

A Cartesian-octree adaptive front-tracking solver for immersed elastic capsules

4.1 Introduction

The numerical study of membrane-enclosed fluid objects, or capsules, has seen tremendous interest over the past three decades due to the wide range of applications in the biomedical and bioengineering world. Indeed, numerical simulation of capsule dynamics in viscous flows is crucial to better characterize and understand blood flow through capillary microcirculation and develop applications such as targeted drug delivery [43], migration of cancerous leukocytes through the microvascular network [18, 161] and cell sorting and cell characterization in microfluidic devices [22, 96]. In particular, the latter application has the potential to speed-up labour-intensive diagnosis procedures or to extract relevant components of biofluids. For instance inertial centrifugation in spiral-shaped microchannels has been shown to efficiently and accurately segregate cells based on their size, and could be applied to perform non-destructive blood plasma extraction [52, 55, 192].

The study of capsules from a mechanical point of view was paved in 1981 by the pioneering analytical work of Barthès-Biesel & Rallison [21], who derived from the thin-shell theory a time-dependant expression for the deformation of an elastic capsule in a shear flow in the limit of small deformations. A decade later, Pozrikidis went beyond the assumption of small deformations, using the Boundary Integral Method (BIM) to investigate finite deformations of elastic capsules in a shear flow [156, 170]. This work was quickly followed by Eggleton & Popel who simulated spherical and biconcave capsules in shear flows using the Front-Tracking Method (FTM) [49]. Capitalizing on the advantages of the BIM – such as a lower computing cost compared to the FTM, and the ability to simulate true Stokes conditions – Pozrikidis investigated the bending resistance of capsules and proposed a

simplified bending model for biological membranes valid for small deformations [157], leading to the first numerical simulation of an RBC based on the thin-shell theory [158]. The work of Pozrikidis was later extended by Zhao et al., who proposed a BIM able to simulate RBCs in complex geometries with up to 30% volume fraction [219]. In the 2000s, Barthès-Biesel and Lac also used the BIM and studied finite deformations of capsules devoid of bending resistance: they considered the effect of the membrane constitutive law and exhibited buckling instabilities [114] as well as the dynamics of two interacting capsules in a shear flow [115].

Despite the major success of the BIM to simulate capsules and biological cells, the FTM is still being developed. Indeed, while the FTM is more computationally intensive than the BIM because it necessitates meshing the whole 3D fluid domain, and while it can require very small time steps to satisfy stability conditions depending on the considered membrane forces; the FTM can handle inertial regimes, thus allowing to examine a wider range of applications. As such, Bagchi uses the FTM to perform two-dimensional simulations of several thousand RBCs in a shear flow [12], allowing the study of RBC interactions at the mesoscale. In the next years, Doddi & Bagchi [45] and Yazdani & Bagchi [212–214] develop respectively three-dimensional implementations of the elastic membrane stress and of the Helfrich’s bending stress for biological membranes, the latter not being limited to small deformations as was the case for the formulation of Pozrikidis. The ability to consider finite Reynolds numbers allowed Doddi & Bagchi to extend the work of Lac et al. on capsule interactions to inertial flows [44]. Their framework was later extended to complex geometries by Balogh & Bagchi [15], enabling them to study the dynamics of hundreds of RBCs in a microvascular network with a hematocrit (volume fraction of RBCs) of 30% [16]. Another variant of the FTM is to use a Lattice-Boltzman fluid solver rather than the traditional PDE-based Navier-Stokes solver: this can bring significant performance improvement especially at low Reynolds number where the Lattice-Boltzman Method (LBM) performs well. For instance, Li & Sarkar extend the work of Barthès-Biesel and Lac on the instabilities of elastic capsules in shear flows using an FTM-LBM solver [121], and Zhang et al. describe a similar framework able to simulate RBCs [216], including cell-cell aggregation phenomena [217]. More recently, Ames et al. [7] harnessed the performance improvements of GPUs and demonstrated an impressive 17 million RBCs simulated in a microvascular network using a similar FTM-LBM framework.

Other methods to simulate biological capsules and vesicles include the RBC model of Fedosov et al. [54]. In the work of Fedosov, the RBC mem-

brane mechanics is not governed by the thin shell theory but rather by a coarse-grained molecular dynamics model. The membrane of the RBC is discretized, with each edge representing several nonlinear springs which correspond to elastic and viscoelastic properties of the membrane of the RBC. The model parameters are found for an extremely fine mesh of over 27000 nodes, where the lengths of the edges correspond to that of biological spectrins. Yet, in Fedosov's model practical RBC simulations are conducted with different model parameters which are intended to display the same mechanical behavior as that obtained with a fine mesh, but using a number of nodes orders of magnitude lower. These coarse-grained mechanical properties of the membrane are shown to lead to results which accuracy lies within the range of experimental measurement errors for the specific cases considered. However the range of validity of this coarse-graining step is not obvious and the spatial convergence can be non-monotonous or even not exist (see the transverse diameter plot in figure 1 in [54]), indicating to use this model with care. A last approach to capsule simulations is to adopt a fully Eulerian framework, where the membrane is not discretized with Lagrangian nodes, edges and faces. Instead, the capsule configuration is described using the Eulerian grid employed to solve the Navier-Stokes equations. Removing the need of a Lagrangian grid is a desirable description as the IBM can reduce the spatial and temporal accuracies to first order if no special treatment is implemented. In the Eulerian capsule description, Volume-Of-Fluid (VOF), level-set or phase-field methods can be utilized to track the position of the membrane, similar to what is done in the context of fluid-fluid interfaces [25, 78, 153, 193]. If the considered membrane mechanical behavior is independent of the past configuration of the membrane, for instance if there is no resistance to shear and high resistance to bending, the membrane forces can be computed using techniques developed for surface-tension flows: the local curvature can be computed using height-functions in the case of a VOF description, or by numerically differentiating the level-set or phase-field function near the interface [37]. However, in most biological applications the membrane properties do depend on the past membrane configurations due to its elastic behavior. In such cases a quantity representing the membrane stretch needs to be initialized and advected in the vicinity of the membrane, for instance the left Cauchy-Green deformation tensor. Ii et al. have demonstrated that this approach is possible and scalable [92–94], although more comparisons with the FTM are needed in order to evaluate the performance and the accuracy of the Eulerian methods, especially for long-lasting simulation. An in-depth description of the fully Eulerian method to simulate elastic capsules is presented in Chapter 6.

Concomitant to these developments of capsule simulations, the IBM gained great popularity in the particle-laden flows community [26, 105, 194], and in the past two decades some adaptive IBM have been proposed in cases of immersed solid particles. In this context, Roma et al. [174] present a two-dimensional adaptive IBM implementation where the Adaptive Mesh Refinement (AMR) is achieved by means of overlapping rectangles – or “patches” – of finer grid cells in the regions of interest, i.e. where higher accuracy of the flow field is needed. This method was later improved and proved second-order accurate by Griffith et al. [68] and Vanella et al. [200]. Previously, Agreasar et al. [3] had used a non-patched adaptive FTM-IBM method in order to simulate axisymmetric circular cells. Their IBM implementation did not use Peskin-like regularized Dirac delta functions: instead the Lagrangian grid on the membrane communicates with the background Eulerian grid via an area-weighted extrapolation. More recently, Cheng & Wachs [31] used the IBM coupled with an LBM solver to achieve adaptive simulations in the case of a single rigid sphere in various flow conditions.

The goal of this chapter is to present an efficient framework to study the dynamics of dilute suspensions of capsules in complex geometries, not limited to non-inertial regimes. As the BIM cannot be used at finite Reynolds numbers, we use the FTM and therefore the whole 3D fluid domain is discretized. Since a vast range of realistic applications consider geometries of sizes orders of magnitude larger than the typical size of a capsule, requiring hundreds of millions to billions of Eulerian grid cells when the Cartesian grid has a constant grid size, we develop an adaptive FTM solver rendering achievable to simulate configurations that were previously out of reach with a constant grid size. We provide the open-source code as part of the Basilisk platform [80, 150, 152, 153].

The chapter is organized as follows: in Section 4.2 we present the problem formulation and the governing equations for both the fluid and the capsule dynamics. We describe the implementation of our numerical model in Section 4.3, emphasizing the finite element membrane model and the FTM method. Numerous validation cases are shown in Section 4.4, for increasingly difficult configurations: validations are performed by comparing our computed results against accurate BIM data available in literature whenever possible, otherwise against other FTM results. Section 4.5 contains new results generated with the present method, where the adaptive mesh capability dramatically improves computational efficiency. In Section 4.6 we summarize our work and discuss the strengths, weaknesses and possible improvements of the present method as well as future perspectives.

4.2 Governing equations

4.2.1 Fluid motion

The fluid phase is assumed Newtonian and incompressible: the fluid surrounding and enclosed by the elastic membranes is described using the mixture Navier-Stokes equations:

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \nabla \cdot (\mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)) + \mathbf{f}_b \quad (4.1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (4.2)$$

where \mathbf{u} is the velocity, p is the pressure, ρ is the constant density and μ is the variable viscosity field, since we will consider non-unity viscosity ratios $\lambda_\mu = \mu_i/\mu_e \neq 1$, with μ_i and μ_e the internal and external viscosities. \mathbf{f}_b denotes the body force containing the membrane elastic and bending force densities acting on the fluid: $\mathbf{f}_b = \mathbf{f}_{\text{elastic}} + \mathbf{f}_{\text{bending}} = (\mathbf{F}_{\text{elastic}} + \mathbf{F}_{\text{bending}})/V$, with V a relevant control volume and $\mathbf{F}_{\text{elastic}}$ and $\mathbf{F}_{\text{bending}}$ the integrated membrane force densities.

4.2.2 Membrane mechanics

We assume that the lipid-bilayer membrane is infinitely thin: please note that this is not a strong assumption for most biological cells, as thickness of the biological membrane (lipid-bilayer) is $5nm$ while an RBC characteristic size is $10\mu m$. A biological membrane undergoing deformation responds with elastic and bending stresses, described with two distinct mechanical models.

The elastic strains and stresses are described using the theory of thin shells[67]. We summarize this framework here, but the interested reader is referred to the work of [21] for more details. In this continuous description of the capsule, we first introduce the projectors $\mathbf{P} = \mathbf{I} - \mathbf{n}\mathbf{n}$ and $\mathbf{P}_R = \mathbf{I} - \mathbf{n}_R\mathbf{n}_R$ onto the current and reference (stress-free) membranes shapes, with \mathbf{I} the identity tensor and \mathbf{n} and \mathbf{n}_R the unit normal vectors to the current and reference membranes configurations, which are both oriented outward. The membrane strains are described using the surface deformation gradient tensor \mathbf{F}_s , derived from the classical deformation gradient tensor \mathbf{F} as follows:

$$\mathbf{F}_s = \mathbf{P} \cdot \mathbf{F} \cdot \mathbf{P}_R. \quad (4.3)$$

The surface right Cauchy-Green deformation tensor \mathbf{C}_s is then defined from \mathbf{F}_s :

$$\mathbf{C}_s = \mathbf{F}_s^T \cdot \mathbf{F}_s. \quad (4.4)$$

4.2. Governing equations

Let the three eigenvalues of \mathbf{F}_s be $\lambda_1, \lambda_2, 0$ associated with the eigenvectors $\mathbf{t}_1, \mathbf{t}_2, \mathbf{n}$. Then the eigenbasis of \mathbf{C}_s is the same as that of \mathbf{F}_s , associated with eigenvalues $\lambda_1^2, \lambda_2^2, 0$. Note that in the stress-free configuration, at the beginning of a typical simulation, $\lambda_1 = \lambda_2 = 1$, and $\mathbf{F}_s = \mathbf{C}_s = \mathbf{P} = \mathbf{P}_R$.

The above quantities are useful to compute the membrane elastic stress, which can be expressed using a surface strain-energy function $W_s(\lambda_1, \lambda_2)$:

$$\sigma_i = \frac{1}{\lambda_j} \frac{\partial W_s}{\partial \lambda_i}, \quad i \neq j. \quad (4.5)$$

In this work, two distinct strain-energy functions corresponding to two membrane elastic laws are used to describe several types of lipid bilayers in various conditions:

1. The Neo-Hookean law, used to describe vesicles and artificial capsules, and which corresponding strain-energy function is

$$W_s^{NH} = \frac{E_s}{6} \left(\lambda_1^2 + \lambda_2^2 + \frac{1}{\lambda_1^2 \lambda_2^2} - 3 \right), \quad (4.6)$$

where E_s denotes the shear modulus.

2. The Skalak law, used to describe the elastic response of RBC membranes, and which strain energy function is

$$W_s^{Sk} = \frac{E_s}{4} (I_1^2 + 2I_1 - 2I_2 + CI_2^2), \quad (4.7)$$

where the invariants $I_1 = \lambda_1^2 + \lambda_2^2 - 2$ and $I_2 = \lambda_1^2 \lambda_2^2 - 1$ have been introduced, as well as the area dilatation modulus C preventing strong area changes and is taken “large” [20, 156] in order to describe the strong area incompressibility of RBCs. Unless otherwise stated, the value $C = 10$ is used in the simulation results presented below.

Once the elastic stress is known, the elastic force exerted by the membrane onto the fluid is simply

$$\mathbf{F}_{\text{elastic}} = \nabla \cdot \boldsymbol{\sigma}, \quad (4.8)$$

although we will follow the approach of [29] and use the principle of virtual work instead of directly computing the divergence of the stress, as explained in Section 4.3.4.

The bending stresses are described using Helfrich’s bending energy per unit area \mathcal{E}_B [77]:

$$\mathcal{E}_B = 2E_b (\kappa - \kappa_0)^2 + E_g \kappa_g \quad (4.9)$$

4.3. Numerical method

where $\kappa = (\kappa_1 + \kappa_2)/2$ is the local mean curvature, $\kappa_g = \kappa_1\kappa_2$ is the Gaussian curvature, and E_b and E_g are their associated bending moduli. κ_1 and κ_2 are the two principal curvatures at a given point of the two-dimensional membrane sheet, and κ_0 is the reference curvature. Then, the bending stresses are derived from the total bending energy taking the variational derivative of Eq. (4.9), to yield the normal bending force per unit area [71]:

$$\mathbf{F}_{\text{bending}}/A = -2E_b(\Delta_s(\kappa - \kappa_0) + 2(\kappa - \kappa_0)(\kappa^2 - \kappa_g + \kappa_0\kappa))\mathbf{n}, \quad (4.10)$$

where the operator Δ_s is the surface Laplacian – or Laplace-Beltrami operator – defined as $\Delta_s = \nabla_s \cdot \nabla_s = ((\mathbf{I} - \mathbf{n}\mathbf{n}) \cdot \nabla) \cdot ((\mathbf{I} - \mathbf{n}\mathbf{n}) \cdot \nabla)$, and A is a relevant control area. Note how E_g has disappeared in the variational formulation because κ_g is a topological invariant [19, 71].

At this point a parallel with surface tension forces is enlightening: the bending energy related to surface tension acting on a droplet is proportional to the area of the interface, and leads to surface tension forces proportional to the curvature, i.e. to the second derivative of the interface geometry. In contrast, as stated above the bending energy related to lipid-bilayer membranes is proportional to the curvature, and thus the corresponding bending force depends on the second derivative of the curvature, i.e. to fourth-order derivatives of the geometry. As such, the numerical simulations of biological capsules subject to bending stresses is a formidable challenge, and the interested reader is referred to the reviews of [71, 72]. Our approach to computing the bending force is described in Section 4.3.4, and corresponds to method E in [72].

4.3 Numerical method

4.3.1 Adaptive Finite Volume solver for the Navier-Stokes equations

Assuming the body force field \mathbf{f}_b is known, Eq. (4.1–4.2) are solved using the open-source platform Basilisk [150]. The viscous term $\nabla \cdot (\mu(\nabla u + (\nabla u)^T))$ is treated implicitly using a multigrid Poisson solver [152, 153], the incompressibility condition is satisfied by the classical projection method of Chorin [32], and the advection term $\mathbf{u} \cdot \nabla \mathbf{u}$ is solved using the second-order Bell-Colella-Glaz upwind advection scheme [23]. To this end, the divergence-free velocity field and the viscosity field are located on cell faces while a velocity field approximately divergence-free, the pressure field, and the body force field are all defined on the cell centers.

Basilisk computes the solution of the Navier-Stokes equations on an octree grid, which allows to coarsen and refine computational cells throughout the simulations while keeping a structured mesh. The coarsening and refinement of grid cells is implemented using a wavelet-based algorithm: for the sake of completeness we present here a short overview of the adaptivity process, and the interested reader is referred to [152, 153, 199] for more in-depth descriptions. Let f be a field of interest which variations in space will govern the size of the grid cells. First, f is downsampled onto a lower level grid by volume-averaging: we call this downsampled field f_d . Then, f_d is upsampled back to the original grid using second-order interpolations, resulting in the downsampled-then-upsampled field f_{du} . Since f and f_{du} are defined on the same (original) grid, the sampling error field $\epsilon^i = \|f^i - f_{du}^i\|$ can be defined in each computational cell i . Finally, ϵ^i is used to decide if cell i should be coarsened or refined based on an adaptivity criterion $\zeta > 0$:

$$\begin{cases} \epsilon^i > \zeta & \Rightarrow \text{refine cell } i \\ \epsilon^i < \frac{2\zeta}{3} & \Rightarrow \text{coarsen cell } i \\ \frac{2\zeta}{3} \leq \epsilon^i \leq \zeta & \Rightarrow \text{leave cell } i \text{ at its current level.} \end{cases} \quad (4.11)$$

This wavelet-based algorithm is very versatile as any field of interest can be used to influence the refinement level of the octree grid. In this study, the adaptivity is based on the velocity field and on both the presence of domain boundaries and the capsule membrane. In other words, the fields of interest that the wavelet adaptivity algorithm considers are: u_x , u_y , u_z , c_s and ξ , where the first three scalar fields are the three components of the velocity field \mathbf{u} , c_s is the fluid volume fraction field (in case of complex geometries), and ξ is a scalar field which varies strongly in the vicinity of the membrane and is constant elsewhere – see Section 4.3.3 for a definition of ξ . The refinement criterion ζ can be different for different fields of interest. In this study, we choose ζ to be very small when applied to c_s and ξ in order to impose the maximum level of refinement in the vicinity of walls and of the membrane (typically we choose $\zeta < 10^{-10}$); while we found by trial and error that having ζ of the order of 1% of the characteristic velocity when applied to u_x , u_y and u_z leads to satisfactory refinement and coarsening of computational cells in the rest of the fluid domain.

4.3.2 Second-order treatment of solid embedded boundaries

In Basilisk, complex geometries are handled using the sharp, second-order and conservative embedded boundaries method of Johansen and Colella

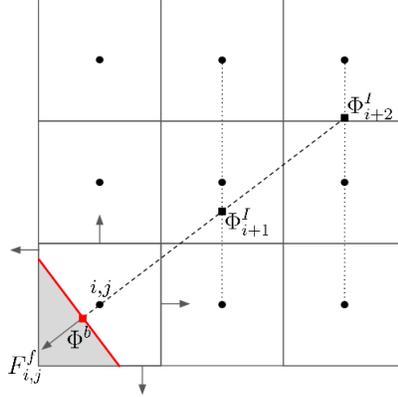


Figure 4.1: A two-dimensional example of an interpolation stencil estimating the boundary flux Φ^b with second-order accuracy. The gray area denotes the solid, the red line denotes the solid boundary, and the arrows denote the five face fluxes that are computed in the cut cell of interest i, j . The circle dots denote the Eulerian cell centers while the square dots show the locations of the data points Φ from which the boundary flux $F_{i,j}^f$ is interpolated. The interpolation line normal to the solid boundary is represented by a dashed line. In case of Dirichlet boundary conditions, Φ^b is known but the data points Φ_{i+1}^I and Φ_{i+2}^I are themselves interpolated with second-order accuracy along the dotted lines using the centers of the Eulerian grid cells.

[100]. In this method, the solid boundaries are assumed to cut the Eulerian cells in a piecewise linear fashion. Boundary conditions are enforced by estimating the flux on the solid boundary using second-order interpolations on a stencil involving the surrounding fluid cells. An example of such interpolation stencil in two dimensions is shown in figure 4.1. This method can be implemented so that only the volume and face fluid fractions are necessary to describe the boundary and recover the boundary flux [152]. As such, at the beginning of the simulation these two fluid fraction fields are generated from either a user-defined level-set function describing the geometry or an STL file.

If no additional treatment is done, it is well known that this class of cut-cell methods suffers from unreasonably strict CFL restrictions due to cells with small fluid volume fractions. Indeed, when a cell is cut by a solid boundary the effective CFL condition becomes $\Delta t < c\Delta x/(f|\mathbf{u}|)$, where c and f are the volume and face fractions, and $|\mathbf{u}|$ is the velocity norm in the considered cell. If c/f is close to zero, the time step Δt may become arbitrarily small. To alleviate this issue, a “flux redistribution” technique is carried out, where the fluxes of problematic small cells are redistributed to

their neighbors, thus preventing Δt from becoming arbitrarily small [35, 58, 154].

4.3.3 Front-Tracking Method (FTM)

Standard FTM formulation

The capsule configuration is described using the Front Tracking Method: we adopt a Lagrangian representation of the capsule, which we cover by a triangulated, unstructured mesh [193, 195]. This Lagrangian mesh communicates with the Eulerian octree grid used to describe the background fluid by means of regularized Dirac-delta functions introduced by Peskin [147], which role is to interpolate velocities from the Eulerian grid to a Lagrangian node; and to spread membrane forces from a Lagrangian node to the Eulerian grid. In this paper we use a cosine-shaped regularized delta function:

$$\delta(\mathbf{x}_0 - \mathbf{x}) = \begin{cases} \frac{1}{64\Delta^3} \prod_{i=1}^3 \left(1 + \cos\left(\frac{\pi}{2\Delta}(x_{0,i} - x_i)\right)\right) & \text{if } |x_{0,i} - x_i| < 2\Delta, \\ 0 & \text{otherwise,} \end{cases} \quad (4.12)$$

where Δ is the length of an Eulerian cell and $\mathbf{x}_0 = [x_0, y_0, z_0]$ corresponds in practice to the coordinates of a Lagrangian node. The prefactor $1/(64\Delta^3)$ ensures that the discrete integral over the whole space $\int_{\Omega} \delta(\mathbf{x}_0 - \mathbf{x}) d\mathbf{x}$ is equal to 1. Then, the velocity \mathbf{u}_0 of a given Lagrangian node located at \mathbf{x}_0 is interpolated from the Eulerian velocity field \mathbf{u} using:

$$\mathbf{u}_0 = \int_{\Omega} \mathbf{u}(\mathbf{x}) \delta(\mathbf{x}_0 - \mathbf{x}) d\mathbf{x} \iff \mathbf{u}_0 = \sum_{i \in \text{stencil}} \mathbf{u}_i \delta(\mathbf{x}_0 - \mathbf{x}_i) \Delta^3, \quad (4.13)$$

where ‘‘stencil’’ denotes the Eulerian cells which center \mathbf{x}_i is such that $\delta(\mathbf{x}_0 - \mathbf{x}_i) \neq 0$, and \mathbf{u}_i is the velocity of a given fluid cell. Similarly, the membrane force \mathbf{F}_0 at a Lagrangian node is spread to a force density field \mathbf{f} using:

$$\mathcal{M}(\text{supp}(\delta)) \mathbf{f} = \int_{\Omega} \mathbf{F}_0 \delta(\mathbf{x}_0 - \mathbf{x}) d\mathbf{x} \iff \mathbf{f}_i = \mathbf{F}_0 \delta(\mathbf{x}_0 - \mathbf{x}_i), \quad (4.14)$$

where $\mathcal{M}(\text{supp}(\delta))$ is the measure of the support of the regularized Dirac-delta function, and is in practice equal to $64\Delta^3$ in three dimensions for the regularized Dirac-delta function we choose in Eq. (4.12).

Once the Lagrangian velocities of all the capsule nodes have been interpolated from the Eulerian velocity field using Eq. (4.13), the position of each node is updated using a second-order Runge-Kutta time integration scheme. The membrane stresses are then computed from the new configuration of the capsule using the methods described later in Section 4.3.4, and transferred to the background fluid using Eq. (4.14). In the context of particle-laden flows, the use of the IBM requires sub-time stepping for the particle advection due to the orders of magnitude difference in the fluid time scale and the solid-solid interactions time scale [194]. This is not necessary for capsule-laden flows described by the FTM, i.e. the time step for the advection of Lagrangian nodes is equal to that of the fluid solver. It follows that the trajectories of each Lagrangian node coincide with the streamlines of the flow and that the triangulations of two interacting capsules can never overlap – provided that the time step is sufficiently small. As such, there is no need for any ad-hoc repulsive force between two approaching capsules or between a capsule approaching a wall, as was the case in e.g. [128]: the non-penetration condition is seamlessly handled by the local flow field. In the latter case of close interaction between a capsule and a wall, however, the definition of the IBM stencil must be altered, as described in the next subsection.

Additionally, the case of capsules of inner viscosity μ_i different from the viscosity of the surrounding fluid μ_e needs special treatment. We adopt the approach developed in the original FTM by Unverdi & Tryggvason [195]: a discrete indicator function I is computed from a discrete “grid-gradient” field $\mathbf{G}(\mathbf{x})$:

$$\mathbf{G}(\mathbf{x}) = \sum_{i \in \mathcal{T}} S_i \delta(\mathbf{x} - \mathbf{x}_i) \mathbf{n}_i, \quad (4.15)$$

where \mathcal{T} denotes the set of all triangles of the discretization of the surface of all the capsules, S_i is the surface area of triangle i , \mathbf{x}_i is the position vector of its centroid and \mathbf{n}_i is its unit *inward* normal vector. In practice, $\mathbf{G}(\mathbf{x})$ is computed by looping over all triangles of the discretizations of all capsules and spreading the quantity $S_i \mathbf{n}_i$ using the regularized Dirac-delta functions introduced previously. As such, $\mathbf{G}(\mathbf{x})$ is non-zero in the union of all the IBM stencils. The discrete indicator function $I(\mathbf{x})$ is computed by solving the following Poisson problem:

$$\Delta I = \nabla \cdot \mathbf{G}. \quad (4.16)$$

Since I is a regularized step function, it should have constant values away from the capsule membranes. To guarantee this property, we only update I

in the cells where \mathbf{G} is non-zero and we re-initialize I to 0 or 1 elsewhere, as suggested in [193].

Adaptive FTM strategy

Our current implementation of the FTM requires all cells in an IBM stencil to be the same size Δ . As a result, all the Eulerian cells around the membrane must be the same size as well, although future studies may lift this restriction. In practice, since the flow physics is happening in the vicinity of the capsule, we need the cell sizes around the membrane to be the smallest grid size of the fluid domain. To enforce this condition, we create a scalar field ξ initialized at each time step to be: (i) 0 if the Eulerian cell does not belong to any IBM stencil; or (ii) a randomly generated value between 0 and 1 otherwise. In other words, the scalar field ξ tags the IBM stencils with noise while the rest of the domain is set to a constant value. Feeding this scalar field to Basilisk’s wavelet adaptation algorithm ensures that all the stencil cells are defined at the finest level, and that no IBM stencil contains Eulerian cells of different cell levels.

4.3.4 Computation of the membrane forces

Computation of the elastic force with the Finite Element Method

In order to compute the nodal elastic forces given by Eq. (4.8), we employ a Finite Element Method (FEM). In most FEM solvers from Engineering applications, the sought quantity is the displacement of a structure under a known applied stress. In the case of biological membranes, we rather seek the internal stress of the membrane under a known displacement [19]. Charrier et al. [29] have been the first to design this specific FEM framework: we base our implementation on their work as well as that of Doddi & Bagchi [45].

Consider an arbitrary triangle T_i on the discretized membrane: in order to compute the elastic force of its three vertices, we first rotate it to a common plane – e.g. the x, y -plane – using a rotation matrix \mathbf{R}_i from the current orientation of the triangle to its orientation in the common plane. Then, we assume the position of the triangle vertices in a stress-free configuration is known in the common plane, and we compute the displacements \mathbf{v}_k of each of the three vertices of T_i . Using linear shape functions, the deformation gradient tensor and the Cauchy-Green deformation gradient tensor

attached to T_i can be computed:

$$\mathbf{F} = \frac{\partial \mathbf{v}_k}{\partial \mathbf{x}^p}, \quad \mathbf{C} = \mathbf{F}^T \mathbf{F}, \quad (4.17)$$

where $(\mathbf{x}^p, \mathbf{y}^p)$ is the basis of the common plane. Note that \mathbf{F} and \mathbf{C} are two-dimensional tensors and correspond to the tangential components of \mathbf{F}_s and \mathbf{C}_s in Eq. (4.3–4.4). By diagonalizing \mathbf{C} and taking the square root of its eigenvalues (\mathbf{C} is symmetric positive definite), we can access the two principal stretch ratios λ_1, λ_2 attached to T_i . Following Charrier et al. [29], the principle of virtual work yields the expression linking the nodal force and nodal displacement at node j :

$$\mathbf{F}_{\text{elastic},j}^P = A_i \frac{\partial W}{\partial \lambda_1} \frac{\partial \lambda_1}{\partial \mathbf{v}_j} + A_i \frac{\partial W}{\partial \lambda_2} \frac{\partial \lambda_2}{\partial \mathbf{v}_j}, \quad (4.18)$$

where A_i is the area of T_i . Rotating Eq. (4.18) back to the current reference frame of T_i , we get the final expression of the contribution of triangle T_i to the elastic force of node j :

$$\begin{aligned} \mathbf{F}_{\text{elastic},j} &= \mathbf{R}^T \mathbf{F}_{\text{elastic},j}^P \\ &= A_i \mathbf{R}^T \left(\frac{\partial W}{\partial \lambda_1} \frac{\partial \lambda_1}{\partial \mathbf{v}_j} + \frac{\partial W}{\partial \lambda_2} \frac{\partial \lambda_2}{\partial \mathbf{v}_j} \right). \end{aligned} \quad (4.19)$$

This FEM implementation is summarized in algorithm 4.1.

Computation of the bending force using paraboloid fits

The computation of $\mathbf{F}_{\text{bending}}$ relies on the local evaluation of: (i) the mean and Gaussian curvatures κ and κ_g , (ii) the Laplace-Beltrami operator of the mean curvature $\Delta_s \kappa$, and (iii) a relevant control area A .

To evaluate κ and κ_g at node i , we blend the approaches of Farutin et al. [53] and Yazdani & Bagchi [212]. A local reference frame is attached to node i , with the z -direction coinciding with the approximate normal vector \mathbf{n}_i . Then, a paraboloid is fitted to node i and its one-ring neighbors. In our triangulated surface, most nodes have six neighbors³, making the system overdetermined and a least-squares method is used. From this paraboloid fitting, we can derive the local mean and Gaussian curvatures – see equations (12) and (13) in [212] –, as well as a refined approximation of \mathbf{n}_i .

³Exactly twelve nodes have five neighbors, since we discretize a spherical membrane by subdividing each triangle of an icosahedron. Each newly created node is projected back to a sphere, and if necessary projected onto a more complex shape, e.g. a biconcave membrane.

Algorithm 4.1 Pseudocode for the Finite Element Method

```

loop over all triangles  $i$ 
  loop over the three nodes  $j$  of  $T_i$ 
    Compute  $\mathbf{x}_j^P = \mathbf{R}\mathbf{x}_j$ 
    Compute the nodal displacement  $\mathbf{v}_j = \mathbf{x}_j^P - \mathbf{x}_{j,t=0}^P$ 
  end loop
  Compute  $\mathbf{F}$ ,  $\mathbf{C}$  from Eq. (4.17)
  Compute the eigenvalues of  $\mathbf{C}$  and  $\mathbf{F}$ , i.e.  $\lambda_1^2$ ,  $\lambda_2^2$ ,  $\lambda_1$ ,  $\lambda_2$ 
  loop over the three nodes  $j$  of  $T_i$ 
    Compute  $\partial\lambda_1/\partial\mathbf{v}_j$ ,  $\partial\lambda_2/\partial\mathbf{v}_j$ 
    Compute  $\mathbf{F}_{\text{elastic},j}^P$  from Eq. (4.18)
    Rotate  $\mathbf{F}_{\text{elastic},j}^P$  to the current orientation of  $T_i$ 
    Add  $\mathbf{F}_{\text{elastic},j}$  to the total elastic force of node  $j$ 
  end loop
end loop

```

This procedure is iterated using the newest normal vector approximation to define the local frame of reference, until satisfactory convergence of \mathbf{n}_i is reached. Our numerical experimentations show that between three to five iterations usually suffice to obtain a converged normal vector.

The same paraboloid fitting method is used to compute $\Delta_s\kappa$, or $\Delta_s(\kappa - \kappa_0)$ in the case of non-zero reference curvature. This time, a paraboloid is fitted to the *curvatures* of node i and its neighbors, and then differentiated to obtain the desired surface Laplacian.

The last term A is necessary to obtain a bending force as opposed to a bending force per surface area. Let A_i denote the nodal area attached to node i : at any time the sum of all nodal areas need to equal the total area of the capsule, i.e. $\sum_{i=0}^N A_i = A_{tot}$ with N the number of Lagrangian nodes and A_{tot} the total area of the discretized surface of the capsule. The Voronoi area of node i enforces this property only for non-obtuse triangles. As such, we adopt the “mixed-area” of Meyer et al. [72, 137] which treats the special case of obtuse triangles separately: if a triangle j is not obtuse, its contribution to the nodal area of its vertices is the standard Voronoi area; while if j is an obtuse triangle, the nodal area of its obtuse vertex is $A_j/2$ while the nodal area of the remaining two vertices is $A_j/4$, where A_j is the area of triangle j .

4.4 Validation Cases

4.4.1 Elastic and bending forces of an isolated membrane

Elongation of a flat elastic membrane

Our first validation case focuses on the computation of the elastic stress in the membrane. To this end, we stretch an isolated flat membrane devoid of bending resistance in one of its principal direction \mathbf{e}_1 while ensuring the principal stress T_2 in the second principal direction \mathbf{e}_2 remains zero. We then analyze the non-zero principal stress $T_1 = (\partial W / \partial \lambda_1) / \lambda_2$ as a function of the principal stretch $e_1 = (\lambda_1^2 - 1) / 2$. This test is repeated for two membranes: the former obeying the neo-Hookean law and the latter obeying the Skalak law. Note that in order to set the principal stress T_2 equal to zero, we impose λ_2 to a value strictly lower than 1, i.e. the membrane is shrunk in the second principal direction:

$$\lambda_2 = \begin{cases} 1/\sqrt{\lambda_1} & \text{for the neo-Hookean law} \\ \sqrt{(1 + C\lambda_1^2)/(1 + C\lambda_1^4)} & \text{for the Skalak law} \end{cases} \quad (4.20)$$

We compare our results to the exact stress derived by Barthès-Biesel et al. [20] in figure 4.2, with $E_s = C = 1$. The data we generate overlaps perfectly with the analytical stress-strain relations, thus validating the implementation of our Finite-Element solver for the elastic membrane stresses. The source code to reproduce this validation case is available online [81].

Bending force of a curved membrane

In order to validate our bending force, we follow the procedure of Guckenberger et al. [72]: considering a biconcave membrane with zero reference curvature, we compare the mean and Gaussian curvatures, Laplace-Beltrami operator of the mean curvature, and total nodal bending force density to analytical expressions derived using a symbolic calculus software. Since the biconcave capsule has a rotational symmetry around the z -axis and a symmetry with respect to the (x, y) -plane, we plot our results according to the angle θ defined in figure 4.3, with θ varying from 0 to $\pi/2$. This biconcave shape is a good candidate to test the bending force since its two principal curvatures are in general not equal to each other and are varying along the surface of the biconcave shape – even changing sign. The following results are obtained with a biconcave membrane discretized by a triangulation containing 5120 triangular elements.

4.4. Validation Cases

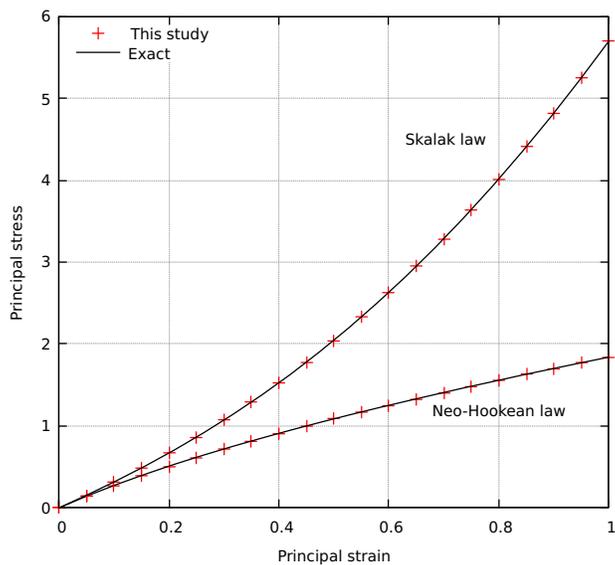


Figure 4.2: Stress-strain response of an isolated flat membrane for the neo-Hookean and Skalak elastic laws. The results from this study are compared to exact expressions derived in [20].

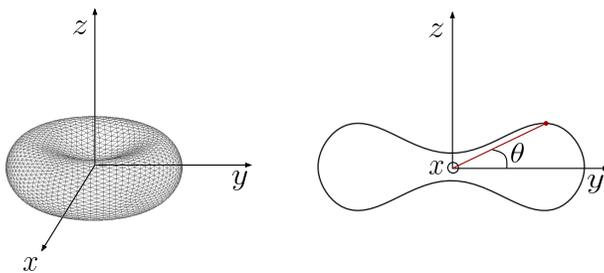


Figure 4.3: Schematic of a biconcave capsule centered at the origin, and definition of the polar angle θ .

We compare our computed mean and Gaussian curvatures to their respective analytical expressions in figure 4.4a: the agreement is very satisfactory for both curvatures. Figure 4.4b shows the Laplace-Beltrami operator of the curvature against its analytical expression: the general trend still matches that of the analytical expression very well, but a few outliers deviate from it by a few percents. The same behavior is observed in figure 4.4c which shows the nodal bending force density. The fact that the behavior of figure 4.4b and figure 4.4c is similar is note surprising, as the nodal bending force density plotted in figure 4.4c directly involves the Laplace-Beltrami operator of the mean curvature shown in figure 4.4b. It is expected to see some small deviations to the theory when taking the Laplace-Beltrami operator of the mean curvature, as we are essentially taking a fourth-order derivative of the geometry of the membrane, and Guckenberger et al. [72] observe a similar noise when performing the same tests (see figures 6c and 8e in [72]). In fact, they show that most other methods perform much worse at computing the Laplace-Beltrami operator of the mean curvature, and hence at computing the total bending force. As such, our implementation of the bending force shows the expected performance. The code to reproduce this test case is available at [82].

4.4.2 Initially spherical capsule in an unbounded shear flow Neo-Hookean elasticity without bending resistance

We now seek validation of the coupling between the membrane solver and the fluid solver. To this end, we consider an initially spherical capsule of radius a in an unbounded shear flow. The elasticity is governed by the neo-Hookean law, and the flow field is initialized to be that of an undisturbed shear flow. As the capsule deforms, we plot the Taylor deformation parameter $D = (a_{max} - a_{min}) / (a_{max} + a_{min})$ as a function of the non-dimensional time $\dot{\gamma}t$, with a_{max} and a_{min} the maximum and minimum radii of the capsule at a given time, and $\dot{\gamma}$ the shear rate. We perform this simulation for various Capillary numbers $Ca = \mu a^2 \dot{\gamma} / E_s$, with E_s the elastic modulus. In this test case, Stokes conditions are intended so we set the Reynolds number to 0.01. At time $t = 0$, the flow field is set to that of a fully developed shear flow: $u_x = \dot{\gamma}y$. The computational box is bi-periodic in the x and z directions, while Dirichlet boundary conditions for the velocity are imposed in the y direction. The length of the computational box is equal to 8 initial radii, the size of the most refined Eulerian cells is set to 1/128 that of the domain length, and the membrane is discretize by 1280 triangles. The non-

4.4. Validation Cases

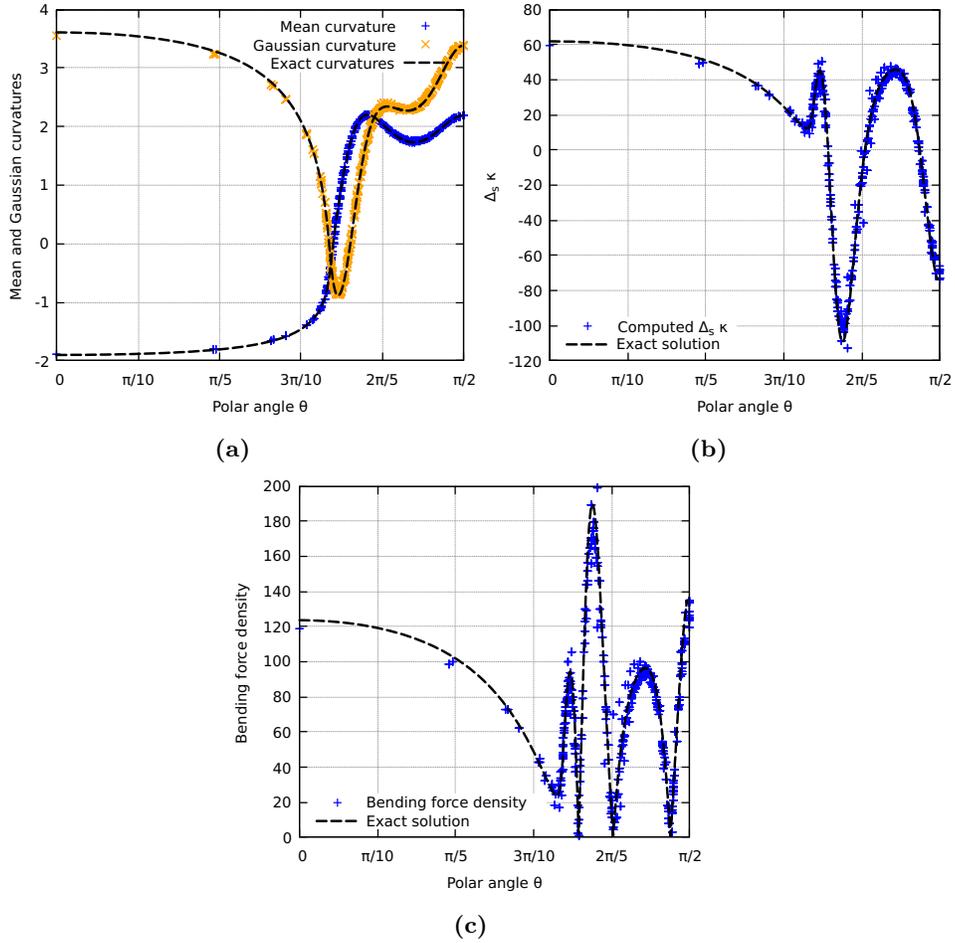


Figure 4.4: Comparison of the computed mean and Gaussian curvatures (top left), Laplace-Beltrami operator of the mean curvature (top right) and nodal bending force density (bottom) to their analytical expressions. All quantities are plotted against the polar angle θ defined in figure 4.3.

4.4. Validation Cases

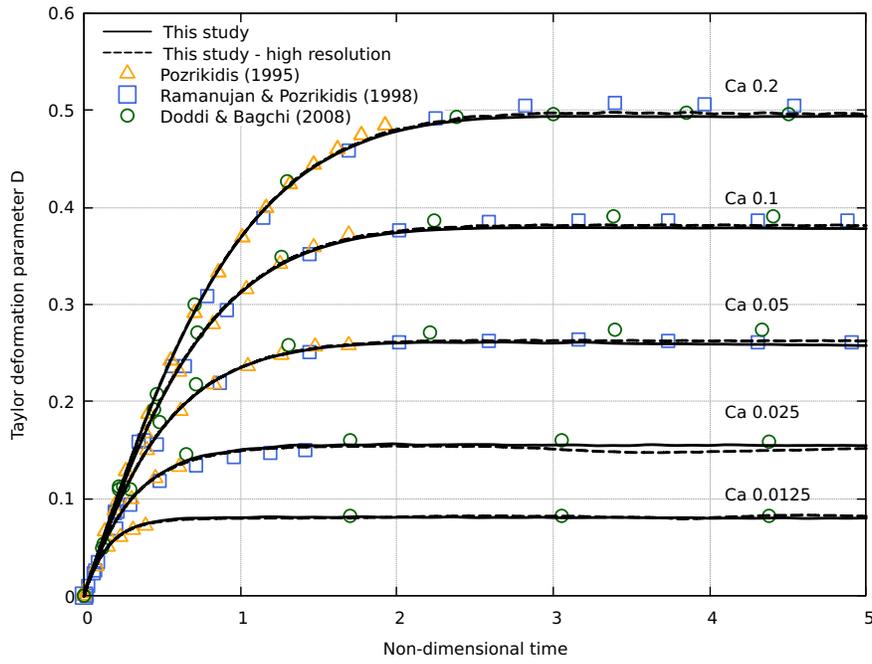


Figure 4.5: Taylor deformation parameter as a function of the non-dimensional time of an initially spherical Neo-Hookean capsule in an unbounded shear flow. The solid line corresponds to 32 Eulerian cells per initial diameter and a Lagrangian discretization using 1280 triangles, while the dashed line corresponds to 64 Eulerian cells per initial diameter and a Lagrangian discretization using 5120 triangles.

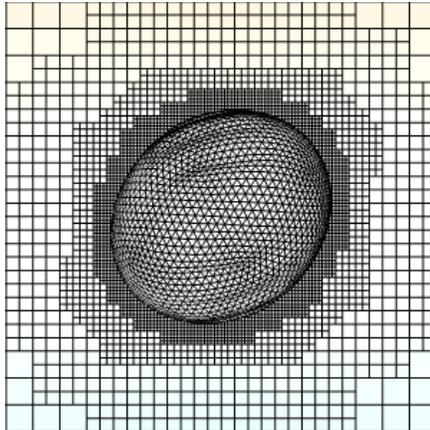


Figure 4.6: A zoomed in snapshot of a buckling membrane at $Ca = 0.0125$ with a Lagrangian discretization comprising 5120 triangles. This behavior is arising due to the absence of bending stresses, and the buckling wavelength is dependant on the Lagrangian discretization. The color field represents the x -component of the velocity.

dimensional time step $\dot{\gamma}\Delta t$ is set to 10^{-3} , except for the Capillary numbers $Ca = 0.025$ and $Ca = 0.0125$ where the time step is decreased to $\dot{\gamma}\Delta t = 10^{-4}$ to stabilize the elastic force computation.

This case has been widely studied in the literature: in figure 4.5 we compare our results to those of [156, 170] who used the BIM, as well as [45] who used the FTM. The agreement is very satisfactory: the steady-state value we obtain for D is well within the range of the reported data, both in the transient regime and once a steady-state is reached. We also show in figure 4.5 the results for a finer triangulation of the membrane with 5120 triangles, and a refined Eulerian mesh with the finest Eulerian cell size corresponding to $1/256$ that of the domain length. The only difference is that the steady state is longer to reach for $Ca = 0.025$ and $Ca = 0.0125$, due to the apparition of buckling instabilities on the membrane as a result of the absence of bending stresses. This buckling instability has been observed both experimentally [109] and numerically [204, 213], although in numerical simulations the wavelength is unphysical and determined by the size of the mesh discretizing the capsule [19]. In our simulations, we do observe the same dependance of the wavelength of the membrane buckles on the Lagrangian mesh element size. An example of this buckling instability is shown in figure 4.6. The code to reproduce this test case is available at [83].

Including bending resistance

We further validate our solver by considering the similar case of a capsule deforming in a shear flow, this time with the addition of a bending force. As in the previous case, an initially spherical, unstressed capsule is placed in a shear flow where the initial velocity field is fully developed. The Capillary number is $Ca = 0.05$, and the non-dimensional bending coefficient $\tilde{E}_b = E_s/(a^2 E_b)$ is chosen equal to 0.0375. The membrane is discretized with 5120 triangles and the same Eulerian resolution as in the previous case is chosen. Due to the stiffness of the bending force, we set the time step to $\Delta t = 10^{-4}$. The Taylor deformation parameter is compared to that of various studies in the literature in figure 4.7. The capsule deforms under the action of the flow field and the Taylor deformation parameter quickly attains a steady state of about $D = 0.15$. We remark that the data reported in the literature is scattered by about 20% which underlines the challenges to simulate Helfrich’s bending force, as was previously noted by [72]. We also note that our results are situated well within the range of the reported data: we are close to the results of Zhu & Brandt [221] and Le et al. [120], and our curve is located in the middle of the reported range that we borrowed from [71, 72]. Given such a wide range of reported literature data, it is difficult to conduct a rigorous quantitative analysis. Nevertheless, we conclude from figure 4.7 that our bending force shows a similar behavior as that of other studies, a claim also supported by the validation case in Section 4.4.4. The code to reproduce this test case is available at [84]

4.4.3 Initially spherical capsule flowing through a constricted channel

To validate our implementation for an elastic capsule in the presence of complex boundaries, we consider the case of a capsule flowing through a constricted square channel proposed by Park & Dimitrakopoulos [145]. The elasticity of the membrane is governed by the Skalak law with the area dilatation modulus C set to 1, the capsule is initially pre-inflated such that its circumference is increased by 5%, and the flow is driven by an imposed uniform velocity field at the inlet and outlet boundaries. We follow [145] and choose the Capillary number to be 0.1, and since Stokes conditions are intended we set $Re = 0.01$.

