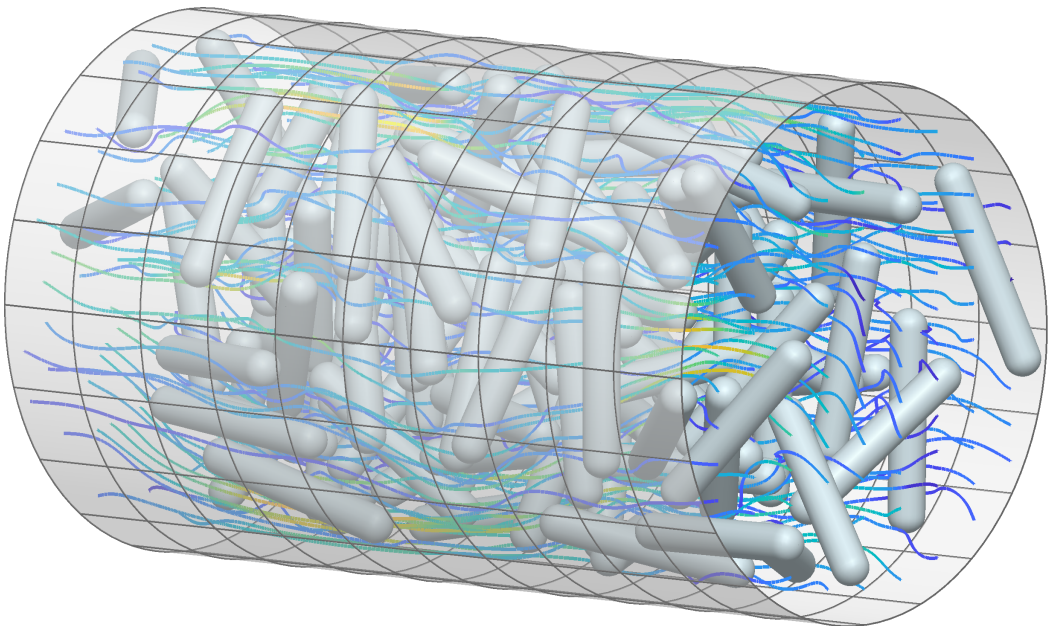


Doctoral Thesis in Applied and Computational Mathematics

Accurate quadrature and fast summation in boundary integral methods for Stokes flow

JOAR BAGGE



Accurate quadrature and fast summation in boundary integral methods for Stokes flow

JOAR BAGGE

Academic Dissertation which, with due permission of the KTH Royal Institute of Technology, is submitted for public defence for the Degree of Doctor of Philosophy on Wednesday the 14th of June 2023, at 2:00 p.m. in F3, Lindstedtsvägen 26, Stockholm.

Doctoral Thesis in Applied and Computational Mathematics
KTH Royal Institute of Technology
Stockholm, Sweden 2023

© Joar Bagge

© Joar Bagge, Anna-Karin Tornberg, Paper A

© Joar Bagge, Anna-Karin Tornberg, Paper B

© Joar Bagge, Tomas Rosén, Fredrik Lundell, Anna-Karin Tornberg, Paper C

© Davood Saffar Shamshirgar, Joar Bagge, Anna-Karin Tornberg, Paper D

© Joar Bagge, Anna-Karin Tornberg, Paper E

© Joar Bagge, Anna-Karin Tornberg, Paper F

ISBN 978-91-8040-608-6

TRITA-SCI-FOU 2023:30

Printed by: Universitetservice US-AB, Sweden 2023

Abstract

This thesis concerns accurate and efficient numerical methods for the simulation of fluid flow on the microscale, known as Stokes flow or creeping flow. Such flows are important, for example, in understanding the swimming of microorganisms, spreading of dust particles, as well as in developing new nano-materials, and microfluidic devices that can be used for on-the-fly analysis of blood samples, among other things.

Flow on the microscale is dominated by viscous forces, meaning that a fluid such as water or air will behave as a very viscous fluid, like e.g. honey. The equations governing the flow, known as the Stokes equations, are linear PDEs, which permits the use of boundary integral methods (BIMs). In these methods, the PDE is reformulated as a boundary integral equation, thus reducing the dimensionality of the computational problem from three dimensions to two dimensions. The boundary integral formulation is well-conditioned, so that high accuracy can be achieved.

We consider two main challenges related to BIMs. The first challenge is that the integrals in the formulation contain integrands that vary rapidly for evaluation points close to the boundary, and cannot be accurately resolved using a standard method for numerical integration. Therefore, special quadrature methods are needed. We consider two such methods: quadrature by expansion (QBX) and the “line extrapolation/interpolation method” (also known as the Hedgehog method). In particular, we consider these methods applied to simulations involving rigid rodlike particles and surrounding walls.

The second challenge is that discretizing the boundary integral formulation leads to a dense linear system, which requires $\mathcal{O}(N^2)$ operations to solve iteratively, where N is the number of unknowns. This becomes too expensive for large systems. A fast summation method, such as the Spectral Ewald (SE) method considered in this thesis, reduces the number of operations required, for example to $\mathcal{O}(N \log N)$. The SE method can also be used for problems with periodic boundary conditions in any number of the spatial directions (arbitrary periodicity).

We also consider an application of these methods to a flow problem involving an inertial spheroid in a parabolic flow profile, and analyze the lateral drift of this spheroidal particle.

The numerical methods studied in this thesis enable fast and accurate computer simulations of e.g. suspensions of rigid particles in three-dimensional Stokes flow, including surrounding walls and arbitrary periodicity.

Sammanfattning

Denna avhandling behandlar noggranna och effektiva numeriska metoder för att simulera strömning på mikroskalan, känt som Stokesflöde eller krypande flöde. Sådana flöden är viktiga till exempel för att förstå hur mikroorganismer simmar och stoftpartiklar sprider sig, liksom för att utveckla nya nanomaterial samt mikrofluidiska enheter för omedelbar blodanalys, bland annat.

Strömning på mikroskalan domineras av viskösa krafter, vilket innebär att en fluid såsom vatten eller luft kommer att bete sig som en mycket viskös fluid, som till exempel honung. De ekvationer som styr strömningen kallas Stokes ekvationer och är linjära PDE:er, vilket innebär att randintegralmetoder kan användas. I dessa metoder omformuleras PDE:n som en randintegralekvation, så att beräkningsproblemets dimensionalitet minskar från tre till två dimensioner. Randintegralformuleringen är välkonditionerad, så att hög noggrannhet kan uppnås.

Vi behandlar två huvudsakliga utmaningar kopplade till randintegralmetoder. Den första utmaningen är att integralerna i formuleringen innehåller integrander som varierar snabbt för evalueringspunkter nära randen, och inte kan lösas upp noggrant med en standardmetod för numerisk integration. Därmed behövs speciella kvadraturmetoder. Vi betraktar två sådana metoder: expansionskvadratur (eng. quadrature by expansion) och "linje-extrapolation/interpolation" (även känt som igelkottsmetoden, eng. Hedgehog method). Metoderna tillämpas specifikt på strömningsproblem innehållande stela stavlika partiklar och omgivande väggar.

Den andra utmaningen är att diskretiseringen av randintegralformuleringen leder till ett tätt linjärt system, som kräver $\mathcal{O}(N^2)$ operationer att lösa iterativt, där N är antalet okända. Detta blir alltför kostsamt för stora system. En snabb summeringsmetod, såsom den spektrala Ewald-metoden som behandlas i denna avhandling, minskar antalet operationer som krävs till exempelvis $\mathcal{O}(N \log N)$. Den spektrala Ewald-metoden kan även användas för problem med periodiska randvillkor i godtyckligt antal rumsriktningar.

Vi tillämpar även dessa metoder på ett strömningsproblem med en trög sfäroid i en parabolisk strömningsprofil, och analyserar driften i sidled hos denna sfäroidiska partikel.

De numeriska metoder som studeras i denna avhandling möjliggör snabba och noggranna datorsimuleringar av exempelvis suspensioner av stela partiklar i tredimensionellt Stokesflöde, inklusive omgivande väggar och godtycklig periodicitet.

Populärvetenskaplig sammanfattning

Matematiska ekvationer kan användas för att beskriva en mängd olika fenomen i vår omvärld. I denna avhandling ligger fokus på strömningsmekaniska fenomen där mikroskopiska partiklar eller droppar rör sig i en vätska eller gas. Till exempel kan det handla om mikroorganismer som simmar i havet, eller stoftpartiklar som sprider sig i atmosfären. Tillämpningar finns även inom industrin (t.ex. tillverkning av nya nanomaterial) och medicinen (t.ex. utveckling av portabla enheter för att analysera blodprover och ställa diagnos utan utrymmeskrävande utrustning).

Sådana mikroskopiska strömningsfenomen kallas Stokesflöden och domineras av viskösa krafter, vilket innebär att de beter sig som en mycket trögflytande vätska. För en mikroorganism i havet är det alltså som att simma i honung.

Ett mål kan vara att beräkna hur en mängd partiklar i en vätska eller gas kommer röra sig. De matematiska ekvationer som beskriver strömningen kan inte lösas för hand, utan måste förenklas och lösas med hjälp av datorer och numeriska metoder. En typ av metoder utgår från randintegraler, vilket innebär att ekvationerna löses på partiklarnas ytor istället för i hela vätskevolymen. Detta gör det lätt att hantera rörliga geometrier, till exempel partiklar som förflyttar sig.

En utmaning kopplad till randintegralmetoder är att få hög noggrannhet i beräkningarna då partiklar kommer nära varandra, eftersom randintegralerna då blir svåra att uppskatta. En annan utmaning gäller att utföra beräkningarna effektivt, så att en stor mängd partiklar kan simuleras inom rimlig tid. Speciella metoder krävs för att hantera dessa utmaningar. Syftet med detta avhandlingsarbete har varit att utveckla sådana noggranna och effektiva metoder.

Contents

Preface		ix
Acknowledgements		xi
 Part I: Introduction and summary		
1 Overview		1
2 Fluid flow on the microscale – Stokes flow		7
2.1 Physics of Stokes flow		7
2.2 The equations of fluid flow		9
2.3 Applicability of Stokes flow		13
2.4 Particle suspensions; resistance and mobility problems		14
2.5 Numerical methods for Stokes flow		15
3 Boundary integral formulations of Stokes flow		21
3.1 Fundamental solutions of Stokes flow		21
3.2 Boundary integral formulations		26
3.3 Solution procedure		31
4 Special quadrature methods		33
4.1 The need for special quadrature methods		33
4.2 Overview of special quadrature methods		38
4.3 Quadrature by expansion (QBX)		39
4.4 The Hedgehog method		41
5 Fast summation methods and periodic problems		45
5.1 Overview of fast summation methods		45
5.2 Periodic problems		48
5.3 The Spectral Ewald method		50
6 Conclusions and outlook		55
References		59

Part II: Scientific papers**Paper A**

J. Bagge and A.-K. Tornberg. Accurate quadrature methods with application to Stokes flow with particles in confined geometries. *The Eleventh UK Conference on Boundary Integral Methods (UKBIM11)*, Nottingham, pp. 15–24, 2017. <http://irep.ntu.ac.uk/id/eprint/31463/>

Paper B

J. Bagge and A.-K. Tornberg. Highly accurate special quadrature methods for Stokesian particle suspensions in confined geometries. *International Journal for Numerical Methods in Fluids*, vol. 93, pp. 2175–2224, 2021. <https://doi.org/10.1002/flid.4970>

Paper C

J. Bagge, T. Rosén, F. Lundell and A.-K. Tornberg. Parabolic velocity profile causes shape-selective drift of inertial ellipsoids. *Journal of Fluid Mechanics*, vol. 926, A24, 2021. <https://doi.org/10.1017/jfm.2021.716>

Paper D

D. S. Shamshirgar, J. Bagge and A.-K. Tornberg. Fast Ewald summation for electrostatic potentials with arbitrary periodicity. *The Journal of Chemical Physics*, vol. 154, 164109, 2021. <https://doi.org/10.1063/5.0044895>

Paper E

J. Bagge and A.-K. Tornberg. Fast Ewald summation for Stokes flow with arbitrary periodicity. *Preprint*, 2022. Submitted and under revision. <https://arxiv.org/abs/2210.01255>

Paper F

J. Bagge and A.-K. Tornberg. Accurate quadrature via line extrapolation and rational approximation with application to boundary integral methods for Stokes flow. *Technical note*, 2023.

Preface

This is a compilation thesis divided into two parts. Part I provides an introduction to the subject and summarizes the work presented in the included papers. Part II consists of six papers:

- Paper A (a conference paper) and Paper B (a journal paper) present an extension to new geometries (flat walls, pipes, rodlike particles) of previous work [6] by Ludvig af Klinteberg (LK) and Anna-Karin Tornberg (AKT) on the quadrature by expansion (QBX) method. Joar Bagge (JB) contributed to the ideas, performed the derivations, implemented the method for the new geometries, performed the numerical experiments, and wrote the manuscript, under the supervision of AKT.
- Paper C (a journal paper) applies the QBX method developed by LK and AKT to a problem in fundamental fluid mechanics, and is a continuation of JB's Master's thesis work [11]. JB contributed to the ideas and derivations, performed the numerical experiments, and wrote a part of the manuscript, in collaboration with Tomas Rosén, Fredrik Lundell and AKT.
- Paper D (a journal paper) concerns the extension of the Spectral Ewald (SE) method, previously developed by Dag Lindbo, LK, Davood Saffar Shamshirgar (DSS) and AKT, to arbitrary periodicity for electrostatics. The first version of the manuscript was written by DSS and AKT. JB further developed the paper in collaboration with AKT and DSS. JB contributed to ideas, derivations, methods and implementations; performed additional numerical experiments; and rewrote parts of the manuscript.

- Paper E (a preprint) extends the previously developed SE method to arbitrary periodicity for Stokes flow, presenting work carried out by JB under the supervision of AKT. JB contributed to the ideas, derivations, methods and implementations; performed the numerical experiments; and wrote the manuscript.
- Paper F is a technical note summarizing recent work on the Hedgehog method for special quadrature carried out by JB under the supervision of AKT. JB contributed to the ideas, derivations, methods and implementations; performed the numerical experiments; and wrote the note.

The contributions of the included papers are also outlined throughout Part I, highlighted as **Paper X** at the ends of Sections 2.1, 3.2, 4.3, 4.4 and 5.3, and also summarized in Chapter 6.

Acknowledgements

First and foremost, I would like to thank my advisor Anna-Karin Tornberg for giving me this opportunity. I have learned a lot from you during these years, and I am grateful for your support, guidance and patience. It has been a long journey, but you have always been a source of inspiration.

I would also like to thank my second advisor Katarina Gustavsson. Thank you for your constant positivity and kindness, and for always being available for a chat.

I am also grateful to Jennifer Ryan, who kindly reviewed the draft of this thesis and provided helpful comments and TikZ templates. My thanks also go to Fredrik Lundell and Tomas Rosén for their enthusiasm and inspiration during our collaboration.

To all my friends and colleagues at the department, past and present, thank you for helping to create such a positive working environment, and for many interesting discussions about math, life, and everything in between. In particular, I want to thank the group members of Professor Tornberg's research group, and my past and present office mates.

I would like to thank the audio group at Ericsson Research for giving me the opportunity to do an internship there, and see examples of boundary integral methods applied to real-world problems.

A big thank you to my family and friends, for always believing in me and supporting me, no matter what. I am lucky to have you. To Emma, my best friend, thank you for being there in good times and bad. You help me to remember what is important in life.

1 Overview

Mathematical equations can be used to describe a wide range of phenomena in the world around us, from the behaviour of atoms and molecules, via the flow of air around a train or airplane, to the motion of the planets and the stars, and the evolution of the universe. Often, the mathematical laws governing these phenomena are formulated as *differential equations*. The solutions to these equations can in most cases not be found simply by pen and paper using methods from calculus. Instead, one must use *numerical methods*, which are based on approximations of the original equations. As computers have grown increasingly powerful, scientists have been able to study more and more complicated problems.

In this thesis, we consider the flow of fluids – i.e., liquids and gases – such as water or air. More specifically, we consider *microscale* flows, i.e., flows with a characteristic length scale somewhere between a micrometer and a millimeter, roughly speaking.¹ On such a small scale, both air and water behave as a viscous² fluid such as honey or glycerol (as will be made clear in Chapter 2). This is the realm of bacteria, plankton, blood cells, pollen, dust particles, and droplets (such as mist). Fluid flow on the microscale is known as *Stokes flow*, and is characterized by viscous forces (due to the internal friction of the fluid) dominating over inertial forces (due to the inertia³ of the fluid), corresponding to a vanishing Reynolds number in the Navier–Stokes

¹The precise meaning of the words “microscale” and “macroscale” vary depending on the context. In this thesis, “microscale” will be understood as roughly between 1 μm and 1 mm, while “macroscale” is larger than 1 mm.

²The *viscosity* of a fluid is its “thickness”, or more formally, its resistance to deformation. For instance, honey has a higher viscosity than water. Viscosity is caused by internal friction in the fluid.

³*Inertia* is the ability of an object to resist changes to its velocity, and is caused by its mass.

equations. The negligible inertia will, for example, make a microorganism come to an abrupt halt if it suddenly stops swimming, unlike a human who will “glide”.

The purpose of this thesis is to develop numerical methods that can be used to perform computer simulations of Stokes flow, with a focus on controllable accuracy and high computational efficiency. Being able to control the accuracy of the methods is important for obtaining reliable simulations, and accuracy requirements may vary depending on the application. A high computational efficiency is necessary in order to enable large simulations of three-dimensional problems with many particles, which is often needed in applications. In this context, efficient methods save time and energy, and may be needed for the computation to be feasible at all.

Typical simulation setup

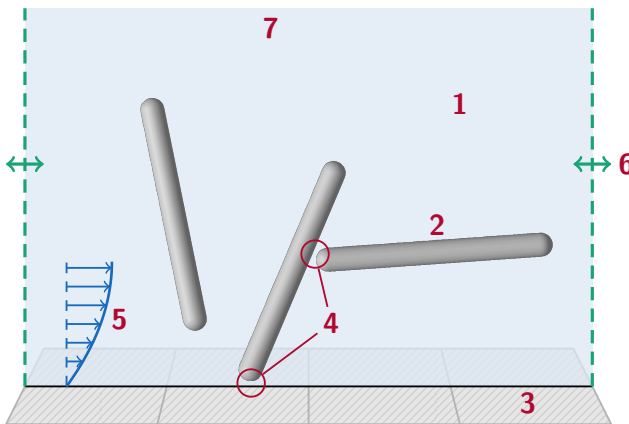


FIGURE 1.1: Schematic of a typical simulation setup.

A typical situation that we might want to simulate is shown in Figure 1.1. The important features in this setup are, referring to the numbers in the figure:

1. The **fluid** itself is assumed to be incompressible and Newtonian⁴. The

⁴In a Newtonian fluid, the viscous stress is proportional to the strain rate, with the constant of proportionality being the viscosity. For example, water and air can both be considered to be Newtonian.

flow is assumed to be Stokes flow, i.e., at zero Reynolds number.

2. The fluid contains **rigid particles**, such as the rodlike particles shown here. Typically, the goal of the simulation is to determine how the particles will move over time, due to the influence of e.g. gravity or a prescribed fluid flow.
3. The fluid domain may be bounded by **rigid walls** in some directions.
4. The particles may come quite close to each other, or to the walls. These **close interactions** are challenging for numerical methods, and special strategies are needed to handle them.
5. A **background flow** may be prescribed, i.e., a flow profile that should hold in the absence of particles. This can be used e.g. to drive the particle motion.
6. **Periodic boundary conditions** may be used in some (or all) spatial directions. Imposing periodicity is a common way of simulating a large collection of particles using limited computational resources.
7. In some (or all) spatial directions, the flow domain may be unbounded and continue into infinity. We call these directions **free**, and if all directions are free, we call the setup a **free-space problem**.

All simulations considered in this thesis are fully three-dimensional in space. In this thesis, we consider a limited number of shapes for the particles and walls, shown in Figure 1.2. For details, see §3 in Paper B. Note that a *spheroid* is a sphere that has either been elongated (called *prolate*) or flattened out (called *oblate*) along one axis; the spheroid in Figure 1.2 is prolate.

