A BOUNDARY INTEGRAL METHOD APPLIED TO STOKES FLOW

A THESIS SUBMITTED TO THE

SCHOOL OF MATHEMATICS AND STATISTICS

UNIVERSITY COLLEGE

UNIVERSITY OF NEW SOUTH WALES

AUSTRALIAN DEFENCE FORCE ACADEMY

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

By

John Roumeliotis

March 2000

© Copyright 2000

by

John Roumeliotis

Certificate of Originality

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of a university or other institute of higher learning, except where due acknowledgement is made in the text.

I also hereby declare that this thesis is written in accordance with the University's Policy with respect to the Use of Project Reports and Higher Degree Theses.

J. Roumeliotis

Acknowledgements

The work in this thesis was undertaken during a Teaching Fellowship at the School of Mathematics and Statistics, University College, UNSW and a continuing Lecturer position at the School of Communications and Informatics, Victoria University of Technology. I gratefully acknowledge the support provided by these institutions.

Thanks are due to my (many) supervisors. Dr. Adam Kucera, who saw merit in the proposal that sought to connect theoretical numerical analysis with a practical investigation of a physical model, and not least for his wonderful conversations, support and friendship; Dr. Glenn Fulford for his helpful insights, guidance, support and continued interest, and Dr. Geoff Mercer for his patience and understanding.

I would also like to acknowledge the friendship and support provided by the staff at the School of Mathematics and Statistics and School of Communications and Informatics. Particularly the guidance of Associate Professor Neil Barnett, Professor Colin Pask, Professor Rowland Sammut, and Professor Sever Dragomir is gratefully acknowledged. As well as the encouragement and guidance provided by Professor Edgar Smith during my undergraduate studies at La Trobe University.

Special thanks are due to my *proxy supervisor* Associate Professor Pietro Cerone. Peter's continual interest, insightful comments and cheeriness helped during the difficult times.

I also thank my wonderful friends for their non-stop support, in particular Peter Vouzas, Dr. Simon Watt and Joe Kovacs.

An especially warm thank you goes to my parents, Βασείλιος and Μαργυρώ Ρουμελιώτη, as well as my brothers Arthur and David and sister-in-law Barbara. It is a pleasure to acknowledge their marvelous support and encouragement.

Most of all, I thank my wife, Katalin Kovacs. Her constant friendship, love, understanding, encouragement and superhuman support are enormously appreciated.

to my wife Katalin Kristy Kovács

and my parents Βασείλιος και Μαργιρώ Ρουμελιώτη

Abstract

This dissertation is an examination of the application of the boundary integral equation method to describe axi-symmetric particle motion in Stokes flow. In integral form, the Stokes flow equations describe the flow at any point using a surface distribution of singularities over given boundaries. For rigid particles, the strength of the singularity distribution is unknown resulting in a Fredholm integral equation of the first kind. For free surfaces, it is the surface velocity which is unknown and this results in an equation of the second kind. The axi-symmetric integral equations are two-dimensional, linear and exhibit a logarithmic singularity via the presence of complete elliptic integrals of the first and second kind.

The work in this dissertation can be divided into two parts. The first is a theoretical investigation of Fredholm integral equations of the first kind and the second part is a study of numerical techniques to simulate axi-symmetric particle (rigid and drop) dynamics in Stokes flow.

The theoretical investigation is undertaken to identify and examine the major issues involved in the numerical inversion of first kind equations. Two solution techniques are compared in terms of their stability and accuracy. One is an expansion-collocation method, the other is based on interpolation-collocation. We show that first kind equations are ill-conditioned and that this manifests in instability with high frequency unknowns. It is shown that the expansion approach can fail for this precise reason and a Shanks transformation (Shanks 1955) is employed to avoid higher order terms as well as increase convergence rate.

The interpolation method uses Hermite interpolation polynomials to allow the nodal behaviour of the unknown to be furnished. We show that this method is sensitive to collocation and present a method to find an optimal collocation strategy. This is done by employing the Peano kernel theory to develop a weighted trapezoid-like integral inequality. Analysis of the inequality bound reveals an optimal gridding scheme, as well as an indication of favourable collocation points. The Peano kernel theory is expanded to account for more general functions and we describe a method to obtain weighted (or product) composite quadrature rules that share the same abundance of

error bounds as Newton-Cotes type rules but have the advantage of being more accurate. We report that, in this case, interpolation-collocation is more accurate and stable than expansion-collocation.

The main results of this work are the presentation and analysis of a highly accurate and efficient algorithm to study axi-symmetric particle interactions in Stokes flow. For rigid particles, a Hermite polynomial method is used to calculate the particle traction. The numerical integration is performed using a combination of Gauss-Legendre and Gauss-Log quadrature in conjunction with a highly accurate polynomial approximation of the complete elliptic integrals. We show that the method is stable and performs well for simple spheroidal geometries, but due to the ill-conditioned nature of the integral representation, is subject to instability for more complicated particle shapes. Two methods, based on particle curvature, are presented that increase stability and, not least, increase accuracy.

Buoyant liquid drop deformation is studied via inversion of the second kind integral equation. This representation is not subject to the ill-conditioning of the first kind equation and we focus on other aspects of the numerical implementation. They are the representation of deformable surfaces, integration and time-stepping of the dynamic simulation. Results for one, two and three drop interactions are given as well as for a toroidal drop simulation. The algorithm is compared to those given in other studies. Three drop and toroidal drop simulations have not been reported previously (to the author's knowledge) and these are compared to the earlier results as well as experimental results reported elsewhere.

Contents

List of Tables

List of Figures

List of Algorithms

Papers published during the author's candidature

From the material in this thesis there are, at the time of submission, three papers which have been published in refereed publications.

- Part of the work from chapter 5 has been published in *Computational Techniques and Applications: CTAC97*, World Scientific in 1997. Co-authors are Glenn R. Fulford and Adam Kucera.
- Apart from the toroidal drop investigation, the work from chapter 5 has been published in the international journal *Computers and Fluids*, volume 29, 2000. Glenn R. Fulford is a co-author.
- The work from appendix C has been published in the *Journal of the Korea Society for Industrial and Applied Mathematics* in volume 3, 1999. Co-authors are Pietro Cerone and Sever S. Dragomir.

A manuscript based on the work in chapter 3 and appendix C has been submitted to the *Journal of Inequalities in Pure and Applied Mathematics* (no co-authors) and the work in chapter 4 (co-authored with Adam Kucera) will be submitted for publication in the latter half of 2000.

In the fullness of time, I hope that the other original contribution, namely the toroidal drop deformation of chapter 5, will also be published.

In addition, other papers published or submitted to refereed publications during the author's candidature are listed below.

- Barnett, N. S., P. Cerone, S. S. Dragomir and J. Roumeliotis (2000). Some inequalities for the dispersion of a random variable whose pdf is defined on a finite interval. Accepted for

publication in *J. Inequal. Pure Appl. Math.* and will appear in Volume 2, Number 1, 2001.

- Barnett, N. S., P. Cerone, S. S. Dragomir and J. Roumeliotis (2000). Some inequalities for the expectation and variance of a random variable whose pdf is n-time differentiable. *J. Inequal. Pure Appl. Math.* **1**(2), Article 21.

- Barnett, N. S., P. Cerone, S. S. Dragomir, J. Roumeliotis and A. Sofo (1999). A survey on Ostrowski type inequalities for twice differentiable mappings and applications. Accepted for publication by *Kluwer Academic Publishers*.

- Cerone, P., S. S. Dragomir and J. Roumeliotis (1999). An inequality of Ostrowski type for mappings whose second derivatives are bounded with applications. *East Asian Math. J.* **15**(1), 1–9.

- Cerone, P., S. S. Dragomir and J. Roumeliotis (1998). An inequality of Ostrowski type for mappings whose second derivatives belong to $L_p(a, b)$ and applications. Submitted for publication in *Mathematical Communications*.

- Cerone, P., S. S. Dragomir and J. Roumeliotis (1999). An inequality of Ostrowski-Grüss type for twice differentiable mappings and applications in numerical integration. *KYUNGPOOK Math. J.* **39**(2), 333–341.

- Cerone, P., S. S. Dragomir and J. Roumeliotis (1999). An inequality of Ostrowski type for mappings whose second derivatives belong to $L_1(a, b)$ and applications. *Honam. Math. J.* **21**(1), 127–137.

- Cerone, P., S. S. Dragomir and J. Roumeliotis (1999). Some Ostrowski type inequalities for n-time differentiable mappings and applications. *Demon. Math.* **32**(4), 697–712.