The results are presented in figure 4.8 and figure 4.9. The qualitative agreement in figure 4.8 is very satisfactory as the capsule shape is visually identical to that of [145]. We draw the reader’s attention to the adaptive

4.4. Validation Cases

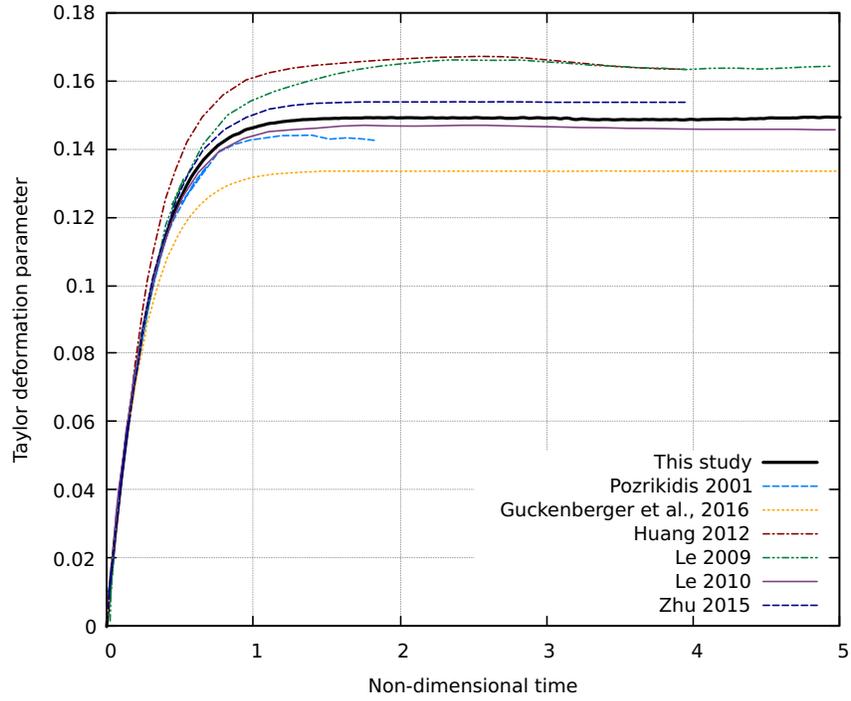


Figure 4.7: Taylor deformation parameter of an isolated capsule undergoing elastic and bending stresses in a shear flow. The capillary number is $Ca = 0.05$ and the non-dimensional bending coefficient is $\tilde{E}_b = 0.0375$.

Eulerian mesh on the right-hand side of figure 4.8: the cells size is imposed to be minimal at the solid boundaries and around the capsule, while everywhere else the adaptivity criterion is governed by the velocity gradients. As a result, the grid cells away from the membrane and from the walls quickly coarsen to up to three levels lower, except in the vicinity of the corners where stronger velocity gradients occur. Figure 4.9 shows the non-dimensional lengths of the capsule in the x -, y - and z -directions with respect to the non-dimensional position of its center x_c/H_c , with x_c and H_c the x -position of the center of the capsule and the half-height of the constriction, respectively. As found by Park & Dimitrakopoulos, the final shape of the capsule is not exactly spherical as it remains shrunk in the x -direction downstream of the constriction. Despite some small deviations during the extreme deformation of the capsule, around $x_c/H_c = -1$, the overall quantitative agreement of the transient shape of the capsule is also satisfactory, especially considering that other authors have reproduced this case with similar or larger deviations from the results reported by Park & Dimitrakopoulos [15, 93]. The code to reproduce this test case is available at [85].

4.4.4 Red blood cell in an unbounded shear flow

The next test case aims at validating the membrane solver when a viscosity ratio $\lambda_\mu = \mu_i/\mu_e$ is different than 1. To this end, we consider an RBC in an unbounded shear flow, with $\lambda_\mu = 5$. The membrane forces include the Skalak elastic law and the Helfrich's bending force. The Capillary number is $Ca = 0.1$, the area dilatation modulus C is chosen equal to 50 and the non-dimensional bending coefficient is $\tilde{E}_b = 0.01$. The reference curvature is $c_0 a = -2.09$ [159, 214], where $a = (3V/4\pi)^{1/3}$ is the radius of the sphere of equal volume as that of the RBC. The initial shape of the RBC is biconcave and is described by the following equations, for an RBC which largest radius is orthogonal to the y direction [160]:

$$\begin{cases} x = ac \cos \phi \cos \psi \\ y = \frac{ac}{2} \sin \phi (\alpha_1 + \alpha_2 \cos^2 \phi + \alpha_3 \cos^4 \phi) \\ z = ac \cos \phi \sin \psi, \end{cases} \quad (4.21)$$

with $\phi \in [0, 2\pi]$, $\psi \in [-\frac{\pi}{2}, \frac{\pi}{2})$, $\alpha_1 = 0.207$, $\alpha_2 = 2.003$ and $\alpha_3 = -1.123$. Since we consider a viscosity ratio, we also define the initial indicator function I as the volume fraction of inner fluid:

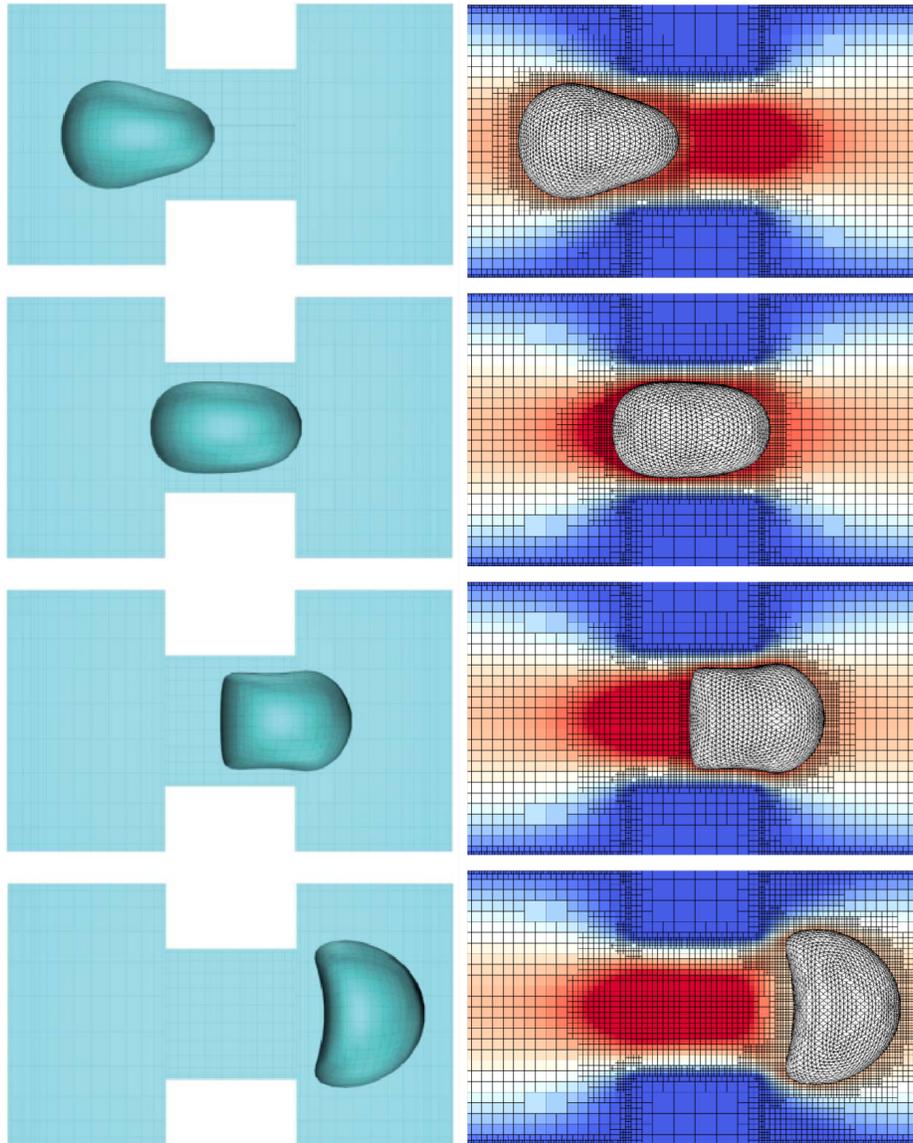


Figure 4.8: Snapshots of the capsule as it flows through the constriction. Left: Park & Dimitrakopoulos [145]; Right: this study.

4.4. Validation Cases

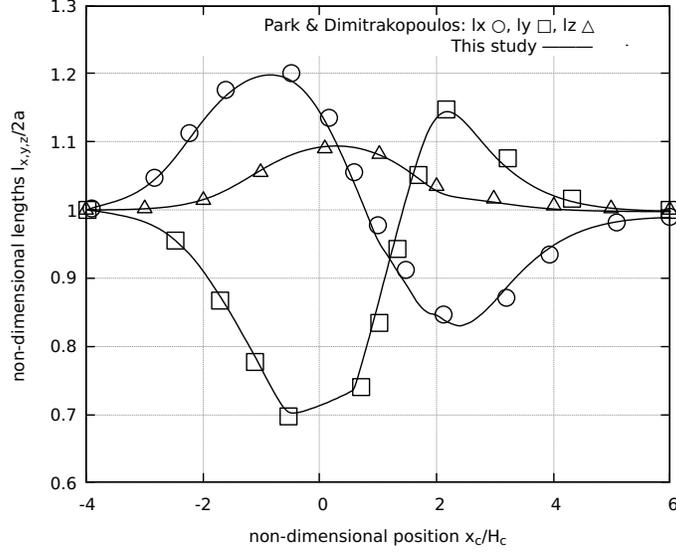


Figure 4.9: Non-dimensional lengths of the capsules in the three directions x , y and z with respect to the non-dimensional x -position of the center of the capsule. Results are compared to [145].

$$I(\mathbf{x}) = \begin{cases} 1 & \text{if } \Phi(\mathbf{x}) < 0 \\ 0 & \text{if } \Phi(\mathbf{x}) > 0 \\ \text{between 0 and 1} & \text{otherwise,} \end{cases} \quad (4.22)$$

where Φ is the level-set alternative formulation of Eq. (4.21):

$$\Phi(x, y, z) = \frac{x^2 + z^2}{(ac)^2} + \frac{4y^2}{(ac)^2} \left(\alpha_1 + \alpha_2 \frac{x^2 + z^2}{(ac)^2} + \alpha_3 \left(\frac{x^2 + z^2}{(ac)^2} \right)^2 \right). \quad (4.23)$$

The initial fluid velocity is set to that of an unbounded shear flow of shear rate $\dot{\gamma}$ with the velocity gradient in the direction of the greater axis of the RBC. The dimensionless time step we use is $\dot{\gamma}\Delta t = 10^{-4}$ and is determined from trial and error.

The qualitative results are presented in figure 4.10, where we include snapshots of the same case from Yazdani & Bagchi [214]. We observe that the RBC is undergoing a tumbling motion, a behavior of RBCs that is not seen without viscosity ratio in this range of Capillary numbers [160, 214]. Moreover, the deformation of the RBC matches qualitatively well that of

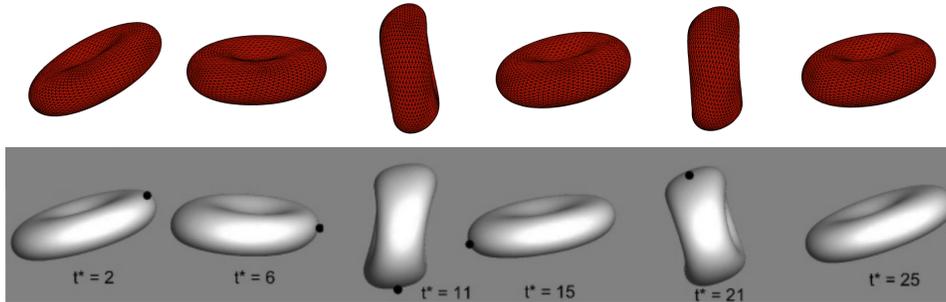


Figure 4.10: Tumbling motion of an isolated RBC in a shear flow. Top: this study; bottom: Yazdani & Bagchi [214].

[214]. However our tumbling period seems slightly shorter than that of [214]: we attribute this small discrepancy to the fact that we may have set different values for the area dilatation modulus C , as [214] only provides a range of values: $C \in [50, 400]$. Nevertheless, the results in figure 4.10 show that in our implementation, the combination of elastic forces, bending forces and viscosity ratio matches well the qualitative results observed in the literature, and that the overall agreement is satisfactory. The code to reproduce this case is available at [86].

4.4.5 Capsules interception in a shear flow

Our last validation case focuses on the interactions of two capsules. Two initially spherical, pre-inflated neo-Hookean capsules are placed in an unbounded shear flow with their initial positions offset in the horizontal and vertical directions as shown in figure 4.11. Since the capsules are offset in the vertical direction, they gain horizontal velocities of opposite signs and their trajectories eventually intercept. This configuration is a good validation candidate since we can compare our results to those obtained by Lac et al. using the boundary integral method [115]. We consider a computational box of size $16a$ where a is the initial radius of the capsules. The finest Eulerian resolution corresponds to the domain being discretized by 512 cells in each direction, and the two membranes are discretized with 5120 triangles. A non-dimensional time step of $\dot{\gamma}\Delta t = 2.5 \cdot 10^{-4}$ is chosen. The Reynolds number is set to 0.01.

Figure 4.12 shows the qualitative comparison of the shape of the two capsules at several stages of the interception. In our simulations, the color field corresponds to the vertical component of the velocity. At each stage,

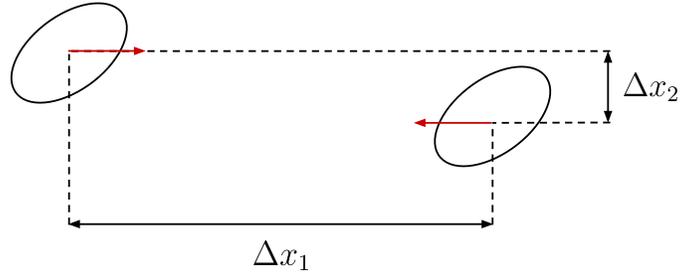


Figure 4.11: Schematic of the two capsules in in the shear flow, prior to the interception. The horizontal and vertical gaps Δx_1 and Δx_2 are defined, and the red arrows represent the velocities of the centers of the capsules.

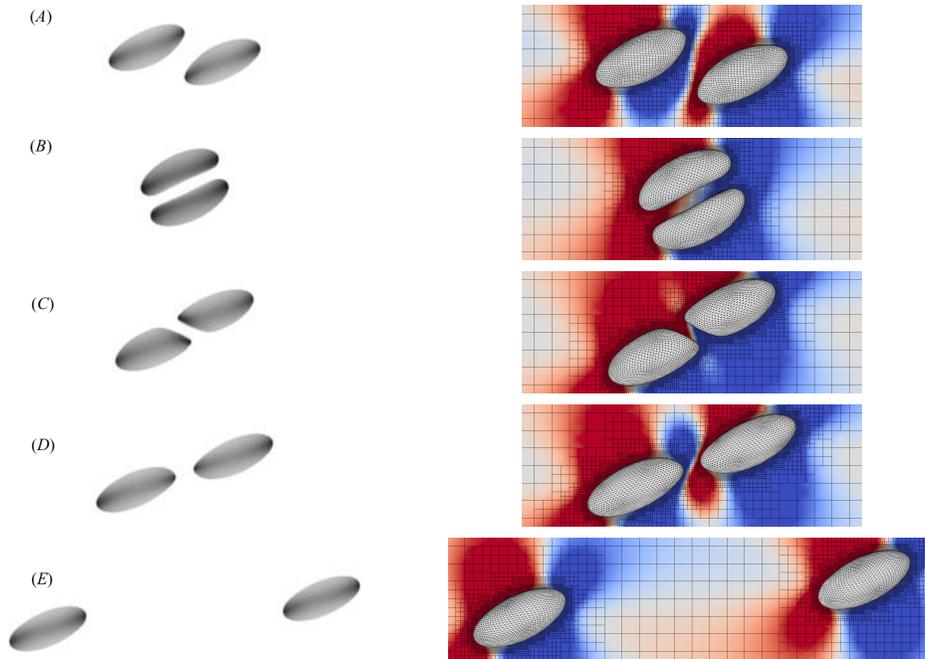


Figure 4.12: Snapshots of the interception of two neo-Hookean capsules in a shear flow. Left: Boundary Integral results of Lac et al. [115]. Right: This study. The color field corresponds to the vertical component of the velocity (rescaled for each snapshot).

4.4. Validation Cases

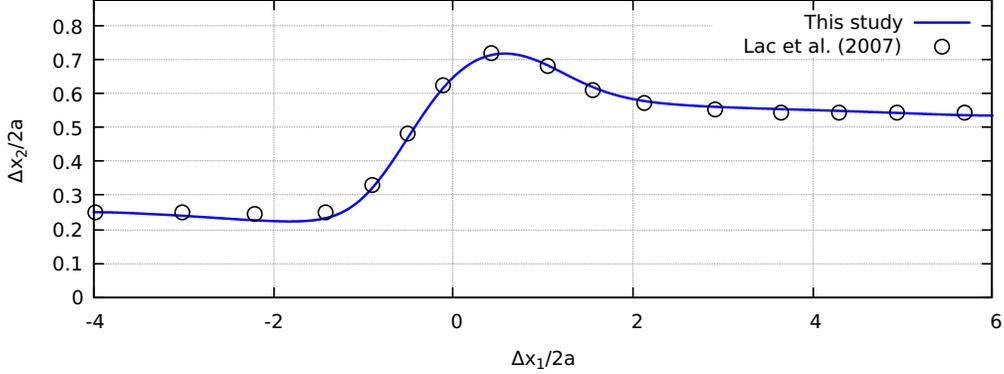


Figure 4.13: Non-dimensional vertical gap $\Delta x_2/2a$ against the non-dimension horizontal gap $\Delta x_1/2a$ between the centers of the two capsules. The results from this study are compared to Lac et al. [115].

there is visually no difference in the shape of the capsules. If we track the center of each capsule throughout the simulation, we can compute their difference Δx_2 in the vertical direction and their difference in the horizontal direction Δx_1 . Normalizing by the initial diameter $2a$ of the capsules, we plot in figure 4.13 the vertical gap between the two capsules as their intercept, and we compare our results to those of Lac et al. [115]. The agreement is very satisfactory: the transient regime is very well captured, both methods showing a maximum non-dimensional vertical gap of about 0.72; and the steady-state reached is about 0.54. Small discrepancies can be observed around $\Delta x_1/2a = -2$ where our vertical gap is slightly lower than that of Lac et al.; and for $\Delta x_1/2a$ between 4 and 6 where our slope is still slightly negative while that of Lac et al. is essentially zero. Those discrepancies are minor and could be explained by our choice of Reynolds number $Re = 10^{-2}$ while the boundary integral method operates in true Stokes conditions. Regarding the adaptive mesh, as stated above we perform this simulation using an equivalent fluid resolution of 64 cells per initial diameter, in a cubic box 8 diameters in length. Our simulation requires about $4.5 \cdot 10^5$ fluid cells, while using a constant mesh size would require about $1.3 \cdot 10^8$ cells. For this specific case, using an adaptive grid therefore reduces the number of fluid cells by a factor of about 300.

A similar configuration was later examined by Doddi & Bagchi [44] in the presence of inertia. In a cubic box of size $H = 4a\pi$, periodic in the x and z directions, for a Capillary number of 0.05 and an initial vertical distance $\Delta x_2/2a = 0.2$, they observed that the capsules don't intercept

4.4. Validation Cases

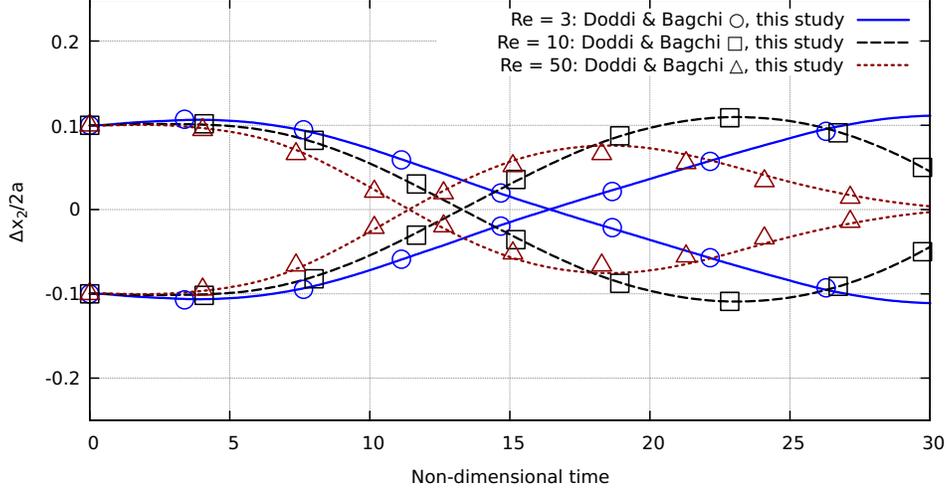


Figure 4.14: Non-dimensional vertical gap $\Delta x_2/2a$ against the non-dimensional time $\dot{\gamma}t$ for Reynolds number of 3, 10 and 50. The results of this study are compared to Doddi & Bagchi [44].

when the Reynolds number $Re = \rho\dot{\gamma}a^2/\mu$ is greater or equal to 3. Instead, the vertical component of the center of the capsules changes sign and the direction of movement is reversed. We reproduce these results from Doddi & Bagchi (figure 8c in [44]): each capsule is discretized with 5120 triangles, the Eulerian equivalent resolution is 40 points per initial diameter, and the non-dimensional time step is $\Delta t = 10^{-3}$. Figure 4.14 shows the vertical position of the centers of the two capsules with respect to time, for Reynolds numbers of 3, 10 and 50, and the generated data is compared to [44]. Our results superimpose very well with [44], in particular for $Re = 3$ and $Re = 10$. For $Re = 50$, the agreement is still very satisfactory although we notice that we predict a slightly larger overshoot around $t = 18$ compared to the results of [44].

Overall, the quantitative agreement in figure 4.13 and figure 4.14 is very satisfactory and it validates our adaptive front-tracking solver for several capsules, both in Stokes conditions and in the presence of inertia. The code for these two cases is available at [87, 88].

4.5 Results

4.5.1 Capsule flowing through a narrow constriction

In their original study, Park & Dimitrakopoulos [145] investigated the case presented in Section 4.4.3 for relatively wide constriction sizes – the half size of the constriction H_c was greater than or equal to the capsule radius a . In this subsection we instead decrease the constriction size to $H_c = a/2$ in order to demonstrate the robustness of our solver in cases of extreme deformations, including close to domain boundaries. As in Section 4.4.3, the capsule is initially circular and pre-inflated such that each distance on the capsule surface is increased by 5%. It obeys the Skalak elastic law with $C = 1$ and $Ca = 0.1$. Since the capsule undergoes extreme deformations, we also consider a bending force in order to suppress unphysically sharp corners: the non-dimensional bending coefficient is set to $\bar{E}_b = 10^{-3}$. In order to resolve well the capsule deformation, we increase the resolution of the triangulation, which now comprises 20480 triangles. We also perform this simulation for two finest Eulerian grid resolutions of 50 and 100 grid cells per initial diameter respectively. The flow is driven by an imposed uniform velocity field at the inlet and outlet boundaries, and the Reynolds number is set to 0.01 to model Stokes flow conditions. For this case the non-dimensional time step is set to $a\Delta t/U = 2.5 \cdot 10^{-5}$, with a the initial radius of the capsule and U the characteristic velocity of the fluid. It appears that this strict restriction on the time step is due to the stiff bending force combined to the very fine discretization we choose.

Qualitative results are shown in figure 4.15. As expected, the deformation of the capsule is considerably greater when the constriction size is halved: the capsule becomes almost flat as it reaches the center of the constriction. Figure 4.15 also confirms that the Eulerian mesh is refined only in the region of interest, as the grid size quickly increases with the distance from the constriction and from the capsule. We show quantitative results in figure 4.16, where the non-dimensional x -, y - and z -lengths of the capsule are plotted against the non-dimensional x -position of the center of the capsule. Unsurprisingly, the capsule vertical length l_y decreases by over a factor of two when the capsule reaches the center of the constriction, before sharply increasing again as the front of the capsule expands while leaving the constriction. The sharp point observed for l_y at $x_c/H_c = 0$ is simply due to the non-locality of the variable l_y : for $x_c/H_c \leq 0$ the maximum height of the capsule is located at its rear, while for $x_c/H_c \geq 0$ it is located at its front. We also note that the maximum decrease of the capsule height $l_{y,\min}^N$ in the

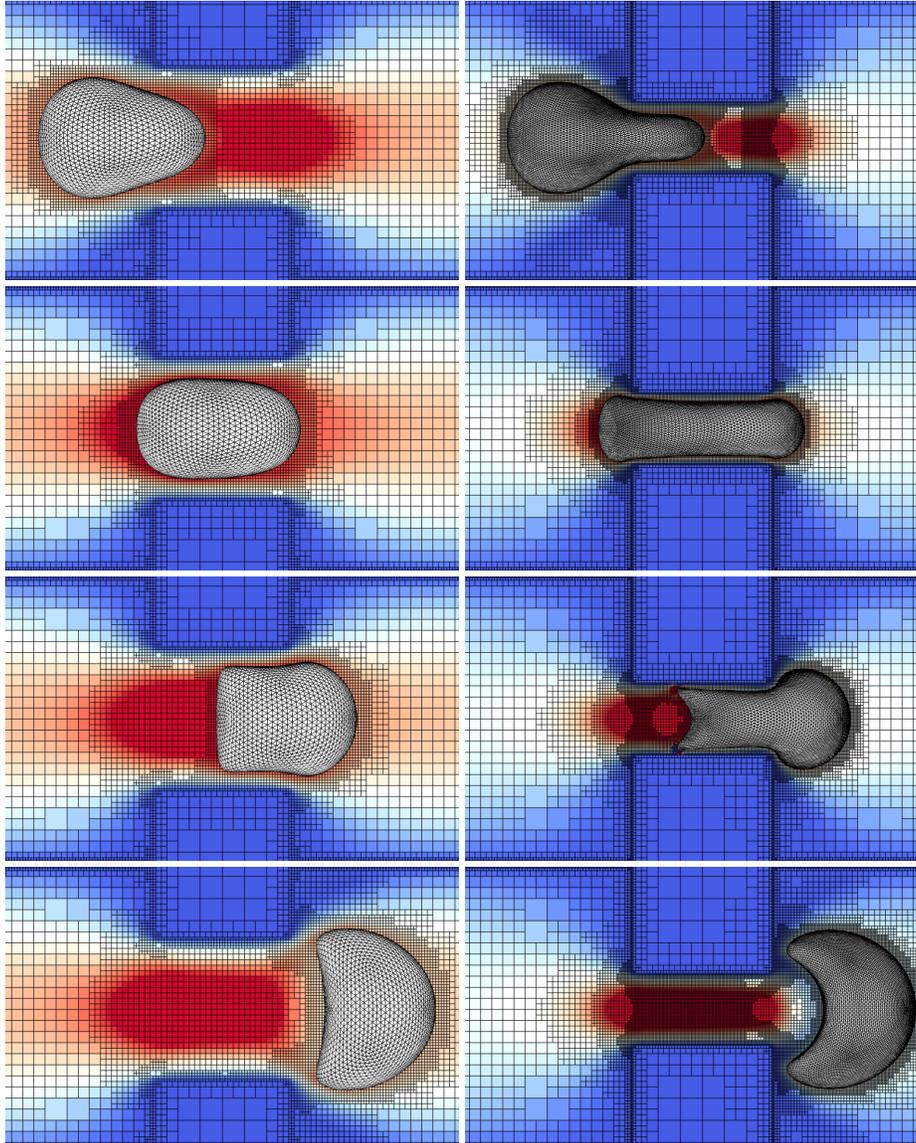


Figure 4.15: Qualitative comparison of a capsule flowing through a constricted channel: when the constriction size is equal to the initial diameter of the capsule (left); and when the constriction size is equal to half of the initial diameter of the capsule, with a finest Eulerian grid resolution of 100 cells per initial diameter (right). Color field: x -component of the velocity field.

4.5. Results

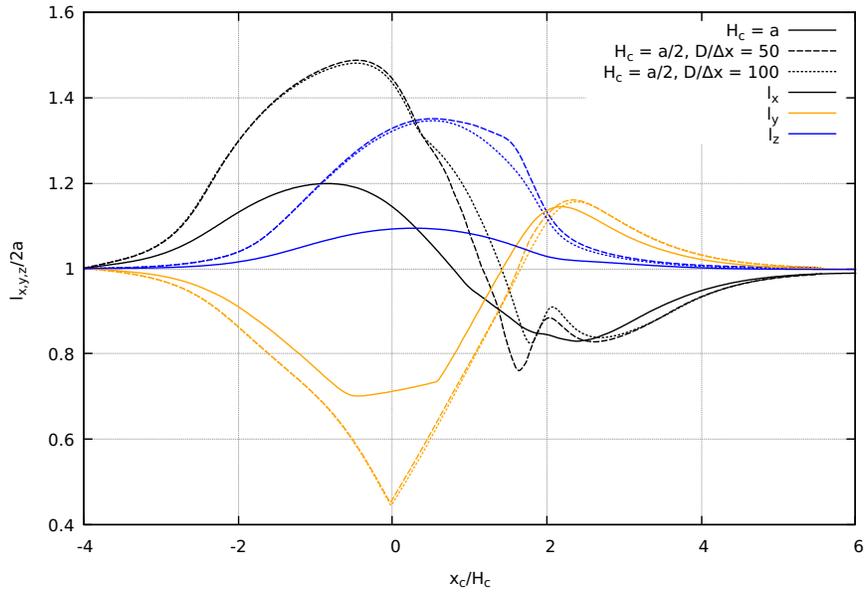


Figure 4.16: Non-dimensional x -, y - and z -lengths of the capsule $l_x/2a$, $l_y/2a$ and $l_z/2a$ with respect to the non-dimensional x -position of the center of the capsule x_c/H_c as it flows through a constriction size equal to: (i) the initial diameter of the capsule (solid line); and (ii) half of the initial diameter of the capsule (dashed line). The dotted line corresponds to a finest Eulerian resolution of 100 Eulerian grid cells per initial diameter (shown for $x_c/H_c = a/2$ only).

case of a narrow constriction is much more pronounced than its counterpart $l_{y,\min}^W$ in the wider constriction. However, the minimum capsule height is not halved when the constriction size is halved, i.e. $l_{y,\min}^N > l_{y,\min}^W/2$. On the other hand, the maximum x -length of the capsule more than doubles in the case of a narrow constriction when compared to the wider constriction size, and the maximum z -length more than triples. Therefore, the capsule preference to elongate in the streamwise direction rather than the spanwise direction is reduced when the constriction is narrower. Finally, the capsule reaches a steady shape after the constriction for $x_c > 6H_c$: the constriction size does not appear to affect this steady shape, which is a slightly deformed sphere compressed in the streamwise direction.

In figure 4.16 the x -, y - and z -lengths of the capsule are shown for two maximum Eulerian resolutions: $D/\Delta x = 50$ and $D/\Delta x = 100$, with D the initial diameter of the capsule and Δx the smallest Eulerian cell size. Relatively minor differences are observed between the two mesh resolutions, indicating that a maximum Eulerian resolution of $D/\Delta x = 50$ is sufficient to capture the underlying physics of this configuration. In terms of performance, conducting the previous convergence study up to $D/\Delta x = 100$ with a constant grid size would have required about $4.2 \cdot 10^7$ fluid cells, while our simulation used less than $4.6 \cdot 10^6$ fluid cells, thus allowing a tenfold reduction in the number of fluid grid cells, and likely reducing the computational resources by a factor of 7 to 10 when accounting for the computational overload due to the complex tree structure of the grid and the adaptivity algorithm.

4.5.2 Capsule-laden flows in large complex channel geometries

It is clear from the previous simulation results presented in this paper that the adaptive mesh refinement is useful to lower the number of cells inside the fluid domain, and thus the amount of computations per time step. However, it can also be desirable to reduce the number of cells *outside* the fluid domain, as they can also be associated with a large computational and memory cost. This is because in cases of complex geometries the computational domain of Cartesian grid methods is by design a bounding box that surrounds the fluid domain. As such, if the volume fraction of the fluid domain in this bounding box is low and if a constant grid size is used, most of the computational cells are located inside the solid walls and a significant amount of memory and computational resources are allocated for these “solid” cells where no physics happens. This is especially true in cases of large, three-dimensional

4.5. Results

channel geometries. For instance, let us consider a helical pipe of radius R_{pipe} connecting an upper and a lower arm of length L_{arm} , where a capsule of radius $R_c = R_{\text{pipe}}/4$ is placed at the top of the geometry. If we assume this geometry is embedded in a bounding box of depth $2R_{\text{helix}} = 10R_{\text{pipe}}$, height $H_{\text{helix}} = 12R_{\text{helix}}$ and length $2L_{\text{arm}} = H_{\text{helix}}$, as shown in figure 4.17a, ensuring 16 grid cells per initial capsule diameter using a uniform Eulerian grid would require over 1.2 billion grid cells, rendering the computation extremely expensive. In contrast, using Basilisk’s adaptive mesh as shown in figure 4.17a allows to reduce the number of computational cells by a factor of about 200, down to less than 6 million grid cells. If only the helix itself is of interest and not the connecting arms, using a uniform Eulerian grid would require about 200 million grid cells, and using our adaptive solver would still reduce the number of grid cells by over a factor of 30.

As a demonstration of the capability of the present solver to handle such large and complex geometries, simulations are carried out in the helical geometry described above and shown in figure 4.17. A neo-Hookean capsule of radius R_c with $Ca = 0.1$ is placed on the upper straight pipe centerline at a distance of $1.5R_{\text{helix}}$ prior to entering the helix. We propose to study the inertial migration of this capsule for three Reynolds numbers: $Re = 10, 50$ and 100 . We set the non-dimensional time step $R_{\text{helix}}\Delta t/U$ to 10^{-3} , with U the characteristic velocity of the fluid; and we choose a finest Eulerian grid resolution corresponding to 16 grid cells per initial diameter. Each case ran for about three days on 96 processors, with around $6 \cdot 10^4$ cells per processor. To analyze the trajectories, we define the distance r^* of the capsule centroid to the helix centerline normalized by the pipe radius R_{pipe} , as well as the number of helical periods $N^* = (H_{\text{helix}} - z)/H_{1p}$, with H_{1p} the height of one vertical period of the helix, i.e. its pitch distance. A slice of the flow field in the helix for $Re = 100$ is shown in figure 4.17c, and a movie of this simulation as well as the code to reproduce it are available at [89]. The trajectories of the capsule are shown in the three-dimensional space in figure 4.17b, in one dimension by showing r^* as a function of N^* in figure 4.18a, and in a cross-section of the pipe orthogonal to the helix centerline in figure 4.18b. Only the path corresponding to the capsule located inside the helix is shown in these figures. Immediately after release, the capsule moves away from the centerline for all Reynolds numbers. The initial overshoot of r^* increases with the Reynolds number, from $r_{\text{max}}^* \approx 0.3$ at $Re = 10$ to $r_{\text{max}}^* \approx 0.6$ at $Re = 100$. After four helical revolutions, the capsule exits the helix with a steady position of $r_{\infty}^* \approx 0.38$ and $r_{\infty}^* \approx 0.45$ for $Re = 50$ and $Re = 100$ respectively. In the case of $Re = 10$, however, a steady state is not yet reached when the capsule exits the helix, but we can extrapolate

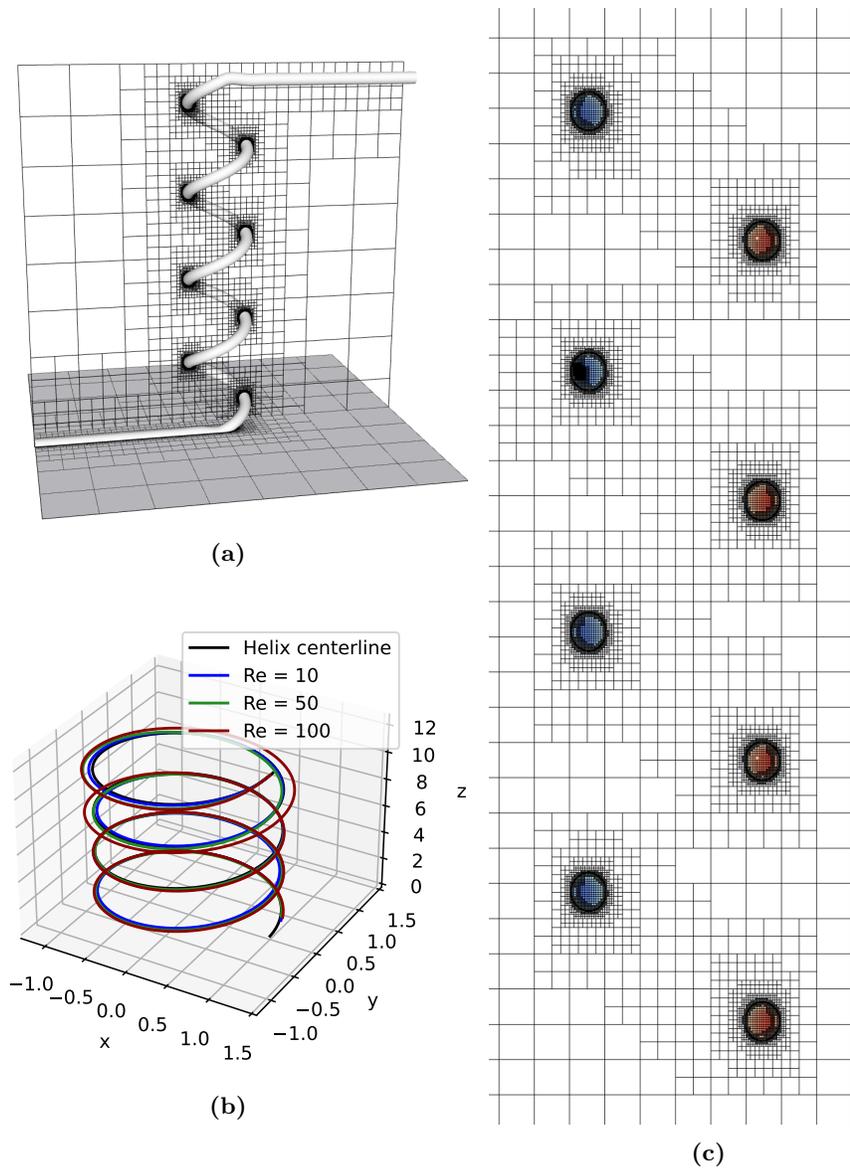
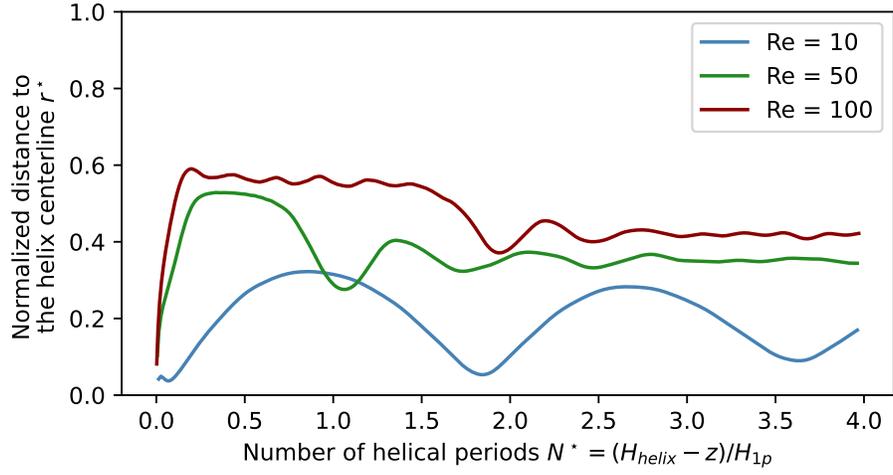
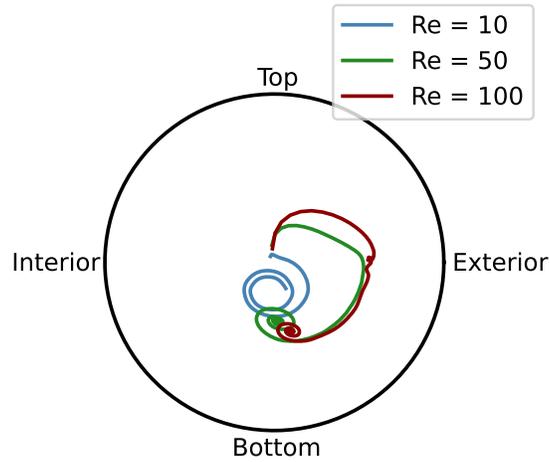


Figure 4.17: Adaptive mesh around a helical geometry and its connecting pipes: (a) full computational box around the whole geometry; (b) three-dimensional trajectory of a capsule at $Re = 10, 50, 100$ (for visual clarity the helix is shrunk in the vertical direction); (c) adaptive mesh and velocity field in the vertical plane. In (c), the color field corresponds to the x component of the velocity, where blue is into the page and orange is out of the page. The capsule is about to cross the vertical plane for the third time, hence the additional small cells inside the third circular cross-section from the top.

4.5. Results



(a)



(b)

Figure 4.18: Trajectory of a solitary capsule in the helix at $Re = 10, 50, 100$: (a) normalized distance from the helix centerline; (b) radial migration in a plane orthogonal to the helix centerline. N^* corresponds to the number of revolutions around the vertical axis of the helix.

the capsule trajectory to find that $r_\infty^* \approx 0.18$ for this Reynolds number. For all Reynolds numbers, the steady position is located in the lower half of the cross-section, at an angle θ from the horizontal line of about $-\pi/2$, although θ increases slightly with the Reynolds number. Interestingly, we note that the capsule transient path is longer for $Re = 100$ than that for $Re = 50$: as can be seen in figure 4.18, for $Re = 100$ the capsule seems to reach an unstable equilibrium at $r^* \approx 0.58$ and $\theta \approx 0$ for as long as 1.5 helical periods, before continuing its spiralling motion towards a stable steady-state location. Further investigation would be necessary to characterize this behavior and determine if, for instance, this unstable equilibrium corresponds to the center of a vortex, but this is not the focus of the present paper. The purpose of this simulation is to show that the present solver is able to simulate large three-dimensional channel geometries, and has the potential to simulate full microfluidic devices.

4.6 Conclusion and perspectives

We have presented an adaptive front-tracking solver to simulate deformable capsules in viscous flows at zero and finite inertia. The membrane mechanics is governed by an elastic and a bending resistance, and a non-unity viscosity ratio is allowed. Moreover, the present solver is compatible with complex STL-defined geometries, thus providing all the ingredients needed to simulate realistic flow configurations of biological cells, including red blood cells, both in-vivo and in-vitro. Numerous validation cases are presented against data available in the literature: we compare our results mainly to the highly accurate boundary integral method in Stokes conditions, and to other front-tracking methods at non-zero Reynolds numbers. Very good qualitative and quantitative agreement is shown in all cases. We then demonstrate the robustness of the present solver in more challenging configurations, as well as its potential to tackle very large, three-dimensional channel geometries relevant to inertial microfluidic applications. Moreover, the present implementation is open-source as part of the Basilisk platform: the documented source code is freely available, as well as the source files to reproduce all the simulations presented in this paper [80].

Although the present adaptive front-tracking solver can simulate all the range of Reynolds numbers, the non-inertial limit is challenging because of the computation of the viscous term. The simulations we show in this paper at $Re = 0.01$ are several times slower to complete than their counterpart at, e.g., $Re = O(1)$ or $Re = O(10)$. As a result, if only the non-inertial

regime is sought, boundary integral solvers likely remain the most efficient method by far. Another challenge is the stiffness of the bending stresses: since the Helfrich’s bending formulation involves such high-order derivatives of the membrane geometry, and since the time integration of the membrane problem is explicit, the time step is controlled by the time scale associated with the bending force whenever bending effects are included. This is a known challenge of computing bending stresses on elastic membranes (see, e.g., p. 40 of [19]). As a result, we have to decrease our time step by one order of magnitude (sometimes even more) whenever the bending stresses are included. Unfortunately, to our knowledge the stability condition associated with the bending force is unknown and it is therefore not possible to stabilize simulations by employing an adaptive time-step strategy, as is already the case in Basilisk with the CFL condition and with the celerity of capillary waves for surface tension stresses. One could investigate the implicit or “approximately implicit” treatment of the immersed boundary method as done by Roma, Peskin & Berger [174].

On the implementation side, the fluid solver from Basilisk is compatible with shared and distributed memory systems – i.e. using OpenMP and MPI libraries –, allowing to run simulations on large supercomputers. Naturally, we have enabled our front-tracking solver to be compatible with MPI as well. However, ensuring a good scaling with the number of capsules is not trivial as the domain decomposition of the Eulerian adaptive mesh is governed by a Z-ordering algorithm. As a result, when the Eulerian mesh is adaptive, the stencils attached to the Lagrangian nodes of a given capsule are most likely containing Eulerian cells handled by several distinct processors. In other words, a single capsule has to exist on many different processors in order to communicate with the background Eulerian mesh. Interpolating velocities from the fluid cells to the Lagrangian nodes, and spreading the Lagrangian forces to the fluid cells thus requires expensive inter-processor communications. Investigating efficient strategies to simulate a large number of capsules with an adaptive mesh is left for future works. That being said, in all the simulations shown in this study with one or two capsules considered, at most 5% of the total computing time is spent in the front-tracking solver. Consequently, unless simulating a large number of capsules – e.g., $O(100)$ –, the bottleneck is still the Navier-Stokes solver. Moreover when a large number of capsules is considered, one could argue that a uniform Eulerian mesh can be more efficient than its adaptive counterpart because a large number of capsules would likely result in a high volume fraction of capsules. An efficient parallel implementation of the present front-tracking solver restricted to uniform Eulerian grids is straightforward and could be

implemented in future studies if dense volume fractions are considered. Another extension to the present solver could be to allow the triangulation of the membrane to be adaptive as well. Such adaptive triangulations have been considered to simulate fluid-fluid interfaces [193], but in the case of elastic membranes special care needs to be given to coarsen or refine the shape functions of a triangle. However only simulations featuring extreme membrane deformation would benefit from an adaptive membrane triangulation, as the front-tracking solver is only taking a few percents of the total computing time. For the applications we seek where reasonable membrane deformations are expected, the gain of an adaptive membrane triangulation is likely close to zero.

Another possible improvement to the present solver would be to allow the support of the regularized Dirac-delta functions to include grid cells of different sizes. Indeed, the current method imposes a constant grid size in the vicinity of the membrane in order to apply the IBM in a straightforward manner, in a similar fashion to our Distributed Lagrange Multiplier/Fictitious Domain method implemented in Basilisk to simulate flows laden with rigid particles [182], but this can result in imposing a finer Eulerian grid resolution around the membrane than what is necessary to properly resolve some parts of the membrane. This scenario typically happens when the capsule is close or will come close to a sharp boundary, such as in the case of the narrow constriction in Section 4.5.1. In figure 4.15, for instance, as the capsule enters the narrow constriction, the Eulerian grid resolution around its tail is much finer than necessary due to the very fine grid resolution needed at the front of the capsule, which is located in a flow with strong gradients and close to sharp boundaries. A conceptually simple way to allow the size of the Eulerian grid cells to change along the membrane would be: (i) to propagate the forcing term or the averaging operator to smaller grid cells located inside the stencil of interest; and (ii) to increase the stencil size if a large grid cell is encountered, such that any stencil size is always four times larger in each direction than its largest grid cell. This adaptive extension of the immersed boundary method is presented in Appendix C and shows promising results.

As demonstrated in Section 4.5.2, the current state of our adaptive solver allows resolved inertial simulations of capsule-laden flows in large three-dimensional geometries for a fraction of the computational cost of that of front-tracking solvers implemented on uniform Cartesian grids. As such, our solver has the potential to provide valuable insight to help develop inertial migration microfluidic devices. The present solver may even allow the simulation of full microfluidic geometries consisting in several stacked layers of

4.6. Conclusion and perspectives

microfluidic channels, such as the spiralling geometries in the experimental work of Fang et al. [52]. This has the potential to provide valuable qualitative and quantitative information about the flow field, capsule dynamics and sorting efficiency of a given realistic microfluidic geometry, thus reducing the number of manufacturing iterations during the design process of such devices. Our medium-term objective is to tackle this type of virtual design problem, considering sub-domains of the full geometry as a first step, while concomitantly investigating the possible improvements stated above.

Chapter 5

Motion and deformation of capsules through a corner in the inertial and non-inertial regimes

5.1 Introduction

Membrane-enclosed fluid objects, or capsules, are everywhere in natural and industrial processes, from red blood cells (RBCs), circulating tumor cells (CTCs) or flowing eggs in biology to encapsulated substances in the pharmaceutical, cosmetic and food industries [19]. The study of microcapsules in particular is of primary importance in a variety of biological applications, such as sorting and enriching solutions of biological microcapsules, e.g. to segregate RBCs or CTCs, as well as efficiently manufacturing capsules enclosing an active substance in the field of targeted drug delivery [11, 113]. In the past decade, microfluidic devices have been shown to accomplish a variety of tasks including cell segregation based on size and deformability [52, 75, 76, 222], concentration enrichment [129, 207, 208] and cell characterization [70, 123, 206]. Moreover, the increase in computing power has recently allowed numerical studies to contribute to the design of microfluidic devices. For example, Zhu et al. [222] numerically investigated an original microchannel geometry consisting of a semi-circular pillar located at the center of a microchannel: their study showed that this design can efficiently segregate cells based on membrane deformability. Recently, experiments were conducted using their microfluidic design and concluded that it can indeed sort cells based solely on membrane stiffness, with relatively high efficacy [76]. With regards to cell characterization, Gubspun et al. [70] proposed a method to determine capsule properties such as the membrane shear modulus by comparing the experimental and numerical “parachute” shape of capsules in a straight microchannel. While the majority of microfluidic

investigations operate in Stokes conditions, in recent years the design and study of inertial microfluidic devices has risen due to their ability to accurately segregate capsules by size and to extract them from their solvent [52, 223]. Inertial focusing in microfluidic devices typically relies on a spiral-shaped channel concentrating heavier capsules to the outer, lower-curvature edge of the channel, while lighter capsules concentrate closer to the inner, higher-curvature edge. A smooth geometry such as a spiral-shaped channel usually does not induce a high strain nor stress on a suspended capsule even in inertial regimes, however little is known about the strains and stresses induced by commonly encountered sharp geometries such as forks and corners on a capsule flowing in the presence of inertia. Moreover, the effect of such sharp geometries on the hydrodynamic interactions of a train of several capsules in inertial regimes is also an open question. More insight in these directions is of practical interest in the design and operation of inertial microfluidic devices because (i) the devices should not compromise the mechanical integrity of the capsules, i.e. it is critical to avoid capsule breakup, and (ii) cell-sorting processes typically operate in very dilute regimes to avoid capsule interactions, while a better understanding of such interactions would allow to operate these devices at a moderate to high concentration optimizing efficacy and throughput.