Numerical methods and the contribution of this thesis

In this thesis, we consider numerical methods based on *boundary integral equations* for solving the equations of Stokes flow (which will be discussed in Section 2.2). Boundary integral methods have an obvious advantage over volume-based methods, namely that only the (two-dimensional) boundary has to be discretized, instead of the entire (three-dimensional) fluid domain. There are, however, challenges regarding the accuracy and speed of these methods. The goal of this thesis is to develop special quadrature methods

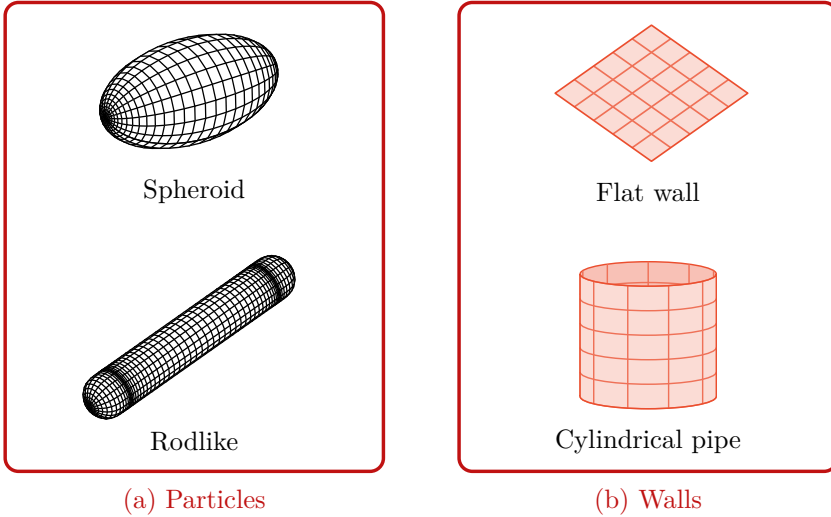


FIGURE 1.2: The geometrical objects considered in the simulations in this thesis (in Papers A–C & F). (Figure from Paper B, reproduced under the Creative Commons Attribution License)

and fast summation methods, to enable accurate and efficient simulations of three-dimensional Stokes flow using boundary integral methods. The included papers detail the development and systematic testing of these methods. The specific challenges that are the main focus of this thesis are the following:

- **Accuracy:** The integrals in the boundary integral formulation must be discretized and computed using a quadrature method. These integrals involve integrands that are singular or nearly singular – i.e., sharply peaked, and increasingly so with decreasing distance between the evaluation point and boundary – especially for the close interactions shown in Figure 1.1 (point 4). They cannot be accurately resolved using standard quadrature methods, and therefore, **special quadrature methods** are needed to get a solution with high accuracy. Such methods are the topic of Chapter 4.
- **Speed:** The linear system that arises when discretizing a boundary integral equation is typically dense, due to the nonlocal nature of the operator. To solve the system efficiently, **fast summation methods**

are needed. Such methods are described in Chapter 5, with a focus on methods that can be used for problems with both periodic and free directions.

Significance and wider applicability

While the focus of this thesis is on the fundamental development of numerical methods, and not primarily on their applications, it is important to understand the target applications. Therefore, we will now briefly discuss possible applications. The main types of applications where this work would be useful are for microscale flow phenomena. The methods developed here may form a part of the next generation simulation tools for such phenomena. In the future, such tools could enable researchers to gain a deeper understanding of microscale flows and design new technology. A few examples of possible applications are:

- Microfluidic devices [100, 101, 120] involve fluids passing through microscopic channels, and can be used e.g. for pregnancy testing [98]; detection of diseases such as COVID-19 [59], HIV [30] and cancer [9]; as well as gene sequencing [120]. Microfluidic devices may be manufactured at a low cost, and permit on-the-fly analysis of samples, without sending them to a central lab. This makes them especially suitable for self-testing or point-of-care testing. They can be cost-effective and environmentally friendly [98], and may be used in remote areas lacking hospitals and labs [30, 101].
- Cellulose nanofibrils [83], which are extracted e.g. from wood, can be used to create extremely strong macroscopic materials, surpassing metals, alloys and glass fibers in strength. This may be used to create sustainable and climate-friendly high-performance building materials. The assembly of macroscale fibers from nanofibrils requires proper alignment, which can be achieved using a microscopic flow-focusing channel.
- In microbiology, there is an interest in understanding how microorganisms, such as plankton and bacteria, swim and behave. These organisms are of importance since they form the basis of the food chain in the oceans [49]. There is also an interest in the possibility to harness the power of collectively swimming microorganisms [109].

Numerical methods and computer simulations can aid the design of new technology, and are cheaper, faster and less energy-consuming than manufacturing physical prototypes.

The numerical methods described in this thesis can also be applied to other fields of physics, such as electrostatics, molecular dynamics, or acoustic and electromagnetic scattering.

Overview of Part I

The remainder of Part I is structured as follows: Chapter 2 introduces the physics, mathematics and numerical methods of Stokes flow, to establish a proper background for what follows. In Chapter 3, the foundation of our boundary integral method for Stokes flow is given, namely, the boundary integral formulations. This chapter forms the basis for the rest of the thesis.

The remaining chapters outline the main topics and contributions of the thesis – i.e., methods to handle the challenges of boundary integral methods. Chapter 4 deals with accurate special quadrature methods, and introduces the quadrature by expansion (QBX) method used in Papers A–C, and also the Hedgehog method studied in Paper F. Chapter 5 discusses fast summation methods and periodic problems, which are the topics of Papers D–E. Finally, conclusions and an outlook are presented in Chapter 6.

2 Fluid flow on the microscale

– Stokes flow

This thesis deals with the topic of microhydrodynamics, i.e., the study of fluid flow in a microscale setting. Such flows are dominated by viscous forces, and are known as *Stokes flow* (or alternatively as “creeping flow” or “linearized viscous flow”). In this chapter, we give an introduction to Stokes flow from a physical, mathematical and numerical viewpoint. For more details, the reader is referred to the books by Kim & Karrila [65] and Pozrikidis [94].

2.1 Physics of Stokes flow

Stokes flow is dominated by *viscous forces* (due to the internal friction of the fluid), rather than *inertial forces* (due to the inertia of the fluid). The ratio of inertial forces to viscous forces is called the *Reynolds number*. For Stokes flow this number is close to zero. In essence, on a small enough length scale, any fluid behaves like a viscous fluid such as honey or glycerol.¹

The lack of inertia in Stokes flow leads to several interesting effects:

- (i) The fluid adjusts *instantly* to the boundaries, and any change in the flow pattern must come from changes at the boundaries (e.g. from one of the boundaries moving).

¹Most types of honey are Newtonian fluids, although some are non-Newtonian [41]. Pure glycerol is also a Newtonian fluid [19]. The viscosity of both honey and glycerol is strongly dependent on temperature.

- (ii) The flow is *reversible*, meaning that if the external forcing is reversed, the flow will be perfectly reversed. In a classical example of this, dye is injected into a viscous fluid confined between two concentric cylinders. As the inner cylinder is rotated, the dye appears to mix. If the inner cylinder is then rotated in the opposite direction, the flow is reversed, and the dye “unmixes”, returning to its original state!²
- (iii) The flow is completely *laminar*, meaning that streamlines are smooth and regular, and there is no turbulence. Furthermore, streamline patterns around symmetric objects are symmetric, as shown in Figure 2.1.

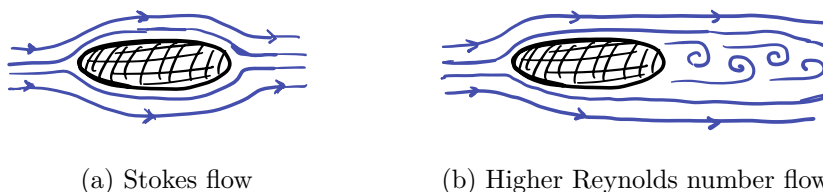


FIGURE 2.1: Streamlines around a stationary object in Stokes flow (a), and in flow at higher Reynolds number (b). The object can be thought of e.g. as a smooth stone in a river. Note that the streamlines are left–right symmetric in (a), but not in (b).

As a result of (ii) above, and the lack of turbulence, true mixing is difficult to achieve on the microscale. Different strategies are required to perform mixing on the microscale compared to the macroscale, and often natural diffusion processes are utilized [112]. Swimming is also different on the microscale, and microorganisms typically propel themselves forward using thin appendages known as *flagella* (from the Latin word for “whip”) [49]. If they stop swimming, they come to a halt instantly due to the lack of inertia.

In **Paper C**, we performed numerical simulations of a situation in which an inertial particle moves through an inertialess fluid.³ In other words, the fluid flow itself is Stokes flow, but the particle inertia may induce effects not seen in normal Stokes flow. (For instance, the motion is no longer

²The reader will find plenty of videos of this phenomenon by searching e.g. for “reversibility of Stokes flow” on YouTube.

³The meaning of this is elaborated upon in Section 2.4, where this situation is case (ii).

reversible.) We found that inertial prolate spheroids have a sideways drift in a parabolic flow, and that the magnitude of the drift depends on the shape of the particle. On the other hand, spheres and oblate spheroids have no such drift, due to their different pattern of motion. Thus, this new drift mechanism could potentially allow for separation and filtering of particles of different shapes.

2.2 The equations of fluid flow

Let us first introduce some notation for the geometry of the flow problem. As shown in Figure 2.2, we let $D_f = D_f(t)$ denote the domain occupied by the fluid itself, which may vary with time as the boundaries (particles) move. The union of all the rigid boundaries (i.e., the surfaces of particles and walls) is denoted by $\Gamma = \Gamma(t)$. The periodic boundaries (where the periodic boundary conditions are enforced) are denoted by Π , and these are fixed.

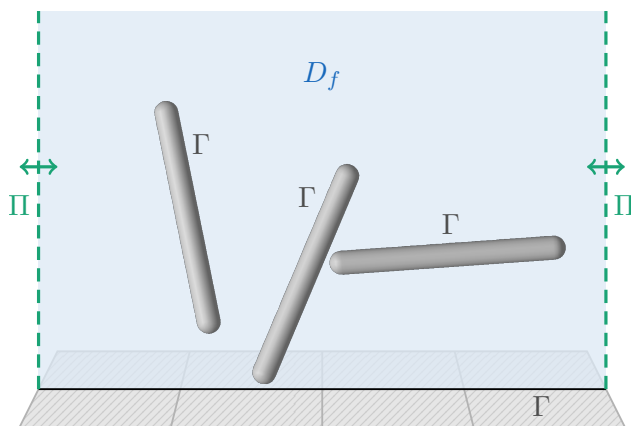


FIGURE 2.2: Notation for the geometry of the flow problem.

Derivation of the Stokes equations

Fluid flow is governed by the *Navier–Stokes equations*, which are a set of partial differential equations (PDEs) based on the physical principles of conservation of momentum (i.e., Newton’s second law, $m\mathbf{a} = \mathbf{F}$) and con-

servation of mass. For an incompressible Newtonian fluid, the momentum conservation equation is⁴

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \mu \nabla^2 \mathbf{u} + \mathbf{f}, \quad (2.2.1)$$

and the mass conservation equation is simply

$$\nabla \cdot \mathbf{u} = 0. \quad (2.2.2)$$

Here, we assume that the density ρ and viscosity μ of the fluid are known, as well as any external body forces $\mathbf{f} = \mathbf{f}(\mathbf{x})$, such as gravity, acting on the fluid. (Note that \mathbf{f} has the physical dimensions of force per unit volume.)

The Navier–Stokes equations (2.2.1)–(2.2.2) can be used to determine the velocity $\mathbf{u} = \mathbf{u}(\mathbf{x}, t)$ and pressure $p = p(\mathbf{x}, t)$ of the fluid, as functions of position \mathbf{x} and time t . The equations hold for $\mathbf{x} \in D_f(t)$, where $D_f(t)$ is the fluid domain as in Figure 2.2.

In hydrodynamics, it is common to introduce *nondimensional* quantities to simplify equations and reduce the number of parameters. This can be done in different ways, and the physical problem under consideration should dictate which choices are made.

For instance, let us consider a small particle moving through a quiescent fluid (i.e., a fluid at rest). In this case, there is a natural characteristic length scale L (namely, the diameter of the particle) and a characteristic speed U (the speed of the particle). We may then introduce the nondimensional quantities

$$\tilde{\mathbf{x}} = \frac{\mathbf{x}}{L}, \quad \tilde{\mathbf{u}} = \frac{\mathbf{u}}{U}, \quad \tilde{t} = \frac{t}{T}, \quad \tilde{p} = \frac{p}{\mu/T}, \quad \tilde{\mathbf{f}} = \frac{\mathbf{f}}{\mu/(LT)}, \quad (2.2.3)$$

where nondimensional quantities are denoted by a tilde above the variable. Here, T is a characteristic time scale. Unless a time scale is imposed using external means (e.g. by a periodically varying background flow), a natural choice is to let $T = L/U$. Inserting the relations (2.2.3) into (2.2.1) allows us to write down the momentum conservation equation on nondimensional form, as

$$\text{Re} \frac{\partial \tilde{\mathbf{u}}}{\partial \tilde{t}} + \text{Re} (\tilde{\mathbf{u}} \cdot \tilde{\nabla}) \tilde{\mathbf{u}} = -\tilde{\nabla} \tilde{p} + \tilde{\nabla}^2 \tilde{\mathbf{u}} + \tilde{\mathbf{f}}, \quad (2.2.4)$$

⁴Throughout this thesis, vectors are in boldface, i.e., $\mathbf{x} = (x_1, x_2, x_3)$, $\mathbf{u} = (u_1, u_2, u_3)$ and so on, and the nabla operator is $\nabla = \left(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \frac{\partial}{\partial x_3} \right)$.

where the nondimensional nabla operator is $\tilde{\nabla} = L\nabla$ (since ∇ has the physical dimensions of inverse length). In (2.2.4), we have introduced the *Reynolds number*

$$\text{Re} = \frac{\rho LU}{\mu}. \quad (2.2.5)$$

As mentioned above, the Reynolds number is the ratio of inertial forces to viscous forces, and it is close to zero for Stokes flow. Letting $\text{Re} \rightarrow 0$ in (2.2.4), we get the momentum equation of Stokes flow, namely

$$-\tilde{\nabla}\tilde{p} + \tilde{\nabla}^2\tilde{\mathbf{u}} + \tilde{\mathbf{f}} = \mathbf{0}. \quad (2.2.6)$$

Returning to the dimensional (physical) quantities, this equation becomes

$$-\nabla p + \mu\nabla^2\mathbf{u} + \mathbf{f} = \mathbf{0}. \quad (2.2.7)$$

Equation (2.2.7) together with the mass equation (2.2.2) are the *Stokes equations*, which govern Stokes flow. Sometimes these equations are called the *steady* Stokes equations to emphasize that they contain no time derivative.

Properties of the Stokes equations

To reiterate, the (steady) Stokes equations are

$$\begin{cases} -\nabla p + \mu\nabla^2\mathbf{u} + \mathbf{f} = \mathbf{0}, \\ \nabla \cdot \mathbf{u} = 0, \end{cases} \quad (2.2.8)$$

where μ is the viscosity and $\mathbf{f}(\mathbf{x})$ is an external body force. The solution to (2.2.8) consists of the velocity $\mathbf{u} = (u_1, u_2, u_3)$ and pressure p at every point of the fluid domain $D_f(t)$. In this thesis, we are mainly interested in the velocity field, since that is what is needed to determine the motion of the fluid and any particles it contains. The pressure is needed to make (2.2.8) a square system (four equations and four unknowns), but it is of no particular interest to us.

If the force field is *conservative*, i.e., $\mathbf{f} = \nabla\varphi$ for some potential φ , we can define $p' = p - \varphi$, and the momentum equation (2.2.7) turns into the homogeneous equation

$$-\nabla p' + \mu\nabla^2\mathbf{u} = \mathbf{0}. \quad (2.2.9)$$

Note that the velocity \mathbf{u} is the same in (2.2.7) and (2.2.9), i.e., it is not affected by adding a conservative force field. Therefore, we will typically

assume that we can set $\mathbf{f} = \mathbf{0}$ in (2.2.8) so that the Stokes equations are homogeneous.

When we took $\text{Re} \rightarrow 0$ in the Navier–Stokes equations (2.2.4), the time derivative as well as the nonlinear term $(\mathbf{u} \cdot \nabla)\mathbf{u}$ disappeared. The lack of a time derivative in the Stokes equations is connected to the instantaneity and reversibility of Stokes flow mentioned in Section 2.1; the flow has no “memory” of the past since it has no inertia.

The lack of a nonlinear term implies that the Stokes equations are linear, unlike the Navier–Stokes equations. The linearity of the Stokes equations makes them easier to analyze theoretically, and one can, for example, show uniqueness of the solution (see for example [65, §2.2.1] or [94, §1.5]). Linearity also implies the *superposition principle*: if we have two solutions (\mathbf{u}_1, p_1) and (\mathbf{u}_2, p_2) to (2.2.8), with forcing terms \mathbf{f}_1 and \mathbf{f}_2 , respectively, then their sum $(\mathbf{u}_1 + \mathbf{u}_2, p_1 + p_2)$ is also a solution to (2.2.8), with forcing term $\mathbf{f}_1 + \mathbf{f}_2$.

Boundary conditions and background flow

In order for (2.2.8) to have a unique solution, boundary conditions are needed. In this thesis, we enforce three types of conditions:

- (i) On the rigid boundary $\Gamma(t)$ (i.e. on the surfaces of particles and walls), *no-slip* boundary conditions are enforced. This means that the fluid should have the same velocity as the boundary, i.e., $\mathbf{u} = \mathbf{u}_\Gamma$ on $\Gamma(t)$.
- (ii) On the periodic boundary Π , periodic boundary conditions are enforced.
- (iii) In the free directions, a far-field condition is needed at infinity. Typically, we require that $\mathbf{u} \rightarrow \mathbf{0}$ as $|\mathbf{x}| \rightarrow \infty$.

Condition (iii) may seem restrictive, but it is still possible to model situations where the flow field does not go to zero at infinity. We do this by introducing a *background flow*, which is a given function $\mathbf{u}_{\text{bg}}(\mathbf{x})$. The background flow must satisfy the Stokes equations (2.2.8), but need not satisfy the conditions (i)–(iii) above. The idea is to solve for a perturbation \mathbf{u}_δ that satisfies (ii) and (iii), such that the total field $\mathbf{u} = \mathbf{u}_{\text{bg}} + \mathbf{u}_\delta$ satisfies (i). In other words, \mathbf{u}_δ should:

- be a solution to the Stokes equations (2.2.8),

- satisfy $\mathbf{u}_\delta = \mathbf{u}_\Gamma - \mathbf{u}_{\text{bg}}$ on $\Gamma(t)$,
- satisfy periodic boundary conditions on the periodic boundary Π , and
- satisfy $\mathbf{u}_\delta \rightarrow \mathbf{0}$ as $|\mathbf{x}| \rightarrow \infty$.

Due to the superposition principle, the total field $\mathbf{u} = \mathbf{u}_{\text{bg}} + \mathbf{u}_\delta$ will also be a solution to the Stokes equations.

2.3 Applicability of Stokes flow

Under what circumstances is it reasonable to assume that the Stokes model (2.2.8) holds? In its derivation, we assumed that $\text{Re} \rightarrow 0$. In reality, the Reynolds number (2.2.5) will never be exactly zero. Is the model still relevant for small, but nonzero, Reynolds numbers?

It has been shown that the difference between the Stokes solution and the solution to the full Navier–Stokes equations is $\mathcal{O}(\text{Re})$ in the vicinity of a particle; this was shown for a sphere by Proudman & Pearson [96], and for a particle of arbitrary shape by Brenner & Cox [21]. Furthermore, Kim & Karrila [65, §1.2.3] show that when the boundaries of the problem move (e.g. when simulating moving particles), an additional error proportional to $\mathcal{O}(\text{Re}^{1/2})$ is introduced when using the steady Stokes equations (2.2.8).

As an example, let us consider a planktonic microorganism swimming in water using flagella. Following [49], let us assume that the organism is of size $L = 10\ \mu\text{m}$ and is swimming at $U = 100\ \mu\text{m/s}$. Since water has density $\rho = 10^3\ \text{kg/m}^3$ and viscosity $\mu = 10^{-3}\ \text{Pa}\cdot\text{s}$, the Reynolds number of the flow around the organism will be $\text{Re} = 10^{-3}$. Assuming a modelling error of $\mathcal{O}(\text{Re}^{1/2})$ as mentioned above, the error would then be around $10^{-3/2}$, i.e., around 3%.

As another example, we consider stirring honey with a thin stick, such as a toothpick. Honey has a density similar to that of water, and the viscosity may range from around $2\ \text{Pa}\cdot\text{s}$ to $40\ \text{Pa}\cdot\text{s}$, depending on water content. Let us assume that $\mu = 10\ \text{Pa}\cdot\text{s}$, and that the stick has characteristic size $L = 1\ \text{mm}$. To get the same Reynolds number as in the example above, i.e., 10^{-3} , we would have to move the stick with speed $U = 1\ \text{cm/s}$. Again, the modelling error would be around 3%.

Both of the above examples are situations in which the Stokes model will give a reasonably good approximation of reality. Of course, the smaller the

Reynolds number, the better the approximation. In general, the Reynolds number (2.2.5) will become smaller if the length scale L or speed U becomes smaller, or if the fluid has a lower density ρ or higher viscosity μ .

2.4 Particle suspensions; resistance and mobility problems

We are interested in simulating rigid particles suspended in a Stokesian fluid, e.g. as in Figure 2.2. Typically, we seek the motion of these particles over time. Thus, we must be able to compute the velocities of the particles. We distinguish between two cases:

- (i) the particles, like the fluid, have no inertia; and
- (ii) the particles, unlike the fluid, have inertia.

In case (i), the particles have no “memory” of the past and will instantly adjust their velocities to any forces currently acting on them. In case (ii), the inertia of the particles causes them to adjust more slowly, and they can continue to move even when no force is acting on them (like a skater sliding on ice).