- Cerone, P., S. S. Dragomir, J. Roumeliotis and J. Sunde (1999). A new generalisation of the trapezoid formula for n-time differentiable mappings and applications. Submitted to *Demon. Math.*

- Cerone, P., J. Roumeliotis and G. Hanna (1999). On weighted three point quadrature rules. Accepted for publication in *Computational Techniques and Applications: CTAC99*.

- Dragomir, S. S., P. Cerone, N. Barnett and J. Roumeliotis (2000). An inequality of Ostrowski type for double integrals and applications for cubature formulae. *Tamsui Oxf. J. Math. Sci.* **16**(1), 1–16.

- Dragomir, S. S., P. Cerone and J. Roumeliotis (2000). A new generalization of Ostrowski's integral inequality for mappings whose derivatives are bounded and applications in numerical integration and for special means. *Appl. Math. Lett.* **13**, 19–25.

- Dragomir, S. S., P. Cerone, J. Roumeliotis and S. Wang (1999). A weighted version of the Ostrowski inequality for mappings of Hölder type and applications in numerical analysis. *Bull. Math. Soc. Sc. Math. Roumanie* **42**(90)(4), 301–314.

- Hanna, G., P. Cerone and J. Roumeliotis (1999). An Ostrowski type inequality in two dimensions using the three point rule. Accepted for publication in *Computational Techniques and Applications: CTAC99*.

- Weber, R. O. and J. Roumeliotis (1995). $i \wedge i \wedge i \wedge i \cdots$. *Austral. Math. Soc. Gazette*, **22**(4), 182–184.

Chapter 1

Introduction

1.1 Stokes flow and its application

Many biological and industrial processes involve fluid flows in which viscosity is large and/or particle lengths are small. An important indicator of flows of this type is the Reynolds number. The Reynolds number, R_e, is defined as

$$R_e = \frac{\rho U L}{\mu}, \tag{1.1}$$

where U is a characteristic velocity scale, L a characteristic length scale and, ρ and μ are the density and viscosity of the fluid respectively. This non-dimensional parameter, whose importance was first recognised by Stokes (1851), is a ratio of the inertial to viscous forces. Thus fluids of sufficient viscosity, slow motion or comprised of small particles will result in a small Reynolds number. In reality the Reynolds number can never vanish, or equivalently inertial forces are never zero, but it has been shown that the zero Reynolds number approximation produces an accurate representation of the flow field in the vicinity of small particles (Batchelor 1967; Stokes 1851). These flow regimes, called Stokes or creeping flows have a number of distinguishing features. They form the basis of the assumptions made in this dissertation, over and above those of normal Newtonian fluids. We list them below:

- For steady flows, pressure and frictional forces are in equilibrium at each instant. Thus the motion of the fluid is not dependent on its temporal history but rather induced and sustained through conditions on boundaries. That is, the motion of a particle depends wholly on the forces subjected to it at each instant; in the absence of these forces the motion

will cease almost immediately (Purcell 1977; Meiners and Quake 1999). This absence of inertia causes the flow to be laminar, and rich complex features such as turbulence and secondary flows do not occur. Lack of inertia also dictates that fluids equilibrate through diffusion and mixing, though the action of stirring is ineffective.

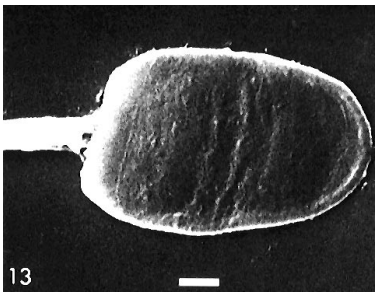
- Stokes flows are completely reversible and this can lead to identification of symmetries in the flow. For example, a simple consideration of a spherical particle falling parallel to a vertical wall dictates that the particle must maintain its horizontal distance from the wall. This argument can be extended to more general symmetric shapes as long as the symmetry axis is parallel to the plane wall. In this case no overall horizontal translation will occur, but depending on orientation, particle rotation can be induced (de Mestre and Russel 1975). Recent experiments (Lecoq *et al.* 1993) have shown that flow field symmetry plays a major role in low (non-zero) Reynolds number motion. Reversibility is further discussed in chapter 5 where the deformation of a liquid toroidal drop is described.

- Since frictional forces dominate, we would expect any disturbances to have long range effects. The magnitude of a disturbance due to a particle at distance r is $O(1/r)$[1]. Thus unlike potential flow, which is inertia driven, the creeping flow field is influenced by particle interactions over large regions. Any numerical implementation must account for this effect. This important factor will be discussed in section 1.3.

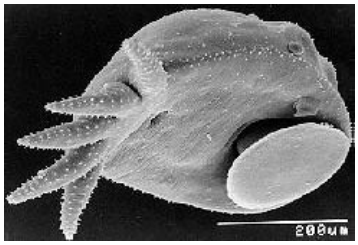
An excellent account of the features of low Reynolds number flows is given in Purcell (1977).

Leal (1980) provides a comprehensive review of how to include weak inertial and non-Newtonian deviations on particle and drop motion. Most of the techniques, which are analytic in nature, have mainly been used to describe drop and particle evolution under idealised conditions. That is, for spherical or *slightly deformed* shapes, simple flows etc. Nevertheless, studies have shown that these small effects can accumulate to describe motion that can never be exhibited by the Stokes class of flows (see for example the experiments of Meiners and Quake (1999) and Steinberger *et al.* (1968) and observations regarding toroidal drops from the theoretical work of Kojima *et al.* (1984)). In spite of this, the Stokes flow equations have been used with great success to predict flow characteristics in many physical problems. Almost any application which involves flow in small regions (such as capillary flow, lubrication) or small particles can be modelled using the Stokes flow equations. Many of these involve industrial and biological processes. Examples involving rigid particles or droplets include filtering (air conditioning, hemodialysis,

[1] See for example the definition of the *Stokeslet* in equation (2.11).



(a)



(b)

Figure 1.1: (a) Taken from Cummins et al. *(1988) showing sperm from the Flying Fox (*Pteropus poliocephalus*). The scale bar is 50nm. (b) A recently described ectosymbiont* Craspedella pedum *from the branchial chamber of the redclaw crayfish* Cherax quadricarinatus. *See Cannon and Sewwll (1995) for details.*

virus removal, feeding), sedimentation, micro-organism propulsion and transport (both flagellum and cilia induced), blood flow, ink-jet printing, micro-dispensing of drugs and spray coating. A good account of the state of the art prior to 1965 can be found in Happel and Brenner (1965). Below we expand on a few recent applications.

There has been much interest in the study of the dynamics of small organisms (see Figures 1.1 and 1.2). At this length scale the Reynolds number is quite small ($\leq 10^{-3}$) and thus inertial effects are negligible. Many of the organisms propel themselves by planar and helical movement of a long slender tail. This geometric feature was first exploited in the foundation work of Hancock (1953). Hancock and later Gray and Hancock (1955) considered the propulsion of micro-organisms by the oscillation of their tails. To simulate the force on the surrounding fluid due to tail motion a distribution of point forces was placed along the centreline. The strength of the distribution was then obtained by considering the conditions on the tail surface. This work was later refined to include the presence of a head (for example in sperm flow) and account for the torque and pitch via the helical, as well as planar, tail movement (Chwang and Wu 1971). The ideas of Hancock and Gray were later adapted to model cilia (Blake 1972; Blake *et al.* 1982). There is now a considerable understanding of cilia motion and their role in micro-fluid transport which include: particle concentration, as in filter feeding (Blake and Fulford 1995); particle elimination, as in mucus transport in the lung and nasal passages (Blake and Fulford 1984; Fulford and Blake 1986b) and organism transport/propulsion (Blake 1971b; Liron and Mochon 1976a; Blake *et al.* 1983). An excellent review which includes both experimental and theoretical investigations is given in Sleigh *et al.* (1988).