In the past four decades, a significant research effort has been invested into the modeling and the study of capsule deformations in non-inertial regimes, primarily because this regime is encountered in microcirculation such as capillary vessels and in traditional microfluidic devices. Using formalism from the thin-shell theory [67], Barthès-Biesel & Rallison first published an analytical solution for the time-dependant deformation of an elastic capsule in an unbounded, creeping shear flow in the limit of small deformations [21]. Over a decade later, Pozrikidis was able to go beyond the assumption of small deformations using a Boundary Integral Method (BIM) [156]. The same method was used to consider finite deformations of sheared capsules which inner and outer fluid viscosities differ [170], as well as to study the contribution of bending stresses [157], allowing to consider RBCs suspended in an unbounded shear flow [158]. Besides unbounded geometries, Zhao et al. [219] simulated RBCs in straight and constricted channels using a spectral BIM. A similar method was later used by Hu et al. [79] to consider an initially spherical capsule flowing through a square channel of width similar to the capsule diameter: the originality of their work is that they performed experiments and showed remarkable agreement between the measured and the computed capsule shape. Concomitantly, Park and Dimitrakopoulos [145] studied the deformation of a capsule with non-unity vis-

cosity ratio flowing through a sharp constriction. More recently, Balogh & Bagchi [15–17] used a Front-Tracking Method (FTM) to analyze the motion and deformation of RBCs through complex geometries resembling capillary vessels found in human microcirculation: their simulations exhibited in particular the well-known cell-free layer observed experimentally between the RBCs and the vessel walls [27, 140].

Regarding the study of flowing capsules in the presence of inertia, the aforementioned analytical theory for small deformations as well as the popular BIM both fall short of accounting for the convective term in the fluid momentum equation. Doddi & Bagchi [44] first studied inertial capsules in the context of two interacting capsules in a shear flow using the FTM. They showed in particular that the two capsules engage in spiralling motions at sufficiently high inertia. The inertial motion of a deformable capsule was then studied in straight microchannels [108, 166], where several equilibrium positions are found away from the channel centerline, along the cross-section diagonals. With regards to curved channels, Ebrahimi & Bagchi [48] recently investigated the migration of a single capsule over an impressive amount of varying parameters: the channel Reynolds number, the capsule deformability, as well as the aspect ratio and curvature of the channel were all varied independently. Their study shows that for sufficiently high inertia, exactly two focusing locations appear near the centers of the vortices of the secondary flow, known as Dean’s vortices. However no mention of the membrane internal strains and stresses is found in their work, as their goal was not to investigate the capsule integrity in such flows.

While straight and curved microchannels are essential components of microfluidic devices, such simple geometries do not account for the numerous junctions, corners and coils commonly found in these devices. To bridge this gap, Zhu & Brandt [221] investigated the non-inertial motion and the deformation of a single elastic capsule in a sharp corner. They showed that the capsule follows the streamlines of the undisturbed flow regardless of membrane deformability. Due to lubrication forces, the capsule velocity decreases when approaching the corner, reaches a minimum along the corner diagonal, and rises back to its steady state with an overshoot increasing with deformability. Similarly, the surface area of the capsule reaches a maximum inside the corner and reaches its steady value with an undershoot more pronounced as deformability is increased. Also reported in their study is the maximum stress in the capsule membrane, which can be used to assess mechanical integrity and characterize the cell mechanical properties. They find that the maximum stress deviation increases and shifts from the front to the top of the capsule with increasing deformability. Wang et al.

5.1. Introduction

[207, 208] later considered the inertial and non-inertial path selection of a single capsule through Y- and T-junctions, both typically encountered in microfluidic geometries. They observe that at high inertia, the capsule does not necessarily favor the daughter branch with the largest flow rate, and that this effect is more pronounced for stiff membranes (corresponding to a low capillary number). Recently, Lu et al. [129] investigated the interaction and path selection of capsules in a T-junction at moderate inertia, with the goal of enriching capsule solutions. When considering a pair of capsules, they show that the leading capsule is weakly affected by the presence of a trailing capsule, but that the reverse is not true. They find that the trailing capsule enters a different branch depending on the initial interspacing distance and on the flow rate split ratio between the two daughter branches of the T-junction. They then consider a train of capsules and find two distinct regimes: (i) the interspacing distance is low and the capsule interaction is high, resulting in an unsteady regime and affecting the trajectories of the capsules, and (ii) the interspacing distance is large and the capsule interaction is low, leaving the capsule trajectories identical to that of a single capsule. Interestingly, they report that the critical interspacing distance between two capsules plotted against the flow rate split ratio of the daughter branches results in a master curve independent of membrane deformability, capsule size, and Reynolds number.

In this chapter, we investigate the inertial and non-inertial motion and the interaction of deformable capsules flowing through a sharp corner, which is a very common geometry in microfluidic devices. As the efficiency of these devices is defined in terms of the capsules throughput, which can be optimized by increasing the flow rate as well as the concentration of capsules, our objective is two-fold: first, we aim to quantify the effect of inertia on the deformation of a single capsule in a microfluidic-relevant geometry, second, we seek to describe the hydrodynamic interactions and deformation differences between leading and trailing capsules when a pair and a train of capsules are considered. The rest of this chapter is organized as follows. In Section 5.2, we describe the governing equations as well as the flow configuration and the considered parameter space. In Section 5.3, we give an overview of our numerical method and we investigate the impact of the inlet length. We analyze the motion of a single capsule in Section 5.4, both in the non-inertial and in the inertial regimes. Section 5.5 is devoted to the analysis of binary interactions of a pair of capsules, where the influence of the initial interspacing distance is investigated. In Section 5.6, we consider a train of ten capsules flowing through the corner and we discuss the velocity and deformation discrepancies between the leading and trailing capsules.

Finally, we conclude in Section 5.7.

The documented source code allowing to reproduce all of the simulations and figures presented in this chapter is freely available online [80].

5.2 Governing equations and problem statement

The capsule membrane Γ is assumed infinitely thin and is surrounded by an incompressible, Newtonian fluid of constant viscosity and density. In all of this study, the capsule inner and outer fluids are assumed identical: in particular their viscosity ratio is unity. The fluid is described by the mass and momentum conservation equations:

$$\nabla \cdot \tilde{\mathbf{u}} = 0 \quad (5.1)$$

$$\frac{\partial \tilde{\mathbf{u}}}{\partial t} + \tilde{\mathbf{u}} \cdot \nabla \tilde{\mathbf{u}} = \frac{1}{\tilde{\rho}} \nabla \tilde{p} + \tilde{\nu} \Delta \tilde{\mathbf{u}} + \frac{1}{\tilde{\rho}} \tilde{\mathbf{f}}_{\mathbf{b}} \quad (5.2)$$

where $\tilde{\mathbf{u}}$ is the velocity field, \tilde{p} is the pressure field, $\tilde{\rho}$ is the density, $\tilde{\nu} = \tilde{\mu}/\tilde{\rho}$ is the kinematic viscosity, $\tilde{\mu}$ is the dynamic viscosity and $\tilde{\mathbf{f}}_{\mathbf{b}}$ is a body term accounting for the action of the membrane on its surrounding fluid. The dimensional quantities are denoted by the \sim symbol. The membrane exhibits elasticity and bending resistance, and its action on the fluid is local, resulting in the following expression for $\tilde{\mathbf{f}}_{\mathbf{b}}$:

$$\tilde{\mathbf{f}}_{\mathbf{b}} = \left(\tilde{\mathbf{f}}_{\text{elastic}} + \tilde{\mathbf{f}}_{\text{bending}} \right) \tilde{\delta}(\tilde{\mathbf{x}} - \tilde{\mathbf{x}}_{\Gamma}), \quad (5.3)$$

where $\tilde{\delta}(\tilde{\mathbf{x}} - \tilde{\mathbf{x}}_{\Gamma})$ is a Dirac distribution that is non-zero on the surface of the membrane.

The shear and area-dilatation membrane stresses are described using the thin-shell theory, and are briefly summarized here. The interested reader is referred to Green & Adkins [67] as well as to the analytical study of Barthès-Biesel & Rallison [21] for more details. We adopt a neo-Hookean law [67], which surface strain-energy function is expressed as:

$$\tilde{W}_s^{NH} = \frac{\tilde{E}_s}{2} \left(\lambda_1^2 \lambda_2^2 + \frac{1}{\lambda_1^2 \lambda_2^2} \right), \quad (5.4)$$

where $\lambda_{1,2}$ are the principal stretches in the two tangential directions, and \tilde{E}_s is a shear modulus. Note the change in convention with respect to Eq. (4.6): for ease of comparison with the existing literature [221], in this chapter \tilde{E}_s is three times lower than in our previous definition of the neo-Hookean law in

5.2. Governing equations and problem statement

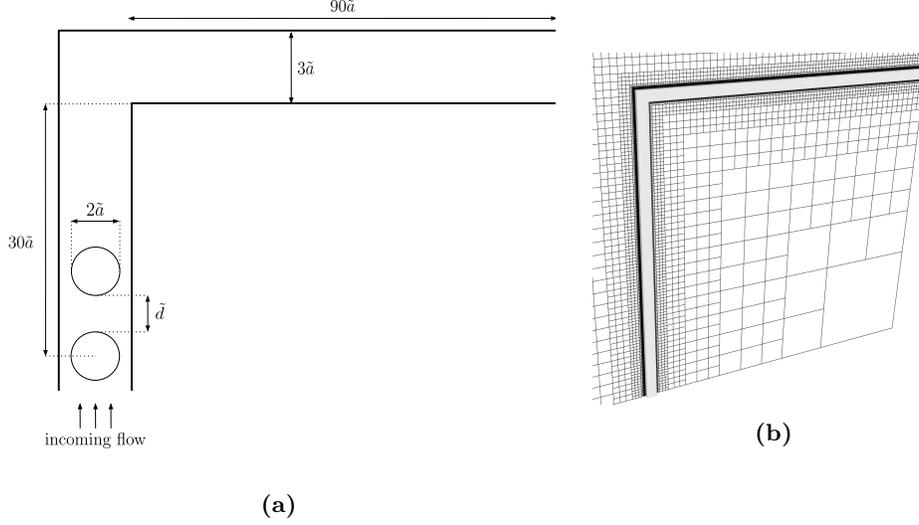


Figure 5.1: (a) Schematic of the geometry of the fluid domain. The channel has a square cross-section of side length $3\tilde{a}$. (b) Visualization of the full channel and the computational grid over the symmetry plane of the channel.

Chapter 4. As a result, the Capillary numbers in this chapter are scaled up by a factor of three. From the previous strain-energy function, the principal stresses $\tilde{\sigma}_{1,2}$ are given by:

$$\tilde{\sigma}_i = \frac{1}{\lambda_j} \frac{\partial \tilde{W}_s^{NH}}{\partial \lambda_i}, \quad i, j \in \{1, 2\}, \quad i \neq j. \quad (5.5)$$

The bending stresses for biological membrane are governed by the Helfrich's bending energy \mathcal{E}_b [77, 220]:

$$\tilde{\mathcal{E}}_b = \frac{\tilde{E}_b}{2} \int_{\Gamma} (2\tilde{\kappa} - \tilde{\kappa}_0)^2 dS, \quad (5.6)$$

where \tilde{E}_b is the bending modulus, $\tilde{\kappa}$ is the mean curvature and $\tilde{\kappa}_0$ is a reference curvature. Taking the variational formulation of Eq. (5.6) leads to the bending force per unit area \tilde{A} :

$$\tilde{\mathbf{f}}_{\text{bending}}/\tilde{A} = -2\tilde{E}_b(\Delta_s(\tilde{\kappa}) + 2(\tilde{\kappa} - \tilde{\kappa}_0)(\tilde{\kappa}^2 - \tilde{\kappa}_g + \tilde{\kappa}_0\tilde{\kappa}))\mathbf{n}, \quad (5.7)$$

where $\tilde{\kappa}_g$ is the Gaussian curvature and \mathbf{n} is the outer normal vector.

At $t = 0$, an initially spherical capsule of radius \tilde{a} is placed in a square channel of width $\tilde{W} = 3\tilde{a}$ at a distance $\tilde{h}_0 = 30\tilde{a}$ from a sharp corner, as

represented in figure 5.1. An average cross-section velocity \tilde{U}_0 is imposed at the inlet boundary, while the outflow boundary condition $\partial\tilde{\mathbf{u}}_n/\partial\mathbf{n} = 0$ is imposed at the outlet boundary. When several capsules are considered, we use the same initial conditions as Lu et al. [129]: a trailing capsule is inserted in the simulation only after the centroid of its preceding capsule has advanced by a distance \tilde{d} . Our problem is governed by the trailing dimensionless numbers:

1. The channel Reynolds number $Re = \tilde{\rho}\tilde{U}_0\tilde{W}/\tilde{\mu}$,
2. The Capillary number $Ca = \tilde{\mu}\tilde{U}_0\tilde{a}/\tilde{E}_s$, representing the ratio of viscous stresses over elastic stresses,
3. The reduced bending stiffness coefficient $E_b = \tilde{E}_b/(\tilde{E}_s\tilde{a}^2)$,
4. The confinement ratio $\beta = 2\tilde{a}/\tilde{W}$,
5. The reduced initial gap between capsules $d_0 = \tilde{d}/2\tilde{a} - 1$.

In this chapter, the Reynolds number Re ranges from 0.01 to 50, the Capillary number Ca varies from 0.075 to 0.35, and the reduced initial gap d_0 is chosen from 0.125 to 1. The reduced bending stiffness E_b and the confinement ratio β are both kept constant, with $\beta = 2/3$ and $E_b = 5 \cdot 10^{-3}$ as proposed by Pozrikidis [160]. The reference curvature $\tilde{\kappa}_0$ is equal to $-2.09/\tilde{a}$ in this study, as is common for some biological membranes such as RBC membranes [159, 214]. In the rest of this chapter, we use the capsule radius \tilde{a} as the characteristic length scale, and we define the characteristic time scale as the radio of the capsule radius over the average cross-section velocity, i.e. $t = \tilde{a}/\tilde{U}_0$.

5.3 Numerical method and validations

We use our adaptive Front-Tracking Method (FTM) to solve the above equations: we provide below a brief overview of the numerical method, while an in-depth description is available in [91]. Eq. (5.1) and Eq. (5.2) are solved using the Finite Volume method on an adaptive octree grid using the open-source software Basilisk [150]. The membrane is discretized using an unstructured triangulation and Eq. (5.5) is solved using a linear Finite Element Method, while Eq. (5.7) is solved using a paraboloid-fitting method. The membrane triangulation and the octree grid communicate by means of

the immersed boundary method [147, 148], where the Dirac distribution in Eq. (5.3) is regularized using a cosine-based formulation:

$$\tilde{\delta}(\mathbf{x}_0 - \mathbf{x}) = \begin{cases} \frac{1}{64\tilde{\Delta}^3} \prod_{i=1}^3 \left(1 + \cos \left(\frac{\pi}{2\tilde{\Delta}} (x_{0,i} - x_i) \right) \right) & \text{if } |x_{0,i} - x_i| < 2\tilde{\Delta} \\ 0 & \text{otherwise} \end{cases}, \quad (5.8)$$

where $\mathbf{x}_0 = [x_{0,1} \ x_{0,2} \ x_{0,3}]$ is the location of a Lagrangian node on the surface discretization of the membrane, and $\tilde{\Delta}$ is the local mesh size of the Eulerian octree grid. Extensive validation of the present numerical method was the focus of our previous study [91] and is therefore not presented here. Nonetheless, the convergence with respect to the Eulerian grid as well as the release distance of the capsule from the corner are investigated below.

In the immersed boundary method, it is well known that the support of the regularized Dirac distribution may extend outside of the fluid domain if the immersed object of interest becomes very close to the domain walls [129, 194, 208]. In order to avoid unphysical loss of momentum for the specific membrane nodes close to the wall, it is important to ensure that none of the supports of the regularized Dirac distribution extend outside of the fluid domain, i.e. that there always exist more than two grid cells between membrane nodes and the domain boundaries. As such, we simulate the dynamics of a capsule for two different grid resolutions in the configuration where it is most deformed and is the closest to the channel wall, as shown in figure 5.2b. Figure 5.2a shows the velocity of the capsule \tilde{V} inside and downstream of the corner for Eulerian resolutions equivalent to 32 and 64 grid cells per initial capsule diameter, as well as the deviation of the velocities in these two configurations. Excellent agreement is found between the velocities computed using the two grid resolutions, with the maximum discrepancy lower than 1% and the average discrepancy over the considered time range of about 0.5%. Moreover, in both configurations it was found that more than 3 grid cells are present in the lubrication layer between the capsule tail and the upper corner wall. These results indicate that an equivalent grid resolution of 32 grid cells per capsule initial diameter is sufficient to obtain converged solutions, and that the present simulations do not suffer from immersed boundary stencils extending outside of the fluid domain.

Next we investigate the influence of the normalized release distance D_c between the initial position of the capsule centroid and the corner. Indeed, after its release the capsule relaxes from a spherical to an equilibrium steady shape and it is important that this steady state is reached before the capsule

5.3. Numerical method and validations

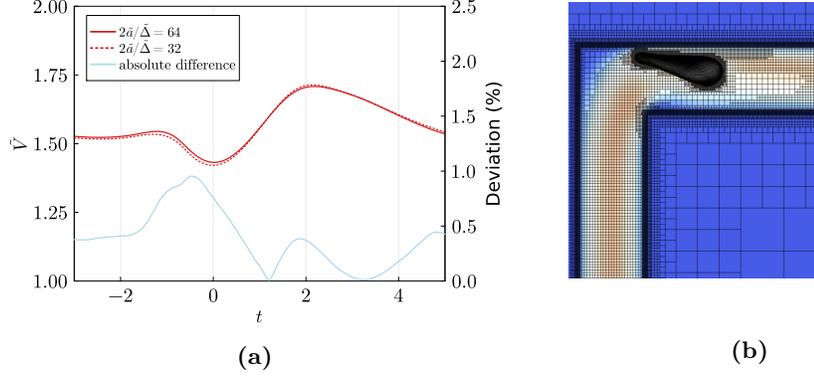


Figure 5.2: (a) Centroid velocity of a capsule at $Ca = 0.35$ and $Re = 50$ for two grid resolutions: 32 grid cells per initial diameter (red dotted line) and 64 grid cells per initial diameter (red solid line). The blue curve denotes the deviation in the centroid velocities for these two grid resolutions. (b) Corresponding shape and grid resolutions of the capsule and the flow field: blue means zero velocity and red means large velocity.

enters the corner. As such, we consider three initial distances $D_c = 15, 30$ and 60 in the most challenging configuration at $Re = 50$ and $Ca = 0.35$, i.e. the capsule is highly deformable and placed in a highly inertial flow. The inlet boundary is located at a distance of $90a$ away from the corner and is therefore sufficiently far away from the capsule to not alter its response. The norm of the capsule centroid velocity \tilde{V} and the reduced capsule area $\mathcal{A} = \tilde{A}/4\pi a^2$ are shown in figure 5.3, where the origin of the reduced time t is chosen at the time the capsule reaches a minimum velocity \tilde{V}_{min} . In figure 5.3a we remark that the capsule velocity \tilde{V} at $D_c = 15$ decreases significantly prior to entering the corner: this is because the initially spherical capsule is located farther away from the channel walls and is therefore advected faster than when it has reached a steady shape. We observe that neither the capsule velocity shown in figure 5.3a nor the normalized area shown in figure 5.3b present a steady state before the capsule enters the corner in the case $D_c = 15$. Therefore a larger initial distance D_c should be used. When considering $D_c = 30$, both the velocity and the normalized area present steady values before the corner. Interestingly, inside and after the corner the capsule velocity and area almost overlap when the capsule is released 15 and 30 initial radii away from the corner, suggesting that the corner resets the dynamics of the capsule regardless of its previous state. The fact that steady values for the velocity and the area of the capsule are reached before the corner for $D_c = 30$ suggests that this initial release distance is suitable for the rest of this study. Interestingly, releasing the capsule at $D_c = 60$

5.4. Motion and deformation of a single capsule

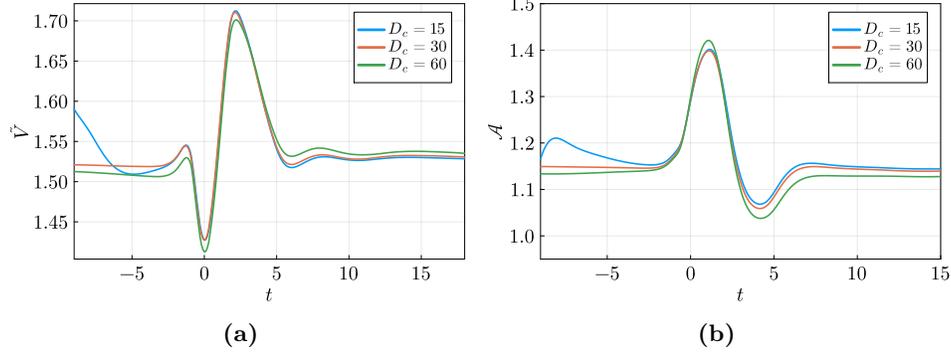


Figure 5.3: Centroid velocity (a) and normalized area (b) of a capsule flowing through a corner from three distinct normalized release distances $D_c = 15, 30$ and 60 , at $Re = 50$ and $Ca = 0.35$.

leads to an unexpected result: the capsule seems to no longer be in a steady motion as its velocity (respectively its normalized area) is slightly decreasing (respectively slightly increasing) prior to entering the corner. This suggests that in this challenging configuration, the relaxation of the capsule from a fixed spherical shape to a steady “parachute” shape occurs over very long time scales. However, the magnitude of the deviations between the capsule velocity and area in the cases $D_c = 30$ and 60 is at most 3%. As the capsule has already reached a pseudo steady state by the time it reaches the corner in the case of $D_c = 30$, and as the aforementioned discrepancies are small, we choose $D_c = 30$ in the rest of this study. Again, this short study of the impact of the initial release distance on the capsule dynamics was performed in our most challenging configuration as we considered our highest Reynolds number and highest Capillary number. The discrepancy between the cases $D_c = 30$ and 60 is less pronounced – sometimes nonexistent – for less deformable membranes and less inertial flows.

5.4 Motion and deformation of a single capsule

We consider the motion of a single capsule through a square duct at $Ca = 0.075, 0.15, 0.25, 0.35$ and $Re = 0.01, 1, 25, 50$, extending the investigation carried out in a non-inertial framework by Zhu & Brandt [221]. In order to establish the influence of the increasing effect of inertia on the motion and the deformation of a single capsule, we first recall the overall dynamics of a capsule moving through a duct corner in the Stokes regime, as detailed in [221]. The capsule once released from its initial position moves along

the center of the channel due to the symmetry of the flow far from the corner. While approaching the corner, the capsule velocity decreases until reaching a minimum in the corner region. The capsule experiences moderate to high deformation (depending on the Capillary number considered) due to the flow acceleration, and its velocity strongly increases; this phenomenon being referred to as the overshoot of velocity. Further away from the corner, the capsule moves in the downstream branch of the duct, relaxing to a steady state (shape and velocity), and moving along the center of the duct.

We investigate the influence of the Reynolds number Re and the Capillary number Ca on the dynamics and the deformation of the capsule, reporting the time evolution of its surface area \mathcal{A} scaled by the initial surface area of the capsule $\mathcal{A}_{\text{sphere}} = 4\pi\tilde{a}^2$, as well as the velocity V of the capsule centroid scaled by its equilibrium velocity V_{eq} before the capsule enters the corner region. In the remainder of this chapter and unless otherwise stated, the time origin is chosen such that $t = 0$ when capsule velocity reaches a global minimum, i.e. $V_{min} = V(t = 0)$. We borrow this convention from Zhu & Brandt [221], as it corresponds to setting the time origin when the capsule is located at the heart of the corner.

5.4.1 Influence of the Reynolds and Capillary numbers

To characterize the dynamics of the capsule as it flows through the corner, we analyze the time evolution of the centroid velocity V and the surface area \mathcal{A} . Figure 5.4 shows the velocity of the capsule centroid for Ca ranging from 0.075 to 0.35. Re is constant for each subfigure of figure 5.4. Conversely, figure 5.5 shows the same data as figure 5.4, but with each subfigure corresponding to a constant Ca . From both figures, we observe a general trend for all cases: the capsule approaches the corner with a steady velocity V_{eq} , then reaches a global minimum V_{min} and a global maximum V_{max} as it flows through the corner, and relaxes back to V_{eq} downstream of the corner. Moreover, we observe in figure 5.4 that the velocity extrema increase with increasing Ca . In the more inertial regimes especially, the maximum velocity deviation of the capsule at $Ca = 0.35$ is close to three times that of the capsule at $Ca = 0.075$.

We note from figure 5.5 that the curves corresponding to $Re = 0.01$ and $Re = 1$ practically overlap, indicating that the capsule motion in low inertial regimes is very similar to that in the non-inertial regime. As the Reynolds number is increased to 25 and 50, major deviations from the non-inertial regime appear. First, as the capsule enters the corner area, a local maximum appears in the capsule velocity, which is independent of the Capillary

number, and is about 1% greater than V_{eq} at $Re = 25$ and 2% greater than V_{eq} at $Re = 50$. This local maximum is due to the migration of the capsule across the centerline of the secondary channel: in this process the capsule is located far away from the channel walls and is therefore less subject to their confinement effect. Then, the minimum velocity V_{min} is reached in the heart of the corner. Interestingly, at small Ca , V_{min} is observed to be independent of the Re , as can be seen in figure 5.5a at $Ca = 0.075$. In contrast, in the case of larger Ca the minimum velocity of the capsule increases slightly with Re . A difference of about 4% is observed for V_{min} as Re increases from 0.01 to 50 for both $Ca = 0.25$ and $Ca = 0.35$.

As the capsule exits the corner area and migrates to the channel centerline, its velocity reaches its maximum value V_{max} which increases with increasing Re and Ca : at $Ca = 0.075$, V_{max} increases by 3% between $Re = 0.01$ and $Re = 50$ while at $Ca = 0.35$, V_{max} increases by about 8% between $Re = 0.01$ and $Re = 50$. Then, the capsule velocity relaxes back to its equilibrium value and its relaxation time increases with increasing Re . Interestingly, velocity undershoots are observed during the relaxation stage in the inertial regime, which magnitude increases with Re . The relaxation time does not depend on Ca .

The time evolution of the normalized capsule area \mathcal{A} is shown in figure 5.6 for fixed Re and in figure 5.7 for fixed Ca . We observe that the area presents a maximum \mathcal{A}_{max} at around $t = 1$ before relaxing to its equilibrium value \mathcal{A}_{eq} . Unsurprisingly, figure 5.6 confirms that a large Ca , i.e. a highly deformable capsule, results in a greater surface area than for lower Ca . Figure 5.6 also shows that the magnitude of the maximum area increases with Ca . Moreover, when large Ca are considered the time evolution of the capsule area presents some undershoots that are more pronounced as Re is increased. Additionally, figure 5.7 reveals that Re has a very strong influence on the deformation of the capsule, especially at large Ca : at $Ca = 0.075$, $\mathcal{A}_{max}/\mathcal{A}_{eq}$ increases from 2% to 8% between $Re = 0.01$ and $Re = 50$, and at $Ca = 0.35$ it increases from 8% to a staggering 22% between $Re = 0.01$ to $Re = 50$. In particular, at $Ca = 0.35$ the maximum capsule area increases from 9% to 40% of the area of a sphere between the non-inertial and the highly inertial regimes. These area deviations are very large and are discussed further in the next section.

5.4.2 Maximum deformation of the capsule

The maximum surface area \mathcal{A}_{max} of the capsule is presented in figure 5.8, as a function of both the Reynolds number and the Capillary number. To

5.4. Motion and deformation of a single capsule

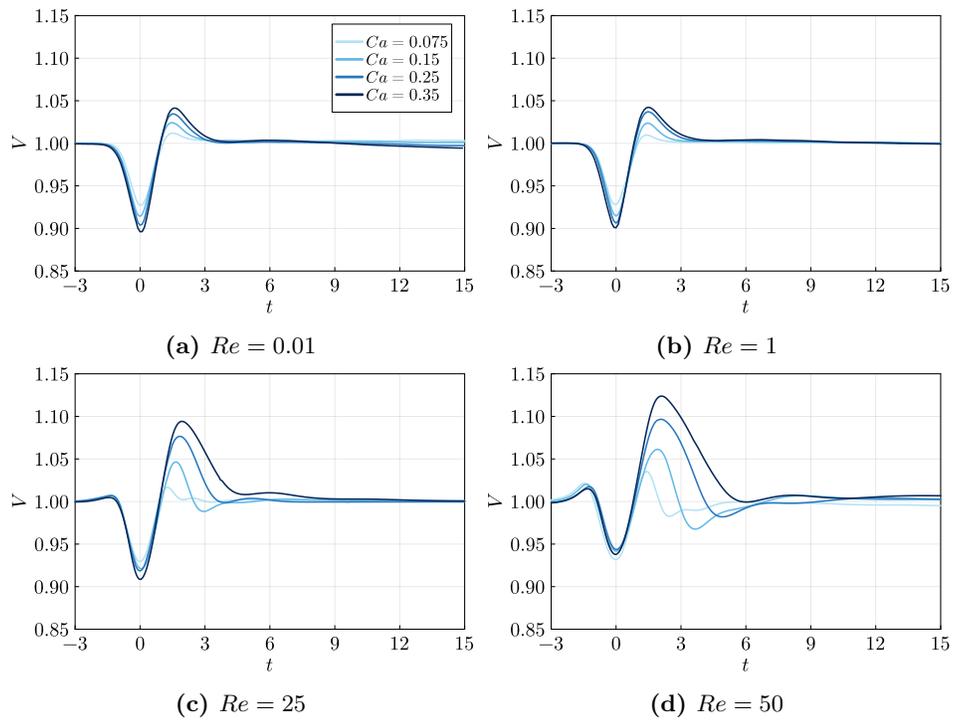


Figure 5.4: Temporal evolution of the capsule centroid velocity V at fixed Reynolds numbers.

5.4. Motion and deformation of a single capsule

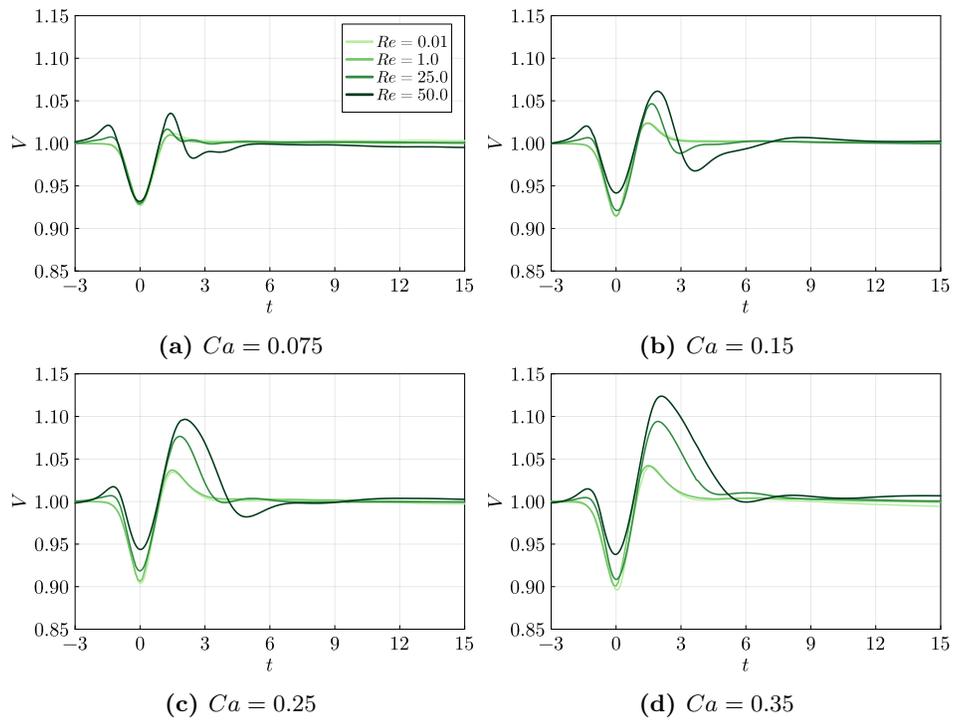


Figure 5.5: Temporal evolution of the capsule centroid velocity V at fixed Capillary numbers.

5.4. Motion and deformation of a single capsule

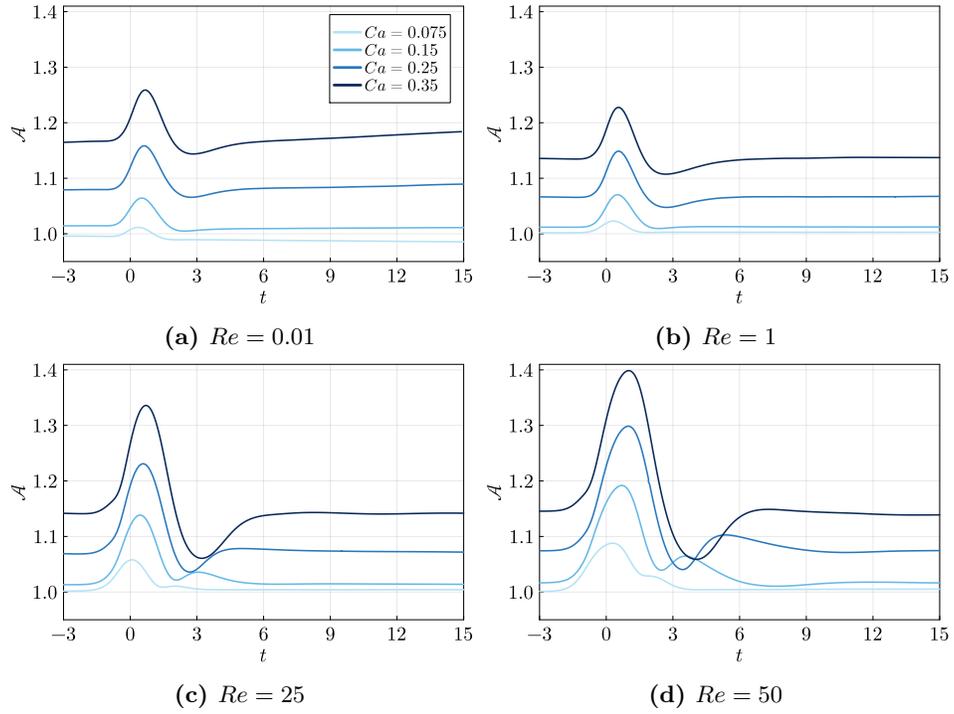


Figure 5.6: Temporal evolution of the capsule surface area \mathcal{A} at fixed Reynolds numbers.

5.4. Motion and deformation of a single capsule

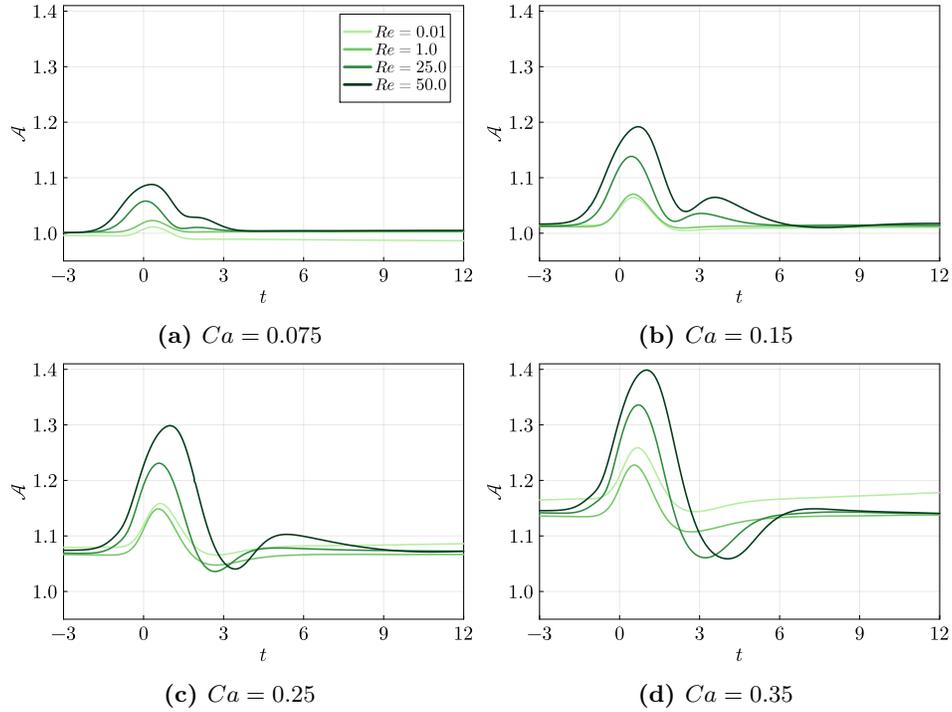


Figure 5.7: Temporal evolution of the capsule surface area \mathcal{A} at fixed Capillary numbers.

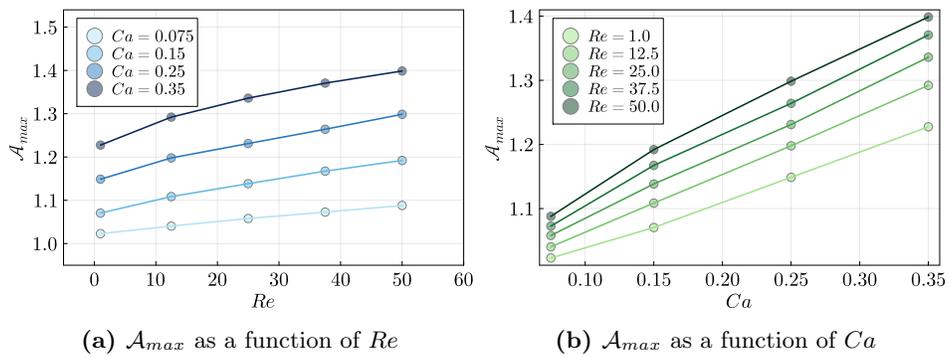


Figure 5.8: Maximum area \mathcal{A}_{max} as a function of Re and Ca for a single capsule passing through the corner.

5.4. Motion and deformation of a single capsule

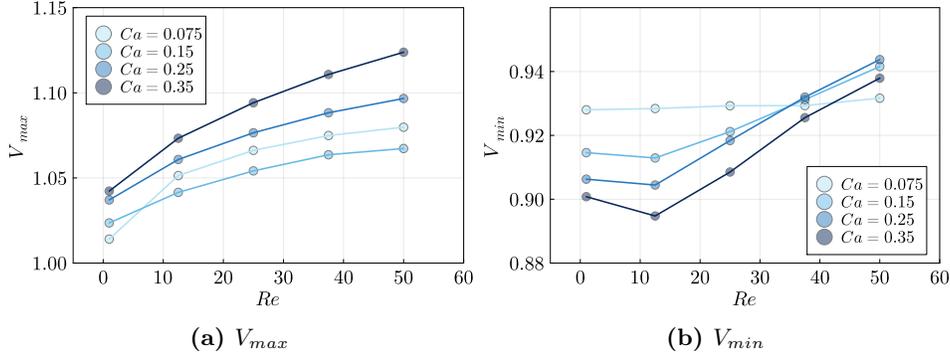


Figure 5.9: Maximum (minimum) velocity V_{max} (V_{min}) as a function of Re and Ca for a single capsule passing through the corner.

better analyze the trends in this figure, we also report the maximum area at intermediate Reynolds numbers, namely at $Re = 12.5$ and 37.5 . The data reported in figure 5.8 clearly exhibits a double linear scaling of \mathcal{A}_{max} with both Ca and Re as long as Ca is below 0.35 – at $Ca = 0.35$, the shape of the curve $\mathcal{A}_{max}(Re)$ is slightly concave. The slope of the scaling is about 0.003 for $\mathcal{A}_{max}(Ca)$ and 1.12 for $\mathcal{A}_{max}(Re)$. This means that the capsule maximum deformation responds proportionally to the Capillary number, but also to the Reynolds number. To our knowledge, this is the first time such a trend has been reported and established for low ($Re = 1$) to moderate ($Re = 12.5, 25, 37.5, 50$) inertial regimes. We believe that this result can be used as a predictive tool for many studies involving single capsules travelling through duct corners, as the maximum deformation observed for a capsule is a measure of its mechanical integrity, which is of major interest in many microfluidic applications.

Additionally, we present in figure 5.9 the maximum and minimum velocity of the single capsule flowing through the corner. In the non-inertial regime, the maximum velocity of the capsule increases with Ca , as shown in figure 5.9a. In inertial conditions we observe that V_{max} increases for Re ranging from 1 to 50. The increase in V_{max} between $Re = 1$ and $Re = 50$ is significant in figure 5.9a, especially for large Ca . For instance, at $Ca = 0.35$, V_{max} increases by about 8% between the non-inertial and the highly inertial regimes. We then consider the evolution of the minimum velocity V_{min} for a single capsule at various Ca and Re in figure 5.9b. In general, we observe that the minimum velocity decreases with Ca in both the non-inertial and the inertial regimes for $Re \leq 25$. In figure 5.9b, we also observe a non-monotonous behavior of V_{min} at low inertia and at sufficiently high Ca : for

5.4. Motion and deformation of a single capsule

$Ca \geq 0.15$, V_{min} first decreases with increasing Re , reaching a minimum for $Re = 12.5$, before increasing sharply at $Re > 12.5$. Overall, we observe from figure 5.9 that the presence of inertia tends to increase both velocity extrema of the capsule, especially at large Ca .

A quantity of practical interest to experimentalists is the maximum stress experienced by the capsule, as it can be used to predict *a priori* if a given geometry can induce plastic deformation or even breakup of the capsule membrane [76]. More specifically, it is the largest eigenvalue $\tilde{\sigma}_2$ of the stress tensor $\tilde{\boldsymbol{\sigma}}$ that can bring insight into the mechanical integrity of the membrane. In figure 5.10, we show the maximum and average values of $\tilde{\sigma}_2$ over the membrane surface as the capsule approaches and flows through the corner at $Ca = 0.35$ and $Re = 1, 25$ and 50 . We observe that $\tilde{\sigma}_{2,avg}$ follows a trend very similar to that of the capsule area observed in figure 5.7d: $\tilde{\sigma}_{2,avg}$ varies smoothly with time, presents a maximum near $t = 1$ and a local minimum near $t = 2.5$, and the value of the maximum deviation from steady state nearly doubles between the low and moderate inertial cases $Re = 1$ and $Re = 50$. We also note that the steady state value of $\tilde{\sigma}_{2,avg}$ prior to entering the corner is independent of Re , as was observed in the case of the capsule surface area in figure 5.7d. In particular, we find by comparing figures 5.7d and 5.10 that at $Ca = 0.35$, a non-dimensional area \mathcal{A} of about 1.14 leads to an average non-dimensional membrane stress of about 0.4. The steady state of the maximum stress $\tilde{\sigma}_{2,max}$, however, increases by about 40% between the low inertial case ($Re = 1$) and the moderate inertial cases ($Re = 25, 50$). Inside the corner, $\tilde{\sigma}_{2,max}$ increases by nearly 75% between $Re = 1$ and $Re = 50$, confirming that a capsule in a moderate inertial regime has a higher risk of breakup than in a low inertial regime.

It is worth noting that for all Re , the value of the maximum stress $\tilde{\sigma}_{2,max}$ is about double that of the average stress $\tilde{\sigma}_{2,avg}$: since we showed previously that $\tilde{\sigma}_{2,avg}$ is closely related to the capsule area – a quantity that is relatively easy to measure experimentally –, this observation can be used by experimentalists as a rule of thumb to estimate the maximum stress in the capsule membrane and assess the mechanical integrity of the membrane.

5.4.3 Evolution of the capsule shape

We now illustrate the temporal evolution of the capsule travelling through the corner. Figure 5.11 shows the outline of the capsule in the symmetry plane $z = 0$ for successive discrete times. The capsule outlines are given for $Ca = 0.075$ and $Ca = 0.35$ and $Re = 0.01, 25$ and 50 . Prior to entering the corner, the capsule adopts a steady shape that is determined by the

5.4. Motion and deformation of a single capsule

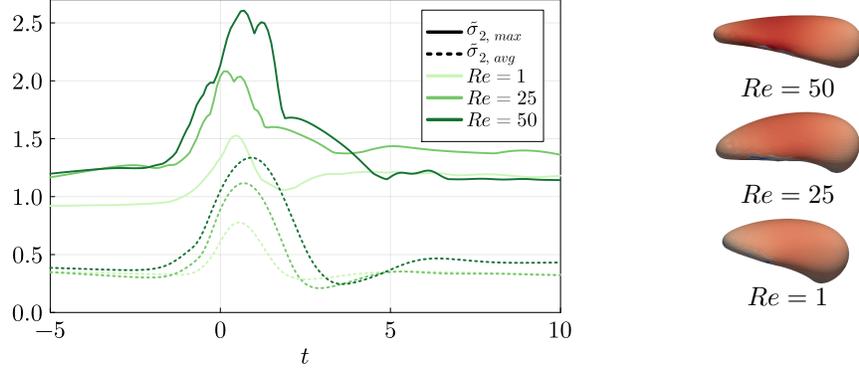


Figure 5.10: Left: Maximum and average tensions in the capsule at $Ca = 0.35$ and $Re = 1, 25$ and 50 . Right: Capsule shape colored by $\bar{\sigma}_2$ when $\bar{\sigma}_{2,max}$ reaches its maximum.

confinement of the walls. In the case of $Ca = 0.35$, we observe the well-known “parachute” shape. Upstream of the corner, the trajectory of the capsule coincides with the centerline of the primary (vertical) channel. As the capsule flows through the corner, the capsule deviates from the channel centerline: in the non-inertial regime, Zhu & Brandt [221] showed that the capsule trajectory closely matches the flow streamlines. We obtain the same conclusion in the inertial regime. When inertia is considered, the capsule trajectory crosses the horizontal centerline of the secondary channel and comes increasingly close to the upper wall as Re increases, before relaxing to the channel centerline.

Figures 5.11a and 5.11b show clear differences in the effects of Ca in the Stokes regime. Increasing Ca from 0.075 to 0.35 causes the equilibrium shape of the capsule to change from an slightly deformed spheroid to a concave “parachute” shape. For a small $Ca = 0.075$, the equilibrium shapes of the capsule remain similar as Re increases from $Re = 0.01$ to $Re = 50$ (see figures 5.11a, 5.11c, and 5.11e). However, the deformation of the capsule becomes more evident inside the corner at higher Re , particularly in figure 5.11e. After passing the corner, the capsule shape returns to its steady spheroid shape observed in the Stokes regime for all values of Re . In the case of a high $Ca = 0.35$, we observe that the equilibrium shape of the capsule is more and more concave as Re increases. Inside the corner, the capsule is highly elongated and presents an increasingly long tail for increasing Re – e.g. figure 5.11f in the case of $Re = 50$. In the highly inertial regime, strong lubrication interactions occur between the capsule and the top wall, resulting in a flat top surface.