A particle moving through a Stokesian fluid will experience a drag force \mathbf{F}_d , which is due to the viscosity of the fluid (i.e., friction from the fluid against the particle surface). This drag, which is also called *viscous resistance*, is proportional to the relative velocity \mathbf{U} of the particle to the fluid: $\mathbf{F}_d = -\mu k \mathbf{U}$, where μ is the viscosity of the fluid and k is a positive coefficient which depends on the shape of the particle. For instance, for a sphere of radius R , the coefficient has the value $k = 6\pi R$, a fact known as Stokes’ law [65, §1.1]. For most particle shapes, the coefficient is not known analytically, but must be computed numerically.

For an inertial particle, i.e., case (ii), the drag force \mathbf{F}_d adds to any external force \mathbf{F}_e (such as gravity), so that the total force acting on the particle is $\mathbf{F} = \mathbf{F}_d + \mathbf{F}_e$. The acceleration \mathbf{a} of the particle can then be computed using Newton’s second law $m\mathbf{a} = \mathbf{F}$ and used to update the velocity \mathbf{U} of the particle. This situation, in which \mathbf{U} (the current velocity of the particle) is known and \mathbf{F}_d (the drag) is to be computed, is known as a *resistance problem*.

On the other hand, for an inertialess particle, i.e., case (i), the total force \mathbf{F} on the particle must be zero since $m = 0$ (no inertia) in Newton's second law, and the particle will adjust its velocity instantly to ensure that it is so. Thus, $\mathbf{F}_d = -\mathbf{F}_e$ is known, and we must compute the particle velocity \mathbf{U} such that $\mathbf{F}_d = -\mu k \mathbf{U}$ holds. This situation, which is the opposite of the one in the previous paragraph, is known as a *mobility problem*.

In general, the rotational motion of the particle also affects the drag, and we may collect the velocity \mathbf{U} and angular velocity $\boldsymbol{\omega}$ of a particle into a combined vector $\bar{\mathbf{U}} = (U_1, U_2, U_3, \omega_1, \omega_2, \omega_3)$. In the same way, we combine the force \mathbf{F}_d and torque $\boldsymbol{\tau}_d$ due to drag into a vector $\bar{\mathbf{F}}_d$. Then, there is a linear relation between $\bar{\mathbf{U}}$ and $\bar{\mathbf{F}}_d$:

$$\bar{\mathbf{F}}_d = \mathbf{R} \bar{\mathbf{U}}, \quad (2.4.1)$$

where the matrix \mathbf{R} is known as the *resistance matrix*. The inverse of the resistance matrix is the *mobility matrix* $\mathbf{M} = \mathbf{R}^{-1}$. These matrices depend on the shape of the particle, and in general, numerical methods must be used to compute them. (More details can be found e.g. in [65, §5].)

When the fluid contains N particles, the vectors $\bar{\mathbf{U}}$ and $\bar{\mathbf{F}}_d$ must be extended with the velocities and force-torques of all particles, such that they have $6N$ elements each. The resistance matrix (and mobility matrix) will then have size $6N \times 6N$, and its entries depend on the position and orientation (and shape) of each particle. However, in practice it is not necessary to compute the matrix explicitly to solve the resistance (or mobility) problem, and it would typically be needlessly computationally expensive to do so. Instead, the computation $\bar{\mathbf{U}} \mapsto \bar{\mathbf{F}}_d$ (or its inverse) is done using the numerical method, without constructing the matrix. We will return to this in Section 3.2, and details are found in Paper B.

2.5 Numerical methods for Stokes flow

In this section, we give an overview of numerical methods for Stokes flow, with a focus on problems containing rigid particles in three-dimensional space.

For certain particle shapes, there exists specialized methods for solving the resistance and mobility problems, and thus computing the motion of the particles. For spherical particles, *Stokesian dynamics* [20, 108, 124] can be used, which is based on a truncated multipole expansion with analytical

formulas available for spheres. A similar method, also based on a multipole expansion, is the *force-coupling method*, which is applicable to spherical [80] and ellipsoidal [74] particles. These methods typically have low accuracy.

For particles of arbitrary shape, the *multiblob method* [12, 35, 111] can be used, in which the particle is discretized using a collection of spherical “blobs” which are restricted to move collectively as a rigid body. While the method is simple and versatile, its accuracy is relatively low [12], although it can be improved by optimizing the placement of the spheres [23].

For a more accurate treatment of particles of arbitrary shape, and complicated geometries in general, one must solve the governing PDEs, i.e., the Stokes equations (2.2.8), to determine the fluid flow around the particles. Numerical methods for solving the PDEs can be divided into volume methods and boundary integral methods, described in more detail in the following subsections.

Volume methods

In volume methods, the PDEs are discretized on a three-dimensional grid in space. The grid may be structured or unstructured, and it may either be fitted to the geometry of the particles (known as *conforming*), or unfitted (*nonconforming*). Volume methods are often based on e.g. finite differences (FD), finite elements (FE) or finite volumes (FV) [126]. There are also spectral methods [27], based e.g. on the Fourier or Chebyshev basis functions.

An example of a **fitted (conforming) method** is the *arbitrary Lagrangian–Eulerian* (ALE) method, which is based on an unstructured grid and the FE method. The ALE method has e.g. been used to simulate sedimentation of spheres [56] and ellipsoids [114]. Since the grid is fitted to the geometry, the no-slip boundary conditions can be accurately represented. However, as the particles move, the grid must be updated in every time step in order to stay fitted to the geometry. Regenerating the grid from scratch (known as *remeshing*) at every time step would incur a prohibitively large computational cost, especially if many moving particles are simulated. Therefore, modern ALE methods usually include algorithms for updating the grid efficiently [56, 69, 104]. Nevertheless, remeshing is still needed at regular intervals as the geometry moves.

Unfitted (nonconforming) methods are sometimes called “fictitious-domain” methods or “domain-embedding” methods, since the actual – complicated – domain is “embedded” into a simpler domain. They use a fixed

simple background grid which is not adapted to the moving geometry, meaning that the boundary will not coincide with the grid points. Therefore, the boundary conditions must be enforced indirectly. Many different methods exist, for example:

- The *immersed boundary method* [91, 92, 84] uses a regularized Dirac delta distribution to transfer data between the fixed background grid and the moving particle grids. The method has been used to simulate sedimenting particles [118, 22, 63, 66].
- The *distributed Lagrange multiplier method* [46, 107] introduces a Lagrange multiplier field in the particle domain, to enforce rigid-body motion within the particle. The *eXtended FE method* [85, 44, 33] also uses Lagrange multipliers, but only on the particle boundary itself.
- The *cut FE method* (CutFEM) [25, 106, 62, 43] is based on Nitsche’s method [88], which adds penalty terms and stabilization terms to weakly enforce the boundary conditions.
- The *lattice-Boltzmann method* [28, 8] is not based directly on the PDEs, but rather on an approximation of them involving fictive “fluid particles” moving on a lattice, with ideas from statistical mechanics. It can also be used to simulate suspensions of solid particles, with boundary conditions handled using a “bounce-back” rule [97, 105, 57] (i.e., the “fluid particles” bounce off the solid particles), or the immersed boundary method [128].

The advantages and disadvantages of volume methods compared to boundary integral methods will be discussed in a separate subsection further below.

Boundary integral methods

In boundary integral methods, the PDEs are reformulated as integral equations over the boundary of the fluid domain (details are given in Chapter 3), i.e., over the surfaces of particles and walls. Thus, only the boundary $\Gamma(t)$ is discretized, using a two-dimensional grid; this makes it easy to treat moving boundaries and means that, for rigid particles, no remeshing at all is needed. The discretization can be done in different ways, leading to different types of methods:

In *collocation* or *Galerkin-type* methods [10, §3] [94, §6] [68, §13], the solution to the integral equation is expressed in terms of basis functions on e.g. a triangulation of the boundary, similar to a two-dimensional FE method. This is a common approach in *boundary element methods* (BEMs) [75, 95, 54]. These methods are able to treat complicated geometries, but the order of accuracy can be limited, depending on the order of the elements.

In *Nyström methods* [10, §4] [68, §12.2], the integral equation is enforced at the quadrature nodes which are used to discretize the integrals. This method is typically used for smooth, parametric surfaces, where it can be combined with high-order quadrature rules to reach a high order of accuracy. A selection of recent works relevant to particle suspensions are: [61, 115, 17, 2, 32, 77, 76]. The Nyström method is the one used in this thesis, in Papers B–C, and we will return to it in Section 3.2 and later.

The methods above generally assume that an explicit representation of the boundary exists, such as a parametrization or triangulation. However, it should be mentioned that boundary integral methods can also be used for implicitly defined boundaries (e.g. using a level-set method); for more details, see e.g. [58].

Regardless of the method of discretization, the solution to the boundary integral equation is defined only on the boundary $\Gamma(t)$ of the fluid domain. The solution to the Stokes equations (2.2.8), e.g. the velocity field, can be computed at any point in the volume $D_f(t)$ in a postprocessing step, by evaluating an extra integral per evaluation point. If only the particle velocities are needed, this postprocessing step is not necessary.

Comparison between volume and boundary integral methods

Applicability and accuracy. Boundary integral methods (BIMs) are possible to use for the Stokes equations (2.2.8), since they are linear PDEs. BIMs cannot be used directly for the Navier–Stokes equations (2.2.1)–(2.2.2), which are nonlinear. There is, however, ongoing research into extending the applicability of BIMs to the full Navier–Stokes equations, see e.g. [3]. Volume methods, on the other hand, can generally be applied to linear and nonlinear PDEs alike, and are thus more widely applicable than BIMs.

For problems on unbounded domains, e.g. solving an exterior flow problem in free space, such as the flow around an obstacle or a set of particles, BIMs have a distinct advantage since only the boundary of the fluid domain – e.g. the particle surfaces – needs to be discretized, and this boundary is

finite. For volume methods, the domain must be truncated, and artificial boundary conditions imposed.

As was mentioned above, volume methods typically have a limited order of accuracy, not least due to the way the boundary conditions are imposed (in particular for unfitted methods). The same is often true of BEMs. Nyström methods, on the other hand, can be implemented with high order accuracy, at least when the boundary is smooth. The accuracy of any BIM hinges on the accuracy of the quadrature rule used to discretize the integral equations. This is one of the challenges with BIMs, since the integrals contain singular functions which require specialized quadrature rules, as will be seen later.

Finally, BIMs have an advantage in that the condition number of the linear system that results after discretization of the integral equations – at least under certain conditions⁵ – remains bounded as the grid size is refined [10, p. 113] [68, p. 282]. This is in contrast to volume methods based on e.g. FD and FE, for which the condition number typically grows as the grid is refined [81].

Computational complexity.⁶ For a volume method with grid spacing h , the number of degrees of freedom will be $N_V = \mathcal{O}(h^{-3})$ since the grid is three-dimensional. The matrix of the resulting linear system is typically sparse, and the system can then be solved in $\mathcal{O}(N_V) = \mathcal{O}(h^{-3})$ operations.

BIMs, due to their reduction of the computational domain from three to two dimensions, require only $N_B = \mathcal{O}(h^{-2})$ degrees of freedom for a surface grid of spacing h . However, the matrix of the linear system is typically dense, and so naive Gaussian elimination would require $\mathcal{O}(N_B^3) = \mathcal{O}(h^{-6})$ operations. Fortunately, due to its bounded condition number, the system can be solved iteratively using e.g. GMRES [99] in a bounded number of iterations N_{iter} [26], reducing the computational cost to $\mathcal{O}(N_{\text{iter}}N_B^2) = \mathcal{O}(N_{\text{iter}}h^{-4})$. Using a fast summation method (see Section 5.1), the matrix–vector product can be reduced from $\mathcal{O}(N_B^2)$ to $\mathcal{O}(N_B \log N_B)$, or even $\mathcal{O}(N_B)$, operations, with a constant that depends on the accuracy requirement.⁷ Thus, the total cost would become $\mathcal{O}(N_{\text{iter}}N_B) = \mathcal{O}(N_{\text{iter}}h^{-2})$, with N_{iter} independent of h .

⁵More specifically, the statement holds for *Fredholm integral equations* of the second kind, to be introduced in Section 3.2.

⁶This discussion is inspired by [2, §3.1] and [58, §1.1].

⁷An alternative to an iterative solver may be to use a fast direct solver [82, 79, 113], which can also reduce the cost of solving the linear system to $\mathcal{O}(N_B)$ operations.

Evaluating the solution at an arbitrary point in space requires a postprocessing step for BIMs, consisting of evaluating an extra integral. For N_{eval} evaluation points, the naive postprocessing cost would be $\mathcal{O}(N_{\text{eval}}h^{-2})$, but using a fast summation method the cost can be reduced to $\mathcal{O}(N_{\text{eval}} + h^{-2})$. The total cost for a BIM is then $\mathcal{O}(N_{\text{eval}} + N_{\text{iter}}h^{-2})$, which is asymptotically better than the result $\mathcal{O}(h^{-3})$ for volume methods, as long as N_{eval} does not scale as h^{-3} or worse.

Finally, we note that while remeshing may be needed for fitted volume methods, it is generally not necessary for unfitted volume methods, nor for BIMs.

Challenges in boundary integral methods

We finish this chapter by outlining two of the major challenges in using boundary integral methods to simulate Stokes flow with rigid particles. Both of these are areas of active research, and have been studied as part of the thesis work. The main contributions of this thesis will be described in Chapters 4 and 5.

- In dense suspensions, the particles will tend to come very close to each other, and to the walls. As will become apparent in Chapter 3, the boundary integrals contain functions with singularities, which become hard to resolve using standard quadrature methods when particles are very close to each other. Special quadrature methods are needed to accurately compute the integrals, and this is the topic of Chapter 4.
- As mentioned above, fast summation methods are necessary to make boundary integral methods computationally competitive. The Spectral Ewald method is a type of fast summation method that can be used for both periodic and free-space problems, and it is the main topic of Chapter 5.

3 Boundary integral formulations of Stokes flow

In this chapter, we will introduce the basis for boundary integral methods for Stokes flow, namely the boundary integral formulation. We start by introducing the fundamental solutions (Green's functions) of Stokes flow, which are at the core of the formulation. Then we continue by introducing the boundary integrals, and discuss different alternative formulations. Finally, we outline a solution procedure, i.e., how the boundary integral formulation can be used to compute relevant quantities of Stokes flow.

The discussions in this chapter will focus on computations in free space; the topic of periodic problems is deferred until Chapter 5.

3.1 Fundamental solutions of Stokes flow

Let us, for the remainder of this chapter – and indeed for the rest of Part I – consider the Stokes equations (2.2.8) with $\mu = 1$, which is the same as nondimensionalizing the equations (but without putting a tilde above all quantities as in (2.2.3)).

The fundamental solutions of Stokes flow represent different physical quantities of the Stokesian flow induced by a *point force*. Thus, let the forcing term in (2.2.8) be given by a point force $\mathbf{f}(\mathbf{x}) = \mathbf{f}^* \delta(\mathbf{x} - \mathbf{x}^*)$. Here, \mathbf{f}^* is a constant vector (the force) and \mathbf{x}^* is the location of the point force; $\delta(\mathbf{x})$ is the Dirac delta distribution. Equation (2.2.8) should hold in the whole of \mathbb{R}^3 , with the condition that the velocity goes to zero as $|\mathbf{x}| \rightarrow \infty$. Then, we have the following fundamental solutions, i.e., solutions to (2.2.8) under the aforementioned assumptions (for details and derivations, see e.g.

[94, §2.2] or [65, §2.4.1]):

- The velocity field is given by

$$\mathbf{u}(\mathbf{x}) = \frac{1}{8\pi} \mathbf{S}(\mathbf{x} - \mathbf{x}^*) \mathbf{f}^*, \quad (3.1.1)$$

where $\mathbf{S}(\mathbf{r})$ is a matrix field (i.e., an order 2 tensor field) known as the *stokeslet* (or sometimes the *Oseen tensor*).¹ The entries S_{jl} of the stokeslet are given by

$$S_{jl}(\mathbf{r}) = \frac{\delta_{jl}}{|\mathbf{r}|} + \frac{r_j r_l}{|\mathbf{r}|^3}, \quad (3.1.2)$$

where the indices j and l range over the set $\{1, 2, 3\}$, and δ_{jl} is the Kronecker delta, defined by $\delta_{jl} = 1$ if $j = l$ and $\delta_{jl} = 0$ otherwise.

At times, we will use index notation and write (3.1.1) as²

$$u_j(\mathbf{x}) = \frac{1}{8\pi} S_{jl}(\mathbf{x} - \mathbf{x}^*) f_l^*. \quad (3.1.3)$$

- The pressure field is given by

$$p(\mathbf{x}) = \frac{1}{8\pi} \mathbf{\Pi}(\mathbf{x} - \mathbf{x}^*) \cdot \mathbf{f}^*, \quad (3.1.4)$$

where $\mathbf{\Pi}(\mathbf{r})$ is a vector field with the components

$$\Pi_j(\mathbf{r}) = 2 \frac{r_j}{|\mathbf{r}|^3}. \quad (3.1.5)$$

¹The terminology varies between authors; Kim & Karrila [65] refer to \mathbf{S} as the “Oseen tensor”, while $\mathbf{S}/8\pi$ is the “Oseen–Burgers tensor”, and the term “stokeslet” is reserved for the solution \mathbf{u} itself, i.e., (3.1.1). However, Pozrikidis [94] refers to \mathbf{S} as the “stokeslet”, or alternatively as the “Oseen–Burgers tensor”. In this thesis, “stokeslet” will refer to the tensor \mathbf{S} .

²In this thesis, Einstein’s summation convention is used, meaning that repeated indices are implicitly summed over $\{1, 2, 3\}$. For instance, $S_{jl} f_l^*$ should be understood as

$$\sum_{l=1}^3 S_{jl} f_l^*.$$

- The vorticity field, i.e., $\boldsymbol{\omega} = \nabla \times \mathbf{u}$, is given by

$$\boldsymbol{\omega}(\mathbf{x}) = \frac{1}{4\pi} \boldsymbol{\Omega}(\mathbf{x} - \mathbf{x}^*) \mathbf{f}^*, \quad (3.1.6)$$

where the *rotlet* has the entries

$$\Omega_{jl}(\mathbf{r}) = \epsilon_{jlm} \frac{r_m}{|\mathbf{r}|^3}. \quad (3.1.7)$$

Here, ϵ_{jlm} is the Levi-Civita symbol (also known as the antisymmetric symbol, or alternating symbol).

- The stress field is given by

$$\sigma_{jl}(\mathbf{x}) = \frac{1}{8\pi} T_{jlm}(\mathbf{x} - \mathbf{x}^*) f_m^*, \quad (3.1.8)$$

where the *stresslet* has the entries

$$T_{jlm}(\mathbf{r}) = -6 \frac{r_j r_l r_m}{|\mathbf{r}|^5}. \quad (3.1.9)$$

Here, σ_{jl} is the stress tensor, which for a Newtonian fluid is given by [94, §1.2]

$$\sigma_{jl} = -p\delta_{jl} + \frac{\partial u_j}{\partial x_l} + \frac{\partial u_l}{\partial x_j}. \quad (3.1.10)$$

(Perhaps the fundamental solution for the pressure, $\mathbf{\Pi}$, ought to be called the “pressurelet” to fit into the naming scheme, but this name does not seem to be used in the literature.)

Not only does the rotlet describe the vorticity field due to a point force, as seen in (3.1.6), but it can also be used to describe the *velocity* field due to a point *torque*, i.e.,

$$\mathbf{u}(\mathbf{x}) = \frac{1}{8\pi} \boldsymbol{\Omega}(\mathbf{x} - \mathbf{x}^*) \boldsymbol{\tau}^*, \quad (3.1.11)$$

where $\boldsymbol{\tau}^*$ is the torque (see e.g. [94, §2.5] or [65, §2.5]).³ This velocity field is also a solution to the Stokes equations.

³Equation (3.1.11) can be derived by writing the torque as a force couple, utilizing (3.1.1), and taking the limit as the separation of the two forces goes to zero. For details, see [65, p. 29].

The stresslet can also produce a velocity field that solves the Stokes equations. Such a field is given by⁴

$$u_j(\mathbf{x}) = T_{jlm}(\mathbf{x} - \mathbf{x}^*)c_{lm}^*, \quad (3.1.12)$$

where \mathbf{c}^* is any constant matrix (see e.g. [94, §3.2]). Below, we will always use source terms of the form $c_{lm}^* = q_l^* n_m^*$ for constant vectors \mathbf{q}^* and \mathbf{n}^* . The typical flow fields generated by a stokeslet (3.1.1), rotlet (3.1.11) and stresslet (3.1.12) are shown in Figure 3.1.