(a)

(b)

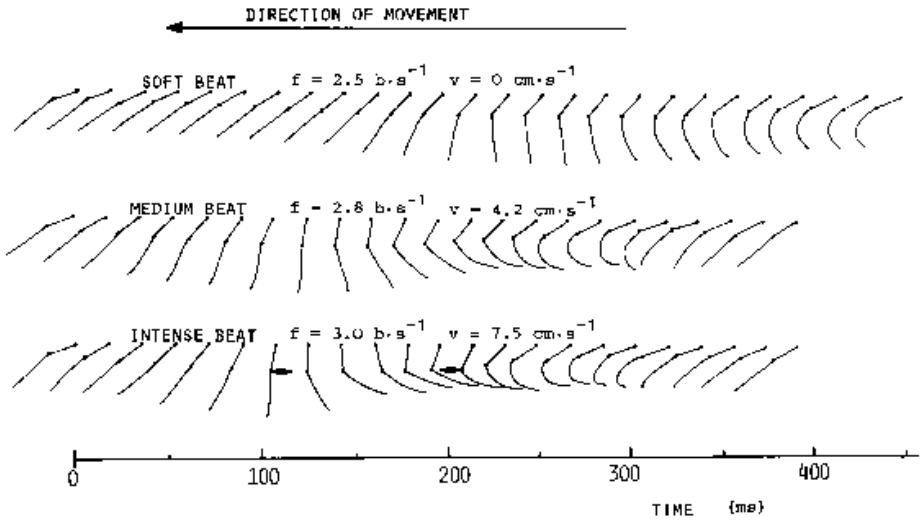


Figure 1.2: Reproduced, with permission, from Kils (1982). (a) Photograph of the Antarctic krill, ***Euphausia superba****, taken during the Antarctic expedition on board the WALTER HERWIG in the austral summer of 1977/78. The krill is a filter feeder and the dynamics of the operation of its filtering basket can be modelled using the Stokes approximation. (b) Beat profile of the krill pleopod for soft, medium and intense beats. Even though the Reynolds number is not small, for intense beats the shape of the pleopod, especially the exo- and endopodite, are reminiscent of cilia bending (Sleigh* et al. *1988) and (Fulford and Blake 1986b, cf. their Figure 2).*

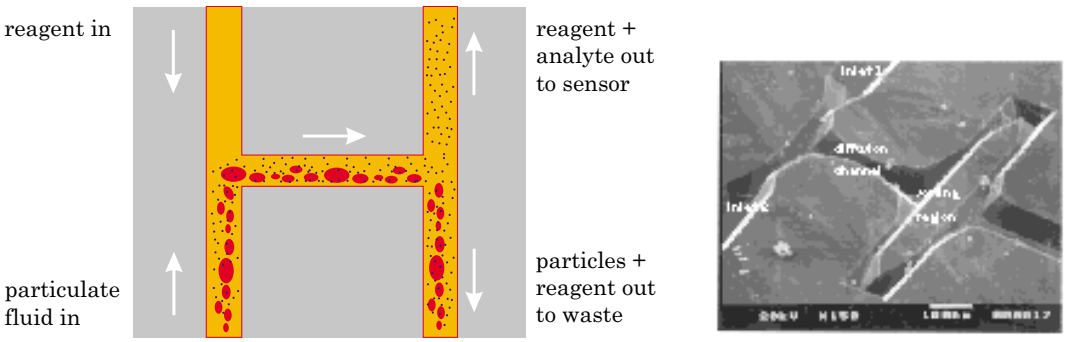


Figure 1.3: H-filter adapted from Brody and Yager (1997). The systems acts as an equilibrator for small molecules (analytes) as they diffuse, larger particles are left in the lower half and are transported away. On the right is a scanning electron micrograph of a Si fabricated H-filter.

Other applications of Stokes flow involve industrial processes such as particle concentration or filtration. For example, micro-biological separation is mainly used to either filter out unwanted particles (as in dialysis and blood virus removal) or concentrate them (as in plasma collection and biological agent detection systems). One way of achieving this is to use a synthetic membrane filter, a standard technique used in the biopharmaceutical industry. For instance, all products derived from a human source require treatment to eliminate pathogens (for example AIDS, CJD). Membranes filter out selected products by distinguishing size and/or shape and improving membrane production and operation would be of great value.

Other ways to concentrate and extract particles have been developed to exploit the characteristics of creeping flow (Brody *et al.* 1996; Brody and Yager 1997). As previously stated, these flows are distinguished by their natural diffusion properties and absence of inertia. Filters have been designed that allow separation based on this diffusion property (see Figure 1.3). The diffusion coefficient is particle dependent and this is used to separate agents in a particulate flow. These designs are quite unique to the micro-scale and will fail to operate efficiently if the scale (and hence Reynolds number) is increased. The H-filter is merely one example of the new devices being manufactured using micro-fabrication techniques. Another is the *no-moving-parts* valve (Forster *et al.* 1995) (see Figure 1.4) which forms an integral part of any pump in a micro-fluidic system. Using fixed geometry to control flow direction was first proposed by Tesla (1920) (see Figure 1.5). This is highly suited to the micro-scale where simplicity in manufacture and control are a requirement for robust operation.

To summarise, we state that insight into the nature of micro-organism behaviour or the mastery of micro-fluidic devices requires a complete *theoretical* understanding of particle behaviour in

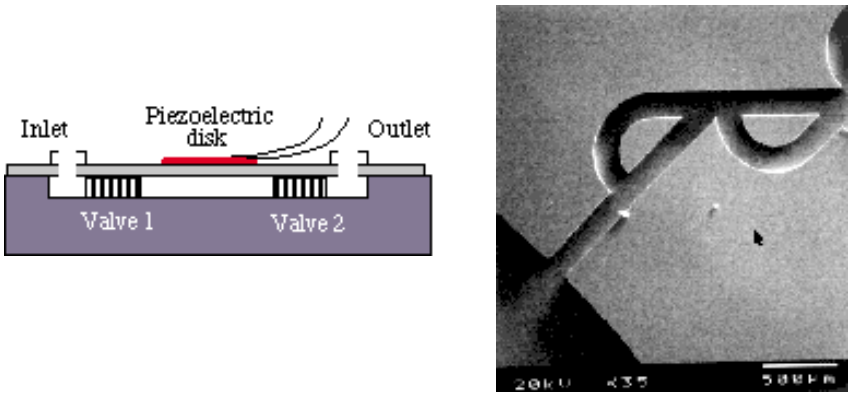


Figure 1.4: No-moving-parts valve taken from Forster et al. *(1995). At left is a side schematic of the pump. At right is a scanning electron micrograph of the valve.*

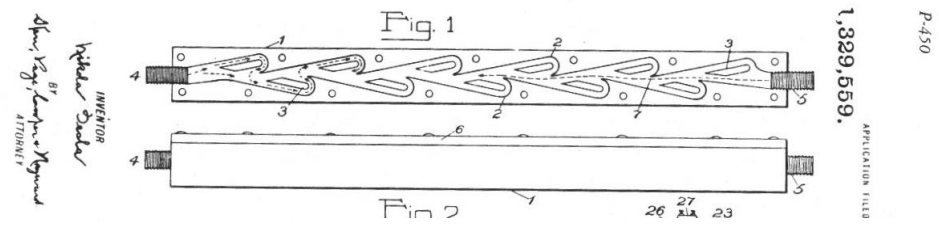


Figure 1.5: Part of the patent of Tesla (1920) showing the design of the fixed geometry valve. Compare with Figure 1.4.

the Stokes flow universe. Work in this field started in 1851 when Stokes considered the viscous forces on a moving sphere and has today evolved into many diverse areas. In the next section we outline some of the methods used to study particle movement and interaction in Stokes flow.

1.2 Rigid particle and drop interactions

The Stokes flow equations, which are discussed in more detail in chapter 2, are

$$\nabla p = \mu \nabla^2 \boldsymbol{u}, \qquad \nabla \cdot \boldsymbol{u} = 0, \tag{1.2}$$

where $\boldsymbol{u}$ is the velocity field and p the pressure field. The most striking feature of these equations are their linearity and it is this that has been exploited over the years to develop analytic solutions to many problems. Special coordinate systems which render (1.2) separable have been used to find the flow field of a translating sphere (Stokes 1851), translating ellipsoid (Oberbeck 1876), rotating ellipsoid (Edwardes 1892), two spheres rotating about their line of centres (Jeffery 1912) and two spheres falling along their line of centres (Stimson and Jeffery 1926). Difficulties arise when attempting to solve for systems of more than one particle and they are attributable to the impracticality of expressing all boundaries as a constant surface in an orthogonal system. Jeffery (1912) and Stimson and Jeffery (1926) managed to use bipolar coordinates to account exactly for two spheres. This method has been employed by others to deal with two idealised boundaries. Brenner (1961) for a sphere translating perpendicular to a plane rigid wall, Haber *et al.* (1973) for two interacting spherical drops, Lee and Leal (1980) for a sphere interacting with an interface, Ganatos *et al.* (1980) for the transverse motion of a sphere between two parallel walls, Kojima *et al.* (1984) for their investigation of spheroidal and toroidal drop deformation and Kim (1987) for flow past three spheres. The last reference is of particular merit since three boundaries are accounted for. They made use of particular addition theorems that transform vector harmonics between different coordinate systems.