5.4. Motion and deformation of a single capsule

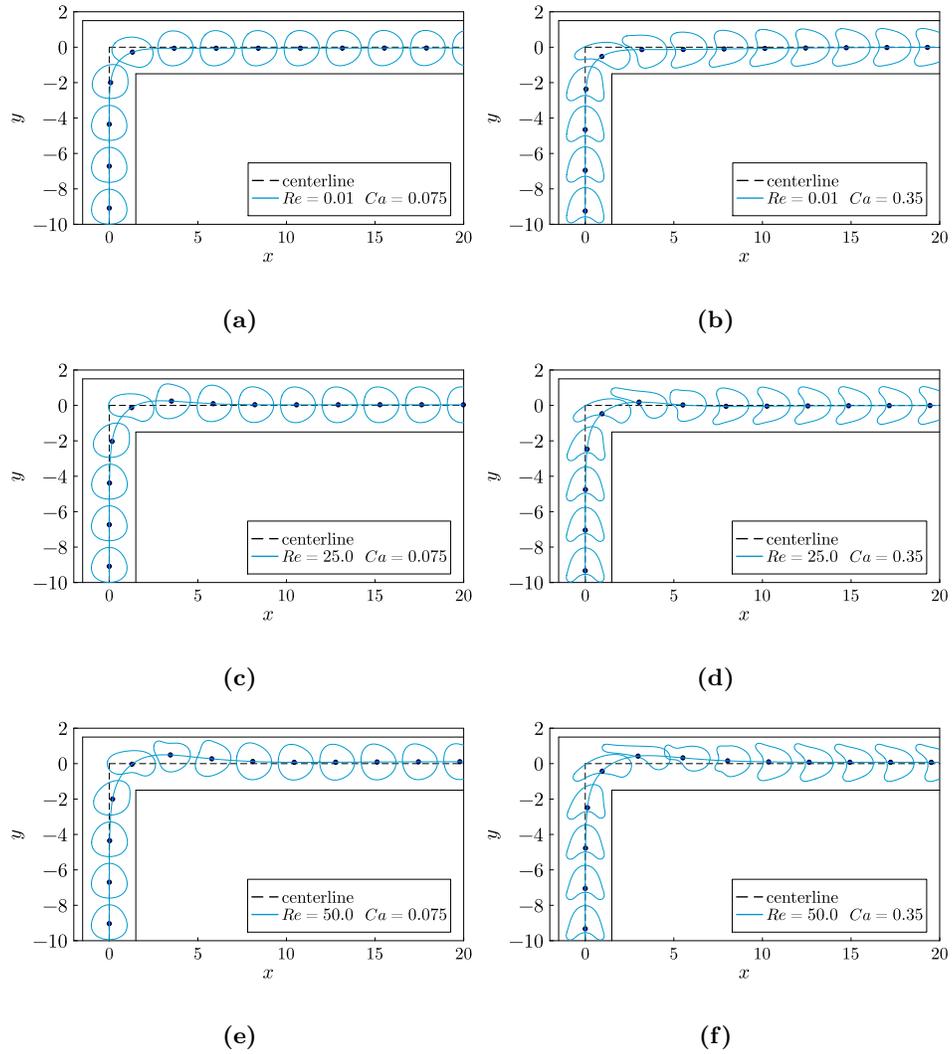
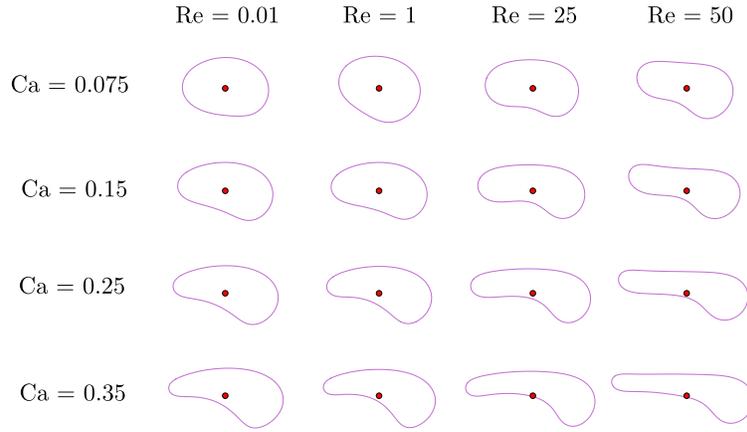
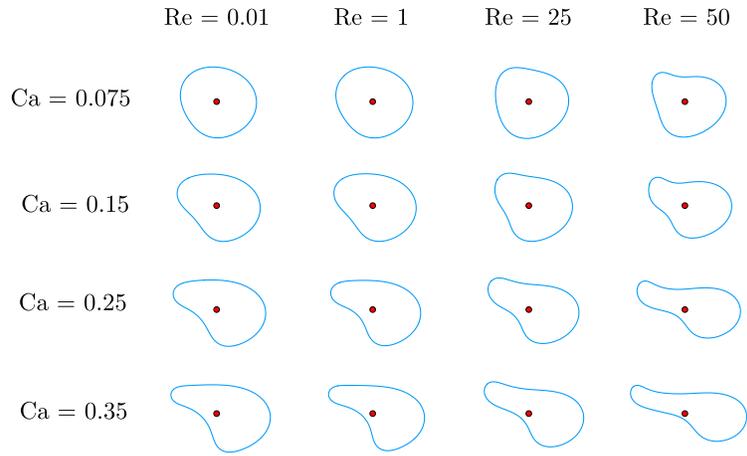


Figure 5.11: Sequence of Capsule outlines for different Ca and Re . The time between each frame is $t = 1.5$.

5.4. Motion and deformation of a single capsule



(a)



(b)

Figure 5.12: Outlines of a single capsule passing a corner with (a) maximal area \mathcal{A} and (b) maximal velocity v_{max} at various Re and Ca .

In figures 5.12a and 5.12b, we present the single capsule outline with the maximum area \mathcal{A}_{max} and the maximum velocity V_{max} inside the corner for all the cases investigated in this section. Inside the corner, the maximum area of the single capsule is reached when it approaches the upper wall and it is quickly followed by the maximum velocity. From figures 5.12a and 5.12b, we observe in particular that a high Re leads to an elongation of the capsule in the streamwise direction, while a high Ca increases the concavity of the capsule. Moreover, we note that the centroid of the capsule moves closer to the rim of the outline at high values of Ca : note that the centroid drawn in figures figure 5.11-5.12b corresponds to the centroid of the three-dimensional capsule, not to that of the two-dimensional outline. The results shown in figures 5.11-5.12b indicate that Ca has a significant effect on capsule deformation, while Re has a more pronounced effect on the trajectory of the capsule as well as its deformation resulting from the lubrication layer against the top wall of the corner. In particular, at high Re , the capsule undergoes significant stretching, which may cause damage or even rupture in microfluidic devices. Understanding the effects of Re on capsule deformation and the resulting damage is crucial in designing efficient and reliable microfluidic devices.

5.4.4 Discussion on the Stokes regime

We observe in figures 5.7 a surprising, non-monotonous behavior of the capsule area around $Re = 1$: at large Ca , the area of the capsule is smaller at $Re = 1$ than at $Re = 0.01$ and $Re = 25$. Additionally, in figure 5.7a the steady area of the capsule at $Re = 0.01$ and $Ca = 0.075$ downstream of the corner is about 1% lower than the initial spherical area of the capsule, indicating a small loss of the internal capsule volume. The cause of these observations may be related to the limitations of the FTM coupled with a sub-optimal choice of numerical parameters in the case of $Re = 0.01$ only. Indeed, the immersed boundary method is known to conserve volume asymptotically rather than to machine precision. In earlier IBM studies involving capsules, the volume loss is always small, typically below 1% [15, 129, 207, 208]. Moreover, Stokes conditions are known to be challenging for PDE-based incompressible Navier-Stokes solvers, as the matrix inverted in the pressure-Poisson problem is less well conditioned at low Re . While it is worth noting that the capsule area in the Stokes regime should be interpreted with caution, these limitations only affect the capsule area and not the centroid velocity. Moreover, our solver was extensively validated in Stokes conditions in [91] and showed excellent agreement with the BIM as

well as other FTM solvers. As such, while further investigation should be conducted in the Stokes regime, it cannot be excluded that at high Ca the capsule area at $Re = 0.01$ is physically slightly greater than that at $Re = 1$. Finally, the main focus of the present work is to investigate the inertial motion and deformation of capsules through a sharp corner, i.e. in conditions where our FTM solver does not suffer from the limitations outlined above.

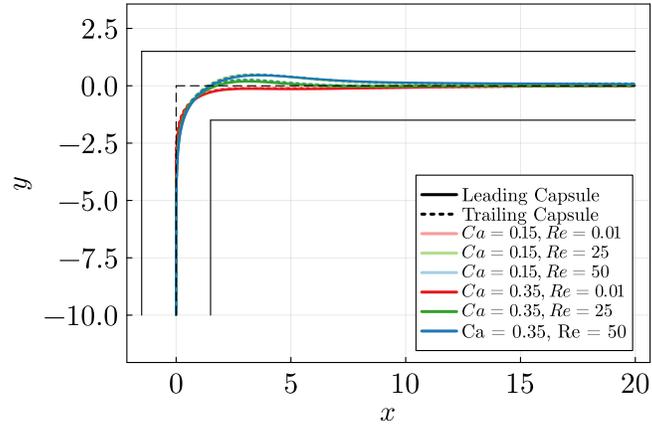
5.5 System of two capsules

In this section, we consider two identical capsules flowing through the corner as we vary the normalized interspacing distance $d = \tilde{d}/2\tilde{a} - 1$ between the capsules as well as the Reynolds and Capillary numbers. Lu et al. [129] previously considered the binary interaction of capsules flowing through a T-junction: they showed that when $d_0 \geq 1.3$ the trailing capsule has minimal impact on the motion of the leading capsule. In contrast, in their T-junction geometry Lu et al. observed that the motion of the trailing capsule is significantly affected by the presence of the leading capsule. To gain insight into the physical features relevant to capsule interactions through a corner in the inertial and non-inertial regimes, we select small values for the normalized interspacing distance $d_0 = 1, 1/2$, and $1/4$ and we examine phenomena such as migration, dynamics and deformation of the leading and the trailing capsules.

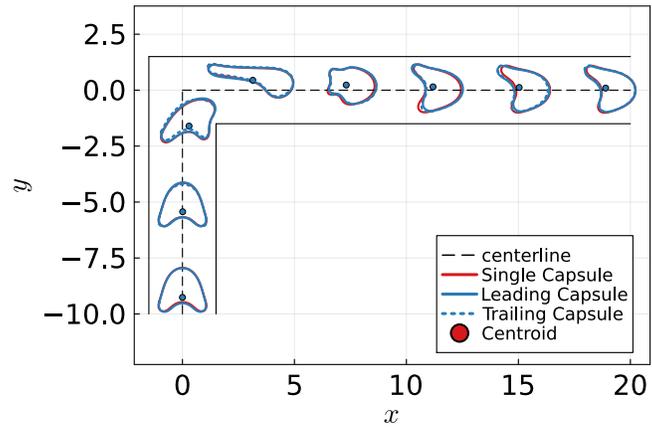
5.5.1 Qualitative analysis: trajectory and capsule shape

We first analyze the trajectory and the qualitative shapes of the pair of capsules as they flow through the corner. Figure 5.13a shows the trajectory of the capsules at $Re = 0.01, 25$ and 50 and $Ca = 0.15$ and 0.35 . We note that all curves corresponding to the same Ca overlap: Ca has no impact on the path of either the leading or the trailing capsule. Likewise, we observe no significant difference in the trajectories of the leading and the trailing capsules, unlike the strikingly different paths reported in the case of a T-junction [129]. In fact, the key parameter that controls the capsule trajectory is the Reynolds number. As Re increases, the inertia drives the capsule closer to the upper channel wall, as observed in Section 5.4 in the case of a single capsule. We then illustrate the capsule shape on the symmetry plane $z = 0$ in figure 5.13b for the most deformed capsule configuration corresponding to $Ca = 0.35$ and $Re = 50$ with an initial interspacing distance $d_0 = 0.25$. We compare the outlines of the leading and the trailing capsules to that of a single capsule in the same conditions. Qualitatively, the deformation of

5.5. System of two capsules



(a)



(b)

Figure 5.13: (a) Trajectory of the two capsules at different Ca and Re . (b) Outlines of the leading and trailing capsules at $Ca = 0.35, Re = 50, d_0 = 0.25$, with comparison to a single capsule.

5.5. System of two capsules

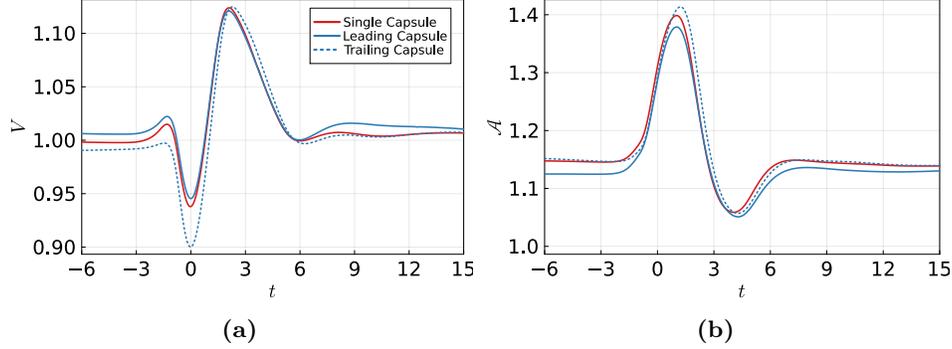


Figure 5.14: Temporal evolution of the velocity V and the area \mathcal{A} of capsules at $Ca = 0.35$, $Re = 50$ with $d_0 = 0.25$: a comparison of the leading, trailing and a single capsule.

interacting capsules is not significantly different than that observed in the case of a single capsule. Perhaps more surprisingly, the qualitative outlines of the leading and the trailing capsules are also very similar, almost overlapped, even in the strongly interacting configuration corresponding to $d_0 = 0.25$. Note that this qualitative shape analysis relies on the outline of the capsule in the plane of symmetry $z = 0$, while the actual three-dimensional shape of the leading and trailing capsules may differ more strongly.

5.5.2 Quantitative analysis: velocity and membrane area

We now compare the temporal evolution of the velocity of the centroids of the capsules as well as the time evolution of their surface areas, as plotted in figure 5.14. To simplify the identification of interaction features, we first focus on the most deformed configuration corresponding to $Ca = 0.35$, $Re = 50$ and $d_0 = 0.25$. For reference, we also plot the evolution of a single capsule under the same conditions in red. Throughout the remainder of this chapter, and unless otherwise stated, the velocity of interacting capsules is normalized by the equilibrium velocity V_{eq} of a single capsule for the same Capillary and Reynolds numbers. This normalization choice allows for an unbiased comparison between the velocities of the leading and the trailing capsules. In this section we also denote the reduced velocity of the single capsule by V_s , that of the leading capsule by V_l and that of the trailing capsule by V_t . Similarly, we denote by \mathcal{A}_s , \mathcal{A}_l , \mathcal{A}_t the normalized areas of respectively the single, leading and trailing capsules.

In figure 5.14a, we observe that the velocity of the leading capsule is affected by the presence of the trailing capsule before it reaches the corner,

5.5. System of two capsules

as it is about 1% higher than that of a single capsule. However, the extrema of V_l as it flows through the corner closely match those of V_s . After the corner, V_l is about 2% larger than V_s but slowly relaxes back to V_s further downstream. With regards to the trailing capsule, we note that its velocity is more markedly affected by the presence of the leading capsule. Prior to reaching the corner, V_t is about 1% lower than V_s , but inside the corner its minimum value is 4% lower than V_s . However, the maximum of V_t is identical to that of both V_l and V_s . Downstream of the corner, V_t quickly relaxes back to V_s and maintains a similar value thereafter, eventually converging to V_{eq} . The time evolution of the surface areas of the pair of capsules is shown in figure 5.14b. The normalized area of the leading capsule \mathcal{A}_l is clearly influenced by the presence of the trailing capsule, as was observed above in the case of its velocity. The steady and maximum areas of the leading capsule are about 2% lower than that of the single capsule. In contrast, the steady area of the trailing capsule closely matches that of the single capsule upstream and downstream of the corner, while its maximum value is about 1% higher than that of the single capsule. We postulate that the small interspacing distance between the two capsules disturbs the wake behind the leading capsule, which tends to mitigate its deformation and therefore decreases its surface area. Conversely, as the wake of the trailing capsule is unaffected, the discrepancies between its area and that of the single capsule are less pronounced.

We then present the time evolution of the velocity and area of the leading and the trailing capsules at various Ca , Re and d_0 . We first focus on the velocity of the capsules, displayed in figure 5.15 for $Ca = 0.15$ and 0.35 and for $d_0 = 0.5$ and 1 . The velocity of both capsules displays a minimum at $t = 0$ and a maximum at $t \approx 2$ at $Ca = 0.15$ and $Ca = 0.35$. The extrema of the velocity are more pronounced as Ca increases. The effects of the initial interspacing distance d_0 on these extrema are less evident but still present: the velocity maxima of both the leading and the trailing capsules are increased by about 1% as d_0 is halved from 1 to 0.5 . Interestingly, the relaxation time of V_t to V_{eq} is significantly reduced when compared to that of V_l : about 3 time units in the case of V_t with respect to more than 10 time units in the case of V_l . Capsule velocities in the inertial regimes at $Re = 25$ and 50 and at $Ca = 0.15$ and $Ca = 0.35$ are plotted in figure 5.15b and figure 5.15c for $d_0 = 1$ and 0.25 , respectively. The results are similar to that of the non-inertial regime: Ca enhances the velocity deviations and the extrema are more pronounced in the case of the trailing capsule. Surprisingly, we note that figure 5.15b and figure 5.15c display very similar behaviors: therefore, the interspacing distance does not seem to impact the capsule velocities

5.5. System of two capsules

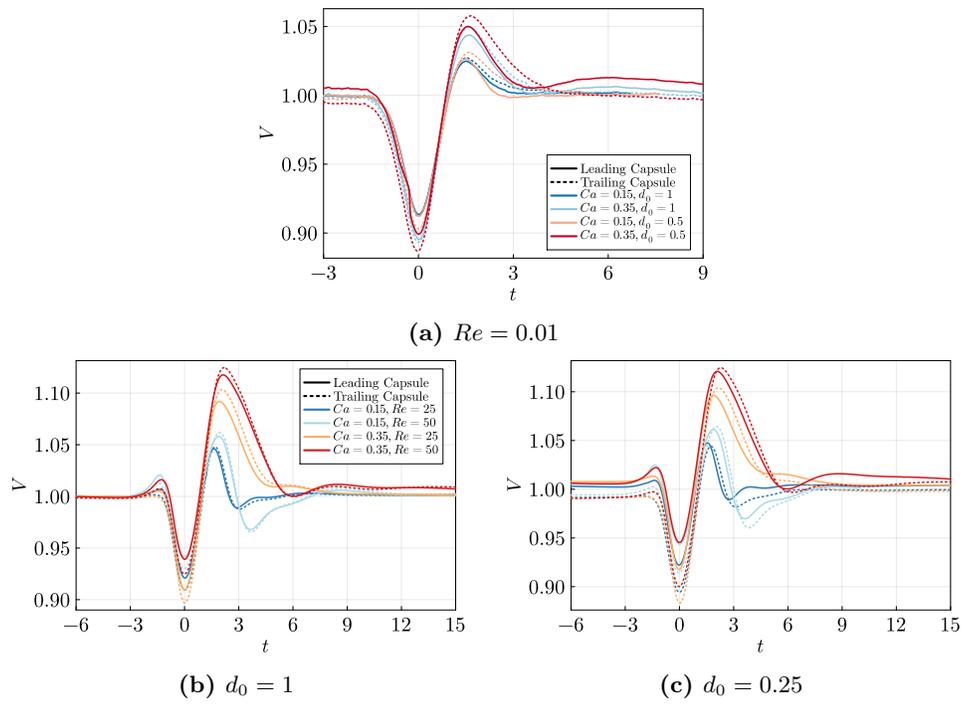


Figure 5.15: Temporal evolution of V of the leading and trailing capsules at different Ca , Re and d_0 .

5.5. System of two capsules

d_0		$Re = 0.01$	$Re = 25$	$Re = 50$
1	$Ca = 0.15$	1.065	1.138	1.193
	$Ca = 0.35$	1.263	1.334	1.399
0.5	$Ca = 0.15$	1.065	1.135	1.186
	$Ca = 0.35$	1.247	1.323	1.383
0.25	$Ca = 0.15$	1.068	1.129	1.180
	$Ca = 0.35$	1.236	1.308	1.379

Table 5.1: Maximum area \mathcal{A}_{max} of the leading capsule at different Ca , Re and d_0 .

d_0		$Re = 0.01$	$Re = 25$	$Re = 50$
1	$Ca = 0.15$	1.068	1.143	1.201
	$Ca = 0.35$	1.271	1.342	1.417
0.5	$Ca = 0.15$	1.070	1.144	1.204
	$Ca = 0.35$	1.277	1.345	1.414
0.25	$Ca = 0.15$	1.069	1.148	1.204
	$Ca = 0.35$	1.277	1.344	1.41

Table 5.2: Maximum area \mathcal{A}_{max} of the trailing capsule at different Ca , Re and d_0 .

inside the corner: its effects are bounded to the capsule velocities upstream and downstream from the corner. We will come back to this observation in Section 5.5.3.

When analyzing the capsule areas for varying Re , Ca and d_0 , a similar behavior is found: the area of the trailing capsule is consistently greater than that of the leading capsule, and increasing Capillary and Reynolds numbers and decreasing the initial interspacing distance enhance this phenomenon. In particular we report in Table 5.1 the maximum areas of the leading capsule and in Table 5.2 that of the trailing capsule. As can be seen from Table 5.1 and Table 5.2, the maximum area of the leading capsule exceeds that of the trailing capsule by up to 5%. The full time-dependant data is provided in Appendix D.1.

5.5.3 Time evolution of the interspacing distance

We now analyze the time evolution of the interspacing distance between the two confined capsules considered in this section. Figure 5.16 shows the time-

5.5. System of two capsules

dependent interspacing distance for $Ca = 0.15$ and 0.35 , $Re = 25$ and 50 and $d_0 = 1, 0.5$ and 0.25 . In this figure, we note that in all cases, the interspacing distance decrease immediately after the trailing capsule is released. This is due to the fact that upon release, the trailing capsule is spherical and therefore located farther away from the channel walls than is the leading capsule, resulting in its initial acceleration before a steady shape is found – typically within less than five time units. In the case where $d_0 = 1$, the interspacing distance d is steady until the leading capsule approaches the corner, reaches a minimum then a maximum value inside the corner and becomes steady again as the trailing capsule leaves the corner region. Interestingly, the steady interspacing distance after the corner is up to 10% greater than its steady value prior to the corner, suggesting that the corner separates the two capsules. Moreover, the initial interspacing distance is greater in the case $Re = 25$ than in the case $Re = 50$: this is only an artifact of our release mechanism. Indeed, the steady “parachute” shape of the capsule is deployed faster at $Re = 50$ than at $Re = 25$, leading to a shorter initial acceleration phase of the trailing capsule towards the leading capsule at $Re = 50$ than at $Re = 25$. When $d_0 = 0.5$ and $d_0 = 0.25$, we observe that the interspacing distance steadily increases until the capsules reach the corner region where it displays the same behavior as in the case of $d_0 = 1$, and continues to increase downstream of the corner. While a steady value of d is not clearly reached within the considered time range, we can extrapolate the trend and conclude that the interspacing distance seems to saturate to values ranging from 0.6 to 0.8 depending on Re , Ca and d_0 . Therefore, the pair of confined capsules we consider exhibit a minimum stable interspacing distance d_{min} . Moreover, we note that the slope of d is greater in the case of lower initial interspacing distances, suggesting that the relative velocity of the capsules is a function of their interspacing distance. To investigate further this behavior, we show in figure 5.17 the velocity of the two capsules at $Ca = 0.35$, $Re = 50$ and d_0 ranging from 0.25 to 1. We observe that the velocity of the trailing capsule is lower than that of the leading capsule prior to entering and downstream of the corner, and that the velocity difference increases with decreasing interspacing distance. This velocity difference confirms the above observations in terms of interspacing distance, in particular that a lower interspacing distance results in a greater relative velocity between the two capsules, i.e. an enhanced repulsive behavior. Moreover, we note in figure 5.17 that the difference in velocity minima between the leading and the trailing capsules is always greater than the difference between their velocity maxima. As a result, the residence time of the trailing capsule inside the corner region is always greater than that of

5.6. Train of ten capsules

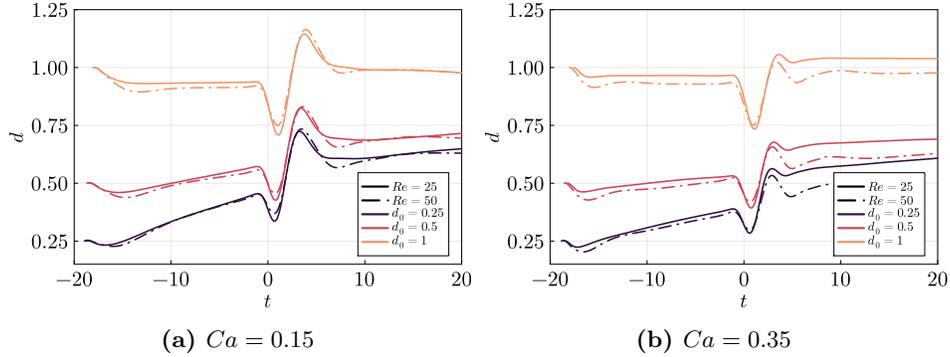


Figure 5.16: Temporal evolution of d for different initial interspacing distance d_0 and Reynolds number Re .

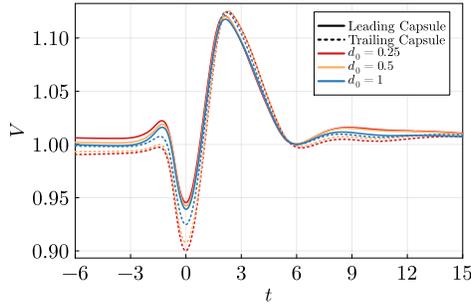


Figure 5.17: Effects of the initial interspacing distance d_0 on the evolution of the capsules velocities V at $Ca = 0.35$, $Re = 50$.

the leading capsule, and the corner tends to separate the pair of capsules. The present analysis of the binary interaction of capsules through a corner reveals that the two considered capsules do interact in this geometry, affecting their motion and deformation. In particular, the trailing capsule tends to be more deformed than the leading capsule, and the corner tends to separate the pair of capsules. A natural question that arises is that of the accumulation of such effects if more than two capsules are considered.

5.6 Train of ten capsules

In this last section, we investigate the behavior of a train of ten capsules flowing through the corner. We insert each capsule using the same procedure employed in the previous section: a new initially spherical capsule appears at a distance $D_c = 30$ radii from the corner as soon as the preceding capsule has

5.6. Train of ten capsules

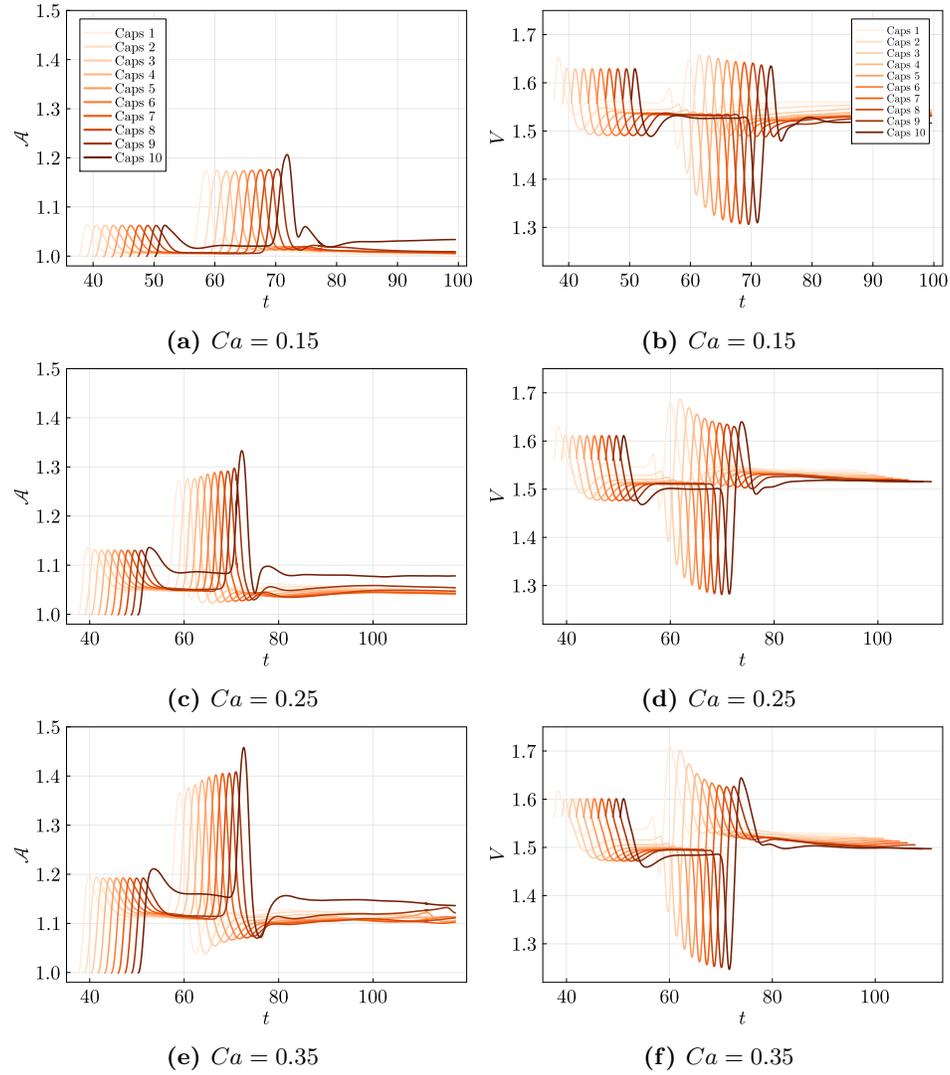


Figure 5.18: Time evolution of the reduced areas and velocities of ten capsules at $Re = 50$ and $d_0 = 1/8$ for $Ca = 0.15, 0.25$ and 0.35 .

5.6. Train of ten capsules

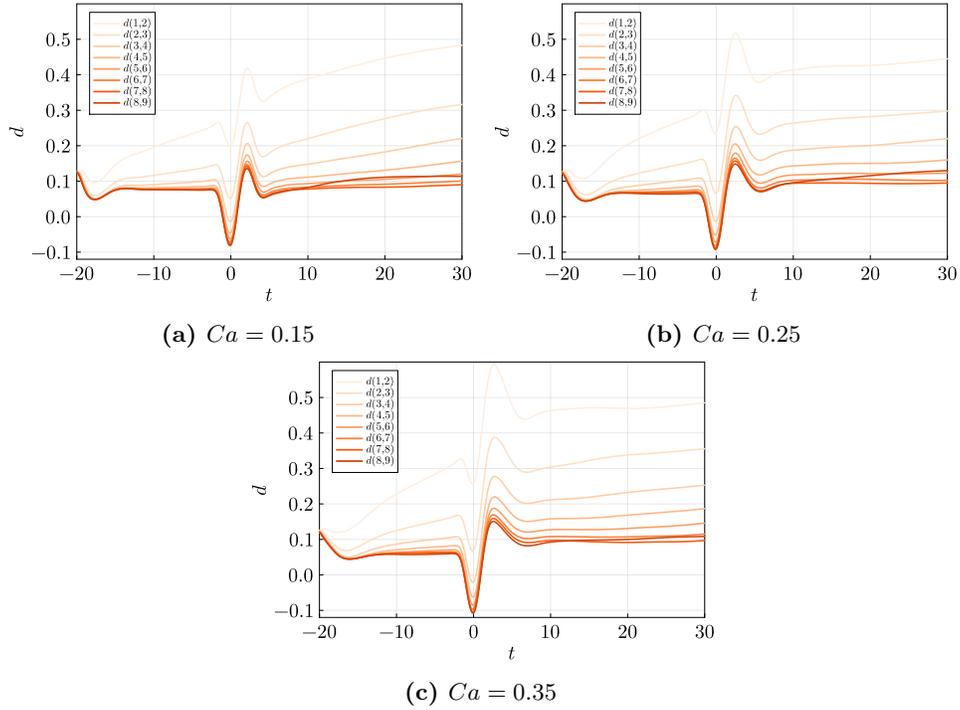


Figure 5.19: Temporal evolution of d for a train of 10 capsules at $Re = 50$ and $d_0 = 0.125$ for (a) $Ca = 0.15$, (b) $Ca = 0.25$ and (c) $Ca = 0.35$.

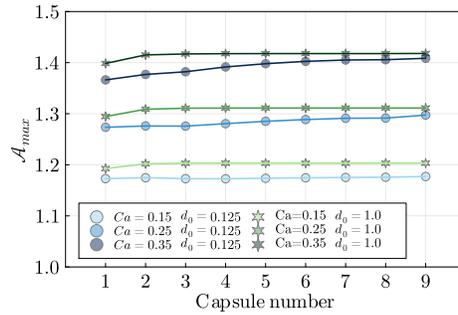


Figure 5.20: A_{max} as a function of the capsule number.

5.6. Train of ten capsules

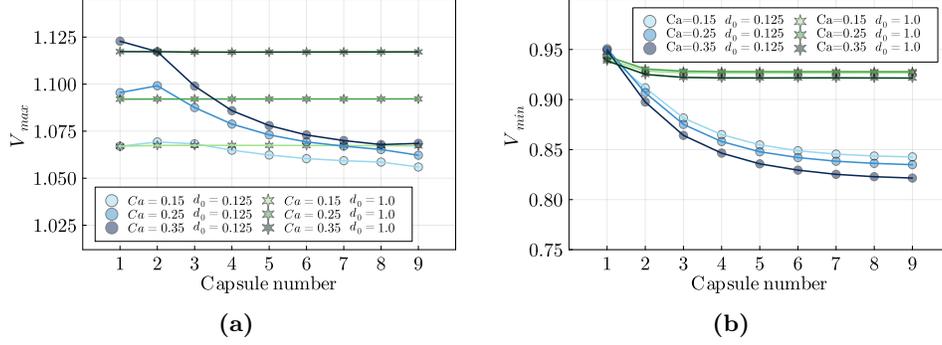


Figure 5.21: (a) V_{max} and (b) V_{min} as a function of the capsule number.

advanced by a reduced distance $\tilde{d} = 2\tilde{a}(1 + d_0)$. The capsules are removed from the computational domain when they are less than one initial diameter away from the outflow boundary. Our goal is to determine if the findings of the previous binary capsule analysis accumulate when more than two capsules are considered, especially with regards to the increased surface area of the capsules and the separating effect reported in Section 5.5. As such, we plot in figure 5.18 the normalized area and velocity of each capsule of the train at $Re = 50$, $d_0 = 0.125$ and Ca ranging from 0.15 to 0.35. The same figure obtained in the case of $d_0 = 1$ is provided in Appendix D.2. In figure 5.18, the darkness of the color corresponds to the position of the capsule in the train: darker means increasing capsule number i.e. further downstream along the capsule train. As mentioned in Section 5.5, the initial peaks in the area and velocity of the capsule are insertion artifacts and do not contribute to the physics that is the focus of this section. We observe in figure 5.18 that the behavior of the last capsule is significantly different than that of the rest of the train. In Section 5.5 we hypothesized that the difference in areas of the leading and the trailing capsules is due to the fact that the wake of the leading capsule is significantly affected by the presence of the trailing capsule. The present observation in figure 5.18 corroborates this statement: all of the capsules in the train see their wake affected by a trailing capsule, except in the case of the last capsule. As a result, its deformation is greater and extends closer to the channel walls, thus decreasing its velocity. We also remark in figure 5.18 that this effect is enhanced with increasing Ca . While noteworthy in the case of a pair of capsules, this effect is less pertinent to the study of a train of capsules, as only the core of the capsule train is relevant to typical microfluidic applications. As such, in the remainder of this section our analysis is focused on the first ninth capsules of the train.

5.6. Train of ten capsules

As expected, a steady state is reached in the straight channel prior to the corner for each capsule and for all Ca . While the steady area remains constant with increasing capsule number, i.e. as we move further downstream in the train of capsules, we observe that the velocity of the capsules decreases. In particular the difference between the steady velocity of the first and ninth capsules increases with increasing Ca . As the capsules enter the corner region, they display the familiar pattern previously described in Section 5.4 and Section 5.5, before relaxing to steady values. The shape of the deviation pattern is strikingly similar across different capsules of the train, regarding both the velocity and the area of the capsules, except that they are shifted in time and magnitude. More precisely, the area curves are shifted upwards with increasing capsule number while the velocity curves are shifted downwards with increasing capsule number. As a result, the maximum area of the capsule increases and the velocity extrema decrease with increasing capsule number. This behavior is more pronounced as Ca increases. Additionally, we compare in figure 5.19 the normalized interspacing distance d between each pair of capsules in the train. In figure 5.19, each curve is shifted in time such that $t = 0$ corresponds to d_{min} inside the corner. For all Ca , we observe that the interspacing distance $d(1, 2)$ between the first and the second capsules increases to a steady value close to 0.5, and that the corner has marginal effects on the downstream evolution of $d(1, 2)$: this behavior is identical to the case of two capsules studied in the previous section. However, as we move downstream in the train of capsules, d increases slower and slower prior to the corner until it remains constant for capsule numbers greater than 7, at a steady value $d \approx 0.7$ that decreases only marginally with increasing Ca . After the transient regime due to the corner, $d(i, i + 1)$ for capsule numbers i greater than 7 reaches a steady state that is slightly higher than prior to entering the corner. In other words, the corner tends to increase the interspacing distance, and therefore exhibits a separating effect. This separating effect is observed regardless of the initial interspacing distance d_0 , as was the case in the previous section when only two capsules were considered.

Finally, in order to investigate further the influence of the capsule number on the capsule dynamics, we plot in figures 5.20-5.21 the maximum area as well as the maximum and minimum velocities of each capsule of the train for varying Capillary numbers and interspacing distances. The difference in minimum velocity (respectively, maximum velocity) between the first and the ninth capsule is about 15% (respectively, about 7%) at $Ca = 0.35$ while it is about 11% (respectively, 2%) at $Ca = 0.15$. Similarly, the difference in maximum area between the first and the ninth capsule is about 4% at

5.7. Conclusion

$Ca = 0.35$ and less than 1% at $Ca = 0.15$. These results correspond to $d_0 = 0.125$, while in the case of $d_0 = 1$ only deviations lower than 1% are observed in the extrema of the capsule area and velocity (except in the case of $Ca = 0.35$ for which velocity deviations of 2% are observed). The very small deviations observed in the case $d_0 = 1$ indicates that for this interspacing distance the capsules interact very weakly. As such, there exist a critical interspacing distance d_c below which capsule interactions are observed, with $0.125 < d_c < 1$.

The fact that d_c is less than 1 can be surprising, as a normalized interspacing distance of $d_0 = 1$ would typically be classified as a strongly interacting regime in other geometries, e.g. in the T-junction investigated by Lu et al. [129]. The main reason for the low interaction we observed is likely due to the short residence time of the capsules in the corner region. Indeed, Lu et al. showed that the residence time is determinant in the path selected by the capsules in a T-junction geometry. Another reason for such a low critical interspacing distance is related to the very confined configuration we study: the capsule shape and behavior is primarily due to the presence of the walls, while the small disturbances of the flow field due to the other capsules only marginally contribute to each capsule dynamics. Future studies could explore the dynamics of a train of capsules in a wider channel, i.e. in a less confined configuration, where each capsule could be more influenced by the wake disturbances of their preceding neighbor.

5.7 Conclusion

In the present work, the inertial and non-inertial dynamics of three-dimensional elastic capsules flowing through a sharp corner are investigated. The capsule trajectory, area, velocity and membrane stress are analyzed in the cases of one, two and a train of ten capsules released upstream of the corner. The channel Reynolds number ranges from 0.01 to 50, the Capillary number representing the ratio of viscous stresses over elastic stresses ranges from 0.075 to 0.35 and the initial normalized interspacing distance between two capsules is varied from 1 to 0.125. The goal of this study is to help provide practical guidelines in order to anticipate capsule breakup and estimate throughput in inertial microchannels.

The case of a single capsule with no inertia was previously studied by Zhu & Brandt [221], who reported that the capsule follows the flow streamlines closely regardless of the Capillary number. In inertial flows, we found that this statement is still valid for all considered Reynolds and Capillary

5.7. Conclusion

numbers. As the streamlines of the inertial flow cross the centerline of the secondary channel – the horizontal channel downstream of the corner –, the capsule position is increasingly close to the top wall for increasing Reynolds number, especially in the case of large Capillary numbers. However no collision between the capsule and the wall of the secondary channel was observed thanks to strong lubrication forces. In their study, Zhu & Brandt also analyzed the velocity of the capsule centroid and the area of the capsule membrane: they found that the capsule velocity decreases in the corner and increases immediately after the corner, with an overshoot increasing with membrane deformability. The area of the capsule was also found to also reach a maximum slightly shifted in time with respect to the minimum of velocity. In the inertial regime, we observed that this behavior is enhanced as the Reynolds number increases. However our results at $Re = 1$ do not differ significantly from results obtained in the non-inertial regime, which corroborates the same observation that was made by Wang et al. [207, 208]. Moreover, at sufficiently high inertia, capsule areas lower to equilibrium areas are observed as the capsule relaxes to its steady state. In other words, immediately after the corner the capsule oscillates around its steady shape. This phenomenon is enhanced as the Capillary number increases. Additionally, we reported that the relation between the maximum area \mathcal{A}_{max} of the capsule and the Reynolds number is linear as long as the Capillary number is kept below 0.35. At $Ca = 0.35$, the relation between \mathcal{A}_{max} and Re is not perfectly linear and the curve $\mathcal{A}_{max}(Re)$ is slightly concave. Moreover, from $Re = 1$ to $Re = 50$, the maximum area increases nearly linearly over the full range of Ca . At $Ca = 0.35$, we compared the membrane stress to the capsule area and found that (i) the time evolution of the average stress presents a strong correlation to that of the membrane area, and (ii) in our configuration, the value of the maximum stress is double that of the average stress. As a result, observing the capsule area experimentally can provide reliable insight into the average stress as well as an estimate of the capsule maximum stress. This finding is of primary importance in the design of microfluidic devices where capsule breakup is to be avoided, as well as in the development of targeted drug delivery methods for which a controlled capsule breakup is sought.

We then investigated the interaction of several capsules in the corner geometry. First, two capsules are considered with varying initial interspacing distances. Similar to the case of a single capsule, neither the trajectory of the leading nor of the trailing capsule is observed to significantly deviate from the flow streamlines. However the velocity of the trailing capsule is found to be lower than that of the leading capsule as well as that of a single

5.7. Conclusion

capsule at the same Reynolds and Capillary numbers. Similarly, the velocity of the leading capsule is greater than that of a single capsule in the same conditions. In contrast, inside the corner the area of the trailing capsule is found to be larger than that of the leading capsule and of the single capsule in the same conditions. However, in the configuration we consider where confinement is strong, the magnitude of these effects is small even for capsules located very close to each other: the velocity of the leading and trailing capsules only deviates by a few percents from that of a single capsule. Then, the time evolution of the interspacing distance d between the pair of capsules was analyzed. Interestingly, we found that capsules initially located at $d_0 \leq 0.5$ or less tend to separate. This suggests that there exists a minimum stable gap d_{min} between two confined capsules. A systematic analysis of this effect is left for future studies. Next, we examined the case of a train of capsules and sought to determine whether the effects observed with a pair of capsule accumulate. While no interaction occurs for a large initial interspacing distance $d_0 = 1$, we found that in the case $d_0 = 1/8$, the steady and extremum areas of the trailing capsules increase by up to 5% and eventually saturate at the tail of the train, around the ninth capsule. In all cases, the corner is found to separate the pair of capsules as well as the capsule train, which can be further evidenced from the analysis of the time evolution of the capsule velocities inside the corner region.

We believe that the present work is a step forward towards providing practical guidelines to avoid capsule breakup in inertial and non-inertial microfluidic experiments. Future works could study capsule membranes exhibiting a strain-hardening elastic behavior, e.g. as described by the Skalak law [184], as well as vary the confinement ratio $\beta = 2\tilde{a}/\tilde{W}$ in order to consider high-throughput microfluidic devices. In the case of lower confinement ratios in particular, we expect to see stronger capsule interactions along with cross-stream capsule migration inside and downstream of the corner. Finally, the present work could also be useful to develop membrane characterization techniques, where viscoelastic membrane properties could be inferred from the time-dependant evolution of a capsule of interest through a corner.

Chapter 6

Towards a fully Eulerian adaptive solver for immersed elastic capsules

6.1 Introduction

In the previous chapter, the front-tracking method was presented together with its Lagrangian discretization of the capsule. This triangulation allowed to compute membrane-related quantities such as strains and stresses in a relatively straightforward fashion and to a high level of accuracy. However, the introduction of the membrane triangulation comes at the expense of communications between the Lagrangian mesh and the Eulerian mesh, achieved by the previously introduced immersed boundary method. The consequences of the back and forth interpolations between the two meshes are multiple. First, the global spatial order of accuracy of the method is reduced to one in the general case. Second, an efficient parallelization of the FTM on adaptive grids is difficult due to the non-trivial Eulerian domain decomposition leading to many IBM stencils existing on several processors, as discussed in Section 4.6. Another limitation of the FTM is of physical consideration: unlike other Eulerian methods, the FTM does not conserve the volumes of the inner and outer fluids to the machine precision, but rather asymptotically. This means that after a very long simulation time, the object of interest represented by the FTM – such as a bubble, a droplet or a capsule – could gain or lose a significant fraction of its initial volume and mass.

The field of interfacial dynamics, which includes the FTM, is home to a second class of methods that don't share the same limitations. In these so-called Eulerian methods, the immersed object is described using a scalar field. In the Volume Of Fluid (VOF) method introduced by Hirt & Nichols [78] and popularized by Brackbill et al. [25], the field c is defined as the volume fraction of the inner phase. In the level-set method introduced by Sussman et al. in the context of droplet dynamics [191] and later extended

to fluid-structure interaction by Gibou et al. [60, 61], the field ϕ is defined such that the interface is exactly the contour-line $\phi = 0$. In both methods, the field c or ϕ is advected using the local velocity field: in the VOF method in a conservative fashion; in the level-set method in a non-conservative fashion. In the case of the VOF method, differentiating the interface geometry – and in fact, finding the precise location of the interface itself – is a complex matter involving a temporary hyperbolic tangent-shaped smoothing of the volume fraction field c , or so-called height functions computed on large stencils and notoriously difficult to implement [151, 153, 171]. The level-set method, however, has the advantage of providing an analytical expression for the interface, thus greatly reducing the challenge of computing derivatives of the geometry needed to express interfacial stresses.

While the VOF and the level-set methods were initially introduced to simulate surface tension-driven flows, both have been extended to immersed capsules and vesicles, and even to fluid-structure interaction problems involving immersed elastic solids. Regarding the latter application, Sugiyama et al. [189] introduce a method relying on VOF framework, while Kamrin et al. [97, 103, 104, 196] introduce the Reference Map Technique (RMT) based on the level-set formulation. In both methods, the elastic strains are accessed by advecting a field quantity: the left Cauchy-Green deformation tensor \mathbf{B} in the work of Sugiyama et al., and the so-called “reference map” $\boldsymbol{\xi}$ in the work of Kamrin et al. Loosely speaking, the role of \mathbf{B} and $\boldsymbol{\xi}$ is to describe by how much the configuration of the elastic solid has deviated from its initial strain-free configuration, at each instant t . This information is then used in the constitutive elastic law of the deformable solid in order to compute the stresses. The main challenge for both methods is to limit numerical diffusion during the advection of \mathbf{B} or $\boldsymbol{\xi}$, otherwise the elastic solid can forget some of its initial strain-free configuration. This problem is handled by using high-order WENO (Weighted Essentially Non Oscillatory) schemes in Sugiyama’s method; and in the RMT, by creating a ghost region into the fluid where the solid strains are extended. With regards to Eulerian formulations applied to immersed capsules and vesicles, Cottet & Maître present two-dimensional simulations of elastic capsules based on the level-set method [37, 38]. In their formulation, the elastic and the bending forces are derived from energy principles, and their pioneering work has paved the way for the emergence of the methods of Sugiyama and Kamrin for immersed elastic solids. In the early 2010, Ii et al. extended Sugiyama’s VOF method to consider an elastic capsule as opposed to a three-dimensional solid [92–94]. They were able to quantitatively reproduce classical results of capsule dynamics, such as the deformation of an elastic capsule in a shear

flow that we presented in figure 4.5. Moreover, they demonstrated that their fully Eulerian solver scales very well with the number of capsules and Eulerian grid cells, by simulating $5 \cdot 10^6$ red blood cells on over 600,000 processors [188]. Their work demonstrates that the fully Eulerian formulation is a viable description of biological capsules. Moreover, at the price of a more challenging differentiation of the membrane geometry and of a slight diffusion of its strains, using a fully Eulerian method based on the VOF formulation has the potential to: (i) describe the capsule in a conservative fashion, (ii) require lower implementation efforts due to the absence of membrane triangulation, and (iii) provide an elegant and efficient parallelization paradigm, as only field quantities described on the Eulerian grid are communicated between processors. Moreover, even though the convergence of the implementation of Ii et al. was first order in space [92], the fully Eulerian formulation does not present an order degenerance due to the IBM, and there is no theoretical argument *a priori* forbidding this formulation to converge with second-order accuracy.

In light of the strengths of the fully Eulerian formulation based on the VOF framework, we present in this chapter an attempt to reproduce the results from Ii et al. [92] on an adaptive grid in the Basilisk platform. In Section 6.2 we describe the fully Eulerian formulation of the fluid-membrane problem, with an emphasis on the computation of the membrane strains and stresses. Then, we present our implementation strategy in Section 6.3 in order to supplement Basilisk with the fully Eulerian membrane solver. Validation cases are shown in Section 6.4. We conclude by summarizing our results and discussing their implications in Section 6.5.

6.2 Governing equations

6.2.1 Eulerian framework

Similar to the front-tracking method previously introduced in Chapter 4, in the Eulerian framework the one-fluid Navier-Stokes equations Eq. (4.1–4.2) are solved, where the body force \mathbf{f}_b in Eq. (4.1) accounts for the membrane stresses exerted on the fluid. In the front-tracking method, the body force \mathbf{f}_b is computed by regularizing point forces located on the nodes of a Lagrangian triangulation of the membrane. The Lagrangian discretization provides a convenient framework to access the strains of the elastic membrane, and to compute its elastic and bending stresses. Unlike the Lagrangian approach, the Eulerian method carries all computations on the structured Eulerian grid, including the elastic strains. In this section we describe the Eulerian

framework and introduce the corresponding equations.