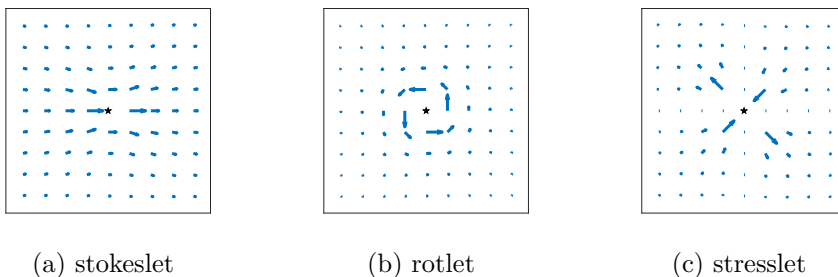


FIGURE 3.1: Typical flow fields generated by the stokeslet, rotlet and stresslet: (a) shows (3.1.1) with $\mathbf{f}^* = (1, 0, 0)$, (b) shows (3.1.11) with $\boldsymbol{\tau}^* = (0, 0, 1)$, and (c) shows (3.1.12) with $c_{lm}^* = q_l^* n_m^*$ and $\mathbf{q}^* = (1, 0, 0)$, $\mathbf{n}^* = (0, 1, 0)$. All plots show the velocity field in the x_1x_2 -plane, with \mathbf{x}^* marked by a black star \star .

Relations between the fundamental solutions

The rotlet and stresslet can be related to derivatives of the stokeslet. For instance, from the relation $\boldsymbol{\omega} = \nabla \times \mathbf{u}$, (3.1.1) and (3.1.6), we see that the rotlet satisfies

$$\boldsymbol{\Omega}(\mathbf{r}) = \frac{1}{2} \nabla \times \mathbf{S}(\mathbf{r}), \quad (3.1.13)$$

or in index notation

$$\Omega_{jl}(\mathbf{r}) = \frac{1}{2} \epsilon_{jmn} \frac{\partial}{\partial r_m} S_{nl}(\mathbf{r}). \quad (3.1.14)$$

⁴We could include a factor $1/8\pi$ in the right-hand side of (3.1.12), but since c_{lm}^* has not been given any physical interpretation here, we choose to absorb the factor $1/8\pi$ into c_{lm}^* . This will make our notation agree with that in Papers A–C & E.

For the stresslet, the relation (3.1.10) implies that

$$T_{jlm}(\mathbf{r}) = -\Pi_j(\mathbf{r})\delta_{lm} + \frac{\partial}{\partial r_m}S_{jl}(\mathbf{r}) + \frac{\partial}{\partial r_l}S_{mj}(\mathbf{r}). \quad (3.1.15)$$

All of the fundamental solutions of Stokes flow can also be related to the fundamental solutions of electrostatics, namely the harmonic kernel $H(\mathbf{r})$ and biharmonic kernel $B(\mathbf{r})$, defined below, and this will turn out to be useful both in the context of special quadrature methods (Chapter 4) and fast summation methods (Chapter 5). The harmonic kernel

$$H(\mathbf{r}) = \frac{1}{|\mathbf{r}|} \quad (3.1.16)$$

is the solution to the Poisson equation $-\nabla^2 H(\mathbf{r}) = 4\pi\delta(\mathbf{r})$, while the biharmonic kernel

$$B(\mathbf{r}) = |\mathbf{r}| \quad (3.1.17)$$

is the solution to the biharmonic equation $-\nabla^4 B(\mathbf{r}) = 8\pi\delta(\mathbf{r})$. (Also note that $H = \frac{1}{2}\nabla^2 B$.) The relations are as follows:

- The stokeslet (3.1.2) satisfies⁵

$$S_{jl}(\mathbf{r}) = (\delta_{jl} - r_j\nabla_l)H(\mathbf{r}), \quad (3.1.18)$$

and

$$S_{jl}(\mathbf{r}) = \left(\delta_{jl}\nabla^2 - \nabla_j\nabla_l\right)B(\mathbf{r}). \quad (3.1.19)$$

The latter relation was noted for example by Beenakker [18] and Claeys [31, App. E, p. 113], and is especially useful since it is easy to represent in Fourier space (we will return to this in Section 5.3).

- The pressure fundamental solution, or “pressurelet”, (3.1.5) satisfies

$$\Pi_j(\mathbf{r}) = -2\nabla_j H(\mathbf{r}). \quad (3.1.20)$$

- The rotlet (3.1.7) satisfies

$$\Omega_{jl}(\mathbf{r}) = -\epsilon_{jlm}\nabla_m H(\mathbf{r}). \quad (3.1.21)$$

⁵Here and below, we use the notation $\nabla_j = \frac{\partial}{\partial r_j}$ for brevity.

- For the stresslet (3.1.9), Tornberg & Greengard [117] noted that

$$T_{jlm}(\mathbf{r}) = [(\delta_{jl} - r_j \nabla_l) \nabla_m + (\delta_{lm} - r_m \nabla_l) \nabla_j] H(\mathbf{r}), \quad (3.1.22)$$

and Fan et al. [40] noted that

$$T_{jlm}(\mathbf{r}) = [(\delta_{jl} \nabla_m + \delta_{mj} \nabla_l + \delta_{lm} \nabla_j) \nabla^2 - 2 \nabla_j \nabla_l \nabla_m] B(\mathbf{r}). \quad (3.1.23)$$

3.2 Boundary integral formulations

In this section, we let V be any volume bounded by a closed surface Γ with outward-pointing unit normal \mathbf{n} , as shown in Figure 3.2. We may think of V as the volume occupied by a particle, and Γ as the surface of the particle, but it could also be an arbitrary volume. In the end, Γ will be selected as the union of all particle and wall surfaces, as it was in Figure 2.2.

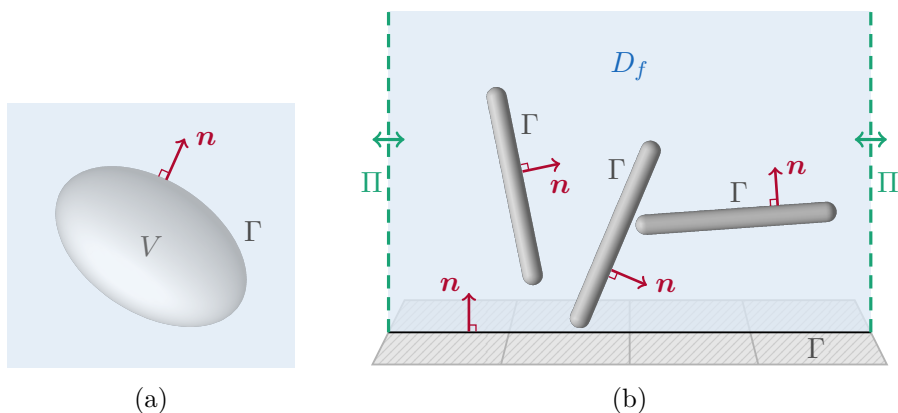


FIGURE 3.2: Notation for the boundary integral formulation: (a) An arbitrary volume V with boundary Γ ; (b) In the end, Γ will be the union of all rigid boundaries.

The direct formulation

The boundary integral formulations for Stokes flow can be motivated by the *Lorenz reciprocal identity*, which states that for any two nonsingular solutions \mathbf{u} and $\bar{\mathbf{u}}$ to the Stokes equation (2.2.8), with accompanying stress

tensors $\boldsymbol{\sigma}$ and $\bar{\boldsymbol{\sigma}}$ (defined through (3.1.10)), respectively, the relation

$$\nabla_l \bar{u}_j \sigma_{jl} = \nabla_l u_j \bar{\sigma}_{jl}. \quad (3.2.1)$$

holds [94, §1.4]. Inserting the fundamental solutions (3.1.1) and (3.1.8) in place of $\bar{\mathbf{u}}$ and $\bar{\boldsymbol{\sigma}}$, respectively, into (3.2.1), we get

$$f_m^* \nabla_l S_{jm}(\mathbf{x} - \mathbf{x}^*) \sigma_{jl}(\mathbf{x}) = f_m^* \nabla_l u_j(\mathbf{x}) T_{jlm}(\mathbf{x} - \mathbf{x}^*). \quad (3.2.2)$$

This relation holds everywhere except at $\mathbf{x} = \mathbf{x}^*$, where the solutions are singular. Removing the arbitrary constant vector \mathbf{f}^* and integrating the remaining relation over any volume V (not including \mathbf{x}^*) bounded by a closed surface Γ , then using the divergence theorem, we find [94, §2.3]

$$\int_{\Gamma} n_l(\mathbf{y}) S_{jm}(\mathbf{y} - \mathbf{x}^*) \sigma_{jl}(\mathbf{y}) dS(\mathbf{y}) = \int_{\Gamma} n_l(\mathbf{y}) u_j(\mathbf{y}) T_{jlm}(\mathbf{y} - \mathbf{x}^*) dS(\mathbf{y}), \quad (3.2.3)$$

where \mathbf{n} is the unit normal of Γ pointing out of V . In (3.2.3), we have renamed the integration variable from \mathbf{x} to \mathbf{y} , in order to keep the notation here consistent with Papers A–F.

Based on (3.2.3), we define two integral operators, namely

$$\mathcal{S}_j[\boldsymbol{\varphi}](\mathbf{x}) = \int_{\Gamma} S_{jl}(\mathbf{x} - \mathbf{y}) \varphi_l(\mathbf{y}) dS(\mathbf{y}), \quad (3.2.4)$$

$$\mathcal{D}_j[\mathbf{q}](\mathbf{x}) = \int_{\Gamma} T_{jlm}(\mathbf{x} - \mathbf{y}) q_l(\mathbf{y}) n_m(\mathbf{y}) dS(\mathbf{y}), \quad (3.2.5)$$

where $\boldsymbol{\varphi}$ and \mathbf{q} are functions defined on Γ , and the kernels S_{jl} and T_{jlm} are as in (3.1.2) and (3.1.9), respectively. We call \mathcal{S}_j the *single layer potential* and \mathcal{D}_j the *double layer potential*.⁶ Using the fact that $\varphi_j = \sigma_{jl} n_l$ is the surface force exerted on Γ , we can now write (3.2.3) as

$$\mathcal{S}[\boldsymbol{\varphi}](\mathbf{x}^*) = -\mathcal{D}[\mathbf{u}](\mathbf{x}^*), \quad (3.2.6)$$

where we have used the properties $\mathcal{S}(-\mathbf{r}) = \mathcal{S}(\mathbf{r})$ and $S_{lj} = S_{jl}$ of the stokeslet (3.1.2), and $\mathcal{T}(-\mathbf{r}) = -\mathcal{T}(\mathbf{r})$ and $T_{mj l} = T_{j l m}$ of the stresslet (3.1.9).

⁶These names are borrowed from electrostatics, where the corresponding integrals represent the electrostatic potential from a distribution of charges, and dipoles, respectively.

Equation (3.2.6) is valid for any \mathbf{x}^* *outside* of the selected volume V . For a point \mathbf{x}^* *inside* the volume V , one can show that [94, §2.3]⁷

$$-8\pi\mathbf{u}(\mathbf{x}^*) + \mathcal{S}[\varphi](\mathbf{x}^*) = -\mathcal{D}[\mathbf{u}](\mathbf{x}^*). \quad (3.2.7)$$

As \mathbf{x}^* approaches the boundary Γ , the double layer potential \mathcal{D} has different limits depending on which side of the boundary \mathbf{x}^* comes from [94, §4.3], namely

$$\lim_{\varepsilon \rightarrow 0^+} \mathcal{D}[\mathbf{q}](\mathbf{x} \pm \varepsilon \mathbf{n}) = \mathcal{D}[\mathbf{q}](\mathbf{x}) \mp 4\pi\mathbf{q}(\mathbf{x}), \quad \mathbf{x} \in \Gamma, \quad (3.2.8)$$

where $\mathcal{D}[\mathbf{q}](\mathbf{x})$ on the boundary is an improper integral (known as the principal value integral), since the integrand is singular (the stresslet is singular). Recall here that the normal vector \mathbf{n} points out of V . For a point $\mathbf{x}^* \in \Gamma$, we thus get

$$-4\pi\mathbf{u}(\mathbf{x}^*) + \mathcal{S}[\varphi](\mathbf{x}^*) = -\mathcal{D}[\mathbf{u}](\mathbf{x}^*). \quad (3.2.9)$$

Note that (3.2.6), (3.2.7) and (3.2.9) can be summarized as (we now drop the star on \mathbf{x}^*)

$$\mathcal{S}[\varphi](\mathbf{x}) + \mathcal{D}[\mathbf{u}](\mathbf{x}) = \begin{cases} \mathbf{0}, & \text{if } \mathbf{x} \text{ is outside } V, \\ 4\pi\mathbf{u}(\mathbf{x}), & \text{if } \mathbf{x} \text{ is on } \Gamma, \\ 8\pi\mathbf{u}(\mathbf{x}), & \text{if } \mathbf{x} \text{ is inside } V. \end{cases} \quad (3.2.10)$$

Note that the jump comes only from the double layer potential \mathcal{D} ; the single layer potential is continuous across Γ .

The boundary integral formulation represented by (3.2.10) is called the *direct formulation*, and leads to the boundary integral equation (3.2.9). It is useful for instance for interior Dirichlet problems, i.e., when $\mathbf{u} = \mathbf{g}$ is known on Γ and one seeks the solution \mathbf{u} inside V . Then, (3.2.9) can be solved for φ on Γ , and subsequently \mathbf{u} inside V can be computed from (3.2.10). However, as it stands, (3.2.10) cannot represent a nonzero exterior flow. This limitation can be removed by considering *indirect*, or *generalized*, formulations.

⁷The reader comparing our (3.2.7) with equation (2.3.11) in [94] will find that there is a difference in signs. This is because we have selected the normal vector \mathbf{n} to be outward-pointing, while [94] makes the opposite choice. One should also keep in mind that $\mathbf{T}(\mathbf{y} - \mathbf{x}) = -\mathbf{T}(\mathbf{x} - \mathbf{y})$ when comparing equations.

Indirect (generalized) formulations

Indirect formulations are based on noting that the single layer potential (3.2.4) and double layer potential (3.2.5), considered separately, are perfectly fine solution to the Stokes equations, for arbitrary functions φ and \mathbf{q} . Thus, we can choose any of the following representations:

- Single layer representation: $\mathbf{u}(\mathbf{x}) = \mathcal{S}[\varphi](\mathbf{x})$
- Double layer representation: $\mathbf{u}(\mathbf{x}) = \mathcal{D}[\mathbf{q}](\mathbf{x})$
- Mixed representation: $\mathbf{u}(\mathbf{x}) = \alpha\mathcal{S}[\mathbf{q}](\mathbf{x}) + \beta\mathcal{D}[\mathbf{q}](\mathbf{x})$ for given $\alpha, \beta \in \mathbb{R}$.

Not all formulations lead to a well-conditioned linear system, and the choice of formulation will typically depend on the type of problem (interior or exterior, Dirichlet or Neumann boundary conditions, etc). In this thesis, we consider exterior problems with no-slip boundary conditions, and the double layer representation is then a suitable choice, as it is relatively simple and leads to a well-conditioned system.

Unfortunately, the double layer representation is “incomplete”, in the sense that it cannot represent all kinds of Stokes flows. In fact, it can only represent flows that exert no net force or torque on Γ .⁸ This is related to the presence of a nontrivial nullspace in the double layer operator, i.e., there are nonzero functions \mathbf{q} that result in a zero flow field $\mathcal{D}[\mathbf{q}](\mathbf{x})$. For this reason, a “completion” term $\mathcal{V}(\mathbf{x})$ due to Power & Miranda [93] is added to the formulation. This term consists of a number of stokeslets and rotlets, which can represent nonzero forces $\mathbf{f}^{(\alpha)}$ and torques $\boldsymbol{\tau}^{(\alpha)}$ (c.f. (3.1.1) and (3.1.11)):

$$\mathcal{V}(\mathbf{x}) = \sum_{\alpha} \frac{1}{8\pi} \left(\mathcal{S}(\mathbf{x} - \mathbf{x}^{(\alpha)}) \mathbf{f}^{(\alpha)} + \boldsymbol{\Omega}(\mathbf{x} - \mathbf{x}^{(\alpha)}) \boldsymbol{\tau}^{(\alpha)} \right), \quad (3.2.12)$$

where $\mathbf{x}^{(\alpha)}$ is an arbitrary point outside of the fluid domain (e.g. inside a particle).

⁸The net force exerted on Γ by the fluid is given by [94, §1.2]

$$F_j = \int_{\Gamma} \varphi_j \, dS = \int_{\Gamma} \sigma_{jl} n_l \, dS, \quad (3.2.11)$$

where σ_{jl} is the stress tensor associated with the flow. Using (3.1.10) and the divergence theorem, the net force can also be expressed as $F_j = \int_V (-\nabla_j p + \nabla^2 u_j) \, dV = -\int_V f_j \, dV$; compare with (2.2.8) with $\mu = 1$.

The net torque on Γ is consequently $\boldsymbol{\tau} = \int_{\Gamma} \mathbf{x} \times \boldsymbol{\varphi} \, dS$.

The completed double layer formulation is thus

$$\mathbf{u}(\mathbf{x}) = \mathcal{D}[\mathbf{q}](\mathbf{x}) + \mathcal{V}(\mathbf{x}), \quad (3.2.13)$$

with \mathcal{D} as in (3.2.5) and \mathcal{V} as in (3.2.12). Letting \mathbf{x} approach Γ leads to the boundary integral equation

$$\mathbf{u}(\mathbf{x}) = \mathcal{D}[\mathbf{q}](\mathbf{x}) - 4\pi\mathbf{q}(\mathbf{x}) + \mathcal{V}(\mathbf{x}), \quad \mathbf{x} \in \Gamma. \quad (3.2.14)$$

In this equation, $\mathbf{q}(\mathbf{x})$ – called the *double layer density* – is the unknown. There are two cases (c.f. Section 2.4):

- (i) For the mobility problem, the force and torque on each particle are known, so the completion term \mathcal{V} is known. The left-hand side of (3.2.14), i.e., the motion of the particles, is unknown. To close the system, the rigid body motion of the particles is coupled to \mathbf{q} . For details, see §2.1.2 in Paper B.
- (ii) For the resistance problem, the velocities of the particles are known, so the left-hand side of (3.2.14) is known. The resulting forces and torques on the particles, i.e., $\mathbf{f}^{(\alpha)}$ and $\boldsymbol{\tau}^{(\alpha)}$, are unknown. They are coupled to \mathbf{q} according to the formulas in §2.1.1 in Paper B.

In both cases, the integral equation (3.2.14) becomes a *Fredholm integral equation* of the second kind, meaning that it is of the form [10, §1.1.4]

$$\lambda q(x) - \int_{\Gamma} K(x, y)q(y) dy = g(x), \quad (3.2.15)$$

where $\lambda \neq 0$, $K(x, y)$ and $g(x)$ are given and $q(x)$ is unknown.⁹ As mentioned in Section 2.5, the discretization of this kind of integral equations leads to a linear system which remains well-conditioned as the discretization is refined [10, p. 113] [68, p. 282].

In the preparatory work for **Paper B**, it was discovered that a single completion source (force $\mathbf{f}^{(\alpha)}$ and torque $\boldsymbol{\tau}^{(\alpha)}$) per particle was not sufficient for particles of high aspect ratio (elongated particles). The reason is that it

⁹Note that the jump term $4\pi\mathbf{q}(\mathbf{x})$ in (3.2.14), which comes from the double layer potential, is crucial in making (3.2.14) a Fredholm integral equation of the second kind. Without this term, it would be an equation of the first kind [10, §1.1.5], which does not have quite as nice properties.

leads to a completion term \mathbf{V} , and a solution $\mathbf{q}(\mathbf{x})$, that varies rapidly along the boundary and therefore becomes hard to resolve (for details, see §7.2.1 in Paper B). Our solution was to place multiple completion sources along the centerline of the particle, in a similar fashion as [61]. The completion sources should be placed symmetrically about the center of the particle, since otherwise the point forces would introduce a torque about the center.

3.3 Solution procedure

To solve a resistance or mobility problem involving rigid particles in Stokes flow, the following steps are needed:

1. Solve the boundary integral equation (3.2.14) to determine the double layer density $\mathbf{q}(\mathbf{x})$ on the surfaces of particles and walls, Γ .

The integral equation looks different depending on whether it is a resistance or mobility problem being solved. In both cases:

- We solve the integral equation iteratively using GMRES, meaning that a current guess for $\mathbf{q}(\mathbf{x})$ will be available in each iteration, so that (3.2.14) can be evaluated.
 - The double layer potential $\mathcal{D}[\mathbf{q}](\mathbf{x})$, which is given by (3.2.5) and appears in (3.2.14), must be computed using a quadrature rule. Since the stresslet kernel T_{jlm} , given by (3.1.9), is singular on Γ , **special quadrature** is needed (Chapter 4).
 - We use the Nyström method, meaning that (3.2.14) is enforced at the quadrature nodes.
2. Once $\mathbf{q}(\mathbf{x})$ is known, it can be used to:
 - compute the forces and torques acting on each particle (for the resistance problem), or the velocity of each particle (for the mobility problem), using the relations from Paper B, §2.1.1–2.1.2;
 - evaluate the flow field $\mathbf{u}(\mathbf{x})$ at any point, using (3.2.13). Special quadrature may be needed also here, if \mathbf{x} is close to the boundary.
 3. Update the position and orientation of each particle using the equations of motion for a rigid body (see e.g. [11, §3]).