As an investigative tool to study particle interactions, the use of special coordinate systems alone is rather limited. The boundaries (at most two or three) must have idealised shapes and be fixed. For example the solution derived by Haber *et al.* (1973) is accurate only for large drop separations and/or small deformations and that of Lee and Leal (1980) for a flat interface. Nevertheless, these techniques have led to the development of new areas including the Oseen correction and an excellent understanding of particle inertia (Oseen 1927; Stewartson and Howarth 1960; Steinberger *et al.* 1968; Lovalenti and Brady 1993). A number of exact solutions are listed in

Lamb (1932) and Happel and Brenner (1965).

To overcome the difficulty of treating only two boundaries, Brenner and Happel (1958) popularised the method of reflections. Since (1.2) are linear, the velocity and pressure fields can be written as a sum with each component representing the disturbance of a particular boundary. For example, for n particles with surfaces $S_1, S_2, \ldots, S_n$, we may write

$$\boldsymbol{u} = \boldsymbol{u}_1 + \boldsymbol{u}_2 + \cdots + \boldsymbol{u}_n,$$

with boundary conditions

$$\boldsymbol{u}_i = \begin{cases} \mathbf{0}, & \boldsymbol{x} \in S_1 \\ \vdots & \vdots \\ \boldsymbol{U}_i, & \boldsymbol{x} \in S_i \\ \vdots & \vdots \\ \mathbf{0}, & \boldsymbol{x} \in S_n \end{cases} \qquad \text{for } i = 1, \ldots, n.$$

Solving for $\boldsymbol{u}_i$ will generate a flow field at all other boundaries. Hence, each subsequent velocity is itself written as a sum with the boundary condition chosen in such a way as to cancel the *reflection* caused by the previous term. Thus, a series is generated which accounts for particle interactions after successive iterates. This solution converges slowly for large interactions but has been shown to be accurate when compared to the known solution of Stimson and Jeffery (1926). The technique has been used successfully to investigate, for example, the interaction of: a sphere moving parallel to a cylindrical wall (Brenner and Happel 1958); three spheres (Kynch 1959); walls on an arbitrary particle (Brenner 1962); a number of particles along the axis of a cylinder (Sonshine and Brenner 1966); two plane walls on the motion of a sphere (Sano and Hasimoto 1978); a particle and a surface contaminated by a surfactant (Shail 1983); particle motion near a fluid interface (Yang and Leal 1984) and the flow past a pair of stagnant cap bubbles (Lerner and Harper 1991).

The method of reflections, although quite general for particle interactions, is cumbersome and again restricted to ideal geometries. A much more powerful approach is the use of singularities to describe types of flow. This is particularly suited to the Stokes equations since their linearity allows construction of singularity distributions that represent very general flows.

The fundamental singularity of Stokes flow, called the *Stokeslet*[2], is given by

$$G_{ij}(\boldsymbol{x}) = \frac{1}{8\pi\mu}\left(\frac{\delta_{ij}}{|\boldsymbol{x}|} + \frac{x_i x_j}{|\boldsymbol{x}|^3}\right), \qquad \boldsymbol{x} \in \mathbb{R}^3 \tag{1.3}$$

[2]Termed by Hancock (1953).

and represents the disturbance at $\boldsymbol{x}$ due to a point force. For any rigid body, $\mathcal{S}$ translating with velocity $\boldsymbol{u}$, we can show that (Lorentz 1897) (see chapter 2)

$$u_j(\boldsymbol{x}) = -\int_{\mathcal{S}} G_{ij}(\boldsymbol{x}-\boldsymbol{y}) f_i(\boldsymbol{y})\, d\mathcal{S}(\boldsymbol{y}), \qquad \boldsymbol{x} \in \mathcal{S}. \tag{1.4}$$

That is, the disturbance in the flow is represented by point forces distributed with density $\boldsymbol{f}$ over the surface. Naturally, if there are many bodies then $\mathcal{S}$ becomes the collection of surfaces.

The elegance and power of equation (1.4) may be appreciated when one considers different flow regimes. If the problem of interest is in a half-space instead of an infinite medium, or particles exhibit rotation instead of translation then one only needs to find the appropriate fundamental singularity which would be substituted for $\boldsymbol{G}$ in (1.4).

This approach led to a flurry of activity in finding Green's functions for different prescribed flows. The fundamental singularity for rotation, called the *rotlet*, was identified by Batchelor (1970b) and that for a half-space by Blake (1971a). Blake and Chwang (1974) later established a number of fundamental and higher order singularities for the half-space including the 1/2-space rotlet. The fundamental singularity in the presence of an interface was found by Aderogba and Blake (1978a, 1978b) and recently a *blinking Stokeslet*, where the point force periodically changes its point of application, was evaluated by Blake *et al.* (1998). Chwang and Wu (1975) are worthy of mention for their work in exploring and tabulating a number of fundamental and higher singularities for many prescribed flows. There are of course other fundamental solutions, for example for periodic motion or flow in a region with periodic boundaries or two dimensional flow. Details may be found in Hasimoto and Sano (1980) and Pozrikidis (1992).

A major use of these singularities is in the so-called *singularity method*. The integral (1.4) is quite difficult to evaluate analytically and this method involves moving the singularity distribution from the surface to some chosen space-curve, often the centreline. Gray and Hancock (1955) first used this technique, using the Stokeslet, to study the movement of spermatozoa. The Stokeslets were distributed along the centreline of the organisms to approximate the flow induced by their propulsion. If the body is slender, as in spermatozoa, then a natural perturbation scheme in the slenderness ratio arises. The singularity method coupled with slender bodies is known as slender body theory (see Blake (1974)). There are countless examples, a few of which we give here, where slender body theory, or the more general singularity method, have been used to study flows in infinite domains (Tuck 1964; Taylor 1969; Tillett 1970; Batchelor 1970a; Keller and Rubinov 1976) or with plane boundaries (de Mestre and Russel 1975; Katz *et al.* 1975; Russel *et al.* 1977; Liron and Blake 1981; Barta and Liron 1988) or with free surfaces (Blake and Fulford 19

Fulford and Blake 1983; Yang and Leal 1983; Yang and Leal 1984; Fulford and Blake 1986a).

Finally we note that although the singularity method is essentially an approximation from a surface distribution to a line distribution of singularities, it has been used to produce exact solutions. The most well known being the application of slender body theory to obtain the exact solution for a spheroid. Chwang and Wu (1975) construct a number of exact solutions using the singularity method.

For more details regarding the singularity method we refer the reader to de Mestre (1974), Fulford (1984), Kim and Karrila (1991) and Pozrikidis (1992).

1.3 The boundary integral method

The methods of the previous section have proven to be invaluable tools that lend insight into a great variety of problems. Their limitation is that analytic approximations are incapable of reproducing the rich complex behaviour inherent in many physical problems of interest. For example, for large deformations in drop-drop or particle-drop interactions, or mobility problems involving complex rigid surface shapes. Experiments of complex drop deformations are shown in Kojima *et al.* (1984) and Manga and Stone (1993). A numerical solution is, most likely, the only option available if one wishes to capture behaviour of this type.

The two most common approaches are the finite element method and the boundary integral equation method. The finite element method involves approximating the governing equations (1.2) by a linear system that spans the entire domain. The linear system thus formed is solved via inversion. The resulting matrix is often sparse and this can be exploited by specialised solvers. For exterior problems in higher dimensions, this method would require a very large grid. An infinite grid is out of the question, so the finite region needs to be large enough to cater for the long range effects of creeping flow. This method can also be difficult to implement for complicated geometries. Tracking large time dependent drop deformations, for example, would be extremely difficult. Another problem with the finite element method is the evaluation of quantities that may not necessarily be required. For example, the pressure field is often not needed in studies where the hydrodynamic force is of interest. Additionally, the velocity may only be required at a few prescribed locations and not at all points in the domain. The problem of infinite domains can be circumvented by employing the *infinite element method.* This has been done to solve Laplace's equation (Bettess 1977) and has been extended to other governing equations, see Bettess (1992)

and Gerdes (2000). This method still retains the other deficiencies of the finite element method (difficulties with temporal dependence and evaluating terms that may not necessarily be of interest) and has the additional complication of finding appropriate infinite element shape functions to suit the governing equation, see Gerdes (2000) for details. Even so, it would be a useful exercise to compare the boundary integral method and the infinite element method for rigid body creeping flow problems.