In the absence of nodes, edges and triangles, the membrane shape is described using an indicator function c , called “Volume of Fluid function” – often abbreviated “VOF function” – or simply “color function”. This function was introduced by Brackbill et al. [25] and is defined to be 0 in the fluid grid cells located outside the membrane and 1 in the fluid grid cells located inside the membrane. Brackbill et al. initially defined $c = 1/2$ in the cells cut by the interface – the elastic membrane in our case – but this method was later improved by defining c as the volume fraction of the inner fluid.

$$\begin{cases} c = 0 & \text{outside the capsule} \\ c = 1 & \text{inside the capsule} \\ 0 < c < 1 & \text{in interfacial cells.} \end{cases} \quad (6.1)$$

The scalar field c thus tags the location of the capsule in an Eulerian fashion, and the location of the membrane can be inferred from the fluid grid cells where $0 < c < 1$. The VOF function is useful to define fluid properties that jump across the membrane, such as the viscosity field using an arithmetic average as follows:

$$\mu = c\mu_i + (1 - c)\mu_e \quad (6.2)$$

where μ_i and μ_e are the constant viscosities in the interior and the exterior of the capsule, respectively. In time-dependent problems, the VOF function is advected as a passive tracer:

$$\frac{\partial c}{\partial t} + \mathbf{u} \cdot \nabla c = 0, \quad (6.3)$$

however the numerical scheme to solve Eq. (6.3) should be chosen with care in order to minimize diffusion. For this reason, geometric schemes are popular to advect the VOF function since they essentially eliminate numerical diffusion [179].

In the Eulerian framework describing elastic membranes, it is useful to introduce a neighborhood around the membrane in order to define membrane-specific quantities on the Eulerian grid. An indicator function $\tilde{\Gamma}$ can be introduced from the VOF function in order to formally define this neighborhood:

$$\tilde{\Gamma} = \begin{cases} 1 & \text{if } |\nabla \tilde{c}| > 0 \\ 0 & \text{otherwise,} \end{cases} \quad (6.4)$$

where \tilde{c} can be the VOF function c itself or a mollified version of it, depending on the desired thickness of the membrane neighborhood. An example of

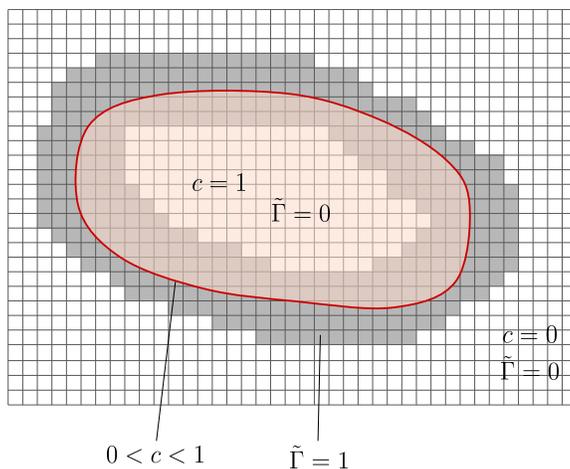


Figure 6.1: Schematic of the Eulerian description of a capsule: the red line denotes the location of the membrane, located in cells where $0 < c < 1$; while the shaded area denotes the location of the membrane neighborhood corresponding to $\tilde{\Gamma} = 1$.

the Eulerian description of the membrane and its neighborhood is sketched in figure 6.1. In the following, we refer to $\tilde{\Gamma}$ as “membrane neighborhood” or “membrane region” indistinctively.

6.2.2 Membrane mechanics

Within the context of the Eulerian framework, we present in this section the expression of the body force term \mathbf{f}_b appearing in Eq. (4.1) and representing the action exerted by the membrane onto its surrounding fluid. The following equations are introduced by Li et al. [92–94] based on the original work of Barthès-Biesel & Rallison [21]. We remind the reader that the equations in Section 6.2.2 pertaining to the membrane mechanics are only valid in three dimensions, i.e. for two-dimensional membranes. However, for code development and validation purposes, it is useful to implement a two-dimensional model in the first place, where the membrane is one-dimensional. As such, we present in Section 6.2.2 a formulation of the membrane mechanics valid for one-dimensional membranes embedded in a two-dimensional space. To our knowledge, there is no such formulation in the literature, despite its importance for the development and validation stages of the numerical solver.

Standard formulation

Similar to the Lagrangian description introduced in Chapter 4, the membrane strains are described using the surface deformation gradient tensor

$$\mathbf{F}_s = \mathbf{P} \cdot \mathbf{F} \cdot \mathbf{P}_R, \quad (6.5)$$

where \mathbf{P} and \mathbf{P}_R are projectors onto the current and reference configurations of the membrane, respectively. The tensor \mathbf{F} stores the deformation of the current configuration \mathbf{x} with respect to the reference configuration \mathbf{X} of a three-dimensional elastic solid:

$$\mathbf{F} = \partial \mathbf{x} / \partial \mathbf{X}. \quad (6.6)$$

Because we only consider membrane elasticity as opposed to the elasticity of a three-dimensional solid, in the following we omit the subscript s and we denote the surface deformation gradient by \mathbf{F} . This tensor has no meaning far away from the membrane and is therefore defined only in the membrane region $\tilde{\Gamma}$.

In the Eulerian description, the membrane strains are computed by advecting the left surface Cauchy-Green deformation tensor \mathbf{B}

$$\mathbf{B} = \mathbf{F} \cdot \mathbf{F}^T, \quad (6.7)$$

using the advection equation

$$\frac{\partial \mathbf{B}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{B} = \mathbf{B} \cdot \nabla_s \mathbf{u} + (\nabla_s \mathbf{u})^T \cdot \mathbf{B}, \quad (6.8)$$

where $\nabla_s = \mathbf{P} \cdot \nabla$ on the right-hand side is the surface gradient operator. If the membrane is in a known state at $t = 0$ – e.g. at $t = 0$ the membrane is in its strain-free configuration $\mathbf{B} = \mathbf{I}$ – Eq. (6.8) fully describes the strains of the membrane for all $t \geq 0$.

In order to express the membrane load \mathbf{f}_b we first need to define the elastic stress tensor $\boldsymbol{\sigma}_e$ as a function of the strain. To this end, we follow Barthès-Biesel & Rallison [21]: knowing that the stress tensor has a zero eigenvalue in the direction \mathbf{n} normal to the membrane, then the other two non-zero eigenvalues – the principal stresses $\sigma_{1,2}$ – in the directions $\mathbf{e}_1, \mathbf{e}_2$ are expressed as

$$\sigma_i = \frac{1}{\lambda_1 \lambda_2} \lambda_i \frac{\partial W}{\partial \lambda_i} \quad i \in \{1, 2\} \quad (\text{no summation}). \quad (6.9)$$

6.2. Governing equations

From the principal stresses, we can express the stress tensor $\boldsymbol{\sigma}$ using the rotation matrix \mathbf{R} from the canonical basis to the shared eigenbasis $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{n})$ of $\boldsymbol{\sigma}$, \mathbf{B} and \mathbf{F} :

$$\begin{aligned}\boldsymbol{\sigma}_e &= e^{-a} \mathbf{R} \cdot \left(\frac{\partial W}{\partial a} (\mathbf{e}_1 \mathbf{e}_1 + \mathbf{e}_2 \mathbf{e}_2) + \frac{\partial W}{\partial b} (\lambda_1^2 \mathbf{e}_1 \mathbf{e}_1 + \lambda_2^2 \mathbf{e}_2 \mathbf{e}_2) \right) \cdot \mathbf{R}^T \\ &\Leftrightarrow \boldsymbol{\sigma}_e = e^{-a} \left(\frac{\partial W}{\partial a} \mathbf{P} + \frac{\partial W}{\partial b} \mathbf{B} \right),\end{aligned}\quad (6.10)$$

where the two invariants a, b have been introduced:

$$\begin{cases} a = \log(\lambda_1 \lambda_2) = \frac{1}{2} \log \left(\frac{1}{2} \text{Tr}(\mathbf{B})^2 - \frac{1}{2} (\text{Tr}(\mathbf{B}^2)) \right) \\ b = \frac{1}{2} (\lambda_1^2 + \lambda_2^2) = \frac{1}{2} \text{Tr}(\mathbf{B}) - 1. \end{cases}\quad (6.11)$$

For simplicity, in this chapter we only consider the neo-Hookean law, which strain energy function is defined in Eq. (4.6). The elastic stress tensor $\boldsymbol{\sigma}_e$ can then be expressed solely in terms of the left surface Cauchy-Green deformation tensor \mathbf{B} and its invariants, as well as the surface projector \mathbf{P} :

$$\boldsymbol{\sigma}_e = \frac{E_s}{3} e^{-a} (\mathbf{B} - e^{-2a} \mathbf{P})\quad (6.12)$$

If a bending stress is considered, we follow the bending model of Pozrikidis [157]. In this model, the bending stress $\boldsymbol{\sigma}_b$ is expressed as a function of the transverse shear tension \mathbf{q} and the normal vector \mathbf{n}

$$\boldsymbol{\sigma}_b = \mathbf{q} \mathbf{n},\quad (6.13)$$

with \mathbf{q} a function of the bending moments \mathbf{m}

$$\mathbf{q} = (\nabla_s \cdot \mathbf{m}) \mathbf{P}.\quad (6.14)$$

Pozrikidis chooses a linear bending model and expresses the bending moments \mathbf{m} as follows:

$$\mathbf{m} = E_b (\boldsymbol{\kappa} - \kappa_R \mathbf{P}),\quad (6.15)$$

where

$$\boldsymbol{\kappa} = -\nabla_s \mathbf{n} = -\nabla \mathbf{n}\quad (6.16)$$

is the so-called ‘‘Cartesian curvature tensor’’, and

$$\kappa_R = -\text{Tr}(\nabla \mathbf{n})/2\quad (6.17)$$

is the reference curvature. This linear bending model is equivalent to the Helfrich bending model in the limit of small deformations [71].

Finally, the body force \mathbf{f}_b is expressed as the surface divergence of the stress tensor localized around the membrane:

$$\mathbf{f}_b = |\nabla \tilde{c}| \nabla_s \cdot (\boldsymbol{\sigma}_e + \boldsymbol{\sigma}_b).\quad (6.18)$$

Expressing the surface Jacobian

Following Ii et al. [93, 94], the numerical stability of the solver can be improved by removing the dependency of \mathbf{B} on the surface Jacobian J of the membrane. The surface Jacobian represents the local ratio of current area over the reference area, and is expressed as:

$$J = \lambda_1 \lambda_2 = \sqrt{\frac{1}{2} (\text{Tr}(\mathbf{B})^2 - \text{Tr}(\mathbf{B}^2))}. \quad (6.19)$$

While J could be computed from \mathbf{B} at every time step, we follow Ii et al. [94] and advect J on its own, following:

$$\frac{\partial J}{\partial t} + \mathbf{u} \cdot \nabla J = (\nabla_s \cdot \mathbf{u})J. \quad (6.20)$$

Consequently, the modified left Cauchy-Green deformation tensor \mathbf{G} is introduced, as well as its advection equation:

$$\mathbf{G} = \mathbf{B}/J \quad (6.21)$$

$$\frac{\partial \mathbf{G}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{G} = \mathbf{G} \cdot \nabla_s \mathbf{u} + (\nabla_s \mathbf{u})^T \cdot \mathbf{G} - (\nabla_s \cdot \mathbf{u})\mathbf{G}. \quad (6.22)$$

Noting that $e^a = J$, the expression of the elastic stress tensor σ_e derived in Eq. (6.12) for the neo-Hookean elastic law becomes

$$\sigma_e = \frac{E_s}{3} \left(\mathbf{G} - \frac{1}{J^3} \mathbf{P} \right). \quad (6.23)$$

Simplification in the case of one-dimensional membranes

The case of two-dimensional simulations leads to a simplification of the above expression of the elastic stress tensor. Indeed, if the membrane is one-dimensional, there exists only one principal stretch λ_1 . Moreover, by definition, we have

$$J \equiv \lambda_1 \quad (6.24)$$

and

$$\mathbf{B} = \begin{bmatrix} \lambda_1^2 & \\ & 0 \end{bmatrix} = J^2 \mathbf{P}. \quad (6.25)$$

Therefore, the elastic stress tensor σ_e can be expressed in terms of the surface Jacobian J and the projector \mathbf{P} only; and neither \mathbf{B} nor \mathbf{G} need to be considered.

Ensuring in-plane strains and stresses

In order to enforce zero normal elastic strains and stresses, additional constraints are placed on \mathbf{G} and $\boldsymbol{\sigma}_e$. At all time in the simulation, the following constraints need to be verified, and are manually enforced at every time step:

$$\mathbf{n} \cdot \mathbf{G} = \mathbf{G} \cdot \mathbf{n} = 0 \quad \iff \quad \mathbf{G} = \mathbf{P} \cdot \mathbf{G} \cdot \mathbf{P}, \quad (6.26)$$

$$\mathbf{n} \cdot \boldsymbol{\sigma}_e = \boldsymbol{\sigma}_e \cdot \mathbf{n} = 0 \quad \iff \quad \boldsymbol{\sigma}_e = \mathbf{P} \cdot \boldsymbol{\sigma}_e \cdot \mathbf{P}. \quad (6.27)$$

6.2.3 Full system of equations

In summary, we solve the incompressible, Newtonian Navier-Stokes equations coupled with the Eulerian membrane mechanics framework presented above. The thirteen equations and definitions describing this problem are provided below:

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \nabla \cdot (\mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)) + \mathbf{f}_b \quad (6.28)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (6.29)$$

$$\frac{\partial c}{\partial t} + \mathbf{u} \cdot \nabla c = 0 \quad (6.30)$$

$$\mu = c\mu_i + (1 - c)\mu_e \quad (6.31)$$

$$\mathbf{f}_b = |\nabla \tilde{c}| \nabla_s \cdot (\boldsymbol{\sigma}_e + \mathbf{q}\mathbf{n}) \quad (6.32)$$

$$\boldsymbol{\sigma}_e = \frac{E_s}{3} \left(\mathbf{G} - \frac{1}{J^3} \mathbf{P} \right) \quad (6.33)$$

$$\mathbf{P} = \mathbf{I} - \mathbf{n}\mathbf{n} \quad (6.34)$$

$$\mathbf{n} = \nabla c \quad (6.35)$$

$$\mathbf{q} = E_b (\nabla_s \cdot (\boldsymbol{\kappa} - \kappa_R \mathbf{P})) \mathbf{P} \quad (6.36)$$

$$\boldsymbol{\kappa} = -\nabla \mathbf{n} \quad (6.37)$$

$$\frac{\partial J}{\partial t} + \mathbf{u} \cdot \nabla J = (\nabla_s \cdot \mathbf{u})J \quad (6.38)$$

$$\frac{\partial \mathbf{G}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{G} = \mathbf{G} \cdot \nabla_s \mathbf{u} + (\nabla_s \mathbf{u})^\top \cdot \mathbf{G} - (\nabla_s \cdot \mathbf{u}) \mathbf{G}. \quad (6.39)$$

$$\frac{\partial \kappa_R}{\partial t} + \mathbf{u} \cdot \nabla \kappa_R = 0 \quad (6.40)$$

$$\mathbf{n} \cdot \mathbf{G} = \mathbf{G} \cdot \mathbf{n} = 0 \quad (6.41)$$

6.3 Numerical method and implementation strategy

6.3.1 Definition of the membrane region $\tilde{\Gamma}$

Our first step is to define a neighborhood of the membrane. Since the membrane is located in grid cells where the VOF function has a fractional value, $0 < c < 1$, a neighborhood of width Δx could be simply be defined using the indicator function

$$\Gamma = \begin{cases} 1 & \text{if } 0 < c < 1 \\ 0 & \text{otherwise.} \end{cases} \quad (6.42)$$

In practice, finite differences and finite volume operations will be performed in this neighborhood, and Γ needs to be over three grid cells-wide. Therefore, we define the mollified function \tilde{c} by averaging c from its neighbors. In practice, we repeat this averaging twice, and the resulting membrane region $\tilde{\Gamma}$ defined in Eq. (6.4) is 5 to 7 grid cell-wide, which is enough to allow several differentiations of the membrane quantities. This operation is done at every time step, right after the advection of the VOF function.

6.3.2 Invariance along the normal direction

In the membrane region $\tilde{\Gamma}$, surface quantities such as J , \mathbf{G} and $\boldsymbol{\sigma}_e$ are extended to the three-dimensional space. This is necessary to be able to differentiate these quantities on a structured Cartesian grid, and it is the

core of the Eulerian method. Of course, none of these surface quantities physically exist away from the membrane: their values in $\tilde{\Gamma}$ are extrapolations from the membrane surface. As such, for any surface quantity q , we require:

$$\begin{cases} \frac{\partial q}{\partial \mathbf{n}} = 0 & \text{in } \tilde{\Gamma} \\ q = 0 & \text{everywhere else,} \end{cases} \quad (6.43)$$

where \mathbf{n} is the vector normal to the membrane.

In order to enforce Eq. (6.43), an elegant idea introduced by Chen et al. [30] and Peng et al. [146] – both improving the level-set method for surface tension-driven flows – is to use a PDE approach. In this approach a quantity q known at the interface is extended outward using the following advection equation:

$$\frac{\partial q}{\partial \tau} + S(\mathbf{x})\mathbf{n} \cdot \nabla q = 0, \quad S(\mathbf{x}) = \begin{cases} 1 & \text{if } c = 1 \\ 0 & \text{if } c \in (0, 1) \\ -1 & \text{if } c = 0 \end{cases}, \quad (6.44)$$

where τ is a pseudo time and the normal vector $S(\mathbf{x})\mathbf{n}$ acts as a propagation velocity of magnitude 1, since $|\mathbf{n}| = 1$, with $S(\mathbf{x})$ ensuring q is propagated away from the membrane. Moreover, if the resolution of Eq. (6.44) is performed using an upwind scheme, there is no need to specify boundary conditions nor an initial guess for q away from the interface, as the information is strictly propagated away from the interface along the normal direction. Our Eulerian method relies on Eq. (6.44) to extend J , κ_R as well as all the components of \mathbf{G} , $\boldsymbol{\sigma}_e$, $\boldsymbol{\kappa}$ and even the normal vector itself \mathbf{n} , which is known to greater accuracy on the interface. In the latter case, an initial guess is necessary away from the interface, and we choose the normalized gradient of the smoothed volume fraction field $\mathbf{n}_0 = \nabla \tilde{c} / |\nabla \tilde{c}|$. The upwind solver iterates as long as the extended quantity q is converged to a desired criterion, or until reaching a maximum number of iterations.

6.3.3 Advection equations

We now focus on the numerical advection of the VOF function c , the surface Jacobian J , the surface left Cauchy-Green deformation tensor \mathbf{G} and the reference curvature κ_R . First, the VOF function needs special treatment as a desirable property of the numerical advection of c is to conserve volume to machine precision. To this end, a geometric advection scheme is used, and was already implemented and validated in Basilisk at the beginning of this study [153].

Regarding the advection of J , \mathbf{G} and κ_R , limiting numerical diffusion during the advection step is not as critical as in the case of the VOF function thanks to the extension algorithm introduced in Section 6.3.2. Indeed, a consequence of the extension of J , \mathbf{G} and κ_R is to overwrite the cells in the outer membrane region to the value of the nearby interfacial cells, effectively suppressing numerical diffusion in the outer membrane region. It is a similar approach to that employed by Jain et al. for their improved RMT [97]. To numerically solve Eq. (6.38–6.40), we choose the upwind advection scheme introduced by Bell et al. [23] and already implemented in Basilisk to compute the advective term of the Navier-Stokes equation. This scheme solves

$$\frac{\partial f}{\partial t} + \nabla \cdot (f\mathbf{u}) = S(f, \mathbf{x}, \mathbf{u}, J, \mathbf{G}), \quad (6.45)$$

where f is a scalar field and S is a source term possibly depending on f , \mathbf{x} , \mathbf{u} and even J and \mathbf{G} . The form of Eq. (6.45) is equivalent to that of Eq. (6.38–6.40) thanks to the incompressibility condition of the fluid. In practice, Eq. (6.45) is applied simultaneously to J , κ_R and all of the components of \mathbf{G} , while all the source terms are pre-computed before the simultaneous advectations begin.

6.3.4 Face-centered acceleration term

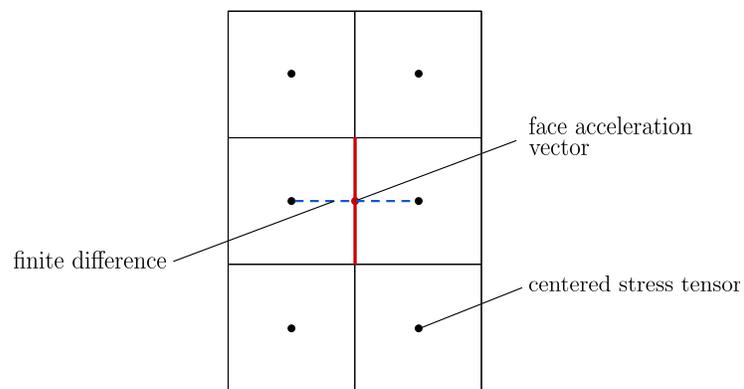
Once the surface quantities are advected, the last step of the Eulerian capsule solver is to compute the surface divergence of the total membrane stress $\boldsymbol{\sigma} = \boldsymbol{\sigma}_e + q\mathbf{n}$ according to Eq. (6.32). For well-balancing reasons [151], in Basilisk the acceleration vector is stored on the cell faces as opposed to the cell centers. Consequently, we extend to three-dimensions the scheme of Lopez-Herrera [125], and we adapt it to surface differentiation operators.

We aim to compute $\nabla_s \cdot \boldsymbol{\sigma}$ on the cell faces from a centered tensor $\boldsymbol{\sigma}$. Let's consider the x -component of $\nabla_s \cdot \boldsymbol{\sigma}$: it is composed of space-derivatives of various components of $\boldsymbol{\sigma}$ along the three directions:

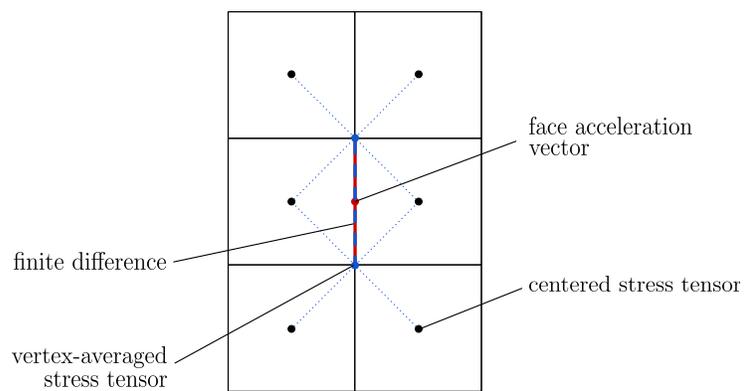
$$\begin{aligned} (\nabla_s \cdot \boldsymbol{\sigma})_x = & [(1 - n_x^2)\partial_x\sigma_{xx} - n_x n_y \partial_y \sigma_{xx} - n_x n_z \partial_z \sigma_{xx}] \\ & + [(1 - n_y^2)\partial_y\sigma_{xy} - n_y n_x \partial_x \sigma_{xy} - n_y n_z \partial_z \sigma_{xy}] \\ & + [(1 - n_z^2)\partial_z\sigma_{xz} - n_z n_x \partial_x \sigma_{xz} - n_z n_y \partial_y \sigma_{xz}], \end{aligned} \quad (6.46)$$

with $\mathbf{n} = [n_x \ n_y \ n_z]$ the vector normal to the membrane, and where the shortcut notation $\partial/\partial i = \partial_i$, $i \in \{x, y, z\}$ is introduced. In Eq. (6.46), all of the derivatives along the direction normal to the face of interest – the x -direction in this example – are computed to second order accuracy with

6.3. Numerical method and implementation strategy



(a)



(b)

Figure 6.2: Numerical scheme to obtain a face-centered differentiation of a cell-centered tensorial quantity. (a) Differentiation along the normal of the considered face (simple finite differencing). (b) Differentiation along a tangential direction to the considered face (involves vertex-averaging before differentiating).

a simple finite difference scheme, as illustrated in figure 6.2a. However, in order to obtain a face-centered differentiation of $\boldsymbol{\tau}$ in the directions tangential to the face of interest, a vertex-averaging procedure is first employed, as shown in figure 6.2b. This is equivalent to taking a finite difference of the face-averaged quantity from the two faces above and below the face of interest.

6.4 Validation cases

In this section, we compare the results produced by the above implementation of the fully Eulerian capsule solver against analytical solutions and findings reported in the literature. We discuss the ability of the present implementation to reproduce expected results and we make modifications to parts of the solver when applicable.

6.4.1 Accuracy of normal vectors

We begin our validations by ensuring that our PDE-based extension procedure of membrane quantities of the interfacial cells leads to satisfactory performance. For simplicity the results in this section are generated in two dimensions, but the extension to three dimensions is straightforward and shows identical performance.

We consider a circular membrane, where the normals everywhere in the membrane region $\tilde{\Gamma}$ are initialized as the gradient of the mollified VOF function, except in interfacial cells where the normals are computed with the height functions method [153]. The PDE presented in Section 6.3.2 is solved for each component of the normal vector, and the L_2 and L_∞ errors between the analytical and extended normal vectors \mathbf{n}_a and \mathbf{n}_e are computed for increasing grid resolutions, from 8 points per diameter to 128 points per diameter. We also display the errors of the normal vectors \mathbf{n}_{VOF} computed from the mollified VOF function, and we plot the results in figure 6.3. First, we notice that the L_∞ -error of \mathbf{n}_{VOF} does not converge, both in the case of interfacial cells and in the case of cells located away from the interface. This strong limitation of differentiating the VOF function was acknowledged early on in the literature [165] and led to a variety of methods aiming to compute the interface normals [136, 153, 165, 171]. The L_2 -error on \mathbf{n}_{VOF} , however, converges at a first order rate. On the other hand, the L_∞ -error on \mathbf{n}_e anywhere in the membrane region does converge with first order accuracy, and its L_2 -error converges with second order accuracy. Considering the absolute value of the error instead of its convergence rate,

at a typical resolution of 32 points per diameter the L_2 - and L_∞ -errors are of the same order of magnitude in the interfacial cells, and they are about 5 times lower for \mathbf{n}_e in the rest of the membrane region.

The results in figure 6.3 are satisfactory: they validate our implementation of the PDE-based extension method presented in Section 6.3.2, and they show that the L_2 - and L_∞ -errors on the extended normal vectors do converge in space at second and first orders of accuracy, respectively.

6.4.2 Stretching of a flat membrane

Our next validation case pertains to the stretching of a flat membrane. Because the membrane is represented in an Eulerian fashion, the desired membrane stretch has to be imposed by the surrounding flow field. The following two cases aim at validating the advection equations of the quantities J and \mathbf{G} describing the membrane stretch, as well as the computation of the stress tensor and of the body force applied to the fluid.

Constant strain

First, we impose an elongational flow field such that the stretch is 1 everywhere in the membrane. If the membrane is located at $y = 0$ as shown in figure 6.4, a corresponding flow field that produces the desired membrane stretch and guarantees the incompressibility condition $\partial u/\partial x + \partial v/\partial y = 0$ is $\mathbf{u} = [u, v, w] = [x, -xy, 0]$. In that case, Eq. (6.38) reduces to

$$\frac{\partial J}{\partial t} + x \frac{\partial J}{\partial x} = J. \quad (6.47)$$

Because the strain is constant, $\partial J/\partial x = 0$ and the unique solution such that $J(t=0) = 1$ is $J(t) = e^t$. We would find the same solution if we applied the method of characteristics to the full equation Eq. (6.47) without noticing this simplification. Eq. (6.39) is subject to similar simplifications, and we find:

$$\mathbf{G} = \begin{bmatrix} e^t & & \\ & 0 & \\ & & e^{-t} \end{bmatrix}. \quad (6.48)$$

Note that \mathbf{G} is diagonal in virtue of Eq. (6.41), otherwise extra non-zero diagonal components would appear. Similarly, the stress tensor is computed from Eq. (6.23):

$$\boldsymbol{\sigma}_e = \begin{bmatrix} e^t - e^{-3t} & & \\ & 0 & \\ & & e^{-t} - e^{-3t} \end{bmatrix}. \quad (6.49)$$

6.4. Validation cases

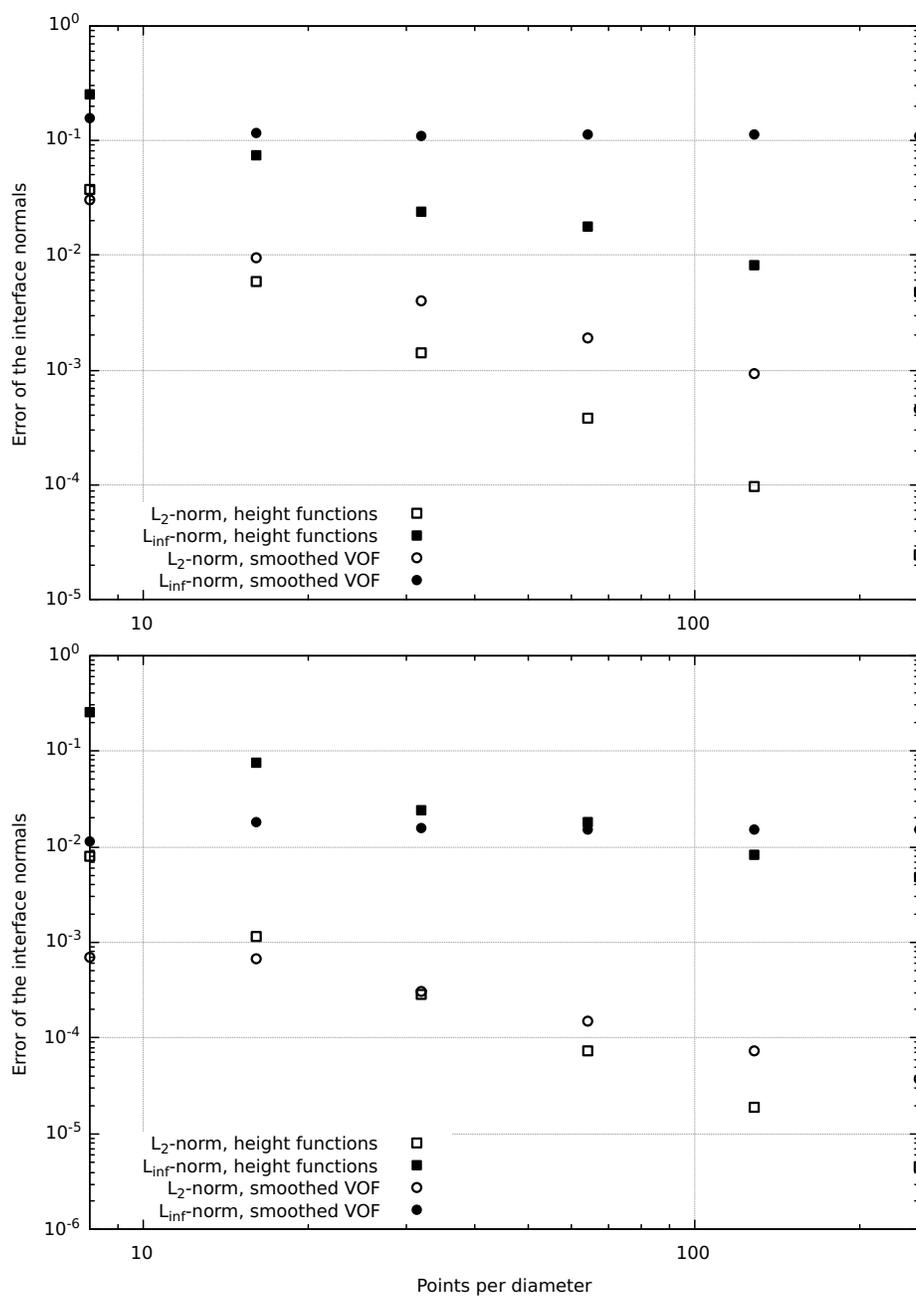


Figure 6.3: L_2 -error and L_{∞} -error on the normal vectors for a circular membrane, in the whole membrane region $\bar{\Gamma}$ (top) and restricted to the interfacial cells (bottom). Two methods to compute the normal vectors in the interfacial cells are compared: (i) from the normalized gradient of the mollified VOF function, and (ii) from the height functions method presented in [153].

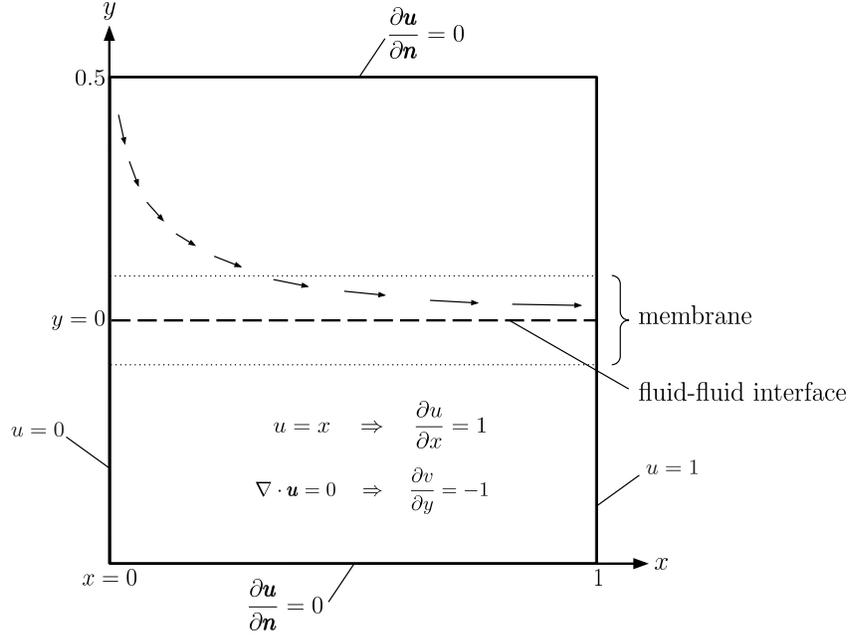


Figure 6.4: Elongational flow field leading to a constant membrane stretch.

Because $\boldsymbol{\sigma}_e$ is independent of space, the body force applied to the fluid is zero in this simple flow configuration. Nevertheless, this case allows us to validate the temporal terms of the advection equations for J and \mathbf{G} , and we plot in figure 6.5 the comparison between the results of the present solver and the above analytical solutions.

We can see from figure 6.5 that our solver compute strains and stresses that perfectly overlap the curves corresponding to the analytical solutions. Considering the simplifications this test case introduced, we can conclude that the time-dependent terms of the advection equations of J and \mathbf{G} and the computation of the diagonal components of \mathbf{G} and $\boldsymbol{\sigma}_e$ are correct.

Linear strain

We then consider an elongational flow resulting in a space-varying membrane stretch, in order to validate the space derivatives of the advection equations of J and \mathbf{G} . A flow leading to a linearly increasing stretch in the x -direction and fulfilling the incompressibility condition is $\mathbf{u} = [x^2/2, -xy, 0]$ as rep-

6.4. Validation cases

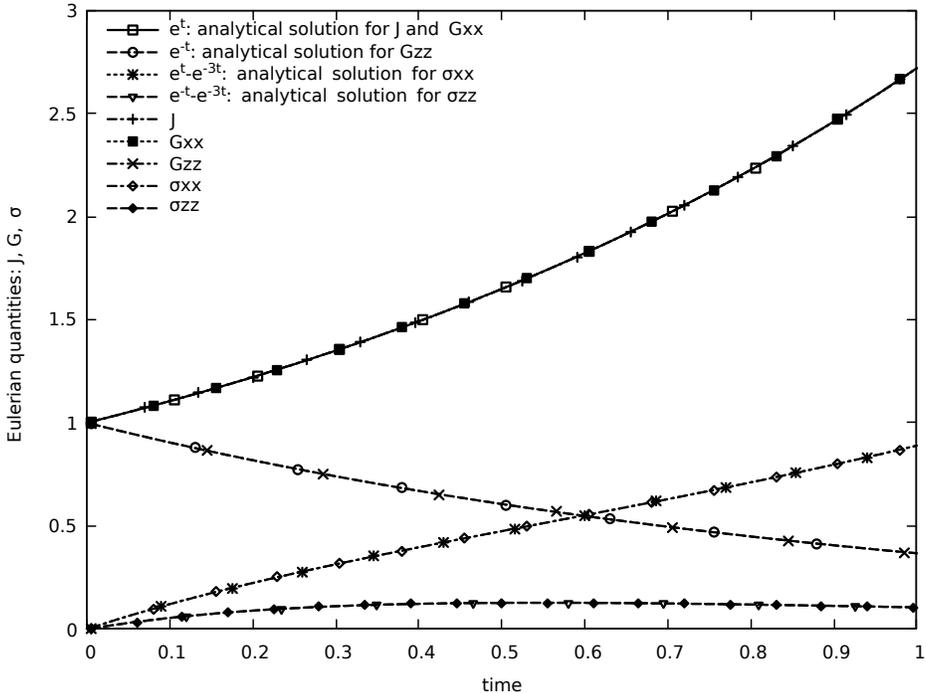


Figure 6.5: Strains and stresses in a flat horizontal membrane located at $y = 0$ and subject to a constant stretch.

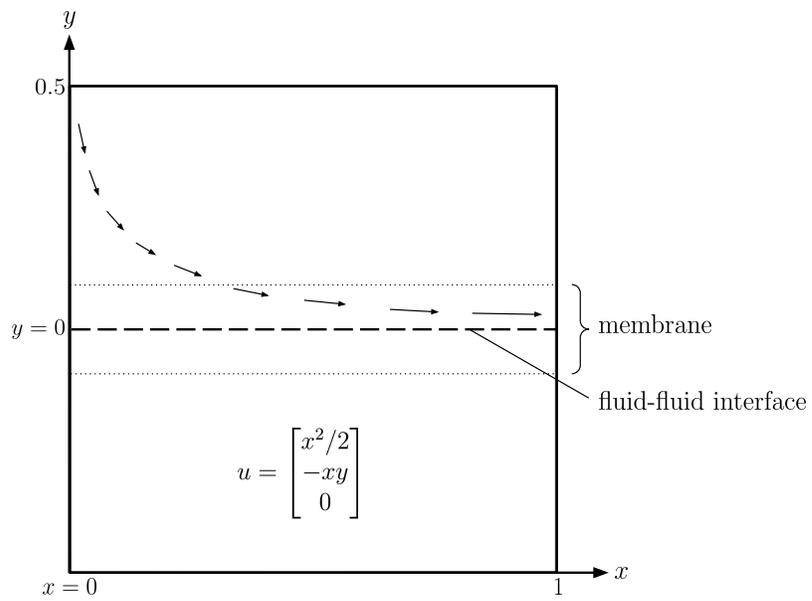


Figure 6.6: Elongational flow field leading to membrane stretch varying linearly in the x -direction.

6.4. Validation cases

resented in figure 6.6. In this case, Eq. (6.38) reduces to

$$\frac{\partial J}{\partial t} + \frac{x^2}{2} \frac{\partial J}{\partial x} = xJ, \quad (6.50)$$

and the method of characteristics yields

$$J(x, t) = \left(1 + \frac{xt}{2}\right)^2. \quad (6.51)$$

Similarly, applying the method of characteristics to Eq. (6.39) and ensuring that the in-plane constraint Eq. (6.41) is verified, we get

$$\mathbf{G} = \begin{bmatrix} \left(1 + \frac{xt}{2}\right)^2 & & \\ & 0 & \\ & & \frac{1}{\left(1 + \frac{xt}{2}\right)^2} \end{bmatrix} = \begin{bmatrix} J(x, t) & & \\ & 0 & \\ & & \frac{1}{J(x, t)} \end{bmatrix}, \quad (6.52)$$

and the stress tensor $\boldsymbol{\sigma}_e$ takes the form

$$\boldsymbol{\sigma}_e = \begin{bmatrix} J(x, t) - \frac{1}{J(x, t)^3} & & \\ & 0 & \\ & & e^{-xt} - \frac{1}{J(x, t)^3} \end{bmatrix}. \quad (6.53)$$

With the elastic stress tensor $\boldsymbol{\sigma}_e$ now varying in space, the elastic body force \mathbf{f}_b , or elastic acceleration \mathbf{a} transferred to the fluid, is non-zero:

$$\mathbf{a}/|\nabla\tilde{c}| = \nabla_s \cdot \boldsymbol{\sigma}_e = \begin{bmatrix} \frac{1}{3} \left(t\sqrt{J(x, t)} + 3tJ(x, t)^{\frac{9}{2}} \right) \\ 0 \\ 0 \end{bmatrix} \quad (6.54)$$

The quantitative comparison of J , \mathbf{G} , $\boldsymbol{\sigma}_e$ and \mathbf{a} produced by our solver to their analytical expressions is shown in figure 6.7. All of them match very well the analytical solutions, thus validating the advection equations for J and \mathbf{G} , as well as the computation of $\boldsymbol{\sigma}_e$ and \mathbf{a} ; at least when the membrane is aligned with one of the axes and the off-diagonal components of the tensors are all zero. In order to further test our solver, we can rotate the frame of reference of this case, resulting in a tilted membrane as well as non-zero off-diagonal components in \mathbf{G} , $\boldsymbol{\sigma}_e$ and \mathbf{a} . As long as the imposed elongational flow is rotated accordingly, all of the above quantities are still expressed according to Eq. (6.51–6.54) in the frame of reference aligned with the membrane.

6.4. Validation cases

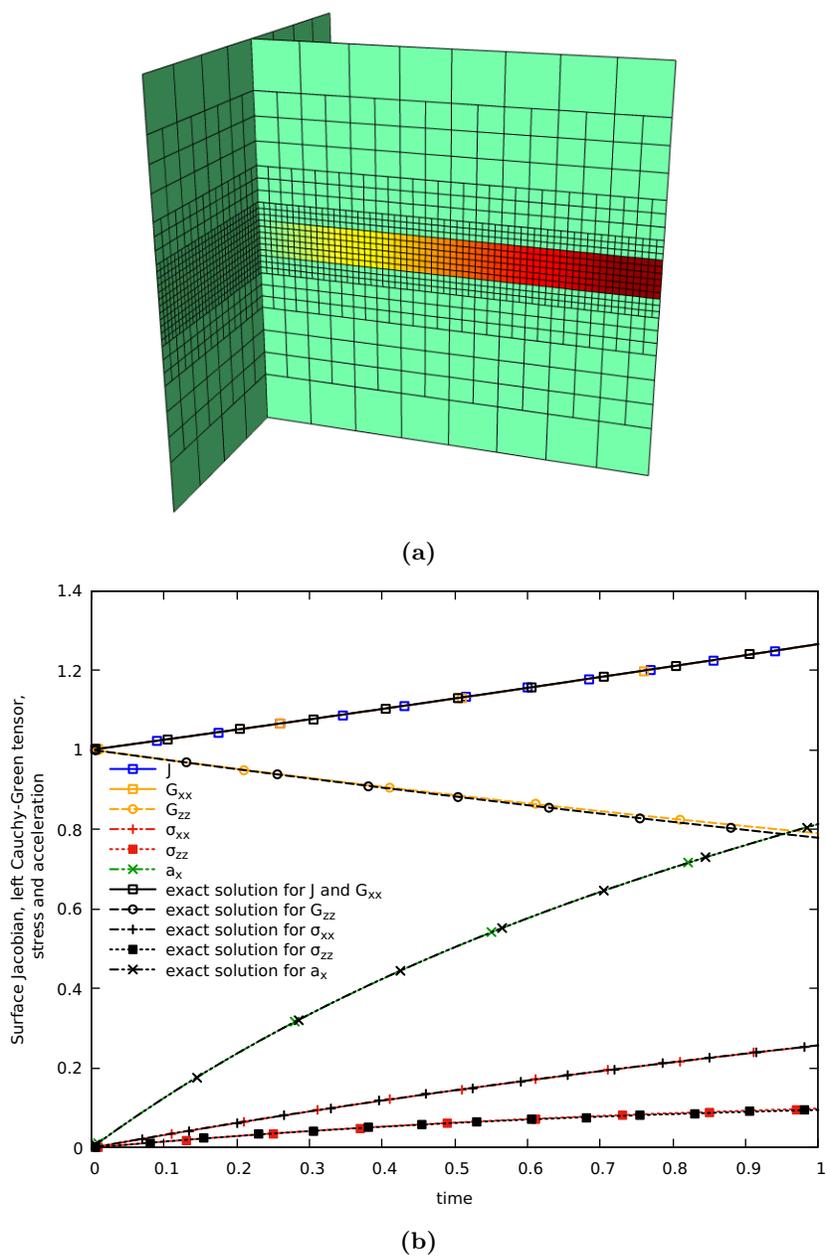
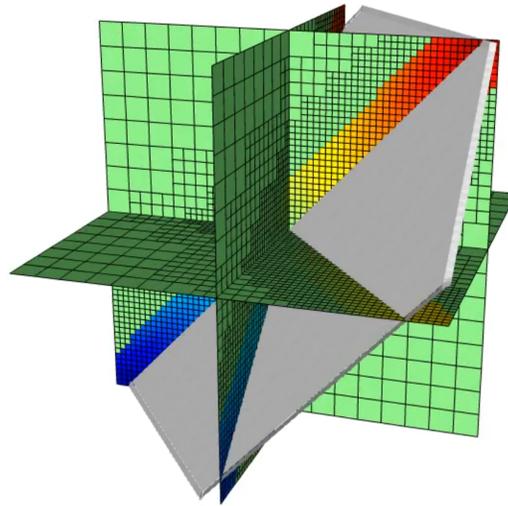
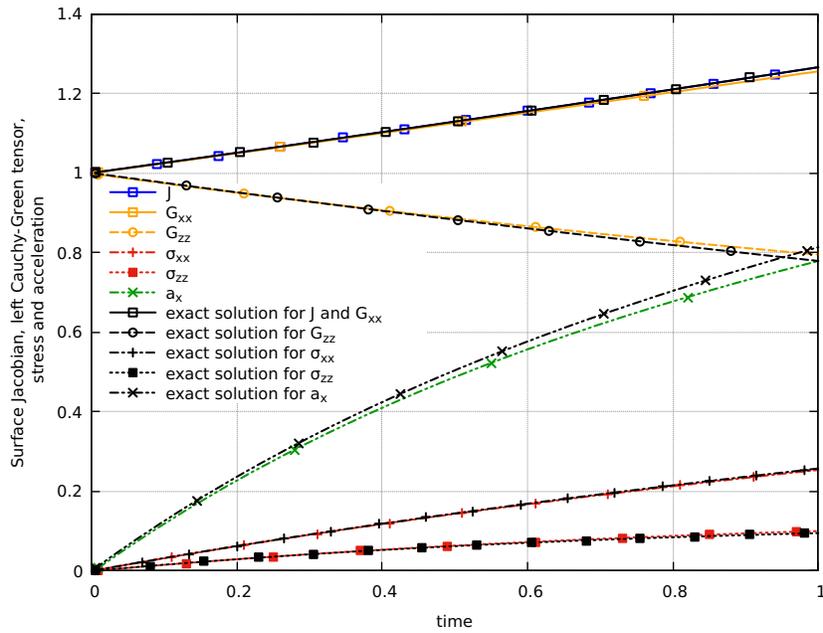


Figure 6.7: (a) Surface Jacobien J around a horizontal flat membrane (not shown) subject increasing linearly along the x -direction. (b) Corresponding strains, stresses and elastic acceleration and comparison with their analytical solutions, sampled at $x = 0.25$

6.4. Validation cases



(a)



(b)

Figure 6.8: (a) Surface Jacobian J around a rotated flat membrane subject to a stretch increasing linearly along the membrane. The membrane is represented in grey. (b) Corresponding strains, stresses and elastic acceleration and comparison with their analytical solutions, sampled at $x = 0.25$.

We perform this case and show the results in figure 6.8. The agreement between the surface Jacobian J , the left Cauchy-Green deformation tensor \mathbf{G} and the elastic stress $\boldsymbol{\sigma}_e$ and their analytical solutions is satisfactory. Thus, our implementation of the advection equations for J and \mathbf{G} as well as the computation of $\boldsymbol{\sigma}_e$ is validated. However, the computed acceleration \mathbf{a} deviates from its analytical solution, and the error accumulates in time leading to a 5% decrease in the predicted value. This discrepancy is rather significant for such a simple, one-way coupling test case. We will come back to this observation in Section 6.5.

6.4.3 Capsule deformation in a simple shear flow

After validating separately the various components of the Eulerian solver, we bring them together in one test case. In this subsection, we simulate the classical capsule in a shear flow problem presented in Section 4.4.2: an initially spherical capsule subject to neo-Hookean elastic membrane forces is placed in a shear flow and its Taylor deformation parameter is measured with respect to time. The Capillary number is 0.2 and we choose a Reynolds number of 0.01 in order to compare our method to Stokes flow solvers. A snapshot of the deformed capsule is shown in figure 6.9.

The results for the transient regime are shown in figure 6.10a for two different non-dimensional time steps $\Delta t_1 = 10^{-3}$ and $\Delta t_2 = 10^{-4}$; and in figure 6.10b for two different grid resolutions $D/\Delta x_1 = 64$ and $D/\Delta x_2 = 128$, with D the initial diameter of the capsule. We note that the results are fully converged in time, and close to be converged in space. In all cases, the Taylor deformation parameter is away from the reference values of Pozrikidis [156] and Eggleton & Popel [49] by about 15% near the end of the transient regime. Such deviations in a simple case like this shear flow are not acceptable, and a considerable amount of time and resources was invested in attempting to improve the behavior of our fully Eulerian method. The influence of the domain size, the method for computing normal vectors, the size of the kernel $|\nabla\tilde{c}|$ during the force-spreading step, the size of the membrane region $\tilde{\Gamma}$, the extension of membrane quantities away from the interface, the computation of J as an invariant of \mathbf{B} or via the advection equation Eq. (6.38), the adaptive mesh, even a slight modification in the advection equation of \mathbf{G} ⁴: all of these factors were tested independently

⁴A typographical error was suspected in [92–94], with the transpose operator being switched from $(\nabla_s \mathbf{u})^T \cdot \mathbf{G}$ to $\mathbf{G} \cdot (\nabla_s \mathbf{u})^T$ in Eq. (6.39). Instead of a misprint, this could be an unconventional choice for $\nabla \mathbf{u}$, with $(\nabla \mathbf{u})_{ij} \equiv \partial u_j / \partial x_i$ in [92–94] instead of the more conventional $(\nabla \mathbf{u})_{ij} \equiv \partial u_i / \partial x_j$.