The steps above must be done in each time step. The discretization of the boundary integrals is done using high-order quadrature rules, such as the trapezoidal rule (which has spectral accuracy for periodic integrands) and Gauss–Legendre quadrature. Details are found in §3 of Paper B.

Fast summation methods (Chapter 5) can be used both in step 1 (to evaluate the integral equation at the quadrature nodes) and in step 2 (to evaluate the flow field at any point).

4 Special quadrature methods

Quadrature methods, i.e., methods for numerically approximating integrals, are needed to discretize and compute the integrals appearing in the boundary integral formulation, such as the double layer potential given by (3.2.5). Since the stresslet kernel T_{jlm} (3.1.9) in this integral is singular at $\mathbf{r} = \mathbf{0}$, special quadrature methods are needed to accurately compute the integral.

This chapter starts with a motivation of why standard quadrature methods, such as the trapezoidal rule or Gauss–Legendre quadrature, are not sufficient when evaluating integrals with singular or nearly singular (rapidly varying) integrands, in Section 4.1. Then, in Section 4.2, we give an overview of available special quadrature methods, with a focus on methods applicable to three-dimensional simulations. Two special quadrature methods are given special attention: the quadrature by expansion (QBX) method, which is used in Papers A–C, is described in Section 4.3, and the Hedgehog method, which is studied in Paper F, appears in Section 4.4.

4.1 The need for special quadrature methods

A one-dimensional toy example

Let us first consider a one-dimensional example to illustrate why special quadrature methods are needed. Say that we are interested in computing the integral

$$I(x) = \int_{-1}^1 \frac{1}{\sqrt{x + |y|}} dy, \quad (4.1.1)$$

where $x \geq 0$ is a parameter. (The symbols x and y here are chosen to reflect their roles in the double layer potential (3.2.5).) The integral in question

does, of course, have a closed-form expression – this is a toy example, after all – and it is given by

$$I(x) = 4 \left(\sqrt{x+1} - \sqrt{x} \right). \quad (4.1.2)$$

Note that this value is well-defined for any $x \geq 0$. In particular, for $x = 0$ the value of the integral becomes $I(0) = 4$.

The integrand of the integral (4.1.1), i.e., $f(y) = 1/\sqrt{x+|y|}$, is shown as a function of y for some different values of the parameter x in Figure 4.1.

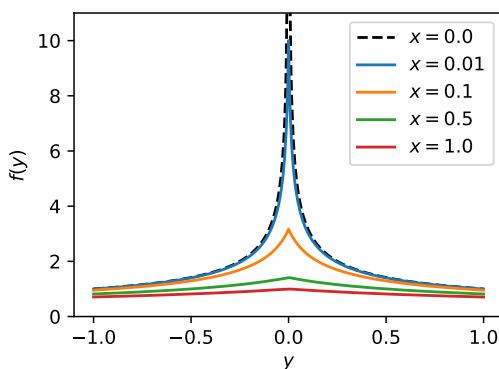


FIGURE 4.1: Plot of the integrand of (4.1.1).

For $x = 0$, the integrand becomes $1/\sqrt{|y|}$, which is singular at the point $y = 0$; for this reason, we call $x = 0$ the **singular case**. Nevertheless, as we saw from (4.1.2), the improper integral corresponding to this case has a well-defined value, namely $I(0) = 4$.

For $x > 0$, the integrand is not singular anywhere (although in this example it is not differentiable at $y = 0$). However, for small but nonzero x , the integrand is very peaked, as seen e.g. for $x = 0.01$ in Figure 4.1. We call this case (small but nonzero x) the **nearly singular case**.

Both the singular and nearly singular cases are challenging for a standard quadrature rule, since the integrand is very peaked and rapidly varying with sharp gradients in both cases, and therefore hard to resolve. (The singular case has the additional difficulty that the integrand cannot be evaluated at $y = 0$, but this is easily avoided by not placing a quadrature node at that

point.) To illustrate this, let us introduce a standard quadrature rule for computing the integral. Since the integrand is not differentiable at $y = 0$, we will split the integration interval into two pieces, $[-1, 0]$ and $[0, 1]$, and apply an $N/2$ -point Gauss–Legendre quadrature rule on each subinterval. This means that, in total, we will have N quadrature nodes $\{y_i\}$ with corresponding quadrature weights $\{w_i\}$, and the integral is approximated as

$$I_N(x) = \sum_{i=1}^N f(y_i)w_i \approx I(x). \quad (4.1.3)$$

The quadrature error decreases exponentially as $N \rightarrow \infty$ for piecewise smooth integrands, assuming they are well resolved. This is illustrated for the integral $\int_{-1}^1 \cos(y) dy$ in Figure 4.2.

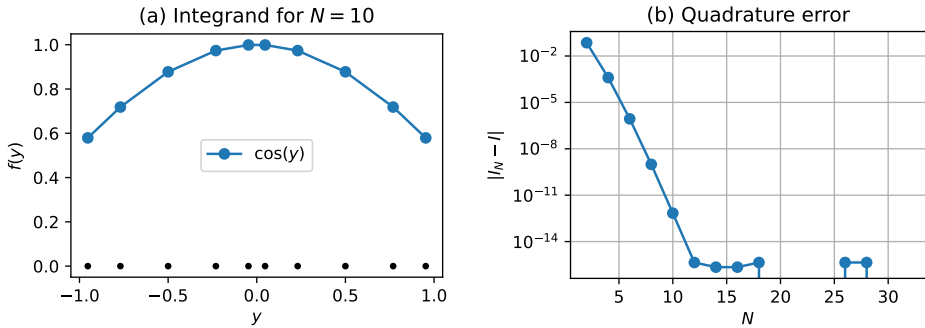


FIGURE 4.2: (a) Gauss–Legendre quadrature nodes y_i and integrand values $f(y_i)$, for $N = 10$ nodes. (b) Quadrature error $|I_N - I|$ as a function of N . The integrand is $f(y) = \cos(y)$.

The corresponding plots for our test integral (4.1.1), i.e., with the integrand $f(y) = 1/\sqrt{x + |y|}$, are shown in Figure 4.3. For large values of x , the integrand is well resolved using relatively few quadrature nodes, and the error decreases exponentially as $N \rightarrow \infty$, similar to the $\cos(y)$ case. However, for both the singular and nearly singular cases, the integrand becomes hard to resolve, and the quadrature error decays very slowly as N increases. As can be seen e.g. for $x = 0$ or $x = 0.01$ in Figure 4.3 (b), simply increasing N is not a viable strategy to reduce the error. Instead, special quadrature methods are needed.

Note that, when computing higher-dimensional integrals, we use a tensorial quadrature rule, i.e. a one-dimensional quadrature rule such as (4.1.3) is

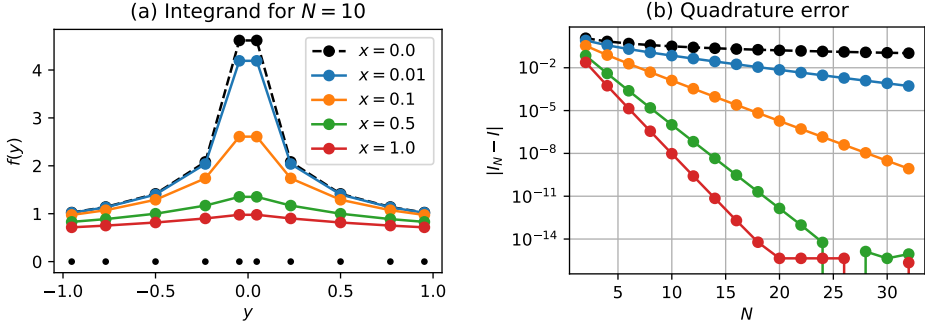


FIGURE 4.3: (a) Gauss–Legendre quadrature nodes y_i and integrand values $f(y_i)$, for $N = 10$ nodes. (b) Quadrature error $|I_N - I|$ as a function of N . The integrand is $f(y) = 1/\sqrt{x + |y|}$.

employed along each coordinate direction. In this context, equation (4.1.3) may represent e.g. either Gauss–Legendre quadrature or the trapezoidal rule. If then N quadrature nodes are used in each direction, we need in total $\mathcal{O}(N^d)$ points for a d -dimensional integral. Recall that the boundary integral formulation reduces the computational dimension by one. Thus, for the three-dimensional case considered below, d would be two.

The three-dimensional case

Equipped with some understanding from the one-dimensional toy example, we are now ready to consider the evaluation of the double layer potential (3.2.5) in a three-dimensional flow problem. Let us for clarity reproduce the aforementioned equation here:

$$\mathcal{D}_j[\mathbf{q}](\mathbf{x}) = \int_{\Gamma} T_{jlm}(\mathbf{x} - \mathbf{y}) q_l(\mathbf{y}) n_m(\mathbf{y}) dS(\mathbf{y}). \quad (3.2.5)$$

We consider as an example Γ to be the surface of a rodlike particle, discretized using a combination of Gauss–Legendre quadrature and the trapezoidal rule (as in Paper B). The trapezoidal rule is used in the azimuthal (circular) direction of the rod since it is spectrally accurate for periodic integrands. Gauss–Legendre is used in the other (axial) direction, where the parametrization goes from one tip to the other and is thus not periodic. For details, see §3.1 of Paper B. The standard quadrature rule for the particle given by this discretization will be called the **direct quadrature**.

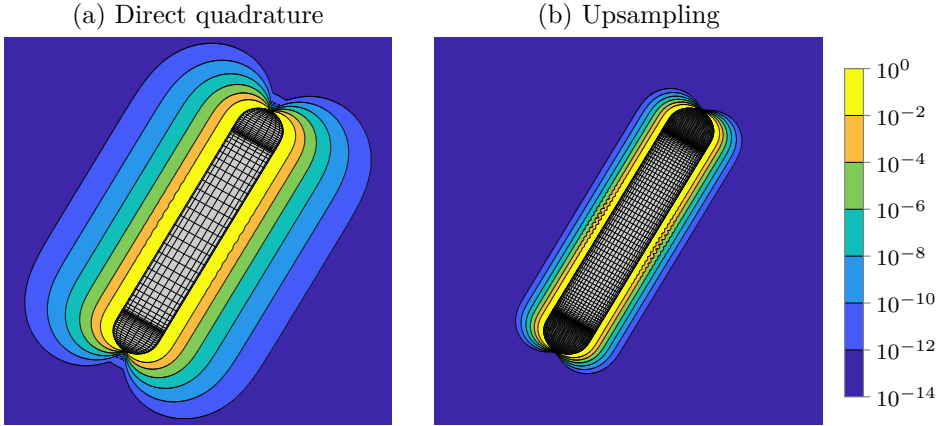


FIGURE 4.4: Quadrature error when evaluating the double layer potential around a rodlike particle (a) using direct quadrature, and (b) using direct quadrature with upsampling.

The quadrature error when using direct quadrature to evaluate (3.2.5) is shown in Figure 4.4 (a). Note that the *singular case* corresponds to an evaluation point \mathbf{x} on the particle surface Γ (since then $\mathbf{x} = \mathbf{y}$ for some point $\mathbf{y} \in \Gamma$), while the *nearly singular case* corresponds to the point \mathbf{x} being close to, but not on, Γ (since then $|\mathbf{x} - \mathbf{y}|$ can become small, but remains nonzero). The singular case occurs when solving the boundary integral equation (step 1 in Section 3.3), since we will then evaluate the layer potential at the quadrature nodes. The nearly singular case occurs e.g. if we want to evaluate the flow field very close to a particle or wall (in step 2 in Section 3.3), or if two particles are very close to each other when solving the boundary integral equation (close interactions).

In this context, the double layer density \mathbf{q} in (3.2.5) is not known analytically, but only as values on the particle grid. These values can be interpolated onto a refined grid upon which the layer potential is computed with direct quadrature, a procedure known as **upsampling**. This would be similar to increasing N in the one-dimensional toy example, and the result of doing this is shown in Figure 4.4 (b). Recall that the computational work in this three-dimensional case scales as $\mathcal{O}(N^2)$, where N is the number of grid points along each direction on the surface. Upsampling moves the error contours closer to the particle surface, but as in the one-dimensional example, it is not very effective for evaluation points very close to the par-

tile surface. For those points, special quadrature methods are needed to accurately compute the layer potential.

4.2 Overview of special quadrature methods

We here give a short overview of existing special quadrature methods. This section is to a large extent a summary of §1.1 in Paper B.

For problems in two-dimensional space (in which Γ would be a curve rather than a surface), there exist time-tested special quadrature methods such as the one by Helsing & Ojala [53], which is accurate and efficient. This method has been used e.g. for simulations of clean [89] and surfactant-covered [90] drops, and vesicles [13]. However, this method is based on complex analysis and does not easily generalize to three-dimensional problems.

For three-dimensional problems – which are our main interest in this thesis – there is no universally established method, and the development of new special quadrature methods is still an active research topic. A number of existing methods exist, some of which are applicable only to the singular case (on the surface Γ), and some to both the singular and nearly singular cases. Some examples are:

- Using a *rotated grid* with the north pole at the singularity. This allows the singularity to be resolved analytically, either locally by means of a “floating” partition of unity [24, 132, 133], or globally [5, 64]. The global approach is more accurate, since no partition of unity needs to be resolved, but it is more computationally expensive since it uses data on the whole particle grid.
- *Regularizing* the kernel so that standard quadrature rules can be used, and then applying correction terms [16, 115, 17, 116]. The accuracy is typically somewhat limited by the approximations.
- *Corrected quadrature rules* are based on modifying the weights of some standard quadrature rule, often the trapezoidal rule, close to the singularity in order to increase the accuracy [7, 78, 127, 58].
- *Quadrature by expansion* (QBX) introduced by Klöckner et al. [67] and Barnett [14]. It is based on a local spherical harmonics expansion of the layer potential, akin to a Taylor expansion. Many variations

of the method have appeared. For a more detailed overview, see §1.1 in Paper B. In general, the method can be combined with high-order quadrature rules to achieve high accuracy. It can be computationally expensive. The method will be described further in Section 4.3.

- The *Hedgehog method* (also called “line extrapolation/interpolation method”) is based on polynomial extrapolation or interpolation along a line [132, 86]. The method has been used e.g. to simulate drops [110] and vesicles [77, 76] in Stokes flow. It will be described further in Section 4.4.

4.3 Quadrature by expansion (QBX)

The QBX method was implemented for the computation of the Stokes double layer potential (3.2.5) for rigid spheroidal particles by Klinteberg & Tornberg [6]. The method is based on expressing the kernel of the integral operator in terms of a local spherical harmonics expansion. Since no such expansion is readily available for the stresslet kernel T_{jlm} (3.1.9) of the double layer potential itself, the relation (3.1.22) between the stresslet and the harmonic kernel $H(\mathbf{r}) = 1/|\mathbf{r}|$ is used, together with the so-called *Laplace expansion*

$$H(\mathbf{x} - \mathbf{y}) = \sum_{l=0}^{\infty} \frac{4\pi}{2l+1} \sum_{m=-l}^l r_x^l Y_l^{-m}(\theta_x, \varphi_x) \frac{1}{r_y^{l+1}} Y_l^m(\theta_y, \varphi_y). \quad (4.3.1)$$

Here, $(r_x, \theta_x, \varphi_x)$ and $(r_y, \theta_y, \varphi_y)$ are spherical coordinates of the points \mathbf{x} and \mathbf{y} , respectively, with respect to some chosen expansion center \mathbf{c} , as shown in Figure 4.5. The point \mathbf{c} can be any point, as long as it is not too close to the boundary Γ . Furthermore, Y_l^m in (4.3.1) is the (normalized) spherical harmonics function of degree l and order m , defined for $|m| \leq l$ by [37]

$$Y_l^m(\theta, \varphi) = \sqrt{\frac{2l+1}{4\pi} \cdot \frac{(l-|m|)!}{(l+|m|)!}} P_l^{|m|}(\cos \varphi) e^{im\theta}, \quad (4.3.2)$$

where P_l^m is the associated Legendre function of degree l and order m . For more details on the QBX formulation, see §4 in Paper B.

Note that in (4.3.1), the point \mathbf{y} is a point on Γ , so it is known in advance and its contribution could be precomputed. As a simplified example, let us

consider the Laplace single layer potential $v(\mathbf{x}) = \int_{\Gamma} H(\mathbf{x} - \mathbf{y})q(\mathbf{y}) \, dS(\mathbf{y})$, which by plugging in (4.3.1) can be expressed as the local expansion

$$v(\mathbf{x}) = \sum_{l=0}^{\infty} \sum_{m=-l}^l r_x^l Y_l^{-m}(\theta_x, \varphi_x) k_l^m[q]. \quad (4.3.3)$$

Here, the coefficients $k_l^m[q]$ of the expansion are

$$k_l^m[q] = \frac{4\pi}{2l+1} \int_{\Gamma} \frac{1}{r_y^{l+1}} Y_l^m(\theta_y, \varphi_y) q(\mathbf{y}) \, dS(\mathbf{y}). \quad (4.3.4)$$

Knowing Γ and the density $q(\mathbf{y})$ on Γ , the coefficients $k_l^m[q]$ can be computed, and then the expansion (4.3.3) can be evaluated at any point \mathbf{x} within the ball shown in Figure 4.5.

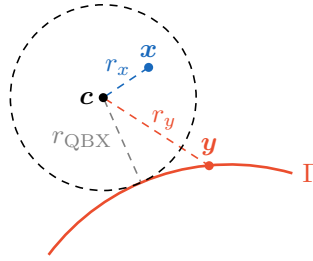


FIGURE 4.5: The points \mathbf{x} , \mathbf{y} and the expansion center \mathbf{c} associated with (4.3.1). The expansion is valid for \mathbf{x} within the ball of radius $r_{\text{QBX}} = \min_{\mathbf{y} \in \Gamma} |\mathbf{y} - \mathbf{c}|$. (Figure from Paper B, reproduced under the Creative Commons Attribution License)

In order to accelerate the evaluation time for QBX, [6] introduced a pre-computation scheme, allowing the expansion coefficients (and also the layer potential at the grid points) to be computed efficiently given the density. The precomputation scheme utilizes some symmetries of the particle, and will, in principle, work for any axisymmetric and mirror symmetric particle shape. (In addition to the spheroids implemented in [6], the precomputation is also used for the rodlike particles introduced in Paper B.)

The QBX method previously implemented for spheroids in [6] is extended in Papers A–B. The contributions of these papers are:

- **Paper A** presents an extension of the method to rigid flat walls discretized with rectangular Gauss–Legendre patches. For singular and

nearly singular evaluation, QBX is used for the nine patches closest to the evaluation point, while direct quadrature is used for all other patches. The precomputation scheme of [6] is adapted, using the symmetries of the rectangular patches.

The accuracy of the method is tested both in the singular (on-surface) and nearly singular (off-surface) cases, and compared with direct quadrature with upsampling in the latter case. It is shown that QBX can achieve high accuracy arbitrarily close to the wall (and on the wall); see Figure 3.8 in Paper A.

- **Paper B** further extends the method to rigid cylindrical pipes (discretized using patches) and rodlike particles. The rodlike particles are constructed to be fully smooth, and the effect of smoothness on the convergence of the local expansions is studied. The precomputation scheme is adapted to the new geometries.

A scheme in which QBX is combined with upsampled direct quadrature is devised, so that QBX is used closest to the boundary Γ , upsampled direct quadrature is used at intermediate distances to Γ , and standard direct quadrature is used when the evaluation point is sufficiently far away. This allows the parameters of the QBX method to be selected for higher computational efficiency.

A parameter selection strategy, designed to ensure that the method as a whole is within a given error tolerance, is presented and thoroughly tested. It is shown that the error is controlled by the given tolerance, as long as the double layer density is well resolved.

Different aspects of the method are tested, e.g. its accuracy and computational complexity. A stresslet identity for the cylindrical pipe and a pair of plane walls is derived, and used to test the accuracy of the method.

4.4 The Hedgehog method

In the Hedgehog method, the layer potential, e.g. the double layer potential (3.2.5), is sampled in a number of points away from the boundary Γ , where it can be evaluated accurately using direct quadrature (possibly with upsampling). The sample points are selected along a line, and a polynomial

interpolant is constructed along this line, and used to approximate the layer potential closer to the boundary. The method is illustrated in Figure 4.6. Here, the potential is sampled in N_{ext} points in the *sampling interval* $[\alpha, \beta]$ along the line, and the polynomial approximant is to be evaluated in the *target interval* $[0, t^*]$.

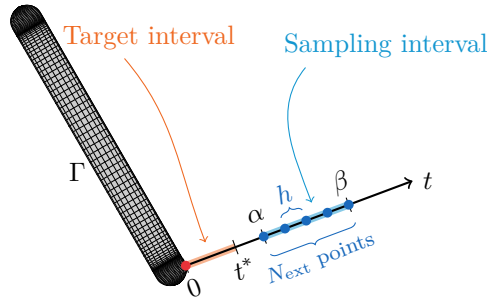


FIGURE 4.6: Illustration of the Hedgehog method and its parameters. In this figure, the sample points are equidistant, but they could be selected according to a different distribution.