Another method which is gaining popularity for use with free surfaces is the *finite volume method*. We briefly mention that this method too relies on discretizing space (into small *volumes*) and so has the obvious drawback for exterior creeping flow problems.

The boundary integral method is the method of choice when solving external problems and those involving complex regions. The dimension of the problem is reduced and integration along complicated curves, if done with care, can be accurately evaluated. To evaluate the unknown, we require some functional approximation. The two methods we will consider are interpolation and spectral expansion.

The spectral approach consists of assuming that the unknown belongs to some inner product space and hence, can be represented as a linear combination of orthogonal basis functions. The coefficients of the expansion, and thus the solution, are obtained via application of the inner product. This method is often simple to formulate, but care needs to be taken with the the size of the expansion and the inner product, which can involve singular and highly oscillatory integrands. Higher order coefficients are notoriously difficult to obtain and an error here will disrupt the entire series (see chapter 3).

The interpolation method involves choosing some grid and approximating the unknown via a piecewise function. The order of the interpolation, distribution of the grid points and integration method are all important points that need consideration. To clarify the issues presented here, the spectral method is compared to the interpolation approach in chapter 3.

We will briefly outline the history of the application of the boundary integral equation to the solution of axi-symmetric Stokes flow problems. To assist in our discussion we present the integral representation of the Stokes equations (derived in appendix A)

$$c(\boldsymbol{x})u_k(\boldsymbol{x}) = -\int_{\mathcal{S}} G_{ik}(\boldsymbol{x}-\boldsymbol{y})f_i(\boldsymbol{y})\,dS(\boldsymbol{y}) - \int_{\mathcal{S}} \Sigma_{ijk}(\boldsymbol{x}-\boldsymbol{y})u_i(\boldsymbol{y})n_j(\boldsymbol{y})\,dS(\boldsymbol{y}), \qquad (1.5)$$

where

$$c(\boldsymbol{x}) = \begin{cases} 1, & \boldsymbol{x} \in \Omega, \\ \frac{1}{2}, & \boldsymbol{x} \in \mathcal{S}, \end{cases} \tag{1.6}$$

and Ω is the domain of the problem with boundary $\mathcal{S}$. The Green's function $\boldsymbol{G}$ is given in (1.3) and $\boldsymbol{\Sigma}$ is the *Stokes-stresslet* given by

$$\Sigma_{ijk}(\boldsymbol{x}) = -\frac{3}{4\pi}\frac{x_i x_j x_k}{|\boldsymbol{x}|^5}.$$

The first integral in (1.5) is known as the single-layer potential, and the second integral is the double-layer potential.

For rigid body problems the integral equation (1.5) simplifies to (1.4), an equation of the first kind in the surface traction $\boldsymbol{f}$ (shown in appendix A). Youngren and Acrivos (1975) solved the first kind equation for a single spheroid. They were able to obtain reasonably accurate solutions, as long as the curvature of the spheroid was not too large. They reported no ill-conditioning with the equation, even though their solution method, as explained in chapter 4 and appendix D, was somewhat imperfect. The instability of the first kind equation, which we show in chapter 4, led to the development of a second kind equation for rigid particles (Power and Miranda 1987). The Stokes-stresslet is itself a solution to the governing equation and in the case of a rigid surface, it is a simple matter to show that the double layer potential satisfies (see appendix A)

$$\int_{\mathcal{S}} \Sigma_{ijk}(\boldsymbol{x}-\boldsymbol{y})u_i(\boldsymbol{y})n_j(\boldsymbol{y})\,dS(\boldsymbol{y}) = -(1-c(\boldsymbol{x}))u_k(\boldsymbol{x}), \tag{1.7}$$

an equation of the second kind. The problem with this equation is that $\boldsymbol{\Sigma}$ cannot describe flows made up of rigid body motions. Indeed, equation (1.7) should be substituted into (1.5) to produce the single-layer, first kind equation (1.4), and is not in itself sufficient to allow solution by any numerical method. The remedy is to *complete* the system by stipulating conditions that are equivalent to the six rigid body motions (Power and Miranda 1987; Kim and Karrila 1991). Liron and Barta (1992) demonstrate the applicability of the equations by numerically solving several rigid particle problems. Also, Power and de Power (1993) use this formulation to investigate the problem of particle motion near a plane wall. Another second kind equation for rigid particles, based on the traction, has been presented by Ingber and Mondy (1993). They demonstrate its use, but there is a requirement to evaluate three normal derivatives of Cauchy principal value integrals. Details regarding the second kind equation formulations can be found in Karrila and Kim (1989) and Kim and Karrila (1991).

It would seem that the price paid to stave off ill-conditioning in first kind equations is the much higher workload in solving those of the second kind. Recently, Ingber and Mammoli (1999)

compared all three formulations on a set of test problems and showed that even though the condition number of the first kind system was large, it did not seem to influence the accuracy. In fact, the first kind equation performed slightly better than the other two and other work suggests that first kind equations are not necessarily worse than second kind (Niessner and Ribaut 1985). In chapter 4 the first kind equation is used, but modifications to the grid and kernel are made in an attempt to prevent ill-conditioning and not least, to increase accuracy.

Free surface and liquid drop problems have received much more widespread attention. This could be due to the fact that the boundary integral equation is of the second kind involving both the single and double layer potential, and hence simpler to solve. Inspection of (1.5) reveals that the double layer term exists in a Cauchy principal value sense which requires careful treatment.

Rallison and Acrivos (1978), Chi and Leal (1989), Stone and Leal (1989), Pozrikidis (1990a) and Manga and Stone (1993) all investigated drop deformation. Their numerical methods have steadily increased in sophistication, for example where Rallison and Acrivos used the trapezoidal rule and constant approximations for the surface and unknown velocity, Pozrikidis and Stone now employ Gauss-Legendre quadrature and higher order approximations for the surface. Both assume a piecewise-linear profile for the velocity. One curious feature that is present in all the work (Rallison and Acrivos 1978; Leal and Lee 1982; Geller *et al.* 1986; Chi and Leal 1989; Stone and Leal 1989; Pozrikidis 1990a; Tjahjadi *et al.* 1992; Manga and Stone 1993; Pozrikidis 1997) is the methodology for evaluating the singular integrals in (1.5). The singular nature of the integrand is almost totally disregarded and only accounted for via a small series expansion. We argue in chapter 4 and appendix D that higher interpolation and better use of Gaussian quadrature will decrease the errors they report. Our results in chapter 5 seem to confirm this.

1.4 Outline of this thesis

The basic equations of Stokes flow that will be used in this dissertation are given in chapter 2. We present the general field equations, fundamental singularities and boundary conditions for both rigid particles and free surfaces. The axi-symmetric integral equation for the rigid particle is also given in this chapter.

In chapter 3 we study and compare two methods for solving a one dimensional integral equation of the first kind. The first method implements a Fourier series expansion, where the coefficients are evaluated via collocation. We investigate the effect of expansion size as well as truncation

on the error. A scheme to accelerate the convergence of the series, Shanks' transformation, is also implemented and examined. In the second method we solve the integral equation via a cubic Hermite interpolant and investigate the effect of collocation on the solution. A Peano kernel inspired method is developed and used to theoretically predict an optimal collocation strategy. Numerical results for both methods are presented and compared.

In chapter 4, the rigid body, axi-symmetric, first kind integral equations are solved. Inspired by the results of chapter 3 we present, in detail, a method based on arbitrary interpolation order, grid distribution and polynomial approximation of the complete elliptic integrals, to find the traction of a translating prolate spheroid. Results for various eccentricities, grid sizes and interpolation order are given. These are then compared with a new, curvature based discretization scheme. The method, coupled with the new gridding procedure is trialled on a purposely chosen *difficult* particle. This particle, which possesses a large curvature and was deliberately chosen to tax the solution scheme, reveals the ill-conditioning of the first kind equation. Finally, we present another method to increase the accuracy of the solution. This involves introducing a weighted, curvature based, kernel into the integral equation.

In chapter 5, we outline an algorithm, using the second kind equation, to solve liquid drop interactions problems. The numerical method involves detailed consideration of the representation, integration and time-stepping. Results, including accuracy checks, are given for various two drop and three drops interactions. Finally, a toroidal drop deformation is presented. To the author's knowledge, there are no published results on the numerical simulation of toroidal drops and hence we compare the deformation to experimental data as well as to those given earlier.