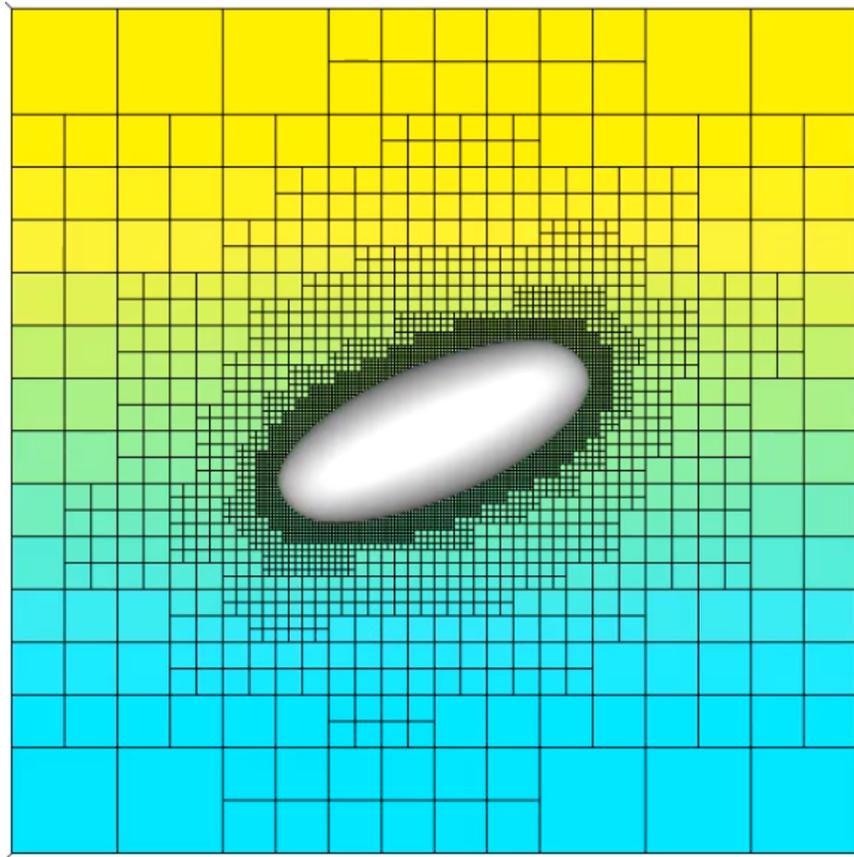
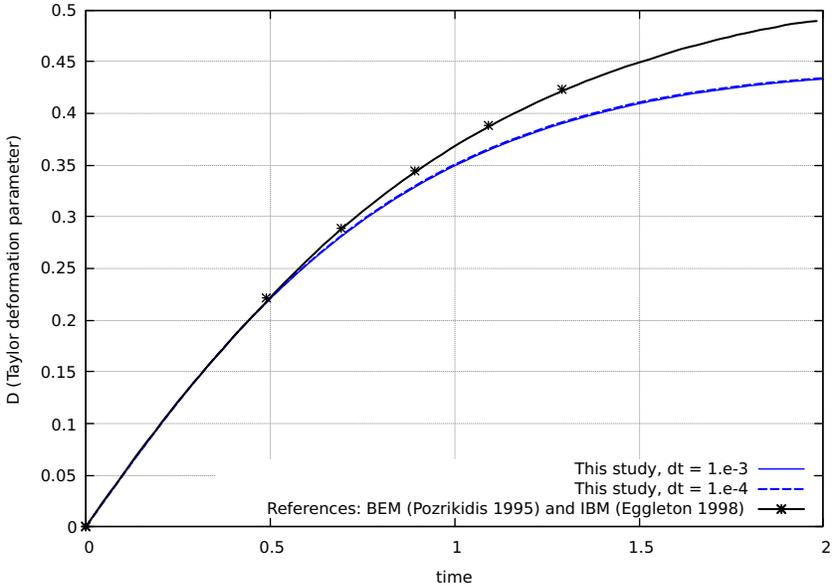
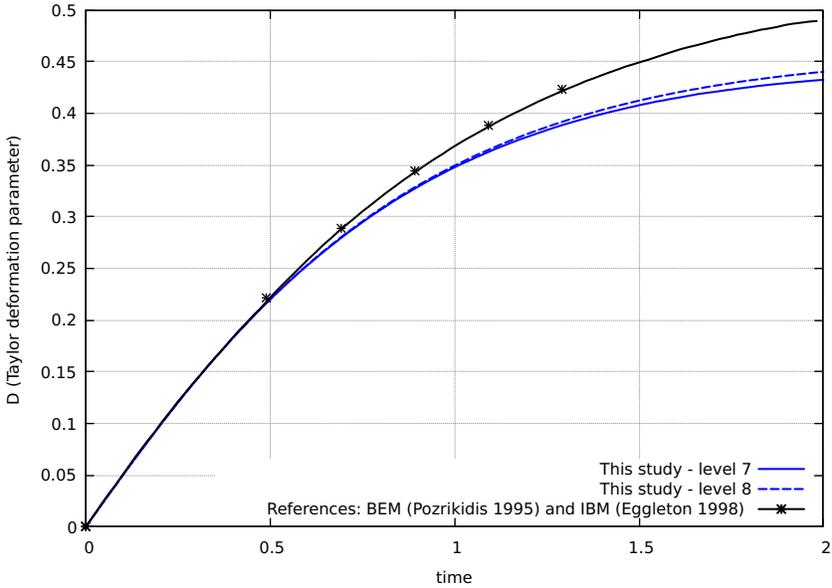


Figure 6.9: A neo-Hookean capsule represented by the fully Eulerian method in a shear flow. The color field represents the magnitude of the x -component of the velocity.

6.4. Validation cases



(a)



(b)

Figure 6.10: Convergence analysis of the fully Eulerien method in space (a) and time (b) for the case of an initially spherical neo-Hookean capsule in a shear flow in three dimensions. The Capillary number is 0.2.

for the aforementioned shear flow configuration, and all of them led to similar (sometimes worse) deviations from the literature. It does not add value to this chapter to display all the plots corresponding to these changes, since they all show a deviation of more than 15% from the reference data. We discuss in the next section the possible causes of such large deviations from the reference values, as well as changes to make in our fully Eulerian implementation in order to improve its accuracy.

6.5 Discussion

Given the rigorous validation process presented in the previous subsections, we rule out the hypothesis of a careless bug in our code leading to the discrepancies observed in Section 6.4.3. Indeed, we showed in Sections 6.4.1 to 6.4.2 that the advection equations of J and \mathbf{G} , the computation of $\boldsymbol{\sigma}_e$ and that of $\nabla_s \cdot \boldsymbol{\sigma}_e$, as well as the extension of the scalars and normal vectors are all computed correctly. However, the small discrepancy between the analytical and computed accelerations in the case of a tilted membrane as seen in figure 6.8 is puzzling, especially because such deviations are not observed in figure 6.7 when the membrane is orthogonal to one of the axes. In other words, when the vectors normal to the interface are arbitrary, an error accumulates over time in the computation of the acceleration vector, or more precisely in the term $\nabla_s \cdot \boldsymbol{\sigma}_e$. We hypothesize that in the fully Eulerian method, computing extremely accurate normal vectors is critical, due to the projection operator $\mathbf{P} = \mathbf{I} - \mathbf{n}\mathbf{n}$ involving quadratic components of the normal vectors and employed every time a surface gradient operator and a surface divergence operator are applied. In other words, our implementation of the fully-Eulerian method could suffer from not sufficiently accurate normal vectors, which error is amplified quadratically every time \mathbf{P} and ∇_s appear in the equations. Indeed, in figure 6.3 we show for that a resolution of 32 to 64 points per diameter, our normal vectors in non-interfacial cells are only slightly more accurate than the normal vectors computed from the mollified VOF function [165]. This initial error on our normal vectors could be amplified at each use of the surface projector and the surface differentiation operator, eventually becoming visible in the body force term in figure 6.8. This accumulated error could undermine the whole method in a two-way coupling simulation, as we experienced in Section 6.4.3. By design or by chance, Li et al. [92, 94] use a method to compute the normal vectors which provides them by default in the whole membrane region $\tilde{\Gamma}$ as opposed to only in interfacial cells. Therefore, they do not need to extend their normal

vectors using the method we presented in Section 6.3.2, and the accuracy of their normal vectors is independent of the distance from the interface.

To improve our implementation of the fully Eulerian method, we could employ the method of Ii et al. to access more accurate normal vectors in $\tilde{\Gamma}$: in this method, the VOF function is smoothed in such way that its profile along the normal direction is a hyperbolic tangent. This so-called MTHINC method – for Multi-dimensional Tangent of Hyperbola Interface Normal Computation – was successful in simulating surface tension-driven flows using the VOF framework, and was designed by Ii et al. in a dedicated study [95]. Alternatively, if the interface is represented by a level-set function ϕ , accurate normal vectors can be accessed anywhere in the membrane region by a simple differentiation of ϕ . If more accurate normal vectors are sought, one could simply employ a higher-order scheme to compute the gradient of ϕ . In fact, Ii et al. presented in 2018 a cell-wall interaction model relying on a modified version of their fully Eulerian method in which normal vectors are computed using a level-set function [93], confirming that the level-set framework can be a viable option as well in the context of elastic capsules. As we previously mentioned, using the level-set framework comes at the expense of losing the conservation property of the method. Ii et al. circumvent this issue by employing the coupled level-set/VOF (CLSVOF) formulation of Sussman & Puckett [190]: this method could be implemented in Basilisk and it would likely improve the results we presented in Section 6.4.3. Before implementing the complicated CLSVOF method, it could be interesting to rely solely on the level-set framework and assess its performance. Indeed, we do not expect the property of exact mass conservation to be critical for the flow configurations we are interested in, i.e. Reynolds numbers ranging from 0 to at most 100.

Besides the inaccuracy of the vectors normal to the capsule likely compromising our current implementation, other reserves with respect to the fully Eulerian method deserve to be formulated in this section. First, we presented in figure 6.10b the spatial convergence of our implementation. In the simple shear flow configuration we considered, our simulation was still not fully converged with 64 grid cells per diameter. 64 and 128 grid cells per diameters are both extremely fine grid resolutions. If such fine resolutions were required to obtain converged simulations, the fully Eulerian solver would be too computationally expensive to be applied to realistic cases featuring big domains and many capsules. Evidently, part of this slow convergence can be attributed to our inaccurate normal vectors, but the fully Eulerian method could also simply require finer grid resolutions close to the membrane than its Lagrangian counterpart. Moreover, the extension

of surface quantities from the interfacial cells to the rest of the membrane region using our PDE-based method presented in Section 6.3.2 can be very expensive: our tests showed that a fully converged extension procedure requires about $O(100)$ iterations. If we were to implement a level-set based method to compute the normal vectors, re-initializing the level-set to a distance function – a step known as “redistanciation” – is also an expensive PDE-based method which cost would add to that of our extrapolation of surface quantities. Finally, if bending forces are considered, the curvature needs to be differentiated twice. For the past 25 years, many very elaborate methods have been proposed in the literature to compute the mean curvature κ of a fluid-fluid interface, such as the height-functions method or the MTHINC method. We can expect that relying solely on an Eulerian grid to compute $\Delta_s \kappa$ even to first order accuracy would be extremely challenging. The lead author of the fully Eulerian method we attempted to reproduce, Satoshi Ii [92], was contacted and kindly replied transparently to our concern about the order of accuracy of the implementation of $\Delta_s \kappa$: they confirmed that their fully Eulerian method leads in theory to a *negative* order of accuracy for $\Delta_s \kappa$, and consequently the bending force as a whole should suffer the same lack of convergence. Therefore, describing an immersed biological membrane subject to elastic and bending stresses by the VOF function alone is not a good strategy. Other Eulerian methods more robust to a direct differentiation of the interface geometry should be preferred, such as the level-set and the phase-field methods.

For these reasons, and after eighteen months of attempting to implement the fully Eulerian method in Basilisk, we decided that describing the capsule in a Lagrangian fashion was a better choice, as this method seems to be more robust and as it had already been implemented successfully by several groups in the context of a constant Eulerian grid. This work eventually led to Chapter 4 and Chapter 5.

Chapter 7

Conclusion

In this thesis, we investigated two types of interaction relevant to the simulations of cell-resolved biological flows. The biological fluids we consider are composed of: (i) highly deformable flowing cells, such as Red Blood Cells (RBCs), and modelled as elastic membranes surrounding an inner fluid⁵; (ii) rigid particles of spherical and non-spherical shapes, such as White Blood Cells and platelets; and (iii) a surrounding Newtonian fluid, such as blood plasma. In Part **I**, the numerical simulations and physical behavior of a collection of dispersed solid particles of arbitrary shape were considered. In Part **II**, numerical methods relevant to immersed elastic capsules were developed, and a detailed study of the physical behavior of capsules flowing through a microfluidic geometry was carried out. In Sections 7.1 and 7.2, we summarize and discuss our findings from Parts **I** and **II**, and we provide recommendations for future works in these specific areas. In Section 7.3, we present general perspectives and we relate our work to promising advances in biology as well as in other fields.

7.1 Granular dynamics

In Chapter 2, we implemented a contact model relevant to the simulation of granular media using the Discrete Element Method. This contact model with “memory” allows to accurately represent dynamic and static granular assemblies, thanks to a spring-dashpot model in the directions normal and tangential to the contact, as well as in the rotational space. Our implementation was validated quantitatively in a series of test cases and the qualitative agreement between our implementation and the literature is very satisfactory. We found, however, that the contact models in DEM require a number of parameters which physical interpretation is not clear: as such, we surveyed the literature and we proposed a procedure to determine the contact parameters in a reproducible and rigorous fashion in the case of spherical particles. The software this model was implemented in will soon

⁵Such objects are referred to as “capsules” in the literature and in this dissertation.

be released on an open-source license, thus allowing everyone to perform accurate DEM simulations of particles of arbitrary shape, as well as rendering our research more reproducible.

In Chapter 3, we used our newly implemented contact model to investigate the dynamics of granular avalanches in a dam break configuration. We started by acknowledging that the literature mainly considers spherical particles while most of the granular materials are composed of particles of arbitrary shape – often non-convex. As a result, the current knowledge in the literature fails to accurately predict the dynamics of realistic granular materials, especially when a close inspection of the granular structure reveals the presence of intricated particles. Therefore, we chose to consider non-convex, cross-shaped particles that are prone to entanglement, and we compared our simulations to experiments. Excellent qualitative and quantitative agreement was shown between our simulations and our experiments, thus further validating our DEM solver. Moreover, several novel behaviors were observed both experimentally and numerically, such as the top-driven collapse and the buckling collapse. We denoted by “intermittent” a regime characterized by the non-reproducibility of the macroscopic outcome of a given configuration. In the intermittent regime, after opening the gate, a granular column was found to either collapse or remain stable in a non-reproducible way. Surprisingly, macroscopic quantities of our avalanches of cross-shaped particles such as the runout distance and the height ratio were found to be very similar to that of avalanches of spherical particles, despite observing very different dynamic processes during the transient regime. Our simulations allowed us to access the microstructure of the granular assembly, and we analyzed the contact network, in terms of the probability density function of the contact forces as well as the fabric anisotropy. Our hope was to distinguish *a priori* the intermittent-stable from the intermittent-collapsed regimes, given a specific microstructure prior to the opening of the gate. We found that neither these macroscopic quantities nor quantities averaged over the whole granular system could be used to discriminate the two regimes, suggesting that the intermittent behavior arises from mesoscale structures such as densely connected clusters of particles. Such a complex three-dimensional and anisotropic microstructure is difficult to analyze from a human perspective: in much simpler cases, such as two- and three-dimensional simulations of disks and spheres, insight was gained from processing the microstructure through a neural network, as these algorithmic frameworks are designed to find regularity in complex datasets. Future research could investigate the intermittent regime of entangled granular media using Machine Learning tools.

During our investigation of granular avalanches of cross-shaped particles, we found that our simulations were not agreeing with our experiments when the contact model parameters were determined using our rigorous procedure introduced in Chapter 2. Indeed, the aforementioned procedure used to set contact model parameters was derived for spherical particles. Even though similar procedures were applied successfully in the case of spheroids [181], our experience shows that it is not correct when the shape of the particles of interest deviates significantly from that of a sphere. As such, we had to perform a calibration analysis using the data from our experiments prior to conducting the simulations relevant to our original study. Unfortunately, numerical investigations that compare their results to experimental data are rare in the DEM community, in particular when dealing with high aspect ratio, non-spherical particles. Our work demonstrates that quantitative DEM studies of non-spherical particles should always be substantiated by experiments, a claim supported by other authors in the community [119, 172]. At the very least, we recommend that all DEM publications clearly indicate the numerical values of the full set of contact parameters in order to promote reproducibility – a practice that is still rare in this field. Another limitation of the DEM is the drastically small time step needed to resolve the time scale of contact. In our study, the time step was about a millionth of the physical time scale of interest. The non-trivial contact detection step of high aspect ratio, non-convex bodies further implies that the computing time for each time step is significant – in the hundreds of milliseconds in the case of our assemblies of $O(10^3)$ particles. As a result, the computing time is extremely long for each simulation – in the order of one to two weeks per simulation depending on the configuration. One way to mitigate this issue is to adopt a shared memory parallel approach instead of our current distributed memory parallel implementation. In other words, instead of using an MPI-based parallelism, relying on OpenMP or on Graphics Processing Units (GPUs) could lead to dramatic performance improvements. More specifically, translating our DEM solver to allow it to run on GPUs could realistically reduce the computing time by one or two orders of magnitude. Indeed, the usage of the “linked-cell” algorithm to optimize the contact detection step⁶ has the side effect of limiting the number of distributed processes in the simulation, but not the number of processes sharing memory, such as GPU threads. As a result, on a hardware architecture such as a GPU where the memory is

⁶This algorithm reduces the number of collision tests from $O(N^2)$ to $O(N)$, with N the number of particles. Its underlying idea is to partition the space in sub-regions: a given particle can only collide with other particles located in the same sub-region or in neighboring sub-regions [168].

shared between all processes, one could assign only one or a few particles per process while maintaining excellent scaling, thus drastically reducing the amount of computations per thread, – and consequently the overall computing time. While the amount of memory on older generations of GPUs could have raised concerns over the viability of a scalable DEM implementation on GPUs, this is not a concern anymore thanks to rapidly advancing hardware technologies. Indeed, a cutting-edge GPU in 2022 can provide around 40GB of memory, and new high-bandwidth technologies such as NVLink allows to increase the memory capacity to over 110GB, thus allowing virtually any DEM simulation to be ported to GPU architectures [163]. This promising parallelism improvement is under current investigation by a member of our group.

Another direction for future research would be in the realm of rheology. In Chapter 3, we have exhibited very complex granular behaviors and our simulations allow to access virtually any quantity of interest: from informations at the scale of a particle to averages at the meso and macro scales. As such, we could build a macroscopic description of our entangled granular medium from the bottom-up: by averaging the granular stress at the mesoscale over a control volume, we would compute a stress tensor that could be used in a continuum simulation – similar to the $\mu(I)$ -rheology stress description – and effectively representing our entangled granular medium as a non-Newtonian fluid. This approach has already been employed to study systems of disk-shaped particles [50, 164], spherical particles [28, 51] and even two-dimensional non-convex particles of low aspect ratio [177]. Due to the complexity of the behaviors we observed, this analysis would first be performed in canonic configurations such as shear flows. We anticipate that even in these simple configurations, valuable insight will arise, especially regarding the topology of the contact force network, the pressure distribution through the granular assembly, and the hypothetical emergence of one or several “backbones” carrying most of the stress.

Finally, with regards to applications to biological flows, two improvements can be made. Firstly, the contact detection between a solid body and a triangulated object is an important feature that is currently under active development by a member of our group. Indeed, while in Part II the capsules follow the streamlines of the flow field and thus do not collide, the motion of dispersed solid phase is usually not exactly following the flow field⁷ [194]. As such, a contact detection algorithm between a solid white blood

⁷This is because a sub-time stepping approach is used in order to prevent the fluid solver to be subject to the same time step restrictions as the DEM solver.

cell and a triangulated RBC need to be implemented in order to prevent any unphysical overlap. The same applies to the contact detection between a solid platelet and a triangulated blood vessel wall, for instance. The second improvement relates to the physics of colliding bodies in a biological context. While we can expect fast binary contacts between two flowing platelets or white blood cells, the same cannot be said if the configuration of interest involves the aggregation of activated platelets at the location of a wound. In this configuration, platelets secrete glycoproteins which enable their adhesion. In this case, a cohesive contact model [1, 141, 173] needs to be adopted and could constitute an innovative research direction in the field of DEM applied to biological systems.

7.2 Immersed elastic capsules

In Chapter 4, an adaptive Front-Tracking Method (FTM) was developed, implemented and validated. After reviewing the literature on numerical simulations for capsule-laden flows, we described the equations governing our problem: the Navier-Stokes equations governing the fluid phase are coupled with the equations governing the elastic and bending stresses in the immersed membrane. The two sets of equations are coupled through of a body-force term in the momentum equation of the Navier-Stokes solver. Then, we presented our adaptive FTM: in summary, the capsule membrane is discretized using a triangulation, where the elastic strains and stresses as well as the membrane curvature can be conveniently computed, by means of a linear Finite Element Method (FEM) and a paraboloid fitting algorithm. The Lagrangian triangulation communicates with the background Eulerian grid by means of the Immersed Boundary Method (IBM), where velocity interpolations and force spreadings occur on the stencils of regularized Dirac distributions centered on the Lagrangian nodes of the membrane triangulation. The adaptive Octree grid used to solve the background fluid is forced at its maximum refinement level on the supports of the regularized Dirac distributions, i.e. in a neighborhood of the membrane. This imposed feature is not restrictive in most cases, because the dynamics of the flow is driven by the boundary layers around the membrane. However, in the rare cases where this restriction may be limiting, we proposed in Appendix C an innovative and promising way to allow the capsule membrane to span different levels of grid refinement. After presenting the numerical implementation of our adaptive FTM, we validated it with a series of test cases of increasing difficulty. We validated the computations of the mem-

brane forces, and then validated the coupling between the membrane and the fluid solvers. We compared our computed results to results published in the literature in many different configurations, featuring: a single capsule and two interacting capsules, the presence of elastic and bending forces, a unity and a non-unity viscosity ratio, and an inertial and a non-inertial flow regime. In all these configurations, our computed results showed excellent qualitative and quantitative agreement with the literature. We then presented two configurations that would be out-of-reach without the adaptive feature of our FTM. In the first configuration, the extreme deformation of a capsule flowing through a narrow constriction is considered. In this case, the Eulerian grid resolution is required to be very small in the constricted region. Our adaptive solver allows to coarsen the rest of the computational domain, thus rendering this simulation achievable. The second demonstration case we considered involved a capsule flowing through a very large and complex three-dimensional geometry. More specifically, we considered the inertial migration of a capsule through a helical channel. Our adaptive solver allowed to refine the Eulerian grid only in the regions of interest, namely around the capsule membrane and the channel walls. This particular case demonstrates the ability of our adaptive FTM solver to realistically simulate resolved capsules flowing through biologically-relevant microfluidic geometries in an inertial regime. In addition to presenting the aforementioned results, we made our FTM solver fully open-source as part of the Basilisk platform. We also provided the necessary scripts allowing to seamlessly reproduce our simulations and our post-processing routines, in an effort to render our research as easily reproducible as possible.

In Chapter 5, we used our newly implemented adaptive FTM solver in order to investigate the non-inertial and inertial motion and deformation of capsules through a sharp corner. The geometry we considered consists of a square channel of width three times the radius of the capsules. In the first part of our analysis, we studied the motion and trajectory of a single capsule through this geometry for four Reynolds numbers ranging from 0.01 to 50, and four Capillary numbers ranging from 0.075 to 0.35. We analyzed the deviation of the capsule centroid velocity from its steady state as well as its normalized area. As expected, the velocity deviations (respectively, the normalized area) displays a minimum (respectively, a maximum) when the capsule is located inside the corner, with increasing extrema as the Capillary and Reynolds numbers increase. For Reynolds numbers varying from 1 to 50, we uncovered a linear dependence of the maximum area with respect to the Reynolds number. This particular result is of striking importance as it can help experimentalists anticipate capsule breakup in inertial microfluidic

devices. We then introduced a second capsule, released in the wake of the primary capsule at a normalized gap d varying from one diameter to one fourth of a diameter. We found that the velocity (respectively the area) of the trailing capsule is lower (respectively larger) than that of a solitary capsule. However these deviations are small, of the order of a few percents, which is likely due to the strong confinement ratio of the configuration we consider. Moreover, we found that when the initial interspacing distance is small, capsules tend to repel each other: in other words, there exists a minimum stable interspacing distance between two confined capsules in a square channel. A legitimate question is that of the accumulation of these behaviors when more than two capsules are considered. As such, we then studied the motion of a train of ten capsules in a very interacting configuration of $d = 0.125$. Both the velocity deviation and the normalized area increase as the capsule of interest is located further downstream in the train of capsules, before reaching steady velocities and deformed areas after the ninth capsule. As sharp corners are very common in microfluidic devices, our study sheds light on the strains and stresses felt by biological cells as they flow through this geometry. In particular, the surprisingly low interactions between leading and following capsules even in inertial regimes is welcome, as it allows to numerically study this system in the simplified case of a solitary capsule. As this study was conducted in a simplified domain, a next promising step would be to consider a larger, more realistic microfluidic channel geometry such as the inertial centrifugation device of Fang et al. [52] or the various zigzag size-segregation devices studied in Bazaz et al. [22].

Prior to implementing our adaptive FTM solver described in Chapter 4, we investigated the adaptive simulations of capsules in a fully Eulerian framework, i.e. without introducing a triangulation of the membrane. We presented this investigation in Chapter 6. After discussing the relevant literature, we introduced the equations describing the capsule and fluid system in the fully Eulerian flavor. While the Navier-Stokes equations remained unchanged from Chapter 4, the equations governing the membrane mechanics differ from those presented in Chapter 4. Indeed, in the fully Eulerian method the membrane strains are computed using a tensorial quantity: the surface left Cauchy-Green deformation tensor \mathbf{B} . As such, the constitutive equations of the membrane are reformulated in Chapter 6 as a function of \mathbf{B} . The Eulerian method we adopted relies on the Volume Of Fluid (VOF) method, where the capsule is represented implicitly using a field storing the volume fraction of the inner fluid, and referred to as the VOF function. At each time step, the VOF function is advected using the updated velocity

field, as is the tensor \mathbf{B} representing the membrane strains. Using this new configuration of the membrane, the stresses are computed using the same constitutive laws introduced in Chapter 4, albeit formulated in terms of \mathbf{B} . The stresses are then transferred to the fluid as a body-force term in the momentum equation, using a regularized two-dimensional Dirac distribution. A significant part of Chapter 6 was dedicated to the validation of our fully Eulerian implementation. We constructed a series of test cases designed to validate specific parts of our numerical method, which our implementation passed successfully. Unfortunately, when comparing our method to the literature in a two-way coupling simulation, we observed discrepancies of the order of 10% to 15%. We believe that these discrepancies arise from accumulated errors in the computation of the vectors normal to the membrane. We figured out that in the fully Eulerian method, the normal vectors need to be known at a very high accuracy because they are used in quadratic forms to project three-dimensional vectorial and tensorial quantities onto the two-dimensional membrane. Moreover, when a bending force is included, the normal vectors need to be differentiated four times. While it is possible to perform such a high-order differentiation of the geometry in case the membrane is defined implicitly using a levelset or a phase-field function, it became clear that the VOF framework is not the best candidate to represent capsules in a fully Eulerian fashion. As such, we hope that the experience that we gained in Chapter 6 can be useful to other researchers investigating a fully Eulerian membrane solver. In particular, a coupled VOF-levelset method was recently proposed [93], which could combine the advantages of both methods – exact volume conservation provided by the VOF method and ease of geometry differentiation provided by the levelset method. As a general comment, it is difficult to compare the numerous methods able to simulate capsules: a study reporting the computational performance and accuracy of the FTM, the phase-field method and the coupled VOF-levelset method all coupled to the same Navier-Stokes solver would be a welcome and valuable contribution to the field of capsules simulations.

Even though our adaptive FTM method performs very well in a wide range of configurations, as discussed in Chapter 4, it can be improved in several areas. First, one of the main limitations of the current method is the stability restriction due to the first order time splitting scheme of the IBM. In the simulations we presented in Part II of this thesis, the time step is set by trial and error: to our knowledge, there is no stability analysis providing an upper bound for the time step that would guarantee the stability of the simulations. Stockie and Wetton [186, 187] analyzed the stability of the IBM in the context of immersed flexible fibers, and an interesting research direc-

tion could be to conduct a similar analysis in the case of capsules subject to elastic and bending stresses. This upper bound would allow to dynamically adapt the time step of the simulation with the stresses experienced by the membrane at each discrete time. While a theoretical upper bound for the time step would certainly be welcome and would allow to always choose its largest allowed value, in our experience this upper bound can still be extremely small, especially in highly refined cases such as the capsule flowing through a narrow constriction presented in Section 4.5.1. One could hope to bypass the stability restriction by coupling the membrane and fluid problems in an implicit fashion, rather than the current first order explicit coupling. Unfortunately, implicit immersed boundary methods lead to linear systems that are very long to invert, which prohibits this research direction until more advances are been made towards preconditionners optimized for such systems [69, 142]. Another possibility would be to use a sub-time stepping method in order to decouple the membrane and the fluid stability conditions, as is done in the particle-laden flow community. The latter approach would imply to evolve the capsule using its own momentum conservation equation. As a consequence, the Lagrangian nodes on the membrane would not exactly follow the flow streamlines, thus likely requiring a contact detection method and an ad-hoc repulsion force between two approaching capsules.

The main strength of our capsule solver is that it is can be used in a wide variety of flow conditions, with the Reynolds number varying from $O(10^{-3})$ to $O(10^2)$. However, because our solver is so general it does not perform as well as other implementations specifically designed for given flow conditions, such as the boundary integral method in the case of non-inertial regimes. For low Reynolds numbers in particular, our implicit viscous solver inverts a linear system that is not well conditioned and we observe a drastic decrease in the performance of our simulations: a case at $Re = 10^{-2}$ can be up to twice as long as the same simulation at $Re = 10$. This issue could be mitigated by coupling our adaptive FTM implementation to a Navier-Stokes solver based on the Lattice Boltzmann Method (LBM). Because the LBM method is fully explicit, it is generally much faster than PDE-based Navier-Stokes solvers where the viscous term is treated implicitly, especially at low Reynolds numbers. As a result, this change would likely dramatically improve the overall performance of our capsule solver at low Reynolds numbers while maintaining good performance at higher Reynolds numbers. An adaptive LBM solver was recently published by a member of our group, and it is also implemented in the Basilisk platform [31], thus reducing to a minimum the implementation efforts needed for an adaptive LBM-based front-tracking method.

Since its introduction in the early nineties, one of the main criticisms against the FTM is its non-trivial parallel implementation. In our case, the Eulerian Octree grid is distributed over several processors using a Z-ordering technique: the grid is mapped to a one-dimensional array that is evenly divided and mapped to the active processors. Consequently the topology of the domain partition between processors is complex, and it changes from one time step to another as a result of the dynamic adaptivity of the grid. Currently, the parallel implementation of our FTM is far from optimal: every capsule exists and is solved on every processor, and only the distributed Eulerian velocities are communicated between processors. Because most of the computing time happens in the Navier-Stokes solver – less than 5% of the simulation time is spent in our FTM for a single capsule simulation –, this naive parallel implementation is harmless when the number of capsules is low. As the number of capsules increases, however, the computing cost as well as the memory cost increase linearly with the number of capsules. Our tests showed that beyond 100 capsules, our current implementation is not viable as most of the computing time is spent in the FTM solver. The case we considered for this performance test is shown in figure 7.1: a bi-periodic suspension of 100 capsules is sheared with the velocity gradient along the vertical direction. The rheology of capsule suspensions is of interest in biology and in many industrial processes [9, 110, 111], and the present adaptive FTM solver could be especially competitive when the considered volume fraction ϕ is lower than 30%, i.e. when the adaptive mesh is not refined in most of the domain and therefore leads to computational speedup. A better parallelization paradigm than the current naive approach would be to divide a given capsule into a collection of small patches and to distribute it to the processors that own the Eulerian grid cells containing its triangulation nodes. To be more precise, a field would be introduced on the whole Octree grid. Instead of being populated with real numbers, this field would contain arrays of Lagrangian nodes, edges and triangles. For the Eulerian grid cells located away from the capsule, the arrays would be empty. For Eulerian grid cells containing a Lagrangian node or located in its IBM stencil, the arrays would contain information about the local patch of the membrane triangulation. The elastic and bending forces would only be computed by the processors owning the Eulerian grid cells containing non-empty arrays, thus dramatically improving the parallel performance. While conceptually simple, this approach would lead to significant implementation challenges, and it is not easily achievable in the current version of Basilisk.

Thanks to our adaptive Eulerian grid, we have been able to fully resolve the lubrication layers between the capsules and the walls of the fluid

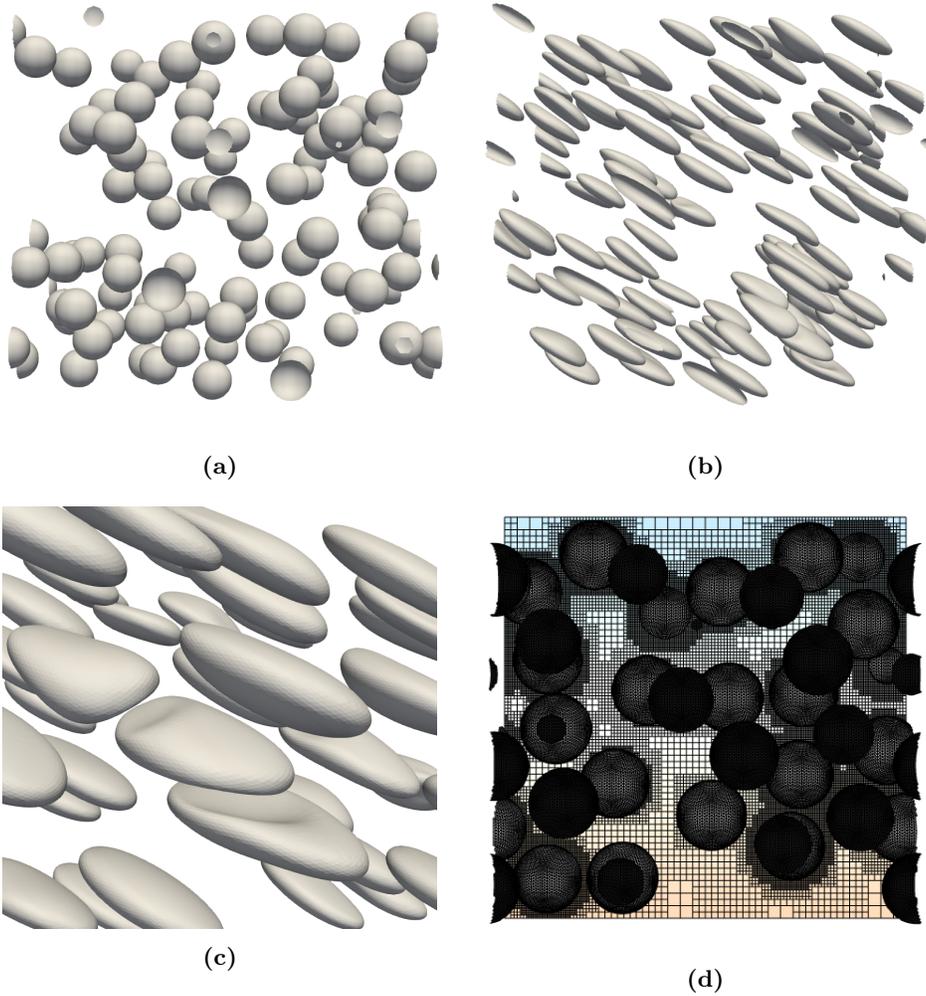


Figure 7.1: Qualitative view of sheared suspensions of 100 initially spherical capsules: (a) initial configuration at a volume fraction $\phi = 5\%$, (b) configuration at $\phi = 5\%$ after 5 reference time units, (c) zoom on interacting capsules located in the lower right corner of (b), and (d) visualization of the adaptive grid at time $t = 0$ and at $\phi = 25\%$. The open capsules are a visualization artifact due to periodic boundary conditions.

domain. However, in more complex configurations such as channel bifurcations, a node from the membrane triangulation can be located so close to the wall that its IBM stencil extends outside of the fluid domain. In this case, according to our implementation, the interpolated velocity outside the fluid domain would be zero, and the body force term spread outside of the fluid domain would be ignored. In other words, the integral over space of this particular regularized Dirac distribution would be strictly less than one, instead of being exactly equal to one. A naive fix would be to manually force this integral to be one by introducing a correction coefficient to all of the cells located at the intersection of the fluid domain and the stencil of interest. We implemented this idea and we conducted a promising preliminary test case in a fork geometry where a capsule collides with the walls over a large surface area, as shown in figure 7.2. At least for the velocity interpolation step, a more elegant approach would be to force symmetry conditions outside of the fluid domain, as is done by default in the ghost fluid method used to represent immersed solid boundaries [57, 134]. Another approach would be to use one-sided immersed boundary kernels in order to ensure that only grid cells inside the fluid domain contribute to the immersed boundary convolution operations [13]. To our knowledge, the issue of regularized Dirac distributions extending outside of the fluid domain is rarely discussed [207] and has not been investigated in the literature. As such, quantifying the introduced errors and assessing the need for such “wall-aware” IBM stencils would be a first step before diving deeper into this approach. Finally, a promising research direction regarding the adaptive FTM implementation would be to investigate further the adaptive IBM stencils presented in Appendix C. Before spending significant resources towards implementing this approach in three dimensions and in parallel, it would be of interest to study its asymptotic behavior theoretically. We would like to prove that the adaptive feature of the IBM stencils does not degrade the convergence nor the stability properties of the overall method.

7.3 Perspectives

The numerical tools presented in this thesis have the potential to provide significant insight into the design of efficient microfluidic tools. In particular, we demonstrated in Chapter 5 that the dynamics of a capsule is significantly affected by the presence of neighbors only in the case of small inter-capsule distances. As such, considering a solitary capsule is sufficient to model microfluidic devices operating in a dilute regime, at least when the width

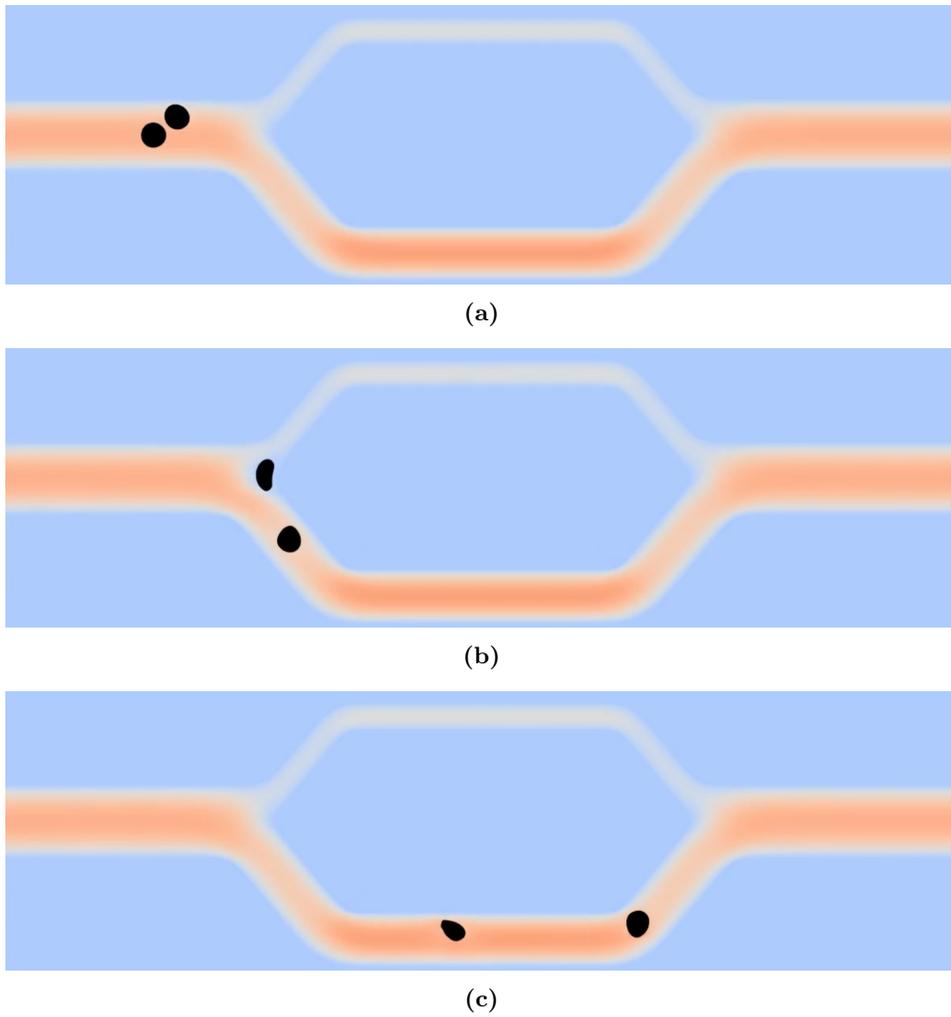


Figure 7.2: (a) Initial pre-collision configuration. (b) Collision of the top capsule with the channel walls. (c) Post-collision configuration.

of the channel is comparable to the diameter of the capsule, as was the case in our study. Simulating the capsule motion through these devices can help find an optimal geometry in order to accomplish a desired task, e.g. cell segregation based on size or membrane deformability. Numerical simulations can also significantly reduce the design schedule of these devices by suppressing the need to manufacture many intermediate prototypes and conduct the associated experiments. The adaptive mesh refinement feature of our solver makes it particularly valuable regarding the aforementioned applications, as the computational resources are focused where they are most needed, i.e. around the solitary capsule.

In some cases, it is necessary to consider a large number of capsules, e.g. to study numerically the in-vivo motion of whole blood. Using cell-resolved computational fluid dynamics tools to simulate such biologically relevant in-vivo systems was pioneered by the work of Zhao et al. [219], later followed by that of Balogh & Bagchi [15]. The former authors were able to gain insight into the partitioning of RBCs through multiple bifurcations, or into the cell-free layer emerging between the vessel walls and the suspended RBCs [16, 17]. When the volume fraction of capsules is higher than 30%, adapting the mesh at each time step becomes less efficient than imposing a fine grid resolution everywhere in the fluid domain. However, as discussed in Section 4.5.2, when the geometry of the fluid domain is large and the ratio of its volume with respect to the volume of its bounding box is low, most of the cells of a uniform Cartesian grid are located in a region of space where there is no fluid. This poor distribution of Eulerian grid cells deteriorates the solver performance, especially its memory cost. As such, our adaptive FTM could be used to simulate systems of complex vascular networks: prior to the beginning of the simulation, at time $t = 0$, the grid is refined everywhere in the fluid domain and coarsened outside; while for $t > 0$ the grid remains unchanged. Of course, in these simulations the number of capsules would drastically increase: Balogh & Bagchi used $O(10^3)$ RBCs in [16, 17], while Ames et al. [7] considered $O(10^6)$ capsules as a proof of concept for their massively parallel implementation on GPUs. Assuming our recommendations outlined in Section 7.2 towards an efficient parallelization of the FTM on octree grids are implemented, we could expect simulating $O(10^3)$ to $O(10^4)$ RBCs⁸. A simulation containing 10^4 RBCs at a biologically relevant hematocrit of 43% with a grid resolution of 32 grid cells along the largest diameter of the biconcave RBC shape would result in over

⁸These numbers would even increase to match those of Ames et al. [7] when Basilisk releases a version compatible with GPUs, which is under current developments.

1.2 billion grid cells located inside the fluid domain. Such a simulation would require $O(10^3)$ CPU processors, which is arguably very computationally intensive but not out-of-reach [139]. Such large simulations would allow to quantitatively study various biological phenomena in realistic conditions. Examples include platelets adhesion to the walls of a blood vessel – a process known as thrombosis – as they collide with other suspended cells such as RBCs, or the advection of chemical components in realistic microcirculation geometries. The latter application is particularly interesting in the context of targeted drug delivery: it is useful to quantify the concentration profile of drugs throughout the capillary network, or to quantify the amount of drugs that remain in the vicinity of a target after their release by a lipid nanoparticle (NLP) [113]. This specific application would require to model the drug particles as point-tracers which are advected by the flow, a feature that would be very easy to implement on top of our adaptive FTM solver as it would reuse its IBM and advecting methods.

Finally, the numerical tools developed in this thesis are general and can be applied to other systems than flowing capsules. If no topology changes are expected, our adaptive FTM method could be used as is to represent surface-tension driven flows, i.e. droplets and bubbles. These flows are of primary importance in a wide range of industrial applications, especially where cooling systems are used, such as in the nuclear energy industry. Moreover, our adaptive FTM could be modified to represent suspensions of one-dimensional flexible fibers. The required modifications would be minimal: the topology of the mesh would need to be changed from a closed surface to an open one-dimensional curve, and a solver for Euler’s beam equations would be used to compute the stress on each fiber [6, 167]. All other components of the adaptive FTM solver developed in this thesis would be unchanged, including our implementation of the adaptive IBM, advection equations, visualization functions, etc. Such suspensions of fibers are found in various industrial and natural processes such as the pulp and paper industry (very dense suspensions) or the microplastic pollution in the ocean (very dilute suspensions). Our adaptive FTM would be particularly well suited to study the latter application.

Bibliography

- [1] Anaïs Abramian, Lydie Staron, and Pierre-Yves Lagrée. The slumping of a cohesive granular column: Continuum and discrete modeling. *Journal of Rheology*, 64(5):1227–1235, 2020. (Cited on pages 59 and 172.)
- [2] Ronald J Adrian and Jerry Westerweel. *Particle image velocimetry*. Number 30. Cambridge university press, 2011. (Cited on page 11.)
- [3] G Agresar, JJ Linderman, Grétar Tryggvason, and KG Powell. An adaptive, cartesian, front-tracking method for the motion, deformation and adhesion of circulating cells. *Journal of Computational Physics*, 143(2):346–380, 1998. (Cited on page 66.)
- [4] J. Ai, J.F. Chen, J.M. Rotter, and J.Y. Ooi. Assessment of rolling resistance models in discrete element simulations. *Powder Technology*, 206(3):269–282, 2011. (Cited on pages 17, 18, and 22.)
- [5] Catherine Alix-Panabières, Heidi Schwarzenbach, and Klaus Pantel. Circulating tumor cells and circulating tumor dna. *Annual review of medicine*, 63:199–215, 2012. (Cited on page 1.)
- [6] Arash Alizad Banaei, Mona Rahmani, D. Martinez, and Luca Brandt. Inertial settling of flexible fiber suspensions. *Physical Review Fluids*, 5, 02 2020. (Cited on page 182.)
- [7] Jeff Ames, Daniel F Puleri, Peter Balogh, John Gounley, Erik W Draeger, and Amanda Randles. Multi-gpu immersed boundary method hemodynamics simulations. *Journal of computational science*, 44:101153, 2020. (Cited on pages 8, 64, and 181.)
- [8] Bruno Andreotti, Yoël Forterre, and Olivier Pouliquen. *Granular media: between fluid and solid*. Cambridge University Press, 2013. (Cited on page 11.)

- [9] Othmane Aouane, Andrea Scagliarini, and Jens Harting. Structure and rheology of suspensions of spherical strain-hardening capsules. *Journal of Fluid Mechanics*, 911:A11, 2021. (Cited on page 177.)
- [10] Electron Microscopy Facility at The National Cancer Institute at Frederick (NCI-Frederick). 2004. URL: https://en.wikipedia.org/wiki/File:Red_White_Blood_cells.jpg. (Cited on pages xiii and 2.)
- [11] You Han Bae and Kinam Park. Targeted drug delivery to tumors: myths, reality and possibility. *Journal of controlled release*, 153(3):198, 2011. (Cited on pages 1 and 103.)
- [12] Prosenjit Bagchi. Mesoscale simulation of blood flow in small vessels. *Biophysical journal*, 92(6):1858–1877, 2007. (Cited on page 64.)
- [13] Rahul Bale, Amneet Pal Singh Bhalla, Boyce E Griffith, and Makoto Tsubokura. A one-sided direct forcing immersed boundary method using moving least squares. *Journal of Computational Physics*, 440:110359, 2021. (Cited on page 179.)
- [14] N.J. Balmforth and R.R. Kerswell. Granular collapse in two dimensions. *Journal of Fluid Mechanics*, 538(1):399–428, 2005. (Cited on pages 32, 33, 37, 40, 42, 43, 44, 48, and 213.)
- [15] Peter Balogh and Prosenjit Bagchi. A computational approach to modeling cellular-scale blood flow in complex geometry. *Journal of Computational Physics*, 334:280–307, 2017. (Cited on pages 8, 64, 85, 105, 124, and 181.)
- [16] Peter Balogh and Prosenjit Bagchi. Analysis of red blood cell partitioning at bifurcations in simulated microvascular networks. *Physics of Fluids*, 30(5):051902, 2018. (Cited on pages 8, 64, and 181.)
- [17] Peter Balogh and Prosenjit Bagchi. The cell-free layer in simulated microvascular networks. *Journal of fluid mechanics*, 864:768–806, 2019. (Cited on pages 105 and 181.)
- [18] Peter Balogh, John Gounley, Sayan Roychowdhury, and Amanda Randles. A data-driven approach to modeling cancer cell mechanics during microcirculatory transport. *Scientific Reports*, 11(1):1–18, 2021. (Cited on page 63.)