If the potential value is available by virtue of some other special method (e.g. QBX) at the point $t = 0$, i.e., where the line touches the boundary surface Γ , this point can be included when constructing the approximant. The problem of approximating the potential in the target interval is then an interpolation problem. If the value at $t = 0$ is not available, the problem is one of extrapolation. The interpolation and extrapolation cases share many features, but one would expect the accuracy in the target interval to become higher when the point at $t = 0$ is included.

The parameters of the Hedgehog method as described here are then as follows:

- The number of sample points N , which is $N = N_{\text{ext}}$ in the extrapolation case, and $N = N_{\text{ext}} + 1$ in the interpolation case.
- The endpoints α and β of the sampling interval, as well as the choice of distribution of the sample points.

(For a fixed distribution, this is two free parameters, α and β . However, one could also consider every sample point as a free parameter of its own. In this case, we would have N_{ext} free parameters here!)

If equidistant sample points with spacing h are used, as in Figure 4.6, the right endpoint β of the sampling interval can be computed as $\beta = \alpha + (N_{\text{ext}} - 1)h$. One can then consider the parameters α and h instead of α and β .

- The upsampling factor κ , which is used when sampling the layer potential in the points in the sampling interval.¹
- The length t^* of the target interval does not affect the method as such, but it controls the region where the method is evaluated.

(For most of Paper F, we set for simplicity $\alpha = t^*$ such that there is no gap between the target and sampling intervals.)

In **Paper F**, this method is studied for a test case consisting of two rigid rodlike particles in Stokes flow. A more detailed description of the method can be found in §2–§4 in Paper F. The contributions of the paper are as follows:

- Rational approximation is considered as an alternative to the polynomial approximation previously used in the Hedgehog method. We use a variant of the *adaptive Antoulas–Anderson* (AAA) algorithm [87] for rational approximation. Spurious poles can pose a problem when using rational approximation, and we discuss ways of dealing with this. We find that rational approximation can reduce the error of the Hedgehog method by about one order of magnitude compared with polynomial approximation, without increasing the computational cost of the method significantly.
- We investigate how much the interpolation case reduces the error of the method compared with the extrapolation case. We find that, in this test case, the error is about one order of magnitude lower for interpolation.
- The accuracy of the Hedgehog method is investigated by varying its parameters for the test case considered here. Strategies for selecting the parameters and estimating the error are discussed.

¹If the surface Γ is discretized using G grid points, upsampling means that the double layer density \mathbf{q} , cf. (3.2.5), is interpolated onto a refined grid consisting of $\kappa^2 G$ points. Direct quadrature is then used on this refined grid.

- The computational cost of the method is discussed and compared with QBX. We find that the Hedgehog method should be able to compete with QBX, although further investigation and more test cases are needed.

For the case of polynomial approximation, we analyze the error contributions of the Hedgehog method, and use this to compute optimal distributions of sample points for a specific case.

5 Fast summation methods and periodic problems

In the numerical method described in Section 3.3, the boundary integral equation (3.2.14) is discretized using N grid points on the boundary Γ , and it is enforced in those same N points. This leads to a linear system of size $3N \times 3N$, which is dense since the double layer potential $\mathcal{D}[\mathbf{q}](\mathbf{x})$ has a slow decay as $|\mathbf{x}| \rightarrow \infty$ (it decays like $|\mathbf{x}|^{-2}$), i.e., all points influence each other. The system is solved iteratively, and in every iteration a dense matrix–vector product is computed, requiring $\mathcal{O}(N^2)$ operations. This matrix–vector product can be accelerated using fast summation methods, so that only $\mathcal{O}(N \log N)$, or even $\mathcal{O}(N)$, operations are needed. This chapter begins with an overview of fast summation methods, in Section 5.1.

The fast summation method that is studied in Papers D–E, i.e., the Spectral Ewald (SE) method, is especially well suited for periodic problems, and therefore periodic problems are also introduced in this chapter, in Section 5.2. The SE method is then outlined in Section 5.3, together with a summary of the contributions of Papers D–E.

5.1 Overview of fast summation methods

The goal of a fast summation method is to compute a sum of the form

$$u(\mathbf{x}_m) = \sum_{n=1}^N G(\mathbf{x}_m - \mathbf{y}_n)q(\mathbf{y}_n) \quad (5.1.1)$$

in M evaluation points (targets) $\{\mathbf{x}_m\}$, where $\{\mathbf{y}_n\}$ is a set of N source points with strengths $\{q(\mathbf{y}_n)\}$. The kernel $G(\cdot)$ describes the propagation

from source to target, and it may be a tensor, e.g. one of the fundamental solutions (Green's functions) of Stokes flow (i.e., the stokeslet, rotlet or stresslet). Indeed, discretizing the double layer potential (3.2.5) will yield a sum of the form of (5.1.1).

Evaluating the sum (5.1.1) directly will require $\mathcal{O}(MN)$ operations. If we assume that the evaluation points coincide with the source points, then $M = N$ and $\mathcal{O}(N^2)$ operations are required to evaluate the sum. A fast summation method reduces this complexity by introducing approximations.

The *fast multipole method* (FMM) [47, 48] works by expanding the kernel G in a *multipole* (far-field) expansion, which allows several source points to be aggregated into a single expansion. In fact, the expansions are very similar to the expansions used in QBX, and are also based on the Laplace expansion (4.3.1). However, in the multipole expansion, the source points \mathbf{y}_n are close to the expansion center and the target points \mathbf{x}_m are far away, so their roles are swapped compared to (4.3.1), and the multipole expansion of (5.1.1) with $G(\mathbf{r}) = H(\mathbf{r}) = 1/|\mathbf{r}|$ becomes

$$u(\mathbf{x}) = \sum_{l=0}^{\infty} \sum_{m=-l}^l \frac{1}{r_x^{l+1}} Y_l^m(\theta_x, \varphi_x) k_l^m[q], \quad (5.1.2)$$

with coefficients

$$k_l^m[q] = \frac{4\pi}{2l+1} \sum_{n=1}^N r_{y_n}^l Y_l^{-m}(\theta_{y_n}, \varphi_{y_n}) q(\mathbf{y}_n). \quad (5.1.3)$$

A hierarchical (octree) algorithm based only on multipole expansions will have complexity $\mathcal{O}(N \log N)$. The FMM also uses local expansions (of exactly the same type as the QBX expansions in Section 4.3) as well as operators to convert from multipole to local expansions, and to translate expansions. This allows the FMM to reduce the complexity further to $\mathcal{O}(N)$.

The original FMM is kernel-specific, and new expansions must be formulated for each new kernel G ; formulations for Stokes flow include [45, 123, 117]. More recently there have also appeared kernel-independent versions of the FMM [131, 130, 129, 125], although these can be expected to be slightly slower than a kernel-specific FMM, as shown in [131].

Another class of methods are based on *Ewald summation* [39], and typically achieve complexity $\mathcal{O}(N \log N)$ through the *fast Fourier transform* (FFT). For example, Particle–Mesh–Ewald (PME) methods [34, 38, 36, 103]

do this by introducing a uniform grid (“mesh”) in space, and interpolate between the source (and target) points and the uniform grid using a *window function*. Any PME method has the following five steps:

- (i) *Gridding*: Source data is interpolated from the (arbitrarily located) source points onto the uniform grid, using a convolution with the window function.
- (ii) *FFT*: A fast Fourier transform is applied to the data on the uniform grid, to go to Fourier space.
- (iii) *Scaling*: The data is multiplied by the Fourier transform of the kernel, which essentially corresponds to computing the original sum, here represented by (5.1.1), in Fourier space.¹
- (iv) *IFFT*: An inverse FFT is applied, to go back to real space.
- (v) *Gathering*: The result is interpolated back from the uniform grid to the desired target points, again using a convolution with the window function.

PME methods mainly differ in the choice of window function. The *Spectral Ewald* (SE) method [70, 72, 4] is a PME method with spectral accuracy, and will be described in more detail in Section 5.3. A related method is the Positively Split Ewald (PSE) method [42], used in Stokesian dynamics. For a more detailed discussion on different fast Ewald summation methods, see §I in Paper D and §1 in Paper E.

An important difference between the FMM and e.g. the SE method is that the FMM is spatially adaptive [29], allowing the hierarchical tree structure to be refined more in areas where the point density is higher. By contrast, the SE method is limited to a uniform spatial grid due to the use of the FFT. For this reason, the SE method is not optimal when the point distribution is highly nonuniform, as noted e.g. in [4] and Papers D–E. In particular, see §VII.D and Figure 16 in Paper D for a comparison.

¹In the scaling step, the data is also multiplied by other factors to undo the convolutions applied in the gridding and gathering steps, but here we give only a basic overview of the steps. Details can be found e.g. in §3 in Paper E.

5.2 Periodic problems

In a periodic problem, one or more of the spatial directions are periodic. Some representative examples are shown in Figure 5.1: (a) shows particles in an unbounded fluid, periodic in all three directions – we call this *triply periodic* (or *fully periodic*) and shorten it as $3\mathcal{P}$; (b) shows particles next to a flat wall, periodic in the two directions parallel to the wall – we call this *doubly periodic*, shortened as $2\mathcal{P}$; (c) shows particles in a pipe, periodic in the direction along the axis of the pipe – we call this *singly periodic*, or in short $1\mathcal{P}$. Finally, a problem which has no periodic directions is called *free-space* and shortened as $0\mathcal{P}$.

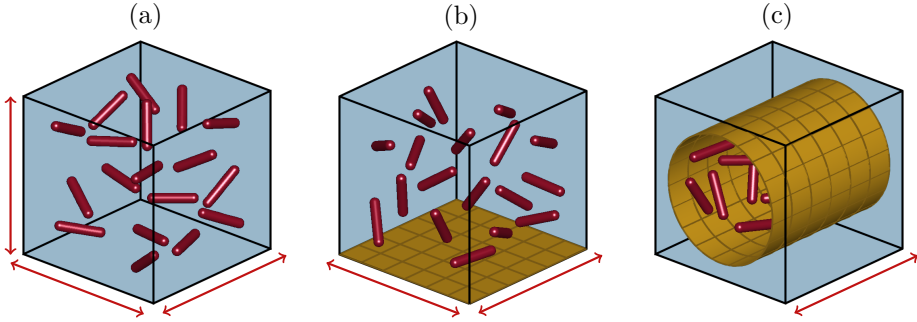


FIGURE 5.1: Examples of periodic problems: (a) particles in a fluid without walls, (b) particles next to a large flat wall, (c) particles in a long pipe. The arrows indicate periodic directions.

The box shown for each of (a), (b) and (c) in Figure 5.1 is called the *primary cell* and corresponds to the computation domain. The periodic boundary conditions enforced on certain faces of the primary cell can be interpreted as a periodic replication of the primary cell, as shown in Figure 5.2. The figure only shows a limited number of the replicas (also called periodic images); in reality, there are an infinite amount of replicas forming a lattice. Following Paper E, this lattice is defined by

$$P_{D\mathcal{P}} = \begin{cases} \{(\bar{p}_1 L_1, \bar{p}_2 L_2, \bar{p}_3 L_3) : \bar{p}_i \in \mathbb{Z}\}, & \text{if } D = 3, \\ \{(\bar{p}_1 L_1, \bar{p}_2 L_2, 0) : \bar{p}_i \in \mathbb{Z}\}, & \text{if } D = 2, \\ \{(\bar{p}_1 L_1, 0, 0) : \bar{p}_i \in \mathbb{Z}\}, & \text{if } D = 1, \\ \{(0, 0, 0)\}, & \text{if } D = 0, \end{cases} \quad (5.2.1)$$

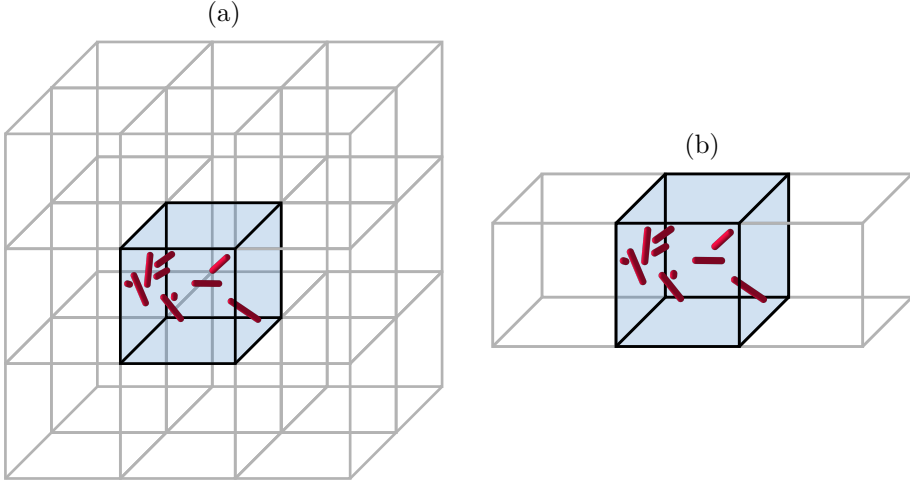


FIGURE 5.2: Illustration of periodic replication of the primary cell, in (a) the triply periodic ($3\mathcal{P}$) case, and (b) the singly periodic ($1\mathcal{P}$) case.

where D denotes the number of periodic directions, and (L_1, L_2, L_3) are the side lengths of the primary cell. Note that, without loss of generality, we have assumed in (5.2.1) that the first D directions are the periodic ones.

The periodic equivalent of the completed double layer formulation (3.2.13) introduced in Section 3.2 is found by periodically replicating the double layer potential $\mathcal{D}[\mathbf{q}](\mathbf{x})$ and completion term $\mathcal{V}(\mathbf{x})$, defined by (3.2.5) and (3.2.12), respectively, i.e.,

$$\mathcal{D}^{D\mathcal{P}}[\mathbf{q}](\mathbf{x}) = \sum_{\mathbf{p} \in P_{D\mathcal{P}}} \mathcal{D}[\mathbf{q}](\mathbf{x} + \mathbf{p}), \quad (5.2.2)$$

$$\mathcal{V}^{D\mathcal{P}}(\mathbf{x}) = \sum_{\mathbf{p} \in P_{D\mathcal{P}}} \mathcal{V}(\mathbf{x} + \mathbf{p}). \quad (5.2.3)$$

The resulting periodic formulation is

$$\mathbf{u}^{D\mathcal{P}}(\mathbf{x}) = \mathcal{D}^{D\mathcal{P}}[\mathbf{q}](\mathbf{x}) + \mathcal{V}^{D\mathcal{P}}(\mathbf{x}), \quad (5.2.4)$$

and the corresponding boundary integral equation becomes

$$\mathbf{u}^{D\mathcal{P}}(\mathbf{x}) = \mathcal{D}^{D\mathcal{P}}[\mathbf{q}](\mathbf{x}) - 4\pi\mathbf{q}(\mathbf{x}) + \mathcal{V}^{D\mathcal{P}}(\mathbf{x}), \quad \mathbf{x} \in \Gamma. \quad (5.2.5)$$

Note that (5.2.2) contains periodic sums of the stresslet T_{jlm} , and (5.2.3) contains periodic sums of the stokeslet S_{jl} and rotlet Ω_{jl} . Due to the slow

decay of these kernels, the periodic sums cannot be summed directly. An Ewald decomposition will split each of the periodic sums into a *real-space part* (that converges fast and can be summed directly) and a *Fourier-space part* (that converges fast in Fourier space). The SE method computes the latter of these. We return to this in Section 5.3.

It should be noted here that the SE method was first developed for $3\mathcal{P}$ problems [70, 72, 5, 1], but has since been extended to $2\mathcal{P}$ problems [71, 73], $1\mathcal{P}$ problems [102] and $0\mathcal{P}$ problems [4]. Papers D–E complete this development by unifying the SE framework for all periodicities and all kernels considered here (harmonic, stokeslet, stresslet, rotlet). The SE method is still most efficient in the $3\mathcal{P}$ case, and introducing free directions will lead to an increase in the runtime.

The FMM, on the other hand, is easiest to formulate and most efficient in the $0\mathcal{P}$ case. It has also been extended to periodic cases, typically based on Ewald-like sums to handle the periodic interactions. Prominent examples are [60, §3.5], where a periodic summation of multipole basis functions is used to compute the periodic far-field contributions in a kernel-specific FMM, and [130], which is based on similar ideas but in the kernel-independent context.

5.3 The Spectral Ewald method

For simplicity, we will here outline the SE method for a specific case, namely for the stokeslet sum in the $1\mathcal{P}$ case. The general case for the Stokes kernels is presented in Paper E.

In the following, we first write down the periodic sum that is to be computed, and then introduce the Ewald decomposition of it, which is the basis for any Ewald summation method. Then, we outline the specifics of the SE method itself, and finally summarize the contributions of Papers D–E.

The periodic sum and Ewald decomposition

The periodic sum of stokeslets, which appears e.g. in (5.2.3), in this case of periodicity in one direction has the general form

$$\begin{aligned} \mathbf{v}^{1\mathcal{P}}(\mathbf{x}) &= \sum_{\alpha} \sum_{\mathbf{p} \in P_{1\mathcal{P}}} \mathbf{S}(\mathbf{x} - \mathbf{x}^{(\alpha)} + \mathbf{p}) \mathbf{f}^{(\alpha)} \\ &= \sum_{\alpha} \sum_{\bar{p}_1 \in \mathbb{Z}} \mathbf{S}(\mathbf{x} - \mathbf{x}^{(\alpha)} + (\bar{p}_1 L_1, 0, 0)) \mathbf{f}^{(\alpha)}, \end{aligned} \quad (5.3.1)$$

where we have ignored the factor $1/8\pi$. This periodic sum is not absolutely convergent (unless the forces $\mathbf{f}^{(\alpha)}$ sum to zero), due to the slow decay of the stokeslet. To make sense of it, an Ewald decomposition of the stokeslet is made, which splits the sum into a real-space part and a Fourier-space part, as mentioned in Section 5.2. The decomposition used here for the stokeslet is due to Hasimoto [50]. One way to derive the decomposition is through a convolution with a *screening function* $\gamma(\mathbf{r}; \xi)$, which depends on a decomposition parameter ξ . The Hasimoto screening function is given by [55]

$$\gamma_H(\mathbf{r}; \xi) = \xi^3 \pi^{-3/2} e^{-\xi^2 |\mathbf{r}|^2} \left(\frac{5}{2} - \xi^2 |\mathbf{r}|^2 \right). \quad (5.3.2)$$

The derivations are simplified by using the relation between the stokeslet and the biharmonic kernel, i.e., (3.1.19). This relation can be represented as $S_{jl}(\mathbf{r}) = K_{jl} B(\mathbf{r})$, where the differential operator K_{jl} is given by

$$K_{jl} = \delta_{jl} \nabla^2 - \nabla_j \nabla_l. \quad (5.3.3)$$

The decomposition of the stokeslet is then given as

$$S_{jl}(\mathbf{r}) = S_{jl}^R(\mathbf{r}) + S_{jl}^F(\mathbf{r}), \quad (5.3.4)$$

where

$$S_{jl}^F = K_{jl} [B * \gamma_H] \quad (5.3.5)$$

$$S_{jl}^R = K_{jl} [B * (\delta - \gamma_H)], \quad (5.3.6)$$

where $\delta(\mathbf{r})$ is the Dirac delta distribution and $*$ denotes convolution. Inserting (5.3.4) into (5.3.1) leads to a decomposition of the periodic sum itself.

An explicit expression for $S_{jl}^R(\mathbf{r}; \xi)$ can be written down; see equation (29) in Paper E. This function has a very fast decay as $|\mathbf{r}| \rightarrow \infty$, and therefore the real-space part of (5.3.1) can be summed directly. The Fourier-space part, which is

$$\mathbf{v}^{1\mathcal{P},F}(\mathbf{x}; \xi) = \sum_{\alpha} \sum_{\bar{p}_1 \in \mathbb{Z}} \mathbf{S}^F(\mathbf{x} - \mathbf{x}^{(\alpha)} + (\bar{p}_1 L_1, 0, 0); \xi) \mathbf{f}^{(\alpha)}, \quad (5.3.7)$$

is converted into an expression in Fourier space using the Poisson summation formula; it becomes

$$\mathbf{v}^{1\mathcal{P},F}(\mathbf{x}; \xi) = \frac{1}{L_1 (2\pi)^2} \sum_{\alpha} \sum_{\mathbf{k}_1 \in \mathcal{K}^1} \int_{\mathbb{R}^2} \widehat{\mathbf{S}}^F(\mathbf{k}; \xi) \mathbf{f}^{(\alpha)} e^{i\mathbf{k} \cdot (\mathbf{x} - \mathbf{x}^{(\alpha)})} d\kappa_2 d\kappa_3, \quad (5.3.8)$$

where $\mathbf{k} = (k_1, \kappa_2, \kappa_3)$, k_1 is a discrete wavenumber belonging to the set $\mathcal{K}^1 = (2\pi/L_1)\mathbb{Z}$, and κ_2, κ_3 are continuous wavenumbers belonging to \mathbb{R} . The analytic expression for the Fourier transform $\widehat{\mathbf{S}}^{\mathbf{F}}(\mathbf{k}; \xi)$ of $\mathbf{S}^{\mathbf{F}}$ is available; see equation (35) in Paper E. The presence of the screening function makes $\widehat{\mathbf{S}}^{\mathbf{F}}$ decay fast as $|\mathbf{k}| \rightarrow \infty$, which makes the Fourier-space sum (5.3.8) convergent. It could be summed directly, but this would require $\mathcal{O}(N^2)$ operations. In the SE method, FFTs are used to reduce the amount of work to $\mathcal{O}(N \log N)$.