In appendix A, the boundary integral equations for a surface in an unbounded, quiescent medium are derived.

In appendix B, the Hermite interpolation polynomials are defined, and a Maple procedure to calculate them is presented.

In appendix C, the Peano kernel theory used chapter 3 is justified and expanded. This is done by proving a series of bounds between a function value and its average in a weighted integral. The weight itself is quite general, thus providing a method of developing general product integration rules. A specific rule and weight, with the methods developed here, is employed in chapter 3.

Calculations involving the axi-symmetric form of the integral equations and complete elliptic

integrals are tabulated in appendix D. In this appendix, we also introduce the polynomial approximation to the complete elliptic integrals and show how their use is superior to the canonical evaluation via the arithmetic-geometric mean.

Finally, in appendix E, we complete the discussion of spectral methods given in chapter 3 by presenting a Maple routine that will produce a sequence of orthonormal polynomials given any integrable positive weight.

Chapter 2

Equations for Stokes Flow

In this dissertation we will consider the slow motion of a body ($\mathcal{S}$) immersed in an otherwise unbounded, stationary and incompressible fluid (Ω). Inertial forces are neglected and the Reynolds number is assumed to be negligible. Such flows are of interest when the fluid viscosity is large and/or length scales[1] are small (Happel and Brenner 1965). We further assume that the fluid is initially quiescent and that disturbances in the flow are induced by particle action – either directly due to prescribed motion or implicit through the action of buoyancy.

The body will be either a rigid particle or a deformable liquid drop. The distinction will be made by application of conditions on the surface $\mathcal{S}$. In the case of a rigid inflexible surface, the velocity is known and the governing equations can be re-cast into a set of coupled linear Fredholm integral equations of the first kind in the unknown traction, $\boldsymbol{f}$. This quantity is the stress experienced by the particle at each point on its surface. Integrating $\boldsymbol{f}$ over $\mathcal{S}$ furnishes the drag

$$D = \int_{\mathcal{S}} \boldsymbol{f} \cdot \boldsymbol{n} \, dS, \tag{2.1}$$

where $\boldsymbol{n}$ is the surface normal. The traction and the drag are known explicitly for simple geometries, see section 1.2, and it is these that will be used to test the accuracy of the numerical method developed in this thesis.

By contrast, the traction for a free surface is a known quantity and it is the velocity that is unknown. In this case the integral equation is of the second kind whose solution yields the surface deformation.

[1] Typically taken as the size of a particle or the interaction distance between particles.

2.1 Stokes flow equations

The Navier-Stokes equations describe the flow field of a fluid at any point $\boldsymbol{x} \in \Omega$ in terms of $\boldsymbol{u}^*$and p^*, the velocity and pressure of the fluid respectively. They are given by

$$\rho\frac{\partial \boldsymbol{u}^*}{\partial t^*} + \rho\left(\boldsymbol{u}^* \cdot \nabla^*\right)\boldsymbol{u}^* = -\nabla^* p^* + \mu{\nabla^*}^2\boldsymbol{u}^*, \tag{2.2}$$

where ρ is the density of the fluid and μ the viscosity. Here, we have adopted a starred notation to denote dimensional quantities. The equation, as written in (2.2), separates the *inertia* terms (left hand side) from the *viscous* terms (right hand side). If the flow field has a given characteristic velocity U, length L and frequency ω, we can define the following non-dimensional quantities

$$\boldsymbol{u} = \frac{\boldsymbol{u}^*}{U}, \qquad \boldsymbol{x} = \frac{\boldsymbol{x}^*}{L}, \qquad p = \frac{p^* L}{U\mu} \qquad \text{and} \qquad t = \omega t^*. \tag{2.3}$$

Thus the non-dimensional form of (2.2) is

$$\alpha^2\frac{\partial \boldsymbol{u}}{\partial t} + R_e\left(\boldsymbol{u} \cdot \nabla\right)\boldsymbol{u} = -\nabla p + \nabla^2\boldsymbol{u}, \tag{2.4}$$

where $\alpha = \sqrt{\rho\omega L^2/\mu}$ is known as the *Womersely parameter* and is a measure of inertial disturbance. The quantity R_e is the Reynolds number as defined in (1.1).

For creeping flow, it is assumed that $R_e \ll 1$. If the temporal disturbances are large enough then we may have $\alpha^2 \sim O(1)$ and (2.4) simplifies to

$$\alpha^2\frac{\partial \boldsymbol{u}}{\partial t} = -\nabla p + \nabla^2\boldsymbol{u}, \tag{2.5}$$

the *unsteady Stokes equations*; see Pozrikidis (1992, p. 1) and Kim and Karrila (1991, p. 147). The validity of this equation lies in the assumption of a high characteristic frequency. If we let, $w = U/L$, which would be the appropriate relation for the problem of a particle in an infinite fluid, then from the definition of α we can show that $\alpha = \sqrt{R_e}$. For flows that exhibit large temporal disturbances of order $O(R_e^{1/2})$, for example particles that begin their motion impulsively, or flows that are highly oscillatory, the use of the unsteady Stokes equation is required. These flows will not be considered here and we refer to Pozrikidis (1992) for their treatment.

Neglecting both the Womersely parameter and the Reynolds number in (2.4) produces the Stokes equations

$$\nabla p = \nabla^2\boldsymbol{u}. \tag{2.6}$$

In addition, for incompressible fluids we also have the continuity equation

$$\nabla \cdot \boldsymbol{u} = 0. \tag{2.7}$$

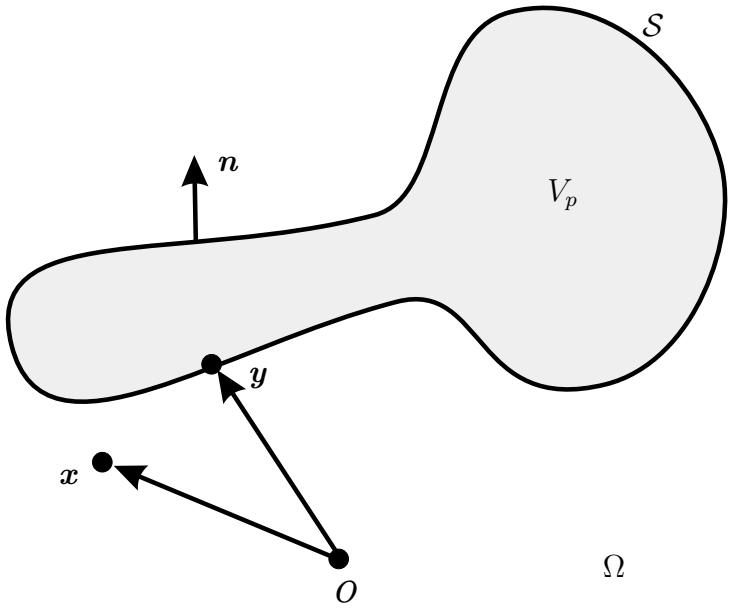


Figure 2.1: Schematic of a particle in an infinite fluid showing geometry and notation used.

In such flows the pressure and viscous forces are in equilibrium throughout the fluid at each instant. Thus, even though the flow may be time dependent, the temporal variable does not appear in the governing equations.

Since we assume that the fluid is initially at rest we impose the far-field condition

$$|\boldsymbol{u}(\boldsymbol{x})| \to 0 \qquad \text{and} \qquad p(\boldsymbol{x}) \to 0 \qquad \text{as} \qquad |\boldsymbol{x}| \to \infty. \tag{2.8}$$

A schematic of the geometry is shown in figure 2.1, where $\mathcal{S}$ is the surface of a particle, V_p its interior and $\boldsymbol{n}$ the unit outward normal to $\mathcal{S}$.

The integral representation of equations (2.6)–(2.8), with a view to establishing a boundary integral equation, is explored in the next section.

2.2 Integral representation

The integral form of the Stokes equations has been known for over a century (Lorentz 1897). From that time their main use came in areas of mathematical analysis where studies of existence, uniqueness and later singularity distributions were the primary focus (Odqvist 1930; Oseen 1927; Ladyzhenskaya 1963).

It was not until much later when Youngren and Acrivos (1975) applied the boundary integral

equation to the resistance problem of finding the traction for a spheroid and cylinder. By modern standards their methods were crude, but the application was original and innovative. Reports that the first kind equation did not exhibit the ill-conditioning often associated with weakly singular kernels instigated much activity in applying the boundary integral equation to a plethora of axisymmetric problems that held current interest.