- [19] Dominique Barthes-Biesel. Motion and deformation of elastic capsules and vesicles in flow. *Annual Review of fluid mechanics*, 48:25–52, 2016. (Cited on pages 2, 69, 74, 82, 100, and 103.)
- [20] Dominique Barthes-Biesel, Anna Diaz, and Emmanuelle Dhenin. Effect of constitutive laws for two-dimensional membranes on flow-induced capsule deformation. *Journal of Fluid Mechanics*, 460:211–222, 2002. (Cited on pages xvii, 68, 77, and 78.)
- [21] Dominique Barthes-Biesel and JM Rallison. The time-dependent deformation of a capsule freely suspended in a linear shear flow. *Journal of Fluid Mechanics*, 113:251–267, 1981. (Cited on pages 63, 67, 104, 107, 144, and 145.)
- [22] Sajad Razavi Bazaz, Asma Mihandust, Robert Salomon, Hossein Ahmadi Nejad Joushani, Wenyan Li, Hoseyn A Amiri, Fateme Mirakhorli, Sareh Zhand, Jesus Shrestha, Morteza Miansari, et al. Zigzag microchannel for rigid inertial separation and enrichment (z-rise) of cells and particles. *Lab on a Chip*, 2022. (Cited on pages 1, 63, and 174.)
- [23] John B Bell, Phillip Colella, and Harland M Glaz. A second-order projection method for the incompressible navier-stokes equations. *Journal of Computational Physics*, 85(2):257–283, 1989. (Cited on pages 69 and 151.)
- [24] Alexis Bougouin, Laurent Lacaze, and Thomas Bonometti. Collapse of a liquid-saturated granular column on a horizontal plane. *Physical Review Fluids*, 4(12):124306, 2019. (Cited on page 59.)
- [25] Jeremiah U Brackbill, Douglas B Kothe, and Charles Zemach. A continuum method for modeling surface tension. *Journal of computational physics*, 100(2):335–354, 1992. (Cited on pages 65, 140, and 143.)
- [26] Wim-Paul Breugem. A second-order accurate immersed boundary method for fully resolved simulations of particle-laden flows. *Journal of Computational Physics*, 231(13):4469–4498, 2012. (Cited on page 66.)
- [27] George Bugliarello and Juan Sevilla. Velocity distribution and other characteristics of steady and pulsatile blood flow in fine glass tubes. *Biorheology*, 7(2):85–107, 1970. (Cited on page 105.)

- [28] David Cantor, Emilien Azéma, Philippe Sornay, and Farhang Radjai. Rheology and structure of polydisperse three-dimensional packings of spheres. *Physical Review E*, 98(5):052910, 2018. (Cited on page 171.)
- [29] JM Charrier, S Shrivastava, and R Wu. Free and constrained inflation of elastic membranes in relation to thermoforming—non-axisymmetric problems. *The Journal of Strain Analysis for Engineering Design*, 24(2):55–74, 1989. (Cited on pages 68, 74, and 75.)
- [30] S Chen, B Merriman, Smereka Osher, and P Smereka. A simple level set method for solving stefan problems. *Journal of Computational Physics*, 135(1):8–29, 1997. (Cited on page 150.)
- [31] Zihao Cheng and Anthony Wachs. An immersed boundary/multi-relaxation time lattice boltzmann method on adaptive octree grids for the particle-resolved simulation of particle-laden flows. *Journal of Computational Physics*, page 111669, 2022. (Cited on pages 66 and 176.)
- [32] Alexandre Joel Chorin. Numerical solution of the navier-stokes equations. *Mathematics of computation*, 22(104):745–762, 1968. (Cited on page 69.)
- [33] P.W. Cleary. DEM prediction of industrial and geophysical particle flows. *Particuology*, 8:106–118, 2010. (Cited on pages 20 and 32.)
- [34] P.W. Cleary and M.L. Sawley. DEM modelling of industrial granular flows: 3D case studies and the effect of particle shape on hopper discharge. *Applied Mathematical Modelling*, 26(2):89–111, 2002. (Cited on page 32.)
- [35] Phillip Colella, Daniel T Graves, Benjamin J Keen, and David Modiano. A cartesian grid embedded boundary method for hyperbolic conservation laws. *Journal of Computational Physics*, 211(1):347–366, 2006. (Cited on page 72.)
- [36] Pedro Costa, Bendiks Jan Boersma, Jerry Westerweel, and Wim-Paul Breugem. Collision model for fully resolved simulations of flows laden with finite-size particles. *Physical Review E*, 92(5):053012, 2015. (Cited on page 17.)
- [37] Georges-Henri Cottet and Emmanuel Maitre. A level set method for fluid-structure interactions with immersed surfaces. *Mathematical*

- models and methods in applied sciences*, 16(03):415–438, 2006. (Cited on pages 65 and 141.)
- [38] Georges-Henri Cottet, Emmanuel Maitre, and Thomas Milcent. Eulerian formulation and level set models for incompressible fluid-structure interaction. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(3):471–492, 2008. (Cited on page 141.)
- [39] P.A. Cundall. Formulation of a three-dimensional distinct element model—Part I. A scheme to detect and represent contacts in a system composed of many polyhedral blocks. *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts*, 25(3):107–116, 1988. (Cited on page 14.)
- [40] P.A. Cundall and O.D.L. Strack. A discrete numerical model for granular assemblies. *Geotechnique*, 29(1):47–65, 1979. (Cited on page 13.)
- [41] Dana Dabiri and Charles Pecora. Particle tracking velocimetry. 2020. (Cited on page 11.)
- [42] Michela Degaetano, Laurent Lacaze, and Jeremy C Phillips. The influence of localised size reorganisation on short-duration bidispersed granular flows. *The European Physical Journal E*, 36(4):1–9, 2013. (Cited on page 213.)
- [43] Mark W Dewhirst and Timothy W Secomb. Transport of drugs from blood vessels to tumour tissue. *Nature Reviews Cancer*, 17(12):738–750, 2017. (Cited on page 63.)
- [44] Sai K Doddi and Prosenjit Bagchi. Effect of inertia on the hydrodynamic interaction between two liquid capsules in simple shear flow. *International journal of multiphase flow*, 34(4):375–392, 2008. (Cited on pages xviii, 64, 90, 91, and 105.)
- [45] Sai K Doddi and Prosenjit Bagchi. Lateral migration of a capsule in a plane poiseuille flow in a channel. *International Journal of Multiphase Flow*, 34(10):966–986, 2008. (Cited on pages 64, 74, and 82.)
- [46] F. Dorai, C. Moura Teixeira, M. Rolland, E. Climent, M. Marcoux, and A. Wachs. Fully resolved simulations of the flow through a packed bed of cylinders: Effect of size distribution. *Chemical Engineering Science*, 129:180–192, 2015. (Cited on page 26.)

- [47] A. Dviugys and B. Peters. An approach to simulate the motion of spherical and non-spherical fuel particles in combustion chambers. *Granular Matter*, 3(4):231–266, 2001. (Cited on pages 19 and 24.)
- [48] Saman Ebrahimi and Prosenjit Bagchi. Inertial and non-inertial focusing of a deformable capsule in a curved microchannel. *Journal of Fluid Mechanics*, 929, 2021. (Cited on page 105.)
- [49] Charles D Eggleton and Aleksander S Popel. Large deformation of red blood cell ghosts in a simple shear flow. *Physics of fluids*, 10(8):1834–1845, 1998. (Cited on pages 63 and 162.)
- [50] N. Estrada, E. Azéma, F. Radjai, and A. Taboada. Identification of rolling resistance as a shape parameter in sheared granular media. *Physical Review E*, 84(1):011306, 2011. (Cited on page 171.)
- [51] N. Estrada, A. Taboada, and F. Radjai. Shear strength and force transmission in granular media with rolling resistance. *Physical Review E*, 78(2):021301, 2008. (Cited on pages 51 and 171.)
- [52] Yaohui Fang, Shu Zhu, Weiqi Cheng, Zhonghua Ni, and Nan Xiang. Efficient bioparticle extraction using a miniaturized inertial microfluidic centrifuge. *Lab on a Chip*, 22(18):3545–3554, 2022. (Cited on pages 1, 63, 102, 103, 104, and 174.)
- [53] Alexander Farutin, Thierry Biben, and Chaouqi Misbah. 3d numerical simulations of vesicle and inextensible capsule dynamics. *Journal of Computational Physics*, 275:539–568, 2014. (Cited on page 75.)
- [54] Dmitry A Fedosov, Bruce Caswell, and George Em Karniadakis. A multiscale red blood cell model with accurate mechanics, rheology, and dynamics. *Biophysical journal*, 98(10):2215–2225, 2010. (Cited on pages 64 and 65.)
- [55] Anirudh Gangadhar and Siva A Vanapalli. Inertial focusing of particles and cells in the microfluidic labyrinth device: Role of sharp turns. *Biomicrofluidics*, 16(4):044114, 2022. (Cited on pages 1 and 63.)
- [56] GDR MiDi [gdrmidi@polytech.univ-mrs.fr http://www.lmgc.univ-montp2.fr/MIDI/](http://www.lmgc.univ-montp2.fr/MIDI/). On dense granular flows. *The European Physical Journal E*, 14:341–365, 2004. (Cited on pages 5 and 12.)

- [57] Reza Ghias, Rajat Mittal, and Thomas Lund. A non-body conformal grid method for simulation of compressible flows with complex immersed boundaries. In *42nd AIAA aerospace sciences meeting and exhibit*, page 80, 2004. (Cited on page 179.)
- [58] Arthur Ghigo, 2021. URL: <http://basilisk.fr/sandbox/ghigo/src/myembed.h>. (Cited on page 72.)
- [59] Arthur R. Ghigo, Stéphane Popinet, and Anthony Wachs. A conservative finite volume cut-cell method on an adaptive Cartesian tree grid for moving rigid bodies in incompressible flows. Working paper or preprint, August 2021. (Cited on page 219.)
- [60] Frédéric Gibou, Liguo Chen, Duc Nguyen, and Sanjoy Banerjee. A level set based sharp interface method for the multiphase incompressible navier–stokes equations with phase change. *Journal of Computational Physics*, 222(2):536–555, 2007. (Cited on page 141.)
- [61] Frederic Gibou, Ronald P Fedkiw, Li-Tien Cheng, and Myungjoo Kang. A second-order-accurate symmetric discretization of the poisson equation on irregular domains. *Journal of Computational Physics*, 176(1):205–227, 2002. (Cited on page 141.)
- [62] Elmer G Gilbert, Daniel W Johnson, and S Sathiya Keerthi. A fast procedure for computing the distance between complex objects in three-dimensional space. *IEEE Journal on Robotics and Automation*, 4(2):193–203, 1988. (Cited on pages 14 and 210.)
- [63] L. Girolami, V. Hergault, G. Vinay, and A. Wachs. A three-dimensional discrete-grain model for the simulation of dam-break rectangular collapses: comparison between numerical results and experiments. *Granular Matter*, 14(3):381–392, 2012. (Cited on pages 26, 33, 38, and 40.)
- [64] L. Girolami, A. Wachs, and G. Vinay. Unchannelized dam-break flows: Effects of the lateral spreading on the flow dynamics. *Physics of Fluids*, 25:043306, 2013. (Cited on pages 26 and 33.)
- [65] Nicolin Govender, Daniel N Wilke, Chuan-Yu Wu, Johannes Khinast, Patrick Pizette, and Wenjie Xu. Hopper flow of irregularly shaped particles (non-convex polyhedra): Gpu-based dem simulation and experimental validation. *Chemical Engineering Science*, 188:34–51, 2018. (Cited on pages 34, 35, 40, and 56.)

- [66] Nick Gravish, Scott V Franklin, David L Hu, and Daniel I Goldman. Entangled granular media. *Physical review letters*, 108(20):208001, 2012. (Cited on pages 32, 34, 60, and 61.)
- [67] Albert Edward Green and John Edward Adkins. Large elastic deformations and non-linear continuum mechanics. 1960. (Cited on pages 67, 104, and 107.)
- [68] Boyce E Griffith, Richard D Hornung, David M McQueen, and Charles S Peskin. An adaptive, formally second order accurate version of the immersed boundary method. *Journal of computational physics*, 223(1):10–49, 2007. (Cited on pages 66 and 219.)
- [69] Boyce E Griffith and Neelesh A Patankar. Immersed methods for fluid–structure interaction. *Annual review of fluid mechanics*, 52:421–448, 2020. (Cited on page 176.)
- [70] Jonathan Gubspun, Pierre-Yves Gires, Clément de Loubens, Dominique Barthes-Biesel, Julien Deschamps, Marc Georgelin, Marc Leonetti, Eric Leclerc, Florence Edwards-Lévy, and Anne-Virginie Salsac. Characterization of the mechanical properties of cross-linked serum albumin microcapsules: effect of size and protein concentration. *Colloid and Polymer Science*, 294(8):1381–1389, 2016. (Cited on page 103.)
- [71] Achim Guckenberger and Stephan Gekle. Theory and algorithms to compute Helfrich bending forces: A review. *Journal of Physics: Condensed Matter*, 29(20):203001, 2017. (Cited on pages 69, 83, and 146.)
- [72] Achim Guckenberger, Marcel P Schraml, Paul G Chen, Marc Leonetti, and Stephan Gekle. On the bending algorithms for soft objects in flows. *Computer Physics Communications*, 207:1–23, 2016. (Cited on pages 69, 76, 77, 79, and 83.)
- [73] E. Guendelman, R. Bridson, and R. Fedkiw. Nonconvex rigid bodies with stacking. In *ACM Transactions on Graphics (TOG)*, volume 22, pages 871–878. ACM, 2003. (Cited on pages 15 and 34.)
- [74] Arthur C Guyton, John Edward Hall, et al. *Textbook of medical physiology*, volume 548. Saunders Philadelphia, 1986. (Cited on pages 1 and 2.)

- [75] Edgar Häner, Matthias Heil, and Anne Juel. Deformation and sorting of capsules in a t-junction. *Journal of Fluid Mechanics*, 885:A4, 2020. (Cited on page 103.)
- [76] Edgar Häner, Doriane Vesperini, Anne-Virginie Salsac, Anne Le Goff, and Anne Juel. Sorting of capsules according to their stiffness: from principle to application. *Soft Matter*, 17(13):3722–3732, 2021. (Cited on pages 103 and 120.)
- [77] Wolfgang Helfrich. Elastic properties of lipid bilayers: theory and possible experiments. *Zeitschrift für Naturforschung c*, 28(11-12):693–703, 1973. (Cited on pages 68 and 108.)
- [78] Cyril W Hirt and Billy D Nichols. Volume of fluid (vof) method for the dynamics of free boundaries. *Journal of computational physics*, 39(1):201–225, 1981. (Cited on pages 65 and 140.)
- [79] X-Q Hu, B Sévénie, A-V Salsac, E Leclerc, and D Barthès-Biesel. Characterizing the membrane properties of capsules flowing in a square-section microfluidic channel: Effects of the membrane constitutive law. *Physical Review E*, 87(6):063008, 2013. (Cited on page 104.)
- [80] Damien P. Huet, 2022. URL: <http://basilisk.fr/sandbox/huet>. (Cited on pages vi, 4, 66, 99, and 107.)
- [81] Damien P. Huet, 2022. URL: http://basilisk.fr/sandbox/huet/tests/lagrangian_caps/uniaxial_stretch.c. (Cited on page 77.)
- [82] Damien P. Huet, 2022. URL: http://basilisk.fr/sandbox/huet/tests/lagrangian_caps/biconcave_curvatures.c. (Cited on page 79.)
- [83] Damien P. Huet, 2022. URL: http://basilisk.fr/sandbox/huet/tests/lagrangian_caps/nh_shear.c. (Cited on page 82.)
- [84] Damien P. Huet, 2022. URL: http://basilisk.fr/sandbox/huet/tests/lagrangian_caps/bending_shear.c. (Cited on page 83.)
- [85] Damien P. Huet, 2022. URL: http://basilisk.fr/sandbox/huet/tests/lagrangian_caps/constricted_channel.c. (Cited on page 85.)
- [86] Damien P. Huet, 2022. URL: http://basilisk.fr/sandbox/huet/tests/lagrangian_caps/rbc_shear.c. (Cited on page 88.)

- [87] Damien P. Huet, 2022. URL: http://basilisk.fr/sandbox/huet/tests/lagrangian_caps/caps_interception.c. (Cited on page 91.)
- [88] Damien P. Huet, 2022. URL: http://basilisk.fr/sandbox/huet/tests/lagrangian_caps/caps_interception_inertia.c. (Cited on page 91.)
- [89] Damien P. Huet, 2022. URL: http://basilisk.fr/sandbox/huet/cases/lagrangian_caps/helix.c. (Cited on page 96.)
- [90] Damien P. Huet, Maziyar Jalaal, Rick van Beek, Devaraj van der Meer, and Anthony Wachs. Granular avalanches of entangled rigid particles. *Phys. Rev. Fluids*, 6:104304, Oct 2021. (Cited on page v.)
- [91] Damien P. Huet and Anthony Wachs. A cartesian-octree adaptive front-tracking solver for immersed biological capsules in large complex domains. *arXiv preprint arXiv:2211.15814*, 2022. (Cited on pages vi, 109, 110, and 124.)
- [92] Satoshi Ii, Xiaobo Gong, Kazuyasu Sugiyama, Jinbiao Wu, Huaxiong Huang, and Shu Takagi. A full eulerian fluid-membrane coupling method with a smoothed volume-of-fluid approach. *Communications in Computational Physics*, 12(2):544, 2012. (Cited on pages 65, 141, 142, 144, 162, 165, and 167.)
- [93] Satoshi Ii, Kazuya Shimizu, Kazuyasu Sugiyama, and Shu Takagi. Continuum and stochastic approach for cell adhesion process based on eulerian fluid-capsule coupling with lagrangian markers. *Journal of Computational Physics*, 374:769–786, 2018. (Cited on pages 85, 147, 166, and 175.)
- [94] Satoshi Ii, Kazuyasu Sugiyama, Shu Takagi, and Yoichiro Matsumoto. A computational blood flow analysis in a capillary vessel including multiple red blood cells and platelets. *Journal of Biomechanical Science and Engineering*, 7(1):72–83, 2012. (Cited on pages 65, 141, 144, 147, 162, and 165.)
- [95] Satoshi Ii, Kazuyasu Sugiyama, Shintaro Takeuchi, Shu Takagi, Yoichiro Matsumoto, and Feng Xiao. An interface capturing method with a continuous function: the thinc method with multi-dimensional reconstruction. *Journal of Computational Physics*, 231(5):2328–2358, 2012. (Cited on page 166.)

- [96] Emel Islamzada, Kerry Matthews, Quan Guo, Aline T Santoso, Simon P Duffy, Mark D Scott, and Hongshen Ma. Deformability based sorting of stored red blood cells reveals donor-dependent aging curves. *Lab on a Chip*, 20(2):226–235, 2020. (Cited on pages 1 and 63.)
- [97] Suhas S Jain, Ken Kamrin, and Ali Mani. A conservative and non-dissipative eulerian formulation for the simulation of soft solids in fluids. *Journal of Computational Physics*, 399:108922, 2019. (Cited on pages 141 and 151.)
- [98] João Janela, Alexandra Moura, and Adélia Sequeira. A 3d non-newtonian fluid–structure interaction model for blood flow in arteries. *Journal of Computational and applied Mathematics*, 234(9):2783–2791, 2010. (Cited on page 8.)
- [99] MJ Jiang, H-S Yu, and D Harris. A novel discrete model for granular material incorporating rolling resistance. *Computers and Geotechnics*, 32(5):340–357, 2005. (Cited on page 22.)
- [100] Hans Johansen and Phillip Colella. A cartesian grid embedded boundary method for poisson’s equation on irregular domains. *Journal of Computational Physics*, 147(1):60–85, 1998. (Cited on page 71.)
- [101] Rimantas Kačianauskas, Darius Markauskas, Algis Džiugys, and Robertas Navakas. Investigation of multi-sphere models in dem simulations of granular flow-comparison with smooth ellipsoids. In: *Proceedings of 2011 Computer Methods in Mechanics*. Warsaw, Poland, 2011. (Cited on page 13.)
- [102] Ken Kamrin. Theory and modeling of granular media, 2019. URL: <http://web.mit.edu/kkamrin/www/granular.html>. (Cited on page 5.)
- [103] Ken Kamrin and Jean-Christophe Nave. An eulerian approach to the simulation of deformable solids: Application to finite-strain elasticity. *arXiv preprint arXiv:0901.3799*, 2009. (Cited on page 141.)
- [104] Ken Kamrin, Chris H Rycroft, and Jean-Christophe Nave. Reference map technique for finite-strain elasticity and fluid–solid interaction. *Journal of the Mechanics and Physics of Solids*, 60(11):1952–1969, 2012. (Cited on page 141.)

- [105] T. Kempe and J. Fröhlich. An improved immersed boundary method with direct forcing for the simulation of particle laden flows. *Journal of Computational Physics*, 231(9):3663–3684, 2012. (Cited on page 66.)
- [106] RR Kerswell. Dam break with coulomb friction: A model for granular slumping? *Physics of Fluids*, 17(5):057101, 2005. (Cited on page 32.)
- [107] Gábor Késmárky, Péter Kenyeres, Miklós Rábai, and Kálmán Tóth. Plasma viscosity: a forgotten variable. *Clinical hemorheology and microcirculation*, 39(1–4):243–246, 2008. (Cited on page 2.)
- [108] Alex Kilimnik, Wenbin Mao, and Alexander Alexeev. Inertial migration of deformable capsules in channel flow. *Physics of fluids*, 23(12):123302, 2011. (Cited on page 105.)
- [109] Ivanka Koleva and Heinz Rehage. Deformation and orientation dynamics of polysiloxane microcapsules in linear shear flow. *Soft Matter*, 8(13):3681–3693, 2012. (Cited on page 82.)
- [110] Heinrich Krüger. *Computer simulation study of collective phenomena in dense suspensions of red blood cells under shear*. Springer Science & Business Media, 2012. (Cited on page 177.)
- [111] Timm Krüger, Badr Kaoui, and Jens Harting. Interplay of inertia and deformability on rheological properties of a suspension of capsules. *Journal of Fluid Mechanics*, 751:725–745, 2014. (Cited on page 177.)
- [112] H. Kruggel-Emden, M. Sturm, S. Wirtz, and V. Scherer. Selection of an appropriate time integration scheme for the discrete element method (DEM). *Computers & Chemical Engineering*, 32(10):2263–2279, 2008. (Cited on pages xiii, 14, 19, 23, 24, 25, and 30.)
- [113] Preeti Kumari, Balaram Ghosh, and Swati Biswas. Nanocarriers for cancer-targeted drug delivery. *Journal of drug targeting*, 24(3):179–191, 2016. (Cited on pages 1, 103, and 182.)
- [114] E Lac, D Barthes-Biesel, NA Pelekasis, and J Tsamopoulos. Spherical capsules in three-dimensional unbounded stokes flows: effect of the membrane constitutive law and onset of buckling. *Journal of Fluid Mechanics*, 516:303–334, 2004. (Cited on page 64.)
- [115] Etienne Lac, Arnaud Morel, and Dominique Barthès-Biesel. Hydrodynamic interaction between two identical capsules in simple shear flow.

- Journal of Fluid Mechanics*, 573:149–169, 2007. (Cited on pages xviii, 64, 88, 89, and 90.)
- [116] Laurent Lacaze, Jeremy C Phillips, and Rich R Kerswell. Planar collapse of a granular column: Experiments and discrete element simulations. *Physics of Fluids*, 20(6):063302, 2008. (Cited on page 33.)
- [117] P-Y Lagrée, Lydie Staron, and Stéphane Popinet. The granular column collapse as a continuum: validity of a two-dimensional navier–stokes model with a μ (i)-rheology. *Journal of Fluid Mechanics*, 686:378–408, 2011. (Cited on pages 5 and 12.)
- [118] E. Lajeunesse, J.B. Monnier, and G.M. Homsy. Granular slumping on a horizontal surface. *Physics of Fluids*, 17:103302, 2005. (Cited on pages 32, 33, 40, 42, 43, 44, 48, 213, 214, and 216.)
- [119] Johann Landauer, Michael Kuhn, Daniel S Nasato, Petra Foerst, and Heiko Briesen. Particle shape matters—using 3d printed particles to investigate fundamental particle and packing properties. *Powder Technology*, 361:711–718, 2020. (Cited on pages 32, 34, 59, and 170.)
- [120] Duc Vinh Le. Effect of bending stiffness on the deformation of liquid capsules enclosed by thin shells in shear flow. *Physical Review E*, 82(1):016318, 2010. (Cited on page 83.)
- [121] Xiaoyi Li and Kausik Sarkar. Front tracking simulation of deformation and buckling instability of a liquid capsule enclosed by an elastic membrane. *Journal of Computational Physics*, 227(10):4998–5018, 2008. (Cited on page 64.)
- [122] Keng-Wit Lim, Kristian Krabbenhoft, and José E Andrade. On the contact treatment of non-convex particles in the granular element method. *Computational Particle Mechanics*, 1(3):257–275, 2014. (Cited on page 33.)
- [123] Tao Lin, Zhen Wang, Ruixin Lu, Wen Wang, and Yi Sui. A high-throughput method to characterize membrane viscosity of flowing microcapsules. *Physics of Fluids*, 33(1):011906, 2021. (Cited on page 103.)
- [124] Xiaoshan Lin and Tang-Tat Ng. Contact detection algorithms for three-dimensional ellipsoids in discrete element modelling. *International Journal for Numerical and Analytical Methods in Geomechanics*, 19(9):653–659, 1995. (Cited on page 34.)

- [125] JM López-Herrera, Stéphane Popinet, and AA Castrejón-Pita. An adaptive solver for viscoelastic incompressible two-phase problems applied to the study of the splashing of weakly viscoelastic droplets. *Journal of Non-Newtonian Fluid Mechanics*, 264:144–158, 2019. (Cited on page 151.)
- [126] G. Lu, J.R. Third, and C.R. Müller. Critical assessment of two approaches for evaluating contacts between super-quadric shaped particles in DEM simulations. *Chemical engineering science*, 78:226–235, 2012. (Cited on pages 15 and 34.)
- [127] G. Lu, J.R. Third, and C.R. Müller. Discrete element models for non-spherical particle systems: From theoretical developments to applications. *Chemical Engineering Science*, 127:425–465, 2015. (Cited on page 34.)
- [128] Libin Lu, Matthew J Morse, Abtin Rahimian, Georg Stadler, and Denis Zorin. Scalable simulation of realistic volume fraction red blood cell flows through vascular networks. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–30, 2019. (Cited on page 73.)
- [129] RX Lu, Z Wang, A-V Salsac, D Barthès-Biesel, W Wang, and Y Sui. Path selection of a train of spherical capsules in a branched microchannel. *Journal of Fluid Mechanics*, 923, 2021. (Cited on pages 8, 103, 106, 109, 110, 124, 125, and 137.)
- [130] G. Lube, H.E. Huppert, R. S. J Sparks, and A. Freundt. Collapses of two-dimensional granular columns. *Physical Review E*, 72(4):041301, 2005. (Cited on page 33.)
- [131] G. Lube, H.E. Huppert, R.S.J. Sparks, and M.A. Hallworth. Axisymmetric collapses of granular columns. *Journal of Fluid Mechanics*, 508(1):175–199, 2004. (Cited on page 33.)
- [132] F. Ludewig and N. Vandewalle. Strong interlocking of nonconvex particles in random packings. *Physical Review E*, 85(5):051307, 2012. (Cited on page 34.)
- [133] Stefan Luding. Introduction to discrete element methods: basic of contact force models and how to perform the micro-macro transition to continuum theory. *European journal of environmental and civil engineering*, 12(7-8):785–826, 2008. (Cited on page 17.)

- [134] Sekhar Majumdar, Gianluca Iaccarino, Paul Durbin, et al. Rans solvers with adaptive structured boundary non-conforming grids. *Annual Research Briefs*, 1, 2001. (Cited on page 179.)
- [135] National Library of Medicine Medicine Plus. Hematocrit, 02-22-2023. URL: <https://medlineplus.gov/ency/article/003646.htm>. (Cited on page 3.)
- [136] T. Ménard, S. Tanguy, and A. Berlemont. Coupling level set/VOF/ghost fluid methods: Validation and application to 3D simulation of the primary break-up of a liquid jet. *International Journal of Multiphase Flow*, 33(5):510–524, 2007. (Cited on page 153.)
- [137] Mark Meyer, Mathieu Desbrun, Peter Schröder, and Alan H Barr. Discrete differential-geometry operators for triangulated 2-manifolds. In *Visualization and mathematics III*, pages 35–57. Springer, 2003. (Cited on page 76.)
- [138] JC Misra, MK Patra, and SC Misra. A non-newtonian fluid model for blood flow through arteries under stenotic conditions. *Journal of biomechanics*, 26(9):1129–1141, 1993. (Cited on page 8.)
- [139] Antoine Morente, Aashish Goyal, and Anthony Wachs. A highly scalable direction-splitting solver on regular cartesian grid to compute flows in complex geometries described by stl files. *Fluids*, 8(3), 2023. (Cited on pages 4 and 182.)
- [140] Bumseok Namgung, Peng Kai Ong, Paul C Johnson, and Sangho Kim. Effect of cell-free layer variation on arteriolar wall shear stress. *Annals of biomedical engineering*, 39:359–366, 2011. (Cited on page 105.)
- [141] Steven T Nase, Watson L Vargas, Adetola A Abatan, and JJ McCarthy. Discrete characterization tools for cohesive granular material. *Powder Technology*, 116(2-3):214–223, 2001. (Cited on page 172.)
- [142] Elijah P Newren, Aaron L Fogelson, Robert D Guy, and Robert M Kirby. Unconditionally stable discretizations of the immersed boundary equations. *Journal of Computational Physics*, 222(2):702–719, 2007. (Cited on page 176.)
- [143] Erfan G Nezami, Youssef MA Hashash, Dawei Zhao, and Jamshid Ghaboussi. A fast contact detection algorithm for 3-d discrete element method. *Computers and Geotechnics*, 31(7):575–587, 2004. (Cited on page 14.)

- [144] Lia Papadopoulos, Mason A Porter, Karen E Daniels, and Danielle S Bassett. Network analysis of particles and grains. *Journal of Complex Networks*, 6(4):485–565, 2018. (Cited on page 56.)
- [145] Sun-Young Park and P Dimitrakopoulos. Transient dynamics of an elastic capsule in a microfluidic constriction. *Soft matter*, 9(37):8844–8855, 2013. (Cited on pages xviii, 83, 86, 87, 92, and 104.)
- [146] Danping Peng, Barry Merriman, Stanley Osher, Hongkai Zhao, and Myungjoo Kang. A pde-based fast local level set method. *Journal of computational physics*, 155(2):410–438, 1999. Section 3: extending quantity off an interface. (Cited on page 150.)
- [147] C.S. Peskin. Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, 25(3):220–252, 1977. (Cited on pages 72 and 110.)
- [148] C.S. Peskin. The immersed boundary method. *Acta Numerica*, 11(1):479–517, 2002. (Cited on page 110.)
- [149] Alexander Podlozhnyuk, Stefan Pirker, and Christoph Kloss. Efficient implementation of superquadric particles in discrete element method within an open-source framework. *Computational Particle Mechanics*, 4(1):101–118, 2017. (Cited on pages 14 and 34.)
- [150] S. Popinet. A quadtree-adaptive multigrid solver for the Serre–Green–Naghdi equations. *Journal of Computational Physics*, 302:336–358, 2015. (Cited on pages 66, 69, and 109.)
- [151] S. Popinet. Numerical Models of Surface Tension. *Annual Review of Fluid Mechanics*, 50:1–28, 2017. (Cited on pages 141 and 151.)
- [152] Stéphane Popinet. Gerris: a tree-based adaptive solver for the incompressible euler equations in complex geometries. *Journal of Computational Physics*, 190(2):572–600, 2003. (Cited on pages 66, 69, 70, and 71.)
- [153] Stéphane Popinet. An accurate adaptive solver for surface-tension-driven interfacial flows. *Journal of Computational Physics*, 228(16):5838–5866, 2009. (Cited on pages xxi, 65, 66, 69, 70, 141, 150, 153, and 155.)
- [154] Stéphane Popinet, 2018. URL: <http://basilisk.fr/src/embed.h#lifting-the-small-cell-cfl-restriction>. (Cited on page 72.)

- [155] Lionel Pournin, Th M Liebling, and Alain Mocellin. Molecular-dynamics force models for better control of energy dissipation in numerical simulations of dense granular media. *Physical Review E*, 65(1):011302, 2001. (Cited on page 21.)
- [156] C Pozrikidis. Finite deformation of liquid capsules enclosed by elastic membranes in simple shear flow. *Journal of Fluid Mechanics*, 297:123–152, 1995. (Cited on pages 63, 68, 82, 104, and 162.)
- [157] C Pozrikidis. Effect of membrane bending stiffness on the deformation of capsules in simple shear flow. *Journal of Fluid Mechanics*, 440:269, 2001. (Cited on pages 64, 104, and 146.)
- [158] C Pozrikidis. Numerical simulation of the flow-induced deformation of red blood cells. *Annals of biomedical engineering*, 31(10):1194–1205, 2003. (Cited on pages 64 and 104.)
- [159] C Pozrikidis. Resting shape and spontaneous membrane curvature of red blood cells. *Mathematical Medicine and Biology: A Journal of the IMA*, 22(1):34–52, 2005. (Cited on pages 85 and 109.)
- [160] Constantine Pozrikidis. Computational hydrodynamics of capsules and biological cells. 2010. (Cited on pages 2, 85, 87, and 109.)
- [161] Daniel F Puleri, Peter Balogh, and Amanda Randles. Computational models of cancer cell transport through the microcirculation. *Biomechanics and Modeling in Mechanobiology*, 20(4):1209–1230, 2021. (Cited on pages 1 and 63.)
- [162] Daniel F Puleri, Sayan Roychowdhury, Peter Balogh, John Gounley, Erik W Draeger, Jeff Ames, Adebayo Adebisi, Simbarashe Chidyagwai, Benjamín Hernández, Seyong Lee, et al. High performance adaptive physics refinement to enable large-scale tracking of cancer cell trajectory. In *2022 IEEE International Conference on Cluster Computing (CLUSTER)*, pages 230–242. IEEE, 2022. (Cited on page 1.)
- [163] “RadiantFriction601”. Why gpu memory matters more than you think, 2022. URL: <https://www.engineering.com/story/why-gpu-memory-matters-more-than-you-think>. (Cited on page 171.)
- [164] Farhang Radjai, Dietrich E Wolf, Michel Jean, and Jean-Jacques Moreau. Bimodal character of stress transmission in granular packings. *Physical review letters*, 80(1):61, 1998. (Cited on pages 50 and 171.)

- [165] Mehdi Raessi, Javad Mostaghimi, and Markus Bussmann. Advecting normal vectors: A new method for calculating interface normals and curvatures when modeling two-phase flows. *Journal of Computational Physics*, 226(1):774–797, 2007. (Cited on pages 153 and 165.)
- [166] Amir Hossein Raffiee, Sadegh Dabiri, and Arezoo M Ardekani. Elastoinertial migration of deformable capsules in a microchannel. *Biomeicrofluidics*, 11(6):064113, 2017. (Cited on page 105.)
- [167] Mona Rahmani, Arash Alizad Banaei, Luca Brandt, and D. Mark Martinez. Stochastic model for predicting the shape of flexible fibers in suspensions. *Phys. Rev. Fluids*, 8:024306, Feb 2023. (Cited on page 182.)
- [168] A.D. Rakotonirina and A. Wachs. Grains3D, a flexible DEM approach for particles of arbitrary convex shape - Part II: Parallel implementation and scalable performance. *Powder Technology*, 324:18–35, 2018. (Cited on pages 12, 33, 34, 170, and 208.)
- [169] Andriarimina Daniel Rakotonirina, Jean-Yves Delenne, Farhang Radjai, and Anthony Wachs. Grains3D, a flexible DEM approach for particles of arbitrary convex shape - Part III: extension to non-convex particles modelled as glued convex particles. *Computational Particle Mechanics*, 5(4):1–30, 2018. (Cited on pages 12, 14, 26, 34, 38, and 212.)
- [170] S Ramanujan and C Pozrikidis. Deformation of liquid capsules enclosed by elastic membranes in simple shear flow: large deformations and the effect of fluid viscosities. *Journal of fluid mechanics*, 361:117–143, 1998. (Cited on pages 63, 82, and 104.)
- [171] Yuriko Renardy and Michael Renardy. Prost: a parabolic reconstruction of surface tension for the volume-of-fluid method. *Journal of computational physics*, 183(2):400–421, 2002. (Cited on pages 141 and 153.)
- [172] Thomas Roessler, Christian Richter, André Katterfeld, and Frank Will. Development of a standard calibration procedure for the dem parameters of cohesionless bulk materials—part i: Solving the problem of ambiguous parameter combinations. *Powder Technology*, 343:803–812, 2019. (Cited on pages 59 and 170.)

- [173] Pierre G Rognon, Jean-Noël Roux, Mohamed Naaim, and François Chevoir. Dense flows of cohesive granular materials. *Journal of Fluid Mechanics*, 596:21–47, 2008. (Cited on page 172.)
- [174] A.M. Roma, C.S. Peskin, and M.J. Berger. An adaptive version of the immersed boundary method. *Journal of Computational Physics*, 153(2):509–534, 1999. (Cited on pages 66, 100, and 219.)
- [175] Leo Rothenburg and Richard J Bathurst. Numerical simulation of idealized granular assemblies with plane elliptical particles. *Computers and geotechnics*, 11(4):315–329, 1991. (Cited on page 34.)
- [176] S. Rémond, JL Gallias, and A. Mizrahi. Simulation of the packing of granular mixtures of non-convex particles and voids characterization. *Granular Matter*, 10(3):157–170, 2008. (Cited on pages 32 and 34.)
- [177] Baptiste Saint-Cyr, J-Y Delenne, Charles Voivret, Farhang Radjai, and Philippe Sornay. Rheology of granular materials composed of nonconvex particles. *Physical Review E*, 84(4):041302, 2011. (Cited on pages 51 and 171.)
- [178] Natzi Sakalihasan, Raymond Limet, and Olivier Damien Defawe. Abdominal aortic aneurysm. *The Lancet*, 365(9470):1577–1589, 2005. (Cited on page 1.)
- [179] Ruben Scardovelli and Stéphane Zaleski. Direct numerical simulation of free-surface and interfacial flow. *Annual review of fluid mechanics*, 31(1):567–603, 1999. (Cited on page 143.)
- [180] LJH Seelen, JT Padding, and JAM Kuipers. Improved quaternion-based integration scheme for rigid body motion. *Acta Mechanica*, 227(12):3381–3389, 2016. (Cited on pages 25 and 212.)
- [181] L.J.H. Seelen, J.T. Padding, and J.A.M. Kuipers. A granular Discrete Element Method for arbitrary convex particle shapes: method and packing generation. *Chemical Engineering Science*, 189:84–101, 2018. (Cited on pages 15, 21, 30, 34, 59, and 170.)
- [182] Can Selçuk, Arthur R Ghigo, Stéphane Popinet, and Anthony Wachs. A fictitious domain method with distributed lagrange multipliers on adaptive quad/octrees for the direct numerical simulation of particle-laden flows. *Journal of Computational Physics*, 430:109954, 2021. (Cited on page 101.)

- [183] M.D. Sinnott, P.W. Cleary, and R.D. Morrison. Is media shape important for grinding performance in stirred mills? *Minerals Engineering*, 24(2):138–151, 2011. (Cited on page 32.)
- [184] R Skalak, A Tozeren, RP Zarda, and S Chien. Strain energy function of red blood cell membranes. *Biophysical journal*, 13(3):245, 1973. (Cited on page 139.)
- [185] L. Staron and E. J. Hinch. Study of the collapse of granular columns using two-dimensional discrete-grain simulation. *Journal of Fluid Mechanics*, 545:1–27, 2005. (Cited on page 32.)
- [186] John M Stockie and Brian R Wetton. Analysis of stiffness in the immersed boundary method and implications for time-stepping schemes. *Journal of Computational Physics*, 154(1):41–64, 1999. (Cited on page 175.)
- [187] John M. Stockie and Brian T. R. Wetton. Stability analysis for the immersed fiber problem. *SIAM Journal on Applied Mathematics*, 55(6):1577–1591, 1995. (Cited on page 175.)
- [188] Kazuyasu Sugiyama, Satoshi Ii, Kazuya Shimizu, Shigeho Noda, and Shu Takagi. A full eulerian method for fluid-structure interaction problems. *Procedia Iutam*, 20:159–166, 2017. (Cited on page 142.)
- [189] Kazuyasu Sugiyama, Satoshi Ii, Shintaro Takeuchi, Shu Takagi, and Yoichiro Matsumoto. A full eulerian finite difference approach for solving fluid–structure coupling problems. *Journal of Computational Physics*, 230(3):596–627, 2011. (Cited on page 141.)
- [190] Mark Sussman and Elbridge Gerry Puckett. A coupled level set and volume-of-fluid method for computing 3d and axisymmetric incompressible two-phase flows. *Journal of computational physics*, 162(2):301–337, 2000. (Cited on page 166.)
- [191] Mark Sussman, Peter Smereka, and Stanley Osher. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational physics*, 114(1):146–159, 1994. (Cited on page 140.)
- [192] Naoki Takeishi, Hiroshi Yamashita, Toshihiro Omori, Naoto Yokoyama, Shigeo Wada, and Masako Sugihara-Seki. Inertial migration of red blood cells under a newtonian fluid in a circular channel. *arXiv preprint arXiv:2209.11933*, 2022. (Cited on pages 1 and 63.)

- [193] Grétar Tryggvason, Bernard Bunner, Asghar Esmaeeli, Damir Juric, N Al-Rawahi, W Tauber, J Han, S Nas, and Y-J Jan. A front-tracking method for the computations of multiphase flow. *Journal of computational physics*, 169(2):708–759, 2001. (Cited on pages 65, 72, 74, and 101.)
- [194] M. Uhlmann. An immersed boundary method with direct forcing for the simulation of particulate flows. *Journal of Computational Physics*, 209(2):448–476, 2005. (Cited on pages 66, 73, 110, and 171.)
- [195] Salih Ozen Unverdi and Grétar Tryggvason. A front-tracking method for viscous, incompressible, multi-fluid flows. *Journal of computational physics*, 100(1):25–37, 1992. (Cited on pages 72 and 73.)
- [196] Boris Valkov, Chris H Rycroft, and Ken Kamrin. Eulerian method for multiphase interactions of soft solid bodies in fluids. *Journal of Applied Mechanics*, 82(4), 2015. (Cited on page 141.)
- [197] G. Van den Bergen. A fast and robust GJK implementation for collision detection of convex objects. *Journal of Graphics, Gpu, and Game Tools*, 4:7–25, 1999. (Cited on pages 14 and 210.)
- [198] G. Van Den Bergen. Proximity queries and penetration depth computation on 3d game objects. In *Game Developers Conference*, volume 170. Citeseer, 2001. (Cited on pages 209 and 210.)
- [199] J Antoon Van Hooft, Stéphane Popinet, Chiel C Van Heerwaarden, Steven JA Van der Linden, Stephan R de Roode, and Bas JH Van de Wiel. Towards adaptive grids for atmospheric boundary-layer simulations. *Boundary-layer meteorology*, 167(3):421–443, 2018. (Cited on page 70.)
- [200] Marcos Vanella, Antonio Posa, and Elias Balaras. Adaptive mesh refinement for immersed boundary methods. *Journal of Fluids Engineering*, 136(4), 2014. (Cited on pages 66 and 219.)
- [201] Vincenzino Vivacqua, Alejandro López, Robert Hammond, and Mojtaba Ghadiri. Dem analysis of the effect of particle shape, cohesion and strain rate on powder rheometry. *Powder Technology*, 342:653–663, 2019. (Cited on page 34.)
- [202] A. Wachs, L. Girolami, G. Vinay, and G. Ferrer. Grains3D, a flexible DEM approach for particles of arbitrary convex shape - Part I:

- numerical model and validations. *Powder Technology*, 224:374–389, 2012. (Cited on pages 12, 19, 26, 34, and 212.)
- [203] Anthony Wachs, Markus Uhlmann, Jos Derksen, and Damien P. Huet. 7 - modeling of short-range interactions between both spherical and non-spherical rigid particles. In Shankar Subramaniam and S. Balachandar, editors, *Modeling Approaches and Computational Methods for Particle-Laden Turbulent Flows*, Computation and Analysis of Turbulent Flows, pages 217–264. Academic Press, 2023. (Cited on page v.)
- [204] J Walter, A-V Salsac, D Barthès-Biesel, and Patrick Le Tallec. Coupling of finite element and boundary integral methods for a capsule in a stokes flow. *International journal for numerical methods in engineering*, 83(7):829–850, 2010. (Cited on page 82.)
- [205] Siqiang Wang, Dzianis Marmysh, and Shunying Ji. Construction of irregular particles with superquadric equation in dem. *Theoretical and Applied Mechanics Letters*, 10(2):68–73, 2020. (Cited on pages 32 and 34.)
- [206] Z Wang, R Lu, W Wang, FB Tian, JJ Feng, and Y Sui. A computational model for the transit of a cancer cell through a constricted microchannel. *Biomechanics and Modeling in Mechanobiology*, pages 1–15, 2023. (Cited on page 103.)
- [207] Z Wang, Y Sui, A-V Salsac, D Barthès-Biesel, and W Wang. Motion of a spherical capsule in branched tube flow with finite inertia. *Journal of Fluid Mechanics*, 806:603–626, 2016. (Cited on pages 103, 106, 124, 138, and 179.)
- [208] Z Wang, Y Sui, A-V Salsac, D Barthès-Biesel, and W Wang. Path selection of a spherical capsule in a microfluidic branched channel: Towards the design of an enrichment device. *Journal of Fluid Mechanics*, 849:136–162, 2018. (Cited on pages 103, 106, 110, 124, and 138.)
- [209] J.R. Williams and R. O’Connor. A linear complexity intersection algorithm for discrete element simulation of arbitrary geometries. *Engineering Computations*, 12(2):185–202, 1995. (Cited on pages 15 and 34.)
- [210] Dong-wen Wu, Yu-meng Li, and Fen Wang. How long can we store blood samples: a systematic review and meta-analysis. *EBioMedicine*, 24:277–285, 2017. (Cited on page 2.)

- [211] Z Yan, SK Wilkinson, EH Stitt, and M Marigo. Discrete element modelling (dem) input parameters: understanding their impact on model predictions using statistical analysis. *Computational Particle Mechanics*, 2(3):283–299, 2015. (Cited on pages [xiv](#), [20](#), [28](#), and [29](#).)
- [212] Alireza Yazdani and Prosenjit Bagchi. Three-dimensional numerical simulation of vesicle dynamics using a front-tracking method. *Physical Review E*, 85(5):056308, 2012. (Cited on pages [64](#) and [75](#).)
- [213] Alireza Yazdani and Prosenjit Bagchi. Influence of membrane viscosity on capsule dynamics in shear flow. *Journal of fluid mechanics*, 718:569–595, 2013. (Cited on page [82](#).)
- [214] Alireza ZK Yazdani and Prosenjit Bagchi. Phase diagram and breathing dynamics of a single red blood cell and a biconcave capsule in dilute shear flow. *Physical Review E*, 84(2):026314, 2011. (Cited on pages [xviii](#), [64](#), [85](#), [87](#), [88](#), and [109](#).)
- [215] Ling Zhan, Chong Peng, Bingyin Zhang, and Wei Wu. A surface mesh represented discrete element method (smr-dem) for particles of arbitrary shape. *Powder Technology*, 377:760–779, 2021. (Cited on pages [15](#) and [34](#).)
- [216] Junfeng Zhang, Paul C Johnson, and Aleksander S Popel. An immersed boundary lattice boltzmann approach to simulate deformable liquid capsules and its application to microscopic blood flows. *Physical biology*, 4(4):285, 2007. (Cited on page [64](#).)
- [217] Junfeng Zhang, Paul C Johnson, and Aleksander S Popel. Red blood cell aggregation and dissociation in shear flows simulated by lattice boltzmann method. *Journal of biomechanics*, 41(1):47–55, 2008. (Cited on page [64](#).)
- [218] F Zhao and BGM van Wachem. A novel quaternion integration approach for describing the behaviour of non-spherical particles. *Acta Mechanica*, 224(12):3091–3109, 2013. (Cited on page [25](#).)
- [219] Hong Zhao, Amir HG Isfahani, Luke N Olson, and Jonathan B Freund. A spectral boundary integral method for flowing blood cells. *Journal of Computational Physics*, 229(10):3726–3744, 2010. (Cited on pages [2](#), [64](#), [104](#), and [181](#).)

- [220] Ou-Yang Zhong-Can and Wolfgang Helfrich. Bending energy of vesicle membranes: General expressions for the first, second, and third variation of the shape energy and applications to spheres and cylinders. *Physical Review A*, 39(10):5280, 1989. (Cited on page 108.)
- [221] Lailai Zhu and Luca Brandt. The motion of a deforming capsule through a corner. *Journal of Fluid Mechanics*, 770:374–397, 2015. (Cited on pages 83, 105, 107, 112, 113, 121, and 137.)
- [222] Lailai Zhu, Cecilia Rorai, Dhruvaditya Mitra, and Luca Brandt. A microfluidic device to sort capsules by deformability: a numerical study. *Soft Matter*, 10(39):7705–7711, 2014. (Cited on page 103.)
- [223] Zhixian Zhu, Dan Wu, Shuang Li, Yu Han, Nan Xiang, Cailian Wang, and Zhonghua Ni. A polymer-film inertial microfluidic sorter fabricated by jigsaw puzzle method for precise size-based cell separation. *Analytica Chimica Acta*, 1143:306–314, 2021. (Cited on page 104.)

Appendix A

On the contact detection of solid bodies of arbitrary shapes

Detecting contacts between potentially colliding bodies is a non-trivial problem responsible for the majority of the computational resources usage of DEM simulations. As detecting contact between two rigid bodies of arbitrary shapes is so expensive, a so-called “broad-phase detection” is implemented in order to call the costly “narrow-phase detection” only when needed. In the following, we present strategies commonly implemented in smooth DEM solvers for the broad- and narrow-phase detections.