It should be noted that for the $k_1 = 0$ term, the integral in (5.3.8) involves a singular integrand, and this case is handled by introducing a modified stokeslet kernel (§3.1 in Paper E). For small but nonzero values of k_1 , the integrand is nearly singular and peaked, which is handled by adaptive upsampling in an adaptive FFT (AFT), see §3.4 in Paper E.

Spectral Ewald: The fast method for the Fourier-space sum

The five steps of the fast method, (i)–(v), were introduced in Section 5.1. The window function is a crucial part of the method, and it is introduced into (5.3.8) by insertion of the identity $\widehat{w}(\widehat{w})^{-2}\widehat{w} = 1$, where $\widehat{w}(\mathbf{k})$ is the Fourier transform of the window function $w(\mathbf{r})$. The first factor \widehat{w} in the identity is used in the gridding step, the factor $(\widehat{w})^{-2}$ is used in the scaling step, and the final \widehat{w} is used in the gathering step. For details, see equations (71)–(75) and also equations (136)–(140) in Paper E.

The window function in the SE method is written as a tensor product $w(\mathbf{r}) = w_0(r_1)w_0(r_2)w_0(r_3)$ of a one-dimensional function $w_0 : \mathbb{R} \rightarrow \mathbb{R}$. In the original formulations of the SE method, e.g. [70, 4], the window function was a truncated Gaussian. One of the novelties of Papers D–E is the use of a new window function, namely a piecewise polynomial approximation of the Kaiser–Bessel window

$$w_{0,\text{KB}}(r) = \begin{cases} \frac{I_0(\beta\sqrt{1 - (r/a_w)^2})}{I_0(\beta)}, & |r| \leq a_w, \\ 0, & |r| > a_w, \end{cases} \quad (5.3.9)$$

where $I_0(\cdot)$ is the modified Bessel function of the first kind and order 0, and $\beta > 0$ is a shape parameter. Here, a_w is the halfwidth of the support of the window function, as shown in Figure 5.3.

The window function is discretized using P points and related to the spacing h of the uniform grid via $Ph = 2a_w$. The shape parameter β can be

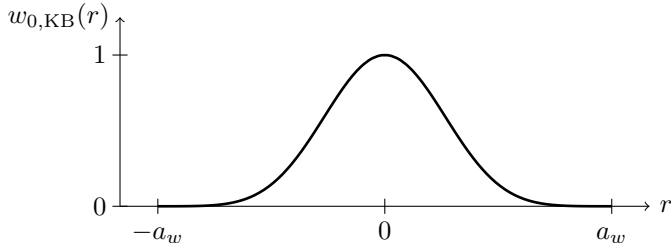


FIGURE 5.3: Plot of the Kaiser–Bessel window function given by (5.3.9).

related to P by balancing different error contributions (§V.B.2 in Paper D). The piecewise polynomial approximation of the window function $w_{0,KB}$ is chosen such that the accuracy associated with the exact Kaiser–Bessel window is retained (§IV.E and §VI in Paper D). It is the smoothness of the window function, and the fact that its support size P can be adjusted separately from the grid spacing h (since a_w is not fixed), that is responsible for the spectral accuracy of the SE method.

Paper contributions

Papers D–E both present a unified framework for the SE method in arbitrary periodicity, for different kernels; Paper D deals with electrostatics (the harmonic kernel), while Paper E deals with Stokes flow (the stokeslet, stresslet and rotlet). The contributions of these papers are:

- In **Paper D**, DSS and AKT introduced modified kernels (based on Vico et al. [119]) to treat the singular integration in problems with free directions; they also introduced an adaptive FFT (AFT) in the $1\mathcal{P}$ and $2\mathcal{P}$ cases, reducing the runtime substantially. The runtime of the algorithm was tested in different periodicities. In the first version of the paper, a new window function based on [15] was tested.

JB and AKT replaced the new window function by a piecewise polynomial approximation of the Kaiser–Bessel window function (PKB window), which has a more optimal bandwidth, and reduces the support and therefore runtime significantly compared to the old Gaussian window. A parameter selection procedure was introduced and tested, two different methods for force computation were compared, and the runtime of the SE method was compared with that of the FMM.

- **Paper E** includes many of the features introduced in Paper D, such as the PKB window and the AFT. New modified kernels are derived, together with new improved truncation error estimates; this improves the performance in the $0\mathcal{P}$ case compared to [4], and makes the truncation error independent of periodicity. Analytical formulas for validation are derived, and the parameter selection procedure from Paper D is adapted. In the numerical results, we show that the method has complexity $\mathcal{O}(N \log N)$, that the pointwise error is well behaved, and we study how the runtime depends on the error tolerance and window function.

6 Conclusions and outlook

In this thesis, we have studied and developed numerical methods for accurate special quadrature and fast summation, that can be applied in the context of boundary integral methods for Stokes flow. These methods can be used to perform fast and accurate computer simulations of suspensions of rigid particles in three-dimensional Stokes flow.

In Papers A–B, the special quadrature method known as quadrature by expansion (QBX), previously implemented for rigid spheroids in Stokes flow in [6], was extended to rodlike particles, flat walls and cylindrical pipes. The precomputation scheme which was adapted from [6] assumes that the geometrical objects are rigid and that symmetries can be used to reduce the amount of precomputations needed. A strategy for selecting the parameters of the method, such that a given error tolerance is met, was designed and systematically tested.

During the development of Paper B, it was discovered that if the rodlike particle geometry were to be made up by a cylinder with spherical caps, the local expansions in QBX would converge very slowly, since the geometry would have a discontinuous second derivative. We chose to avoid this by re-designing the caps to be smoothly connected to the cylinder. An alternative to this could likely be to dyadically refine the grid close to the discontinuity as in [67], possibly using recursively compressed inverse preconditioning (RCIP) [51, 52] to avoid introducing more degrees of freedom.

It was also found that the completion sources (point forces and torques) that are placed inside the particles in the boundary integral formulation must be spread out along the extent of the particle, for example along its centerline, to maintain high accuracy. Otherwise, the solution becomes hard to resolve on the grid. Similar findings were made in [61] for spheroids and

helical particles.

In Paper C, the QBX method from [6] was applied to simulate a problem in fundamental fluid mechanics, namely a single inertial spheroid moving through Stokes flow. It was found that a prolate spheroid drifts sideways in a parabolic flow, while an oblate spheroid does not. The magnitude of the sideways drift was compared with the settling due to gravity. This work was carried out partially in parallel with the work on Papers A–B.

The goal of Papers D–E was to extend the Spectral Ewald (SE) method to arbitrary periodicity for electrostatic problems and Stokes flow, so that efficient simulations can be performed with periodic boundary conditions in any number of the spatial directions. We also implemented various improvements to the SE method, such as a more optimal window function, new modified kernels for singular integration, and parameter selection based on new error estimates.

Our implementation of QBX relies on the precomputation scheme from [6], which is feasible for rigid particles with axisymmetry. For deformable geometries, or geometries without symmetries, this precomputation would not be possible. An alternative would be to build QBX into a fast multipole method (FMM) framework, such as in e.g. [121, 122]. Another option would be to use a different special quadrature method, such as the Hedgehog method studied in Paper F. This method is based on extrapolation or interpolation along a line, and in the paper, we studied rational approximation as an alternative to polynomial approximation when doing this extrapolation/interpolation. We investigated how much the error is reduced when using rational approximation compared with polynomial approximation, and also how much the error is reduced when using interpolation instead of extrapolation. The accuracy and computational cost of the Hedgehog method were compared with those of QBX. Strategies for selecting the parameters of the Hedgehog method and estimating its error were also discussed, although there is certainly room for further improvement and development in this regard.

During the work with Paper F, it was discovered that spurious poles may end up close to the evaluation point when doing rational approximation. These poles disturb the result, increase the error, and make it very hard to predict the error level of the method in a robust way. Therefore, a few methods to deal with spurious poles, and make the results more robust, are discussed in Paper F.

Possible future directions

There are some natural improvements and challenges that could be considered for our current implementation of QBX for rigid rodlike particles and walls:

- The particles could be discretized using patches, in the same way that the walls are, which would allow for a more local application of QBX (or Hedgehog, for that matter), with smaller memory requirements and lower computation times. Work in this direction is currently ongoing within the research group. This would make it possible to do simulations using much longer particles (high aspect ratios), which is of interest in applications such as [83].
- The discretization of walls (and particles) is currently not adaptive. Implementing adaptive discretization would increase the flexibility and allow for local grid refinement.
- It may be possible to develop a fast direct solver for the fixed geometry (i.e., walls), that could be used e.g. in time-dependent simulations to speed up the solve.
- In close interactions, the peaked stresslet kernel in e.g. the double layer potential is only half of the problem, since the layer density (i.e., the quantity being solved for) may itself also become peaked and hard to resolve. Special quadrature only addresses the first issue (peaked kernel), but not the second (peaked density). An adaptive discretization would of course help in this situation, but one could also seek other strategies to avoid increasing the grid resolution.

Another interesting development would be to include drops and bubbles, which are common in microfluidics applications, into the framework, which would allow simulations involving both drops, rigid particles and walls interacting. The formulation for drops could be based on [110], which studied surfactant-covered drops using a boundary integral method and the Hedgehog method (but without walls or rigid particles). Similar simulations have been performed in two dimensions, e.g. in [90].

For the SE method, there is also some room for improvement. Certain aspects of the parameter selection could be improved by further analysis of the errors in the method. The software package published together with

Paper E could also be further optimized, and e.g. implemented purely in C instead of relying on Matlab.

Of course, it would be of interest to apply the new unified SE implementation to particulate Stokes flow with e.g. one or two periodic directions – such as rigid particles in a pipe or close to a flat wall – which was the motivation behind developing the method in the first place. Combining the SE method with the Hedgehog method is then a natural next step. The Hedgehog method has the advantage that it can be applied directly to the real-space part of the Ewald sum, which our current implementation of QBX cannot. Work in this direction could enable fast and accurate three-dimensional simulations of drops, particles and walls in arbitrary periodicity.

Finally, work on Paper F will continue with the goal to publish it in a journal. The parameter selection strategy for the Hedgehog method will then be further developed, possibly employing estimates of the direct quadrature error to select the parameters. The method will be tested on more examples, and we will continue to investigate the possibility to replace QBX in the nearly singular (off-surface) case with the Hedgehog method, in the framework developed in Paper B.

References

- [1] L. af Klinteberg. Ewald summation for the rotlet singularity of Stokes flow. *Preprint*, arXiv:1603.07467, 2016.
- [2] L. af Klinteberg. *Fast and accurate integral equation methods with applications in microfluidics*. PhD thesis, KTH Royal Institute of Technology, Stockholm, 2016. <https://kth.diva-portal.org/smash/record.jsf?pid=diva2:923458>.
- [3] L. af Klinteberg, T. Askham, and M. C. Kropinski. A fast integral equation method for the two-dimensional Navier-Stokes equations. *J. Comput. Phys.*, 409:109353, 2020. DOI:10.1016/j.jcp.2020.109353.
- [4] L. af Klinteberg, D. Saffar Shamshirgar, and A.-K. Tornberg. Fast Ewald summation for free-space Stokes potentials. *Res. Math. Sci.*, 4:1, 2017. DOI:10.1186/s40687-016-0092-7.
- [5] L. af Klinteberg and A.-K. Tornberg. Fast Ewald summation for Stokesian particle suspensions. *Int. J. Numer. Methods Fluids*, 76(10):669–698, 2014. DOI:10.1002/fld.3953.
- [6] L. af Klinteberg and A.-K. Tornberg. A fast integral equation method for solid particles in viscous flow using quadrature by expansion. *J. Comput. Phys.*, 326:420–445, 2016. DOI:10.1016/j.jcp.2016.09.006.
- [7] J. C. Aguilar and Y. Chen. High-order corrected trapezoidal quadrature rules for the coulomb potential in three dimensions. *Computers Math. Applic.*, 49(4):625–631, 2005. DOI:10.1016/j.camwa.2004.01.018.
- [8] C. K. Aidun and J. R. Clausen. Lattice-Boltzmann method for complex flows. *Annu. Rev. Fluid Mech.*, 42:439–472, 2010. DOI:10.1146/annurev-fluid-121108-145519.
- [9] S. Akgönüllü, M. Bakhshpour, A. K. Pişkin, and A. Denizli. Microfluidic systems for cancer diagnosis and applications. *Micromachines*, 12:1349, 2021. DOI:10.3390/mi12111349.

- [10] K. E. Atkinson. *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press, Cambridge, 1997. DOI:10.1017/CBO9780511626340.
- [11] J. Bagge. Numerical simulation of an inertial spheroidal particle in Stokes flow. Master’s thesis, KTH Royal Institute of Technology, Stockholm, 2016. <https://kth.diva-portal.org/smash/record.jsf?pid=diva2:892288>.
- [12] F. Balboa Usabiaga, B. Kallemov, B. Delmotte, A. P. S. Bhalla, B. E. Griffith, and A. Donev. Hydrodynamics of suspensions of passive and active rigid particles: A rigid multiblob approach. *Comm. App. Math. Comp. Sci.*, 11(2):217–296, 2016. DOI:10.2140/camcos.2016.11.217.
- [13] A. Barnett, B. Wu, and S. Veerapaneni. Spectrally accurate quadratures for evaluation of layer potentials close to the boundary for the 2D Stokes and Laplace equations. *SIAM J. Sci. Comput.*, 37(4):B519–B542, 2015. DOI:10.1137/140990826.
- [14] A. H. Barnett. Evaluation of layer potentials close to the boundary for Laplace and Helmholtz problems on analytic planar domains. *SIAM J. Sci. Comput.*, 36(2):A427–A451, 2014. DOI:10.1137/120900253.
- [15] A. H. Barnett, J. Magland, and L. af Klinteberg. A parallel nonuniform fast Fourier transform library based on an “exponential of semicircle” kernel. *SIAM J. Sci. Comput.*, 41(5):C479–C504, 2019. DOI:10.1137/18M120885X.
- [16] J. T. Beale and M.-C. Lai. A method for computing nearly singular integrals. *SIAM J. Numer. Anal.*, 38(6):1902–1925, 2001. DOI:10.1137/S0036142999362845.
- [17] J. T. Beale, W. Ying, and J. R. Wilson. A simple method for computing singular or nearly singular integrals on closed surfaces. *Commun. Comput. Phys.*, 20(3):733–753, 2016. DOI:10.4208/cicp.030815.240216a.
- [18] C. W. J. Beenakker. Ewald sum of the Rotne–Prager tensor. *J. Chem. Phys.*, 85(3):1581–1582, 1986. DOI:10.1063/1.451199.
- [19] G. Bognár and A. Vencl. Experimental investigation of viscosity of glycerol based nanofluids containing carbon nanotubes. *Tribol. Ind.*, 41(2):267–273, 2019. DOI:10.24874/ti.2019.41.02.12.
- [20] J. F. Brady and G. Bossis. Stokesian dynamics. *Ann. Rev. Fluid Mech.*, 20:111–157, 1988. DOI:10.1146/annurev.fl.20.010188.000551.
- [21] H. Brenner and R. G. Cox. The resistance to a particle of arbitrary shape in translational motion at small Reynolds numbers. *J. Fluid Mech.*, 17(4):561–595, 1963. DOI:10.1017/S002211206300152X.

- [22] W.-P. Breugem. A second-order accurate immersed boundary method for fully resolved simulations of particle-laden flows. *J. Comput. Phys.*, 231:4469–4498, 2012. DOI:10.1016/j.jcp.2012.02.026.
- [23] A. Broms, M. Sandberg, and A.-K. Tornberg. A locally corrected multiblob method with hydrodynamically matched grids for the Stokes mobility problem. *J. Comput. Phys.*, 487:112172, 2023. DOI:10.1016/j.jcp.2023.112172.
- [24] O. P. Bruno and L. A. Kunyansky. A fast, high-order algorithm for the solution of surface scattering problems: basic implementation, tests, and applications. *J. Comput. Phys.*, 169(1):80–110, 2001. DOI:10.1006/jcph.2001.6714.
- [25] E. Burman, S. Claus, P. Hansbo, M. G. Larson, and A. Massing. CutFEM: Discretizing geometry and partial differential equations. *Int. J. Numer. Methods Eng.*, 104:472–501, 2015. DOI:10.1002/nme.4823.
- [26] S. L. Campbell, I. C. F. Ipsen, C. T. Kelley, C. D. Meyer, and Z. Q. Xue. Convergence estimates for solution of integral equations with GMRES. *J. Integral Equ. Appl.*, 8(1):19–34, 1996. <http://www.jstor.org/stable/26163188>.
- [27] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. *Spectral Methods*. Springer, Berlin Heidelberg, 2007. DOI:10.1007/978-3-540-30728-0.
- [28] S. Chen and G. D. Doolen. Lattice Boltzmann method for fluid flows. *Annu. Rev. Fluid Mech.*, 30(1):329–364, 1998. DOI:10.1146/annurev.fluid.30.1.329.
- [29] H. Cheng, L. Greengard, and V. Rokhlin. A fast adaptive multipole algorithm in three dimensions. *J. Comput. Phys.*, 155(2):468–498, 1999. DOI:10.1006/jcph.1999.6355.
- [30] C. D. Chin, T. Laksanasopin, Y. K. Cheung, D. Steinmiller, V. Linder, H. Parsa, J. Wang, H. Moore, R. Rouse, G. Umvilighozo, E. Karita, L. Mwambarangwe, S. L. Braunstein, J. van de Wiggert, R. Sahabo, J. E. Justman, W. El-Sadr, and S. K. Sia. Microfluidics-based diagnostics of infectious diseases in the developing world. *Nat. Med.*, 17(8):1015–1019, 2011. DOI:10.1038/nm.2408.
- [31] I. L. A. M. Claeys. *Hydrodynamic transport properties of suspensions of non-Brownian prolate spheroids*. PhD thesis, California Institute of Technology, Pasadena, 1991. <https://resolver.caltech.edu/CaltechETD:etd-01282005-165455>.
- [32] E. Corona and S. Veerapaneni. Boundary integral equation analysis for suspension of spheres in Stokes flow. *J. Comput. Phys.*, 362:327–345, 2018. DOI:10.1016/j.jcp.2018.02.017.

- [33] S. Court and M. Fournié. A fictitious domain finite element method for simulations of fluid–structure interactions: The Navier–Stokes equations coupled with a moving solid. *J. Fluids Struct.*, 55:398–408, 2015. DOI:10.1016/j.jfluidstructs.2015.03.013.
- [34] T. Darden, D. York, and L. Pedersen. Particle mesh Ewald: an $N \cdot \log(N)$ method for Ewald sums in large systems. *J. Chem. Phys.*, 98:10089, 1993. DOI:10.1063/1.464397.
- [35] S. Delong, F. Balboa Usabiaga, and A. Donev. Brownian dynamics of confined rigid bodies. *J. Chem. Phys.*, 143:144107, 2015. DOI:10.1063/1.4932062.
- [36] M. Deserno and C. Holm. How to mesh up Ewald sums. I. A theoretical and numerical comparison of various particle mesh routines. *J. Chem. Phys.*, 109(18):7678–7693, 1998. DOI:10.1063/1.477414.
- [37] C. L. Epstein, L. Greengard, and A. Klöckner. On the convergence of local expansions of layer potentials. *SIAM J. Numer. Anal.*, 51(5):2660–2679, 2013. DOI:10.1137/120902859.
- [38] U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee, and L. G. Pedersen. A smooth particle mesh Ewald method. *J. Chem. Phys.*, 103(19):8577–8593, 1995. DOI:10.1063/1.470117.
- [39] P. P. Ewald. Die Berechnung optischer und elektrostatischer Gitterpotentiale. *Ann. Phys.*, 369(3):253–287, 1921. DOI:10.1002/andp.19213690304.
- [40] X.-J. Fan, N. Phan-Thien, and R. Zheng. Completed double layer boundary element method for periodic suspensions. *Z. angew. Math. Phys.*, 49(2):167–193, 1998. DOI:10.1007/s000330050214.
- [41] C. Faustino and L. Pinheiro. Analytical rheology of honey: a state-of-the-art review. *Foods*, 10(8):1709, 2021. DOI:10.3390/foods10081709.
- [42] A. M. Fiore, F. Balboa Usabiaga, A. Donev, and J. W. Swan. Rapid sampling of stochastic displacements in Brownian dynamics simulations. *J. Chem. Phys.*, 146:124116, 2017. DOI:10.1063/1.4978242.
- [43] T. Frachon. *Cut finite element methods for interface problems*. PhD thesis, KTH Royal Institute of Technology, Stockholm, 2022. <https://kth.diva-portal.org/smash/record.jsf?pid=diva2:1658282>.
- [44] T.-P. Fries and T. Belytschko. The extended/generalized finite element method: An overview of the method and its applications. *Int. J. Numer. Methods Eng.*, 84:253–304, 2010. DOI:10.1002/nme.2914.