The integral equations for the Green's function in an unbounded domain are presented. The corresponding equations for other singularity representations (for example flows with induced periodic motion or rotation, for flows in regions with periodic structures or in half-spaces or bounded by cylinders or parallel planes, for two dimensional regions and many other flow environments) we refer to Hasimoto (1959), Blake and Chwang (1974), Chwang and Wu (1974), Chwang and Wu (1975), Chwang (1975), Liron and Mochon (1976b), Liron and Shahar (1978), Dorrepaal *et al.* (1984), Huang and Chwang (1986), Pozrikidis (1988), Gavze (1990) and Loewenberg (199
An excellent account is given in the book by Pozrikidis (1992).

The linearity of the governing equations (2.6)-(2.7) suggests a super-positioning approach which results in an integral over the surface $\mathcal{S}$. This representation states that the velocity field may be represented by a distribution of singularities over the surface of each body. For the exterior problem we have

$$c(\boldsymbol{x})u_k(\boldsymbol{x}) = -\int_{\mathcal{S}} G_{ik}(\boldsymbol{x}-\boldsymbol{y})f_i(\boldsymbol{y})\,dS(\boldsymbol{y}) - \int_{\mathcal{S}} \Sigma_{ijk}(\boldsymbol{x}-\boldsymbol{y})u_i(\boldsymbol{y})n_j(\boldsymbol{y})\,dS(\boldsymbol{y}), \tag{2.9}$$

where

$$c(\boldsymbol{x}) = \begin{cases} 1, & \boldsymbol{x} \in \Omega, \\ \frac{1}{2}, & \boldsymbol{x} \in \mathcal{S}, \\ 0, & \boldsymbol{x} \in V_p, \end{cases} \tag{2.10}$$

and the usual summation convention has been adopted. The singularities are the Stokeslet

$$G_{ik}(\boldsymbol{x}) = \frac{1}{8\pi}\left(\frac{\delta_{ik}}{|\boldsymbol{x}|} + \frac{x_i x_k}{|\boldsymbol{x}|^3}\right), \tag{2.11}$$

and the Stokes-stresslet

$$\Sigma_{ijk}(\boldsymbol{x}) = -\frac{3}{4\pi}\frac{x_i x_j x_k}{|\boldsymbol{x}|^5}. \tag{2.12}$$

The Stokeslet represents the velocity of the fluid at $\boldsymbol{x}$ in the i^{th} direction due to a unit point force at the origin in the direction k. The stresslet is the analogous singularity for the stress tensor.

For completeness the derivation of (2.9) is included in appendix A.

A key advantage of using (2.9) over the finite element approach is the reduction in the dimension of the problem and its efficacy in dealing with unbounded domains. The price paid is reflected in the singularity of the integrands. In many previous studies, the singular nature of the integrals were either ignored or crudely dealt with. We expand on this point in the next section and in appendix D. More sophisticated methods to evaluate the integrals are developed and implemented in chapters 4 and 5.

2.2.1 Rigid body

If the surface $\mathcal{S}$ is rigid, then the usual no-slip condition applies

$$\boldsymbol{u}(\boldsymbol{x}) = -\boldsymbol{V} \qquad \text{on} \qquad \boldsymbol{x} \in \mathcal{S}. \tag{2.13}$$

Evaluating (2.9) at the surface, substituting (2.13) and applying the divergence theorem on the second integral will result in a first kind integral equation

$$\boldsymbol{V} = \int_{\mathcal{S}} \boldsymbol{G}(\boldsymbol{x}-\boldsymbol{y}) \cdot \boldsymbol{f}(\boldsymbol{y})\, dS(\boldsymbol{y}) \qquad \text{for} \qquad \boldsymbol{x} \in \mathcal{S}, \tag{2.14}$$

in the unknown traction $\boldsymbol{f}$. Again, see appendix A for its derivation.

Axisymmetric integral representation

If one assumes that the particle is axially symmetric and is translating in the direction of its symmetry axis, then the flow field is also axi-symmetric, and with the traction, is independent of the azimuthal angle in a cylindrical polar coordinate system. That is, we have

$$\boldsymbol{f} = (f_2 \cos\theta, f_2 \sin\theta, f_1), \qquad \text{for} \qquad \theta \in [0, 2\pi], \tag{2.15}$$

where f_1 and f_2 are the components of the traction in the axial and radial directions respectively.

Assuming, without loss of generality, that $\mathcal{S}$ has the parametric representation

$$x = r(t)\cos\theta, \quad y = r(t)\sin\theta, \quad z = z(t), \qquad \text{for } a \le t \le b,\ 0 \le \theta \le 2\pi,$$

and choosing $\boldsymbol{x}$ to lie on the $\theta = 0$ plane results in the following form for $\boldsymbol{x}$ and $\boldsymbol{y}$

$$\boldsymbol{y}(t) = (r(t)\cos\theta, r(t)\sin\theta, z(t)), \qquad \text{and} \qquad \boldsymbol{x}(t_0) = (r(t_0), 0, z(t_0)) = (r_0, 0, z_0). \tag{2.16}$$

Hence

$$|\boldsymbol{x}-\boldsymbol{y}|^2(t) = \frac{4r(t)r_0}{k(t)}\Big(1-k(t)\cos^2(\theta/2)\Big), \qquad \text{where} \qquad k(t) = \frac{4r(t)r_0}{(r_0+r(t))^2+(z_0-z(t))^2}. \tag{2.17}$$

Substituting (2.15), (2.16) and (2.17) into (2.14) and integrating analytically in the θ parameter produces a linearly coupled set of one-dimensional integral equations. The azimuthal integration gives rise to complete elliptic integrals of the first and second kind and we refer to appendix D, section D.1 for the details. The equations are

$$\begin{aligned} V = \frac{1}{8\pi\sqrt{r_0}}\int_a^b \Bigg\{ &\frac{2\sqrt{rk}}{(r_0-r)^2+(z_0-z)^2}\Big[\big((r_0-r)^2+(z_0-z)^2\big)F(k)+(z_0-z)^2E(k)\Big]f_1 \\ &+\frac{z_0-z}{(r_0-r)^2+(z_0-z)^2}\sqrt{\frac{k}{r}}\Big[\big(r_0^2-r^2+(z_0-z)^2\big)E(k)- \\ &\big((r_0-r)^2+(z_0-z)^2\big)F(k)\Big]f_2\Bigg\}\sqrt{r'^2+z'^2}\,dt \end{aligned} \tag{2.18}$$

and

$$\begin{aligned} 0 = \int_a^b \Bigg\{ &\frac{\sqrt{rk}(z_0-z)}{(r_0-r)^2+(z_0-z)^2}\Big[\big(r^2-r_0^2+(z_0-z)^2\big)E(k)-\big((r_0-r)^2+(z_0-z)^2\big)F(k)\Big]f_1 \\ &+\sqrt{\frac{k}{r}}\frac{1}{(r_0-r)^2+(z_0-z)^2}\Big[\big(2(z_0-z)^4+3(z_0-z)^2(r_0^2+r^2)+(r_0^2-r^2)^2\big)E(k) \\ &-\big((r_0-r)^2+(z_0-z)^2\big)\big(r_0^2+r^2+2(z_0-z)^2\big)F(k)\Big]f_2\Bigg\}\sqrt{r'^2+z'^2}\,dt, \end{aligned} \tag{2.19}$$

where $F(k)$ and $E(k)$ are the complete elliptic integrals of the first and second kind, respectively.

As $k \to 1$, that is, as $\boldsymbol{y} \to \boldsymbol{x}$ the integrals exhibit a logarithmic singularity. This is because

$$F(k) = \frac{1}{2}\ln\left(\frac{16}{1-k}\right)+\left(\frac{1}{2}\ln\left(\frac{16}{1-k}\right)-1\right)\frac{1-k}{4}+O\big((1-k)^2\big) \tag{2.20}$$

and

$$E(k) = 1+\left(\ln\left(\frac{16}{1-k}\right)-1\right)\frac{1-k}{4}+O\big((1-k)^2\big). \tag{2.21}$$

Approaching the singular point, it is easy to see that E remains bounded while F becomes unbounded. Hence, the coefficient of f_1 in (2.18) and the coefficient of f_2 in (2.19) become unbounded while the other two coefficients are always bounded.