A.1 Broad-phase detection

Consider a granular system of N particles: if collision is tested between every pair of particles, the resulting complexity is $O(N^2)$. Even when testing for collision between two particles is very fast – e.g. in the case of spherical particles – this quadratic scaling would lead to extremely long simulations as the number of particles increases beyond $O(10^3)$. A way to improve this naive approach is to take advantage of the locality of the contact detection problem: two particles can only collide if they are located close to each other. As such, several broad-phase detection strategies can be implemented: their goal is to limit the number of calls to a contact detection algorithm only to the particles j that are located sufficiently close to a target particle i . Figure A.1 illustrates two of these approaches: (i) the cell discretization, or link-cell, and (ii) the bounding boxes.

In the link-cell approach, space is discretized into subregions of size less than one order of magnitude greater than the characteristic length of the particles. If two particles are in contact, they must belong to the same cell or be located in a neighboring cell less than one particle characteristic length away from the cell boundary. In figure A.1, particles that can

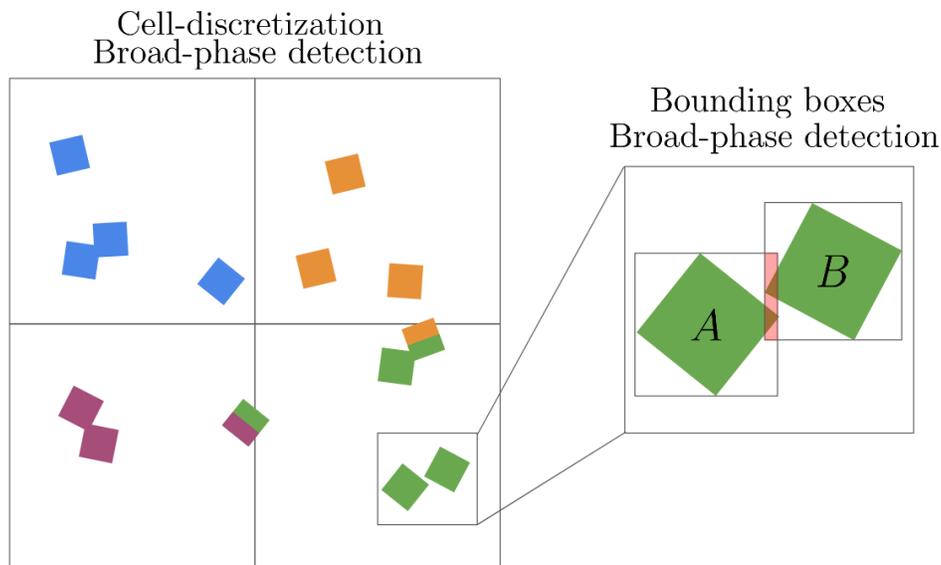


Figure A.1: Two stages of broad-phase detection: space-discretization in cells and bounding boxes. Particles with the same color will undergo the bounding boxes broad-phase detection, and particles which bounded-boxes overlap will launch narrow-phase detection.

potentially collide share the same color: note that particles located close to the cell boundaries are collision candidates for particles of several cells. At the expense of increasing the memory footprint of the DEM solver, this simple link-cell approach reduces the computational complexity of the contact detection problem from $O(N^2)$ to $O(N)$: it is an extremely powerful approach that is implemented in most DEM solver today, including in the solver Grains3D used in this thesis [168].

A finer level of broad-phase contact detection is called “bounding boxes” and is sketched on the right side of figure A.1. In this approach, the contact between two particles A and B is only further investigated if they are contained in overlapping boxes. These bounding boxes can be aligned with the axes of the reference frame – in which cases they are referred to as Axes Aligned Bounding Boxes, or AABB – or rotated so that they match the particle’s body frame – in which case they are referred to as Oriented Bounding Boxes, or OBB. A variant of the bounding boxes approach is to use overlapping circumscribed spheres: this is the strategy implemented in Grains3D.

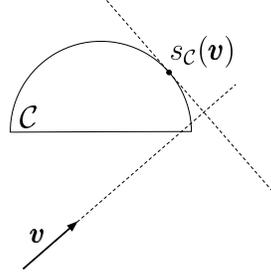


Figure A.2: An example of a support mapping $s_C(\mathbf{v})$ for the convex \mathcal{C} in the direction \mathbf{v} .

A.2 Narrow-phase detection

A.2.1 Prerequisites

The narrow-phase detection in the case of particles of arbitrary convex shapes is performed by the GJK algorithm detailed in the next section. This algorithm uses two mathematical concepts that we define below: the support mapping and the Minkowski sum of convex sets.

First, the support mapping $s_C(\mathbf{v})$ of the convex \mathcal{C} in the direction \mathbf{v} is the farthest point of \mathcal{C} in the direction \mathbf{v} . Formally, we can write

$$s_C(\mathbf{v}) = \sup\{\mathbf{x} \cdot \mathbf{v}, \mathbf{x} \in \mathcal{C}\}, \quad (\text{A.1})$$

where \cdot denotes the inner product. An example of support mapping is shown in figure A.2. Support mappings for common convex shapes, such as spheres, boxes, cylinders and arbitrary polytopes are easily derived: a non-exhaustive list of common support mappings is found in [198].

Then, the Minkowski sum of two convex sets A and B is defined as the set of the sum of every points in A and B :

$$A + B = \{\mathbf{x} + \mathbf{y}, \mathbf{x} \in A \text{ and } \mathbf{y} \in B\}, \quad (\text{A.2})$$

and the concept of Minkowski difference is defined analogously:

$$A - B = \{\mathbf{x} - \mathbf{y}, \mathbf{x} \in A \text{ and } \mathbf{y} \in B\}. \quad (\text{A.3})$$

A sketch of the Minkowski sum and difference of two convex sets is shown in figure A.3.

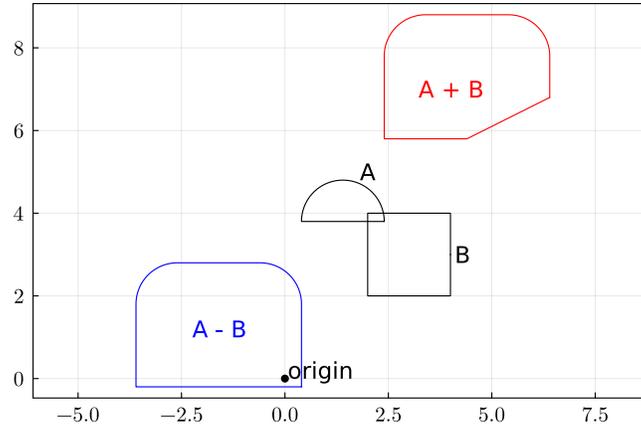


Figure A.3: Minkowski sum and difference of two convex bodies.

A.2.2 The GJK distance algorithm

In 1988, Gilbert, Johnson and Keerthi designed an algorithm which aims to compute the distance and the closest points between two convex objects [62]. In the literature and in this thesis, this algorithm is referred to as the GJK algorithm. At the core of the GJK algorithm lies the following property: the distance between the origin and the Minkowski difference $A - B$ of two convex objects A and B is equal to the distance between A and B . In particular, when two convex objects overlap, their Minkowski difference contains the origin, in which case the GJK algorithm returns zero.

The GJK algorithm is an iterative procedure that produces simplices \mathcal{W}_k contained in the convex $\mathcal{C} = A - B$, and which vertices $\{\mathbf{w}_i, i \in \llbracket 1, d + 1 \rrbracket\}$ lie closer and closer to the origin, where d is the dimension. If A and B are overlapped, the GJK algorithm terminates after a simplex contained in \mathcal{C} and enclosing the origin has been found. Otherwise, termination occurs after a desired accuracy is reached: note that in the case of polytopes, the exact solution is found in a finite amount of steps. A pseudocode for the GJK algorithm and its corresponding first iterations are shown in algorithm A.1 and figure A.4. A more in-depth description of this algorithm, as well as a discussion on termination, pathologic cases and backup procedures are available in [197, 198].

A.2. Narrow-phase detection

Algorithm A.1 The GJK distance algorithm

Choose a maximum number of iterations N_{max} and a tolerance ϵ
 Choose a vector \mathbf{v}_0 pointing towards a convex \mathcal{C} , of support function $s_{\mathcal{C}}$
 Compute the point $w_0 = s_{\mathcal{C}}(-\mathbf{v}_0)$
 Set $\mathcal{W}_0 = \emptyset$
 Set $\mathbf{v}_{-1} = (1 + 2\epsilon)\mathbf{v}_0$
 Set $k = 1$
while $k < N_{max}$ **or** $\|\mathbf{v}_{k-1} - \mathbf{v}_{k-2}\| > \epsilon$ **do**
 Compute \mathbf{v}_k as the point of the simplex $\mathcal{W}_{k-1} \cup w_{k-1}$ that lies closest to the origin
 Set \mathcal{W}_k as the smallest subset of $\mathcal{W}_{k-1} \cup w_{k-1}$ that contains \mathbf{v}_k
 Compute $w_k = s_{\mathcal{C}}(-\mathbf{v}_k)$
 Set $k = k + 1$
end while
 Return w_k , the point of \mathcal{C} closest to the origin

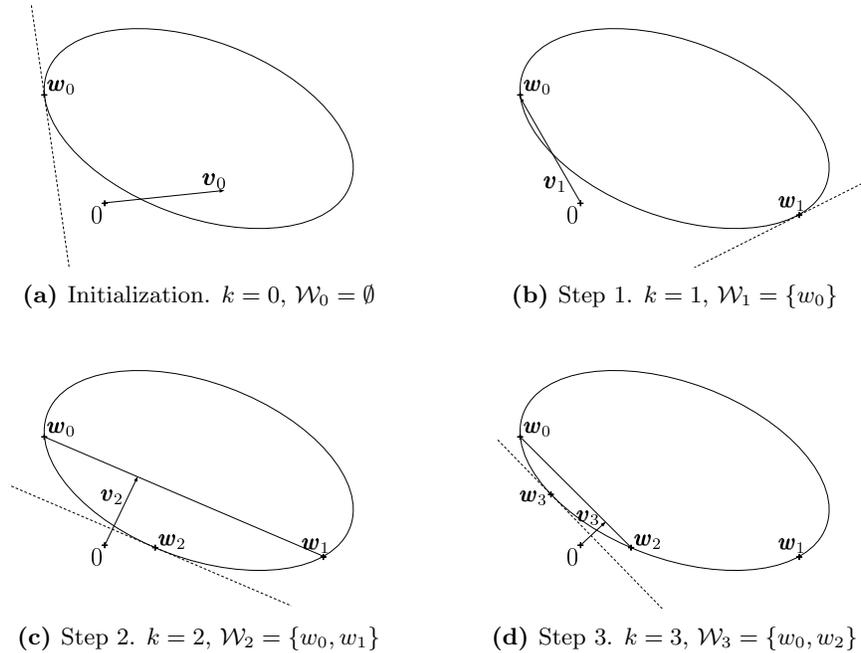


Figure A.4: The first iterations of the GJK distance algorithm.

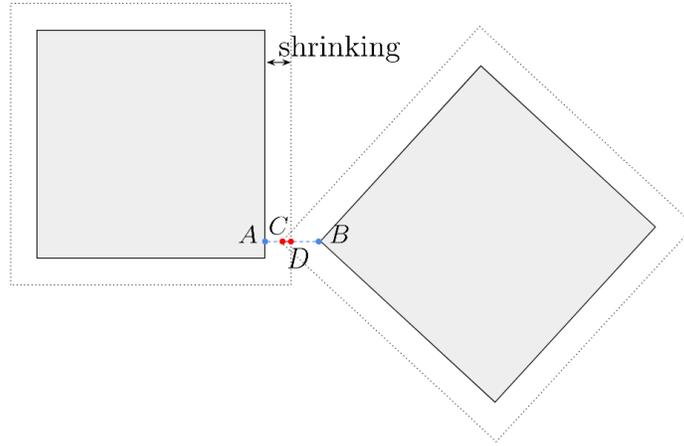


Figure A.5: When two particles are colliding, the access to the contact information is performed by a second call to the GJK algorithm on slightly shrunk versions of the particles. In this schematic (not to scale), the dotted lines represent the original shape of the particles and the solid lines correspond to their shrunk version.

A.2.3 Application to DEM simulations

In smooth DEM simulations, additional information about collision is sought, such as the penetration depth δ_n and the point of application of the contact force. To obtain these quantities, two approaches exist in the literature. The first approach is to call the GJK algorithm again, on a slightly shrunk version of the two particles. In their shrunk configuration, the two particles of interest do not collide anymore and the GJK algorithm returns the two closest points A and B as well as the distance AB , as shown in figure A.5. The closest points A and B are then mapped back to the initial shapes of the colliding bodies, which we denote as C and D in figure A.5, and the penetration depth δ_n is the distance CD . This is the approach adopted in the DEM solver Grains3D used in part I of this thesis [202]. The second approach consists in outputting the last simplex in the GJK algorithm and to extract the closest point from this information, using a procedure called the Expanding Polytope Algorithm (EPA) [180].

Finally, to detect contact between non-convex particles, a property of interest is that non-convex bodies can be decomposed into a set of convex “elementary” bodies. As such, collision between two non-convex particles is performed by calling the GJK algorithm on each pair of their convex elementary particles [169].

Appendix B

Granular avalanches of entangled rigid particles: supplementary material

B.1 Non-dimensional displacement of the center of mass of the whole granular assembly

In a dam break setup, the measurement of the run-out distance may be prone to uncertainties stemming from the dynamics of a few particles located at the front of the avalanche, in particular when the system is small. While we attempted to circumvent this limitation through a systematic image processing routine and were conservative with the associated error bars – see section Section 3.2.3 – Degaetano et al. [42] instead report the position of the center of mass of the granular assembly, a quantity that is far less noisy and less dependent on the dynamics of a few particles at the front of the avalanche. In fact, in their study of two-dimensional avalanches of bidisperse particles, Degaetano et al. obtain clearer trends when plotting the center of mass of the assembly rather than the run-out distance. In this appendix, we investigate if reporting the position of the center of mass of our assemblies of monodisperse non-convex particles rather than their run-out distance yields a similar improvement of the interpretation of our simulation data.

Figure B.1 shows the non-dimensional final displacement of the center of mass as a function of the initial aspect ratio, as well as the non-dimensional run-out distance in our simulations, in our experiments and in the experiments with spherical particles of Lajeunesse et al. [118] and with grit particles of Balmforth et al. [14]. Focusing on the non-dimensional displacement of the center of mass, we observe a very pronounced and clearly visible transition around $H_0/L_0 = 3$ where the slope suddenly changes from 2 to $2/3$. We show below how the numerical values of these slopes can be recovered by means of a simple model. While in figure B.1 the slopes of the run-out

B.1. Non-dimensional displacement of the center of mass of the whole granular assembly

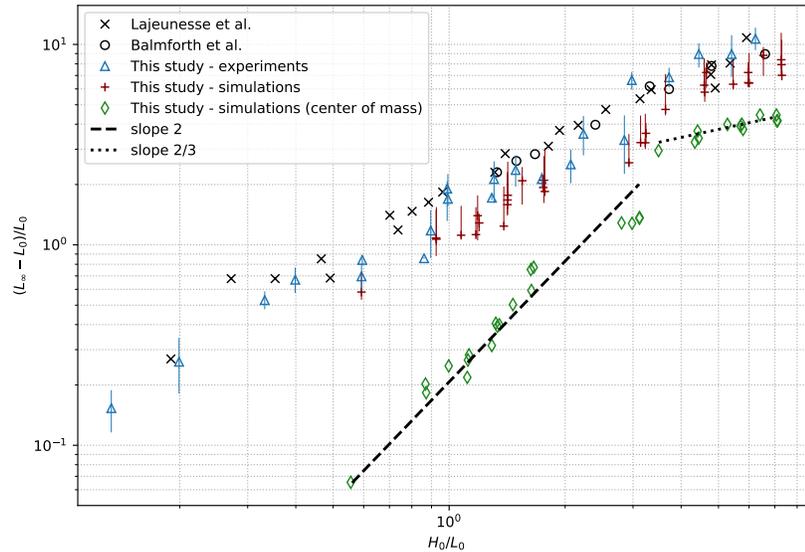


Figure B.1: Non-dimensional final x-displacement of the center of mass of the granular assembly (diamond symbols) and non-dimensional run-out distance (all other symbols) as a function of the initial aspect ratio.

distance in our simulations and experiments seem constant and equal to 1, Lajeunesse et al. consider aspect ratios of up to 20 and reveal that indeed at $H_0/L_0 \approx 3$ a slope transition from 1 to $2/3$ occurs. Figure B.1 confirms that in our simulations of monodisperse non-convex particles the displacement of the center of mass seems to be a more relevant quantity to identify the regime transition at $H_0/L_0 \approx 3$. Therefore, focusing on the center of mass of the assembly improves the interpretation of the simulation data. Please note that this observation is presumably true for any particle shape.

We now show with a simple model how the slopes in figure B.1 can be recovered. For both regimes of small and large aspect ratios, we make the strong assumption that the slope of the avalanched region is constant, as formulated by Lajeunesse et al. [118] and roughly verified experimentally in figure 3.3. This assumption seems to not be valid in the case of buckling collapses – see figure 3.4 – but since this regime occurs far less often than the top-driven collapse we can still consider the assumption of constant slope satisfied.

B.1. *Non-dimensional displacement of the center of mass of the whole granular assembly*

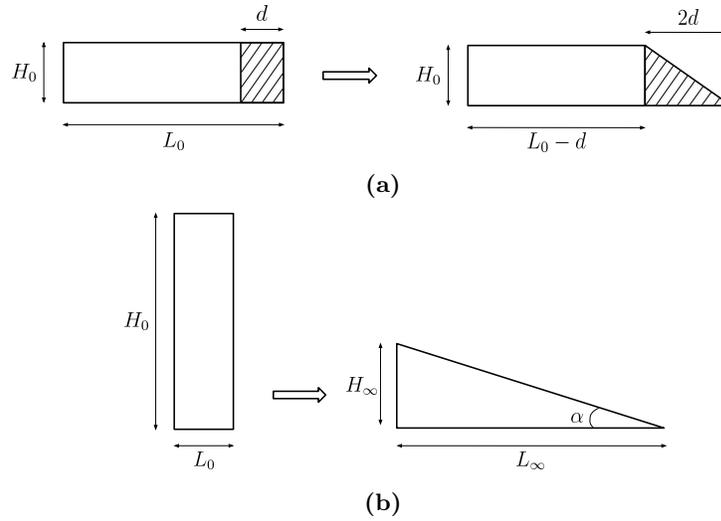


Figure B.2: Schematics of the avalanche profiles of the granular assembly for low (a) and high (b) aspect ratios, under the assumption that the slope of the avalanched profile is constant for a given aspect ratio.

1. **Case of small aspect ratios:**

In this case, the shape of the granular assembly before and after the collapse is represented in figure B.2a. We make the additional assumption that only the particles located at a distance less than d away from the gate experience the avalanche and that d is proportional to H_0 (most likely we have $d \approx H_0$ but this is not needed for the dimensional scaling). This assumption is again roughly justified by the results in our experiments and simulations. The non-dimensional run-out distance $\Delta L/L_0 = (L_\infty - L_0)/L_0$ equals d/L_0 and is therefore proportional to H_0/L_0 , hence the slope of 1 observed in figure B.1. To estimate the non-dimensional x -displacement of the center of mass, $\Delta x/x_0 = (x_\infty - x_0)/x_0$, we need to first compute its final position x_∞ with respect to its initial position $x_0 = L_0/2$. Simply adding the positions of the centers of mass of the plain white rectangle and the hatched triangle on the right hand side of figure B.2a, and weighting

B.2. Probability density functions of the scaled contact force magnitude in various regimes

them appropriately, yields:

$$\begin{aligned}
 x_\infty &= \left(\frac{L_0 - d}{2}\right) \left(1 - \frac{d}{L_0}\right) + \left(L_0 + \frac{d}{3}\right) \left(\frac{d}{L_0}\right) \\
 &= x_0 \left(1 + \frac{1}{3} \frac{d^2}{L_0^2}\right) \\
 &\Rightarrow \frac{\Delta x}{x_0} \propto \left(\frac{H_0}{L_0}\right)^2
 \end{aligned} \tag{B.1}$$

which is in agreement with the slope of 2 observed in figure B.1 for the center of mass for aspect ratios $H_0/L_0 \lesssim 3$.

2. Case of large aspect ratios:

The case of high aspect ratios is schematically represented in figure B.2b, where α denotes the angle of the collapsed profile, which may vary with the aspect ratio H_0/L_0 . Mass conservation yields $H_0 L_0 = H_\infty L_\infty / 2$, and since $H_\infty = \tan \alpha L_\infty$ we obtain:

$$\frac{\Delta L}{L_0} = \frac{L_\infty - L_0}{L_0} = \sqrt{\frac{2H_0}{L_0 \tan \alpha}} \tag{B.2}$$

Lajeunesse et al. have shown experimentally and by means of a force balance model [118] that the non-dimensional run-out distance $\Delta L/L_0$ scales as $(H_0/L_0)^{\frac{2}{3}}$ for large aspect ratios. This result implies that $\tan \alpha$ is proportional to $(H_0/L_0)^{-\frac{1}{3}}$. Focusing now on the center of mass, we have $x_0 = L_0/2$ and $x_\infty = L_\infty/3$, and we use the mass conservation identity again to yield:

$$x_\infty = \frac{1}{3} \sqrt{\frac{2H_0 L_0}{\tan \alpha}} \Rightarrow \frac{\Delta x}{x_0} = \frac{x_\infty - x_0}{x_0} = \frac{1}{3} \sqrt{\frac{2H_0}{L_0 \tan \alpha}} - 1 \tag{B.3}$$

Since $\tan \alpha$ is proportional to $(H_0/L_0)^{-\frac{1}{3}}$ we recover the slope of 2/3 that we observe in figure B.1 for large aspect ratios.

B.2 Probability density functions of the scaled contact force magnitude in various regimes

Figure B.3 shows the average PDFs of $F_c/\langle F_c \rangle$ in the top-driven collapse and in the buckling collapse (a-b) and in the intermittent-stable regime and in the intermittent-collapsed regime (c-d). Each solid line is an average of

B.2. Probability density functions of the scaled contact force magnitude in various regimes

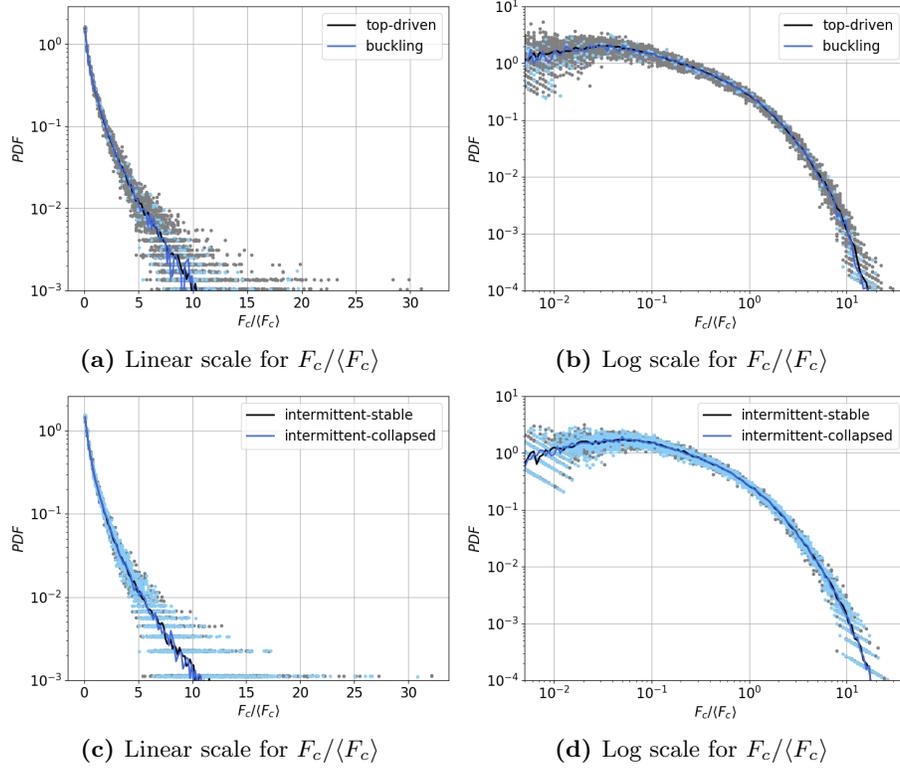


Figure B.3: Probability density functions (PDFs) of the contact forces F_c just before the gate is opened, in the cases of (a-b) top-driven regime (black) with buckling regime (blue); and (c-d) intermittent regime leading to stable (black) and collapsed (blue) behaviours.

N simulations. In figures B.3a and B.3b, $N_{\text{top-driven}} = 50$ and $N_{\text{buckling}} = 6$, while in figures B.3c and B.3d, $N_{\text{stable}} = 26$ and $N_{\text{collapsed}} = 18$. The low number of buckling cases is due to the scarcity of this regime in our simulations – the buckling collapse occurs in about 10% of the simulations located in the buckling region of figure 3.6. Overall, figure B.3 shows that the average PDF of $F_c/\langle F_c \rangle$ in the buckling collapse and the average PDF of $F_c/\langle F_c \rangle$ in the top-driven collapse overlap very well, and so do the average PDF of $F_c/\langle F_c \rangle$ in the intermittent-stable regime and the average PDF of $F_c/\langle F_c \rangle$ in the intermittent-collapsed regime. These results underline the fact that the distribution of force magnitude does not enable us to *a priori* discriminate these regimes from one another and is not a relevant descriptor in predicting the behaviour of the granular avalanche of our cross-shaped entangled particles.

B.3 Raw data for the analysis of the intermittent regime

Table B.1 shows the raw average contact data for a granular column in the intermittent regime. The forces on vertical walls are not included, since the Janssen effect is found to carry on average less than 0.3% of the total weight of the column.

Table B.1: Average data for the intermittent cases about to collapse (top) and remain stable (bottom). The symbol σ stands for the standard deviation of the data.

Collapsed case	Whole space	Half space close to the gate	Quarter space close to the top of the gate
Coordination number	8.20 $\sigma = 0.07$	8.20 $\sigma = 0.07$	7.96 $\sigma = 0.11$
Number of contact points on the ground	79.3 $\sigma = 4.0$	39.3 $\sigma = 3.4$	0
Number of contact points on side walls	169.2 $\sigma = 5.7$	84.3 $\sigma = 3.6$	33.0 $\sigma = 3.4$
Number of contact points on the gate	72.8 $\sigma = 4.5$	72.8 $\sigma = 4.5$	26.1 $\sigma = 3.4$
Stable case	Whole space	Half space close to the gate	Quarter space close to the top of the gate
Coordination number	8.21 $\sigma = 0.07$	8.21 $\sigma = 0.09$	8.02 $\sigma = 0.12$
Number of contact points on the ground	81.2 $\sigma = 4.6$	40.9 $\sigma = 3.1$	0
Number of contact points on side walls	170.3 $\sigma = 5.3$	85.4 $\sigma = 4.0$	30.9 $\sigma = 2.5$
Number of contact points on the gate	72.1 $\sigma = 3.1$	72.1 $\sigma = 3.1$	28.1 $\sigma = 2.7$

Appendix C

Adaptive immersed boundary stencils

C.1 Method

In the default implementation of our adaptive FTM, a constant grid size is imposed on the support of the regularized Dirac distributions. To our knowledge, all adaptive IBM implementations impose this constraint of the support of the regularized Dirac distributions [68, 174, 200]. In most cases, especially in flows featuring fixed solids, this limitation is not constraining because a maximum level of refinement is required along the whole surface of the solid in order to adequately resolve the flow features. However, in some flow configurations, the Eulerian mesh needs to be very fine on a small patch of the surface of the solid, e.g. when resolving the lubrication layer of two colliding bodies [59?]. In the case of capsule dynamics, a similar case was encountered in Section 4.5.1 where the Eulerian cells located near the tail of the capsule do not need to be at a fine level of refinement (see figure 4.15). In order to alleviate this refinement constraint, we discuss in this appendix an implementation strategy allowing the membrane to span different grid levels.

We begin by introducing the terminology related to the adaptive grid. In Basilisk, the grid is a collection of cells of different size, which are arranged in a tree structure, as represented for a two dimensional grid in figure C.1. A cell located at the end of a branch of the tree is called a “leaf” cell, while other cells are referred to as “parent” cells. In figure C.1, filled circles denote leaf cells and open circles denote parent cells. Leaf cells form a perfect tiling of the space, and physical fields such as velocity and pressure are defined on leaf cells only. Computationally, parent cells are used only to point to their children cells.⁹ In this section, we also denote by “stencil” or “IBM stencil” the support of the discrete regularized Dirac distribution, which is – in the

⁹There are some exceptions to this statement, e.g. when prolongation and restrictions operators are used in order to refine or coarsen the grid, or during a multigrid cycle.

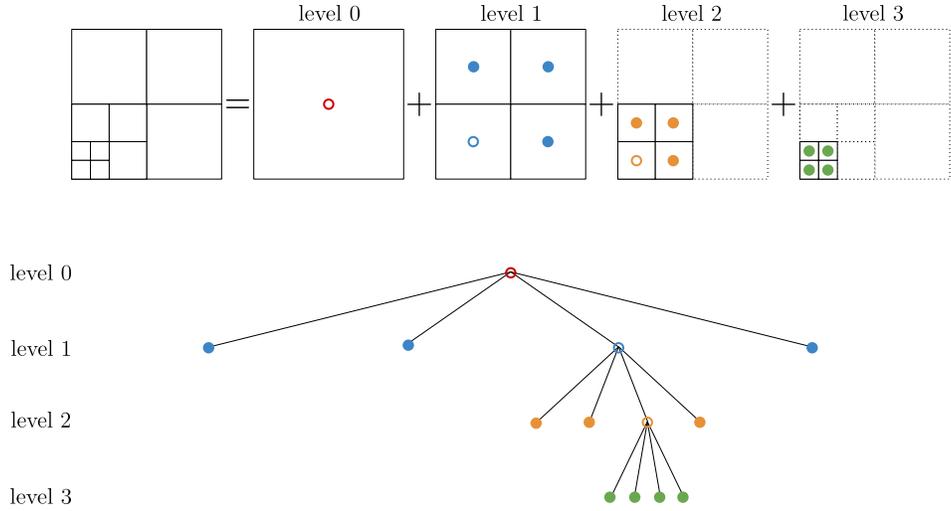


Figure C.1: Tree structure of a quadtree grid. The filled circles represent leaf cells while the open circles represent parent cells.

case of a constant grid – a square of size 4×4 grid cells in two dimensions and a cube of size $4 \times 4 \times 4$ grid cells in three dimensions.

We distinguish two configurations of adaptive IBM stencils: (i) the cell containing the center of the regularized Dirac distribution – in our case, the Lagrangian node on the membrane triangulation – is at the lowest refinement level encountered in the stencil; and (ii) the stencil contains one or more cells at a lower refinement level than that containing the Lagrangian node. These two cases are represented in figure C.2a and C.2b, respectively. The treatment of case (i) is straightforward: if during a velocity interpolation or a force-spreading operation a non-leaf cell is encountered in the IBM stencil, the operation of interest is evenly distributed to the daughter cells. We remind the property of unit integral of the regularized Dirac distribution:

$$\int_{\Omega} \delta(\mathbf{x} - \mathbf{x}_0) d\Omega = 1 \quad (\text{C.1})$$

where Ω is the volume of the computational domain and \mathbf{x}_0 is the location of the Lagrangian node, i.e. the center of the regularized Dirac distribution δ . In order to conserve this property, the velocity is averaged over the daughter cells in the case of a velocity interpolation, while the forcing term is multiplied by the volume of each daughter cell later in the Navier-Stokes solver. A pseudo code for each interpolation and spreading operation is

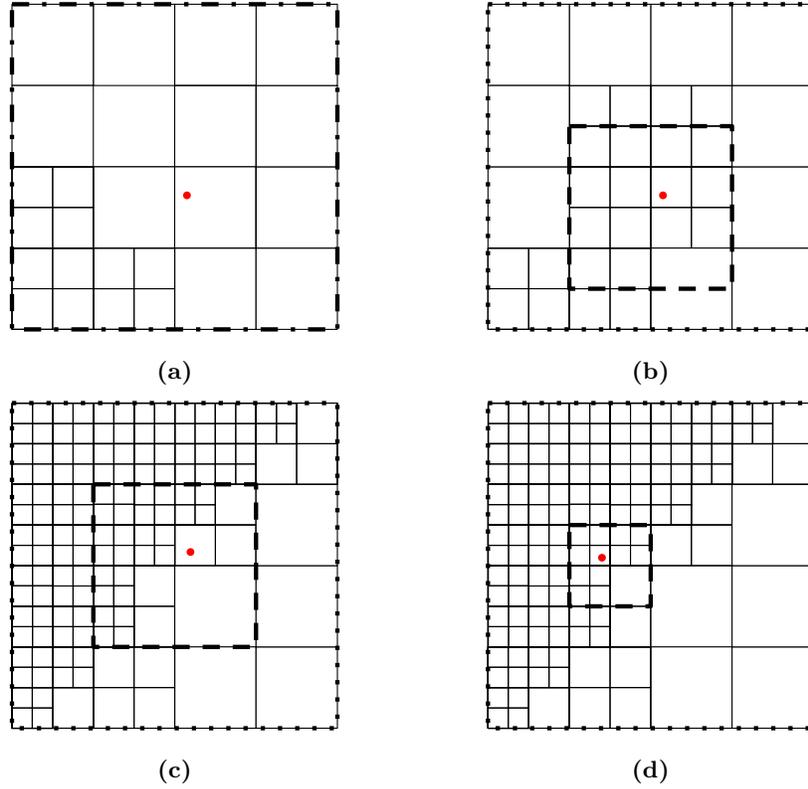


Figure C.2: Examples of adaptive IBM stencils configurations. (a) The cell containing \mathbf{x}_0 is at the largest level of the stencil. (b) The cell containing \mathbf{x}_0 is at the lowest level of the stencil. (c) The cell containing \mathbf{x}_0 is neither at the largest nor the lowest level of the stencil. (d) Two successful increases of the stencil size are necessary to obtain a consistent stencil. The dashed line represents the stencil size based on the level of the cell containing \mathbf{x}_0 , while the dotted line represents the effective stencil size. The red dot corresponds to the location \mathbf{x}_0 of the center of the regularized Dirac distribution.

provided in Algorithms C.1 and C.2.

The treatment of case (ii) is more challenging. If the procedure used for case (i) is directly applied to case (ii), unphysical moments can be introduced in the flow as the isotropy of the stencil can be broken. Such a case is represented in figure C.2b, where a cell extends outside of the IBM stencil represented by a dashed line. To fix this issue, we can transform a case (ii) configuration into a case (i) configuration by requiring that the length of the stencil side is always $4\Delta_s$, where Δ_s is the size of the largest encountered grid cell. The larger stencil is represented in figure C.2 by a dotted line, while the original stencil is represented by a dashed line. Figure C.2c and C.2d display two more commonly encountered configurations. In the case of figure C.2d, the stencil size corresponds to that defined by grid cells two levels of refinement lower than that of the cell containing the Lagrangian node. The size $4\Delta_s$ of the isotropic stencil is determined using a recursive procedure prior to applying the velocity interpolation and force spreading routines presented in Algorithms C.1 and C.2.

Algorithm C.1 Recursive averaging of velocities on leaf cells

```

function AVERAGE_LEAVES_VELOCITIES(grid_cell, velocity_field, averaged_velocity, depth)
  if is_leaf(grid_cell) then
    averaged_velocity.x += velocity_field.x / 2dimension*depth
    averaged_velocity.y += velocity_field.y / 2dimension*depth
    if dimension > 2 then
      averaged_velocity.z += velocity_field.z / 2dimension*depth
    end if
  else
    for  $k \leftarrow 1$  to grid_cell.number_children do
      average_leaves_velocities(grid_cell.children[k], velocity_field,
        averaged_velocity, depth + 1)
    end for
  end if
end function

```

C.2 Proof of concept and discussion

To validate the adaptive IBM method proposed above, we consider a simple configuration featuring a two-dimensional capsule in a shear flow. Three

Algorithm C.2 Recursive spreading of node force onto leaf cells

```

function SPREAD_FORCE_ON_LEAVES(grid_cell, forcing_field, node_force,
depth)
  if is_leaf(grid_cell) then
    forcing_field.x += node_force.x
    forcing_field.y += node_force.y
    if dimension > 2 then
      forcing_field.z += node_force.z
    end if
  else
    for  $k \leftarrow 1$  to grid_cell.number_children do
      spread_force_on_leaves(grid_cell.children[k], forcing_field,
node_force, depth + 1)
    end for
  end if
end function

```

simulations are performed: (i) a constant grid at level 7 is imposed everywhere, (ii) a constant grid at level 6 is imposed everywhere, (iii) an adaptive grid of minimal level 6 and maximum level 7 is imposed and remains fixed in time. A visualization of case (iii) is provided in figure C.3. In all cases, we measure the Taylor deformation parameter D :

$$D = \frac{r_{\max} - r_{\min}}{r_{\max} + r_{\min}}, \quad (\text{C.2})$$

where r_{\max} and r_{\min} are the maximum and minimum radii of the capsule, respectively. The results are plotted in figure C.4. All three considered configurations are in very good agreement, confirming that the proposed adaptive IBM stencils lead to satisfactory results in the simple configuration considered. Moreover, other arbitrary grid refinements were imposed than that shown in figure C.3, such as a coarse mesh for $y < 0$ and a fine mesh for $y > 0$. All lead to similar satisfactory results.

Besides these promising results, further investigation is needed both on the theoretical and computational aspects. First, it is important to study analytically and empirically the order of accuracy of the present method. In particular, when the stencil spans different levels of grid refinement, the velocity interpolation and the force spreading operators are essentially taking the convolution of two functions known at different accuracies: a regularized Dirac distribution which accuracy corresponds to that of the largest cell

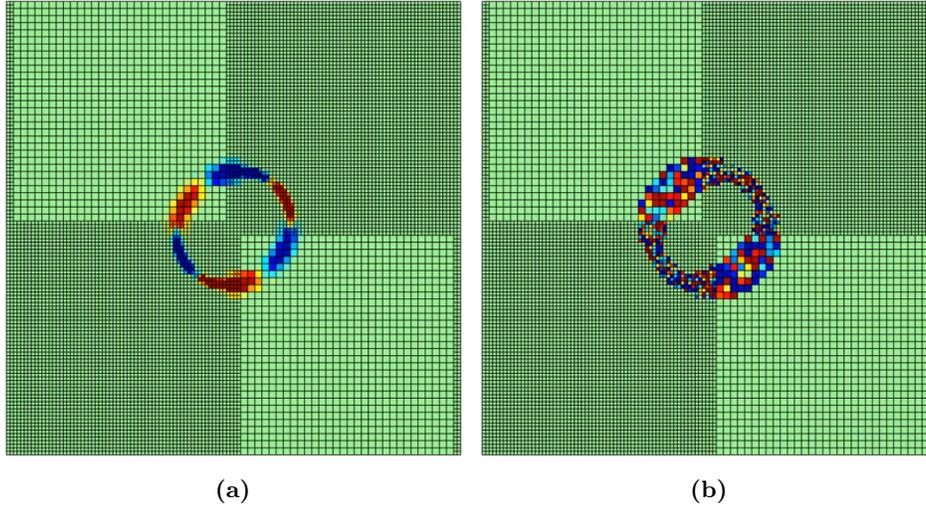


Figure C.3: Visualization of the body force term in the x -direction (left) and the IBM stencils defined by a noisy field (right) in the case of a two-dimensional capsule in a shear flow where a forced grid refinement is imposed in parts of the computational domain.

in the stencil, and the velocity and body force fields which are known to higher accuracies in the remaining cells of the stencil. Another question of theoretical interest is that of the stability: could the proposed adaptive IBM stencils deteriorate the overall stability of the method? These theoretical questions are left for future work due to time constraints.

Computationally, the goal of this adaptive IBM method is to reduce the number of grid cells in order to speed-up the overall simulation. However the current method also requires additional operations to be performed, mainly during two stages: (i) when the effective size of the adaptive IBM stencil is determined, especially in parallel simulation as communications are needed at every call of a recursive procedure¹⁰; and (ii) when performing a velocity interpolation or a force spreading operation over a stencil spanning cells of different sizes, as more than the typical 16 (in two dimensions) or 64 (in three dimensions) cells need to be considered. Estimating the computational benefit and overload of the present adaptive IBM method is important as it allows to decide when it should and should not be used. We expect that both cases can occur depending on specific configurations.

¹⁰In our implementation, we ensured that the exact number of MPI communication events related to the determination of the size of all the IBM stencils is equal to the Δl_{\max} , where Δl is the difference between the maximum and minimum refinement levels in a given stencil, and Δl_{\max} the maximum value of Δl over the whole set of IBM stencils.

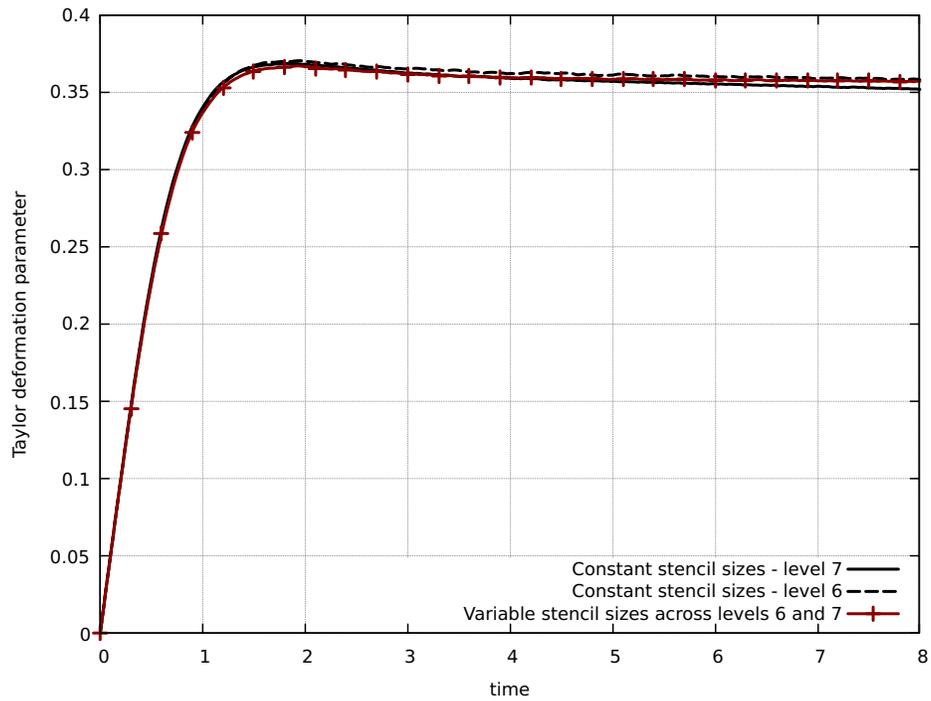


Figure C.4: Taylor deformation parameter of a two-dimensional capsule in a shear flow where a forced grid refinement is imposed in parts of the computational domain. Three cases are considered: (i) the membrane is at a constant fine level of refinement (solid black line), (ii) the membrane is at a constant coarse level of refinement (dashed black line), (iii) the membrane spans two levels of refinement (red line).

Appendix D

Motion and deformation of capsules through a corner in the inertial and non-inertial regimes: supplementary material

D.1 Time evolution of capsule areas in the case of two interacting capsules

Figure D.1 shows the evolution of the areas of the leading and the trailing capsules in the non-inertial regime (figure D.1a), as well as at $Re = 25$ and $Re = 50$ where the initial interspacing d_0 is 1 (figure D.1b) and 0.25 (figure D.1c).

D.2. Train of capsules at large initial interspacings

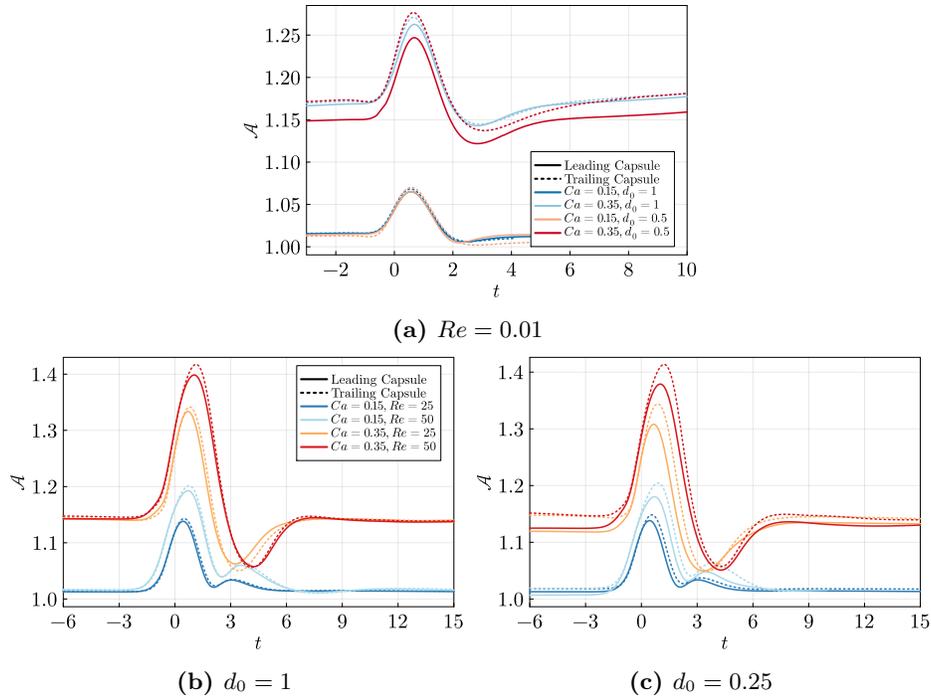


Figure D.1: Temporal evolution of the reduced area \mathcal{A} of the leading and the trailing capsules at various interspacings d_0 and various Capillary and Reynolds numbers Ca and Re .

D.2 Train of capsules at large initial interspacings

We provide in figure D.2 the time evolution of the area and velocity of each capsule in a train of 10 capsules flowing through a corner at $Re = 50$, $Ca = 0.35$ and a reduced initial interspacing between each capsule $d_0 = 0.125$. As can be noted in this figure, the capsules in this regime do not interact as the area and velocity evolution of each capsule is almost identical.

D.2. Train of capsules at large initial interspacings

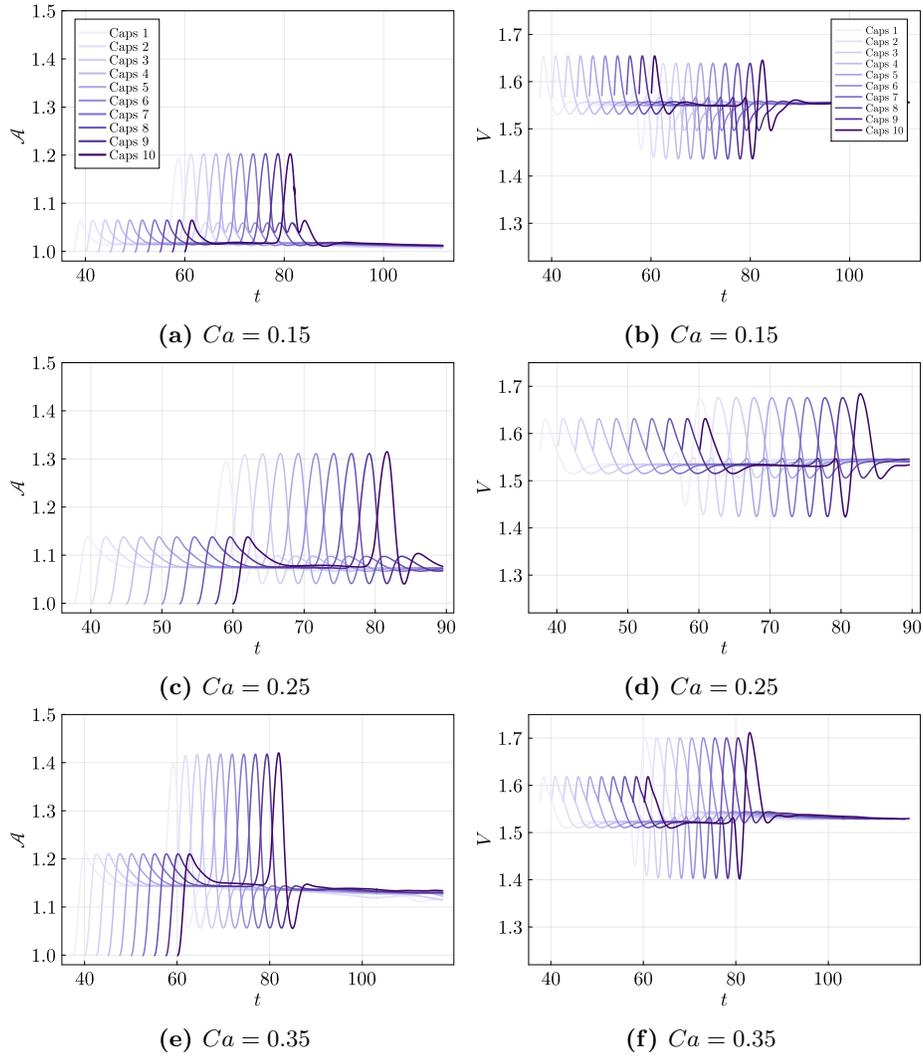


Figure D.2: Time evolution of the reduced areas and velocities of ten capsules at $Re = 50$ and $d_0 = 1$ for $Ca = 0.15, 0.25$ and 0.35 .