- [45] Y. Fu and G. J. Rodin. Fast solution method for three-dimensional Stokesian many-particle problems. *Commun. Numer. Methods Eng.*, 16(2):145–149, 2000. DOI:10.1002/(SICI)1099-0887(200002)16:2<145::AID-CN323>3.0.CO;2-E.
- [46] R. Glowinski, T.-W. Pan, T. I. Hesla, and D. D. Joseph. A distributed Lagrange multiplier/fictitious domain method for particulate flows. *Int. J. Multiphase Flow*, 25:755–794, 1999. DOI:10.1016/S0301-9322(98)00048-2.
- [47] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *J. Comput. Phys.*, 73:325–348, 1987. DOI:10.1016/0021-9991(87)90140-9.
- [48] L. Greengard and V. Rokhlin. A new version of the Fast Multipole Method for the Laplace equation in three dimensions. *Acta Numer.*, 6:229–269, 1997. DOI:10.1017/S0962492900002725.
- [49] J. S. Guasto, R. Rusconi, and R. Stocker. Fluid Mechanics of Planktonic Microorganisms. *Annu. Rev. Fluid Mech.*, 44:373–400, 2012. DOI:10.1146/annurev-fluid-120710-101156.
- [50] H. Hasimoto. On the periodic fundamental solutions of the Stokes equations and their application to viscous flow past a cubic array of spheres. *J. Fluid Mech.*, 5(2):317–328, 1959. DOI:10.1017/S0022112059000222.
- [51] J. Helsing. Integral equation methods for elliptic problems with boundary conditions of mixed type. *J. Comput. Phys.*, 228:8892–8907, 2009. DOI:10.1016/j.jcp.2009.09.004.
- [52] J. Helsing. Solving integral equations on piecewise smooth boundaries using the RCIP method: a tutorial. *Abstr. Appl. Anal.*, 2013:938167, 2013. DOI:10.1155/2013/938167.
- [53] J. Helsing and R. Ojala. On the evaluation of layer potentials close to their sources. *J. Comput. Phys.*, 227(5):2899–2921, 2008. DOI:10.1016/j.jcp.2007.11.024.
- [54] L. Heltai, M. Arroyo, and A. DeSimone. Nonsingular isogeometric boundary element method for Stokes flows in 3D. *Comput. Methods Appl. Mech. Engrg.*, 268:514–539, 2014. DOI:10.1016/j.cma.2013.09.017.
- [55] J. P. Hernández-Ortiz, J. J. de Pablo, and M. D. Graham. Fast computation of many-particle hydrodynamic and electrostatic interactions in a confined geometry. *Phys. Rev. Lett.*, 98(14):140602, 2007. DOI:10.1103/PhysRevLett.98.140602.
- [56] H. H. Hu, N. A. Patankar, and M. Y. Zhu. Direct numerical simulations of fluid–solid systems using the arbitrary Lagrangian–Eulerian technique. *J. Comput. Phys.*, 169:427–462, 2001. DOI:10.1006/jcph.2000.6592.

- [57] H. Huang, X. Yang, and X.-y. Lu. Sedimentation of an ellipsoidal particle in narrow tubes. *Phys. Fluids*, 26:053302, 2014. DOI:10.1063/1.4874606.
- [58] F. Izzo. *High order trapezoidal rule-based quadratures for boundary integral methods on non-parametrized surfaces*. PhD thesis, KTH Royal Institute of Technology, Stockholm, 2022. <https://kth.diva-portal.org/smash/record.jsf?pid=diva2:1701791>.
- [59] M. R. Jamiruddin, B. A. Meghla, D. Z. Islam, T. A. Tisha, S. S. Khandker, M. U. Khondoker, M. A. Haq, N. Adnan, and M. Haque. Microfluidics technology in SARS-CoV-2 diagnosis and beyond: a systematic review. *Life*, 12(5):649, 2022. DOI:10.3390/life12050649.
- [60] I. Kabadshow. *Periodic Boundary Conditions and the Error-Controlled Fast Multipole Method*. PhD thesis, Forschungszentrum Jülich/Universität Wuppertal, Jülich, 2012. <http://hdl.handle.net/2128/4571>.
- [61] E. E. Keaveny and M. J. Shelley. Applying a second-kind boundary integral equation for surface tractions in Stokes flow. *J. Comput. Phys.*, 230(5):2141–2159, 2011. DOI:10.1016/j.jcp.2010.12.010.
- [62] C. E. Kees, J. H. Collins, and A. Zhang. Simple, accurate, and efficient embedded finite element methods for fluid–solid interaction. *Comput. Methods Appl. Mech. Engrg.*, 389:114404, 2022. DOI:10.1016/j.cma.2021.114404.
- [63] T. Kempe and J. Fröhlich. An improved immersed boundary method with direct forcing for the simulation of particle laden flows. *J. Comput. Phys.*, 231:3663–3684, 2012. DOI:10.1016/j.jcp.2012.01.021.
- [64] S. Khatri, A. D. Kim, R. Cortez, and C. Carvalho. Close evaluation of layer potentials in three dimensions. *J. Comput. Phys.*, 423:109798, 2020. DOI:10.1016/j.jcp.2020.109798.
- [65] S. Kim and S. J. Karrila. *Microhydrodynamics : Principles and Selected Applications*. Butterworth–Heinemann, Boston, 1991. DOI:10.1016/C2013-0-04644-0.
- [66] Y. Kim and C. S. Peskin. A penalty immersed boundary method for a rigid body in fluid. *Phys. Fluids*, 28:033603, 2016. DOI:10.1063/1.4944565.
- [67] A. Klöckner, A. Barnett, L. Greengard, and M. O’Neil. Quadrature by expansion: a new method for the evaluation of layer potentials. *J. Comput. Phys.*, 252:332–349, 2013. DOI:10.1016/j.jcp.2013.06.027.
- [68] R. Kress. *Linear Integral Equations*. Springer, New York, 3rd edition, 2014. DOI:10.1017/CBO9780511605345.

- [69] G. Legendre and T. Takahashi. Convergence of a Lagrange–Galerkin method for a fluid–rigid body system in ALE formulation. *ESAIM: M2AN*, 42:609–644, 2008. DOI:10.1051/m2an:2008020.
- [70] D. Lindbo and A.-K. Tornberg. Spectrally accurate fast summation for periodic Stokes potentials. *J. Comput. Phys.*, 229:8994–9010, 2010. DOI:10.1016/j.jcp.2010.08.026.
- [71] D. Lindbo and A.-K. Tornberg. Fast and spectrally accurate summation of 2-periodic Stokes potentials. *Preprint*, arXiv:1111.1815, 2011.
- [72] D. Lindbo and A.-K. Tornberg. Spectral accuracy in fast Ewald-based methods for particle simulations. *J. Comput. Phys.*, 230(24):8744–8761, 2011. DOI:10.1016/j.jcp.2011.08.022.
- [73] D. Lindbo and A.-K. Tornberg. Fast and spectrally accurate Ewald summation for 2-periodic electrostatic systems. *J. Chem. Phys.*, 136(16):164111, 2012. DOI:10.1063/1.4704177.
- [74] D. Liu, E. E. Keaveny, M. R. Maxey, and G. E. Karniadakis. Force-coupling method for flows with ellipsoidal particles. *J. Comput. Phys.*, 228:3559–3581, 2009. DOI:10.1016/j.jcp.2009.01.020.
- [75] Y. Liu. *Fast Multipole Boundary Element Method*. Cambridge University Press, Cambridge, 2009. DOI:10.1017/CBO9780511605345.
- [76] L. Lu, M. J. Morse, A. Rahimian, G. Stadler, and D. Zorin. Scalable simulation of realistic volume fraction red blood cell flows through vascular networks. page 6, Denver, CO, USA, November 2019. Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC '19). DOI:10.1145/3295500.3356203.
- [77] D. Malhotra, A. Rahimian, D. Zorin, and G. Biros. A parallel algorithm for long-timescale simulation of concentrated vesicle suspensions in three dimensions. *Preprint*, <https://users.flatironinstitute.org/~dmalhotra/files/pubs/ves3d.pdf>, 2018.
- [78] O. Marin, O. Runborg, and A.-K. Tornberg. Corrected trapezoidal rules for a class of singular functions. *IMA J. Numer. Anal.*, 34(4):1509–1540, 2014. DOI:10.1093/imanum/drt046.
- [79] P.-G. Martinsson. *Fast Direct Solvers for Elliptic PDEs*. SIAM, Philadelphia, 2019. DOI:10.1137/1.9781611976045.
- [80] M. R. Maxey and B. K. Patel. Localized force representations for particles sedimenting in Stokes flow. *Int. J. Multiphase Flow*, 27:1603–1626, 2001. DOI:10.1016/S0301-9322(01)00014-3.

- [81] J. M. Melenk. On condition numbers in hp-FEM with Gauss–Lobatto-based shape functions. *J. Comput. Appl. Math.*, 139(1):21–48, 2002. DOI:10.1016/S0377-0427(01)00391-0.
- [82] V. Minden, K. L. Ho, A. Damle, and L. Ying. A recursive skeletonization factorization based on strong admissibility. *Multiscale Model. Simul.*, 15(2):768–796, 2017. DOI:10.1137/16M1095949.
- [83] N. Mittal, F. Ansari, K. Gowda.V, C. Brouzet, P. Chen, P. T. Larsson, S. V. Roth, F. Lundell, L. Wågberg, N. A. Kotov, and L. D. Söderberg. Multi-scale control of nanocellulose assembly: transferring remarkable nanoscale fibril mechanics to macroscale fibers. *ACS Nano*, 12(7):6378–6388, 2018. DOI:10.1021/acsnano.8b01084.
- [84] R. Mittal and G. Iaccarino. Immersed boundary methods. *Annu. Rev. Fluid Mech.*, 37:239–261, 2005. DOI:10.1146/annurev.fluid.37.061903.175743.
- [85] N. Moës, J. Dolbow, and T. Belytschko. A finite element method for crack growth without remeshing. *Int. J. Numer. Methods Eng.*, 46(1):131–150, 1999. DOI:10.1002/(SICI)1097-0207(19990910)46:1<131::AID-NME726>3.0.CO;2-J.
- [86] M. J. Morse, A. Rahimian, and D. Zorin. A robust solver for elliptic PDEs in 3D complex geometries. *J. Comput. Phys.*, 442:110511, 2021. DOI:10.1016/j.jcp.2021.110511.
- [87] Y. Nakatsukasa, O. Sète, and L. N. Trefethen. The AAA algorithm for rational approximation. *SIAM J. Sci. Comput.*, 40(3):A1494–A1522, 2018. DOI:10.1137/16M1106122.
- [88] J. Nitsche. Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 36:9–15, 1971. DOI:10.1007/BF02995904.
- [89] R. Ojala and A.-K. Tornberg. An accurate integral equation method for simulating multi-phase Stokes flow. *J. Comput. Phys.*, 298:145–160, 2015. DOI:10.1016/j.jcp.2015.06.002.
- [90] S. Pålsson. *Boundary integral methods for fast and accurate simulation of droplets in two-dimensional Stokes flow*. PhD thesis, KTH Royal Institute of Technology, Stockholm, 2019. <https://kth.diva-portal.org/smash/record.jsf?pid=diva2:1373113>.
- [91] C. S. Peskin. Numerical analysis of blood flow in the heart. *J. Comput. Phys.*, 25:220–252, 1977. DOI:10.1016/0021-9991(77)90100-0.

- [92] C. S. Peskin. The immersed boundary method. *Acta Numerica*, 11:479–517, 2002. DOI:10.1017/S0962492902000077.
- [93] H. Power and G. Miranda. Second kind integral equation formulation of Stokes’ flows past a particle of arbitrary shape. *SIAM J. Appl. Math.*, 47(4):689–698, 1987. DOI:10.1137/0147047.
- [94] C. Pozrikidis. *Boundary integral and singularity methods for linearized viscous flow*. Cambridge University Press, Cambridge, 1992.
- [95] C. Pozrikidis. *A practical guide to boundary element methods with the software library BEMLIB*. Chapman & Hall/CRC, Boca Raton, 2002. DOI:10.1201/9781420035254.
- [96] I. Proudman and J. R. A. Pearson. Expansions at small Reynolds numbers for the flow past a sphere and a circular cylinder. *J. Fluid Mech.*, 2(3):237–262, 1957. DOI:10.1017/S0022112057000105.
- [97] D. Qi and L.-S. Luo. Rotational and orientational behaviour of three-dimensional spheroidal particles in Couette flows. *J. Fluid Mech.*, 477:201–213, 2003. DOI:10.1017/S0022112002003191.
- [98] M. Rahbar, S. Zou, M. Baharfar, and G. Liu. A customized microfluidic paper-based platform for colorimetric immunosensing: demonstrated via hCG assay for pregnancy test. *Biosensors*, 11:474, 2021. DOI:10.3390/bios11120474.
- [99] Y. Saad and M. H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7(3):856–869, 1986. DOI:10.1137/0907058.
- [100] S. Sachdeva, R. W. Davis, and A. K. Saha. Microfluidic point-of-care testing: commercial landscape and future directions. *Front. Bioeng. Biotechnol.*, 8:602659, 2021. DOI:10.3389/fbioe.2020.602659.
- [101] E. K. Sackmann, A. L. Fulton, and D. J. Beebe. The present and future role of microfluidics in biomedical research. *Nature*, 507:181–189, 2014. DOI:10.1038/nature13118.
- [102] D. Saffar Shamshirgar and A.-K. Tornberg. The Spectral Ewald method for singly periodic domains. *J. Comput. Phys.*, 347:341–366, 2017. DOI:10.1016/j.jcp.2017.07.001.
- [103] D. Saintillan, E. Darve, and E. S. G. Shaqfeh. A smooth particle-mesh Ewald algorithm for Stokes suspension simulations: the sedimentation of fibers. *Phys. Fluids*, 17(3):033301, 2005. DOI:10.1063/1.1862262.

- [104] J. San Martín, L. Smaranda, and T. Takahashi. Convergence of a finite element/ALE method for the Stokes equations in a domain depending on time. *J. Comput. Appl. Math.*, 230:521–545, 2009. DOI:10.1016/j.cam.2008.12.021.
- [105] E. Schlauch, M. Ernst, R. Seto, H. Briesen, M. Sommerfeld, and M. Behr. Comparison of three simulation methods for colloidal aggregates in Stokes flow: Finite elements, lattice Boltzmann and Stokesian dynamics. *Comput. Fluids*, 86:199–209, 2013. DOI:10.1016/j.compfluid.2013.07.005.
- [106] B. Schott, C. Ager, and W. A. Wall. Monolithic cut finite element-based approaches for fluid–structure interaction. *Int. J. Numer. Methods Eng.*, 119:757–796, 2019. DOI:10.1002/nme.6072.
- [107] N. Sharma and N. A. Patankar. A fast computation technique for the direct numerical simulation of rigid particulate flows. *J. Comput. Phys.*, 205:439–457, 2005. DOI:10.1016/j.jcp.2004.11.012.
- [108] A. Sierou and J. F. Brady. Accelerated Stokesian Dynamics simulations. *J. Fluid Mech.*, 448:115–146, 2001. DOI:10.1017/S0022112001005912.
- [109] A. Sokolov, M. M. Apodaca, B. A. Grzybowski, and I. S. Aranson. Swimming bacteria power microscopic gears. *PNAS*, 107(3):969–974, 2010. DOI:10.1073/pnas.0913015107.
- [110] C. Sogrentone and A.-K. Tornberg. A highly accurate boundary integral equation method for surfactant-laden drops in 3D. *J. Comput. Phys.*, 360:167–191, 2018. DOI:10.1016/j.jcp.2018.01.033.
- [111] B. Sprinkle, F. Balboa Usabiaga, N. A. Patankar, and A. Donev. Large scale Brownian dynamics of confined suspensions of rigid particles. *J. Chem. Phys.*, 147:244103, 2017. DOI:10.1063/1.5003833.
- [112] T. M. Squires and S. R. Quake. Microfluidics: Fluid physics at the nanoliter scale. *Rev. Mod. Phys.*, 77(3):977–1026, 2005. DOI:10.1103/RevModPhys.77.977.
- [113] D. Sushnikova, L. Greengard, M. O’Neil, and M. Rachh. FMM-LU: A fast direct solver for multiscale boundary integral equations in three dimensions. *Preprint*, arXiv:2201.07325, 2022.
- [114] T. N. Swaminathan, K. Mukundakrishnan, and H. H. Hu. Sedimentation of an ellipsoid inside an infinitely long tube at low and intermediate Reynolds numbers. *J. Fluid Mech.*, 551:357–385, 2006. DOI:10.1017/S0022112005008402.
- [115] S. Tlupova and J. T. Beale. Nearly singular integrals in 3D Stokes flow. *Commun. Comput. Phys.*, 14(5):1207–1227, 2013. DOI:10.4208/cicp.020812.080213a.

- [116] S. Tlupova and J. T. Beale. Regularized single and double layer integrals in 3D Stokes flow. *J. Comput. Phys.*, 386:568–584, 2019. DOI:10.1016/j.jcp.2019.02.031.
- [117] A.-K. Tornberg and L. Greengard. A fast multipole method for the three-dimensional Stokes equations. *J. Comput. Phys.*, 227(3):1613–1619, 2008. DOI:10.1016/j.jcp.2007.06.029.
- [118] M. Uhlmann. An immersed boundary method with direct forcing for the simulation of particulate flows. *J. Comput. Phys.*, 209:448–476, 2005. DOI:10.1016/j.jcp.2005.03.017.
- [119] F. Vico, L. Greengard, and M. Ferrando. Fast convolution with free-space Green’s functions. *J. Comput. Phys.*, 323:191–203, 2016. DOI:10.1016/j.jcp.2016.07.028.
- [120] L. R. Volpatti and A. K. Yetisen. Commercialization of microfluidic devices. *Trends Biotechnol.*, 32(7):347–350, 2014. DOI:10.1016/j.tibtech.2014.04.010.
- [121] M. Wala and A. Klöckner. A fast algorithm for Quadrature by Expansion in three dimensions. *J. Comput. Phys.*, 388:655–689, 2019. DOI:10.1016/j.jcp.2019.03.024.
- [122] M. Wala and A. Klöckner. Optimization of fast algorithms for global Quadrature by Expansion using target-specific expansions. *J. Comput. Phys.*, 403:108976, 2020. DOI:10.1016/j.jcp.2019.108976.
- [123] H. Wang, T. Lei, J. Li, J. Huang, and Z. Yao. A parallel fast multipole accelerated integral equation scheme for 3D Stokes equations. *Int. J. Numer. Methods Eng.*, 70(7):812–839, 2007. DOI:10.1002/nme.1910.
- [124] M. Wang and J. F. Brady. Spectral Ewald Acceleration of Stokesian Dynamics for polydisperse suspensions. *J. Comput. Phys.*, 306:443–477, 2016. DOI:10.1016/j.jcp.2015.11.042.
- [125] T. Wang, R. Yokota, and L. A. Barba. ExaFMM: a high-performance fast multipole method library with C++ and Python interfaces. *J. Open Source Softw.*, 6(61):3145, 2021. DOI:10.21105/joss.03145.
- [126] J. F. Wendt, editor. *Computational Fluid Dynamics*. Springer, Berlin Heidelberg, 3rd edition, 2009. DOI:10.1007/978-3-540-85056-4.
- [127] B. Wu and P.-G. Martinsson. Corrected trapezoidal rules for boundary integral equations in three dimensions. *Numer. Math.*, 149:1025–1071, 2021. DOI:10.1007/s00211-021-01244-1.

- [128] J. Wu and C. Shu. Particulate flow simulation via a boundary condition-enforced immersed boundary–lattice Boltzmann scheme. *Commun. Comput. Phys.*, 7(4):793–812, 2010. DOI:10.4208/cicp.2009.09.054.
- [129] W. Yan and R. Blackwell. Kernel aggregated fast multipole method. *Adv. Comput. Math.*, 47:69, 2021. DOI:10.1007/s10444-021-09896-1.
- [130] W. Yan and M. Shelley. Flexibly imposing periodicity in kernel independent FMM: A multipole-to-local operator approach. *J. Comput. Phys.*, 355:214–232, 2018. DOI:10.1016/j.jcp.2017.11.012.
- [131] L. Ying, G. Biros, and D. Zorin. A kernel-independent adaptive fast multipole algorithm in two and three dimensions. *J. Comput. Phys.*, 196:591–626, 2004. DOI:10.1016/j.jcp.2003.11.021.
- [132] L. Ying, G. Biros, and D. Zorin. A high-order 3D boundary integral equation solver for elliptic PDEs in smooth domains. *J. Comput. Phys.*, 219:247–275, 2006. DOI:10.1016/j.jcp.2006.03.021.
- [133] H. Zhao, A. H. G. Isfahani, L. N. Olson, and J. B. Freund. A spectral boundary integral method for flowing blood cells. *J. Comput. Phys.*, 229:3726–3744, 2010. DOI:10.1016/j.jcp.2010.01.024.