Accurate evaluation of the logarithmic singularity is crucial to the success of any numerical method. A technique in wide spread use to evaluate the unbounded integrals is to define a small

region about the singularity and, using a low order expansion, integrate analytically. All other integrals are then computed via Simpson's rule or a low order Gauss-Legendre rule (Youngren and Acrivos 197 Lee and Leal 1982; Leal and Lee 1982; Koh and Leal 1989; Stone and Leal 1989; Pozrikidis 1990b; Tjahjadi *et al*. 1992; Pozrikidis 1997). In appendix D, section D.2 we argue that proper use of weighted Gaussian quadrature is much more accurate than series expansions. To facilitate their use, we develop a polynomial approximation to the complete elliptic integrals and illustrate their application.

2.2.2 Free surface

If $\mathcal{S}$ is a liquid drop then we need to consider the interior as well as the exterior flow. If we assume that the viscosity of the interior fluid is $\lambda\mu$ and the viscosity of the exterior fluid is μ then the governing equations are

$$\nabla p = \nabla^2 \boldsymbol{u}, \qquad \nabla \cdot \boldsymbol{u} = 0, \tag{2.22}$$

$$\nabla \hat{p} = \lambda \nabla^2 \hat{\boldsymbol{u}}, \qquad \nabla \cdot \hat{\boldsymbol{u}} = 0, \tag{2.23}$$

where $(\boldsymbol{u}, p)$ are the velocity and pressure fields in the exterior fluid (Ω), and $(\hat{\boldsymbol{u}}, \hat{p})$ are the velocity and pressure fields in the interior fluid (V_p).

The boundary conditions are that the velocity and tangential stress are continuous across the interface, and the normal stress at the interface is balanced by a surface tension force. The first condition gives

$$\boldsymbol{u} = \hat{\boldsymbol{u}}, \qquad \boldsymbol{x} \in \mathcal{S}. \tag{2.24}$$

In dimensional variables, we can write the remaining conditions as

$$\Delta \boldsymbol{f}^* = \left(\boldsymbol{\tau}^*_{\text{(outside)}} - \boldsymbol{\tau}^*_{\text{(inside)}}\right) \cdot \boldsymbol{n}, \tag{2.25}$$

where $\Delta \boldsymbol{f}^*$ is the jump in stress across the interface, $\boldsymbol{n}$ the unit normal vector to the drop surface, pointing from the interior fluid to the exterior fluid and $\boldsymbol{\tau}^*$ the modified stress tensor defined as

$$\boldsymbol{\tau}^* = -p^* \delta_{ij} + \frac{\partial u_i^*}{\partial x_j^*} + \frac{\partial u_j^*}{\partial x_i^*}.$$

We note that p^* here is the modified pressure since it has the term $\rho g z^*$ added to account for the effect of gravity. Applying the stress condition to (2.25) gives

$$\Delta \boldsymbol{f}^* = \left(\Delta \rho g z^* + \gamma \nabla^* \cdot \boldsymbol{n}\right) \boldsymbol{n}, \tag{2.26}$$

where $\Delta\rho g z^*$ is the buoyancy term subtracted from the modified pressure and γ the uniform surface tension. Letting a be a characteristic length scale and using the scaling

$$z = \frac{z^*}{a} \qquad \text{and} \qquad \boldsymbol{f} = \frac{\boldsymbol{f}^*}{ga\Delta\rho},$$

produces the non-dimensional stress condition

$$\Delta\boldsymbol{f} = z\boldsymbol{n} + \Gamma(\nabla\cdot\boldsymbol{n})\boldsymbol{n}, \qquad \boldsymbol{x}\in\mathcal{S} \tag{2.27}$$

where

$$\Gamma = \frac{\gamma}{ga^2\Delta\rho},$$

is the inverse Bond number. Details may be readily found in Manga and Stone (1993) and Pozrikidis (1992, p. 147)

Using an approach similar to that in appendix A, the integral equation for the interior flow is

$$c^*(\boldsymbol{x})\hat{u}_k(\boldsymbol{x}) = \frac{1}{\lambda}\int_{\mathcal{S}} G_{ik}(\boldsymbol{x}-\boldsymbol{y})f_i(\boldsymbol{y})\,dS(\boldsymbol{y}) + \int_{\mathcal{S}} \Sigma_{ijk}(\boldsymbol{x}-\boldsymbol{y})\hat{u}_i(\boldsymbol{y})n_j(\boldsymbol{y})\,dS(\boldsymbol{y}), \tag{2.28}$$

where

$$c^*(\boldsymbol{x}) = \begin{cases} 0, & \boldsymbol{x}\in\Omega, \\ \frac{1}{2}. & \boldsymbol{x}\in\mathcal{S}, \\ 1, & \boldsymbol{x}\in V_p. \end{cases} \tag{2.29}$$

Adding (2.9) and (2.28), evaluating at the surface and using the boundary conditions (2.24) and (2.27) results in an integral equation of the second kind

$$u_k(\boldsymbol{x}) = \frac{2}{1+\lambda}\int_{\mathcal{S}} G_{ik}(\boldsymbol{x}-\boldsymbol{y})\Delta f_i(\boldsymbol{y})\,dS(\boldsymbol{y}) \\ - 2\left(\frac{1-\lambda}{1+\lambda}\right)\int_{\mathcal{S}} \Sigma_{ijk}(\boldsymbol{x}-\boldsymbol{y})u_i(\boldsymbol{y})n_j(\boldsymbol{y})\,dS(\boldsymbol{y}) \tag{2.30}$$

in the unknown velocity $\boldsymbol{u}$.

Inspection of (2.12) reveals that the double layer term has an $O(1/r^2)$ type singularity where $\boldsymbol{r} = \boldsymbol{x}-\boldsymbol{y}$. In an axisymmetric regime, this singularity reduces to a $\log r/r$ type. The integral exists in a Cauchy principal value sense and integration of this term, as well as the numerical method used to solve (2.30) are given in chapter 5.

In the next chapter, two potential solution techniques for solving the axisymmetric boundary integral equations are investigated. In order to reduce the complexity of the analysis and to

simplify the identification of key numerical effects, we substitute Symm's integral equation instead of equations (2.18)-(2.19). Symm's equation still exhibits some of the main features of the equations described here - it is a Fredholm integral equation of the first kind with a logarithmic kernel.

Chapter 3

Numerical Investigation of Symm's Integral Equation

3.1 Introduction

A great many problems in the fields of engineering and science can be modelled with partial differential equations. If these problems involve some well defined region, Ω, with known conditions on the boundary Γ, then the equations can be transformed into a Fredholm integral equation of the first or second kind. In this chapter we will consider equations of the first kind

$$g(\boldsymbol{y}) = \int_{\Gamma} K(\boldsymbol{x} - \boldsymbol{y}) f(\boldsymbol{x})\, d\Gamma(\boldsymbol{x}), \tag{3.1}$$

where g represents some known data at the point $\boldsymbol{y} \in \Gamma$ and K is a distribution of source (or sink) terms over the boundary. We seek to find the density of this distribution, f.

The integral equation (3.1) is inherently ill-posed. That is, it can be shown that a small perturbation on g can give rise to an arbitrarily large perturbation in f. To substantiate this point, consider the singular integral

$$\int_0^1 \ln|x - y| e^{inx}\, dx = \frac{i}{n}\left(\ln y - e^{in}\ln(1-y)\right) - \frac{\pi}{n} e^{iny} + O\left(\frac{1}{n^2}\right). \tag{3.2}$$

For large n, infinitesimal changes for the integral (right hand side) correspond to finite changes in the integrand. The problem is compounded if instead we consider

$$\int_0^1 \ln|x - y| n^{\alpha} e^{inx}\, dx = i n^{\alpha-1}\left(\ln y - e^{in}\ln(1-y)\right) - \pi n^{\alpha-1} e^{iny} + O\left(n^{\alpha-2}\right). \tag{3.3}$$

For $0 < \alpha < 1$ and n large, then infinitely small changes for the integral correspond to infinitely large changes in the integrand. Even though (3.2) is within a constant multiple of (3.3), as n becomes large we can see that small errors in g may lead to finite or infinite *errors* in f.

For this reason, numerical methods for solving such equations are often ill-fated. The simple illustration here shows this is often manifested in high frequency terms for the unknown.

Numerical errors are introduced in one of four ways: (i) errors in approximating the surface Γ, (ii) errors in the approximation scheme for the unknown f, (iii) errors in the numerical integration of the kernel and (iv) round off errors. Errors that originate from (i) and (iii) are due to the nature of the model (K) and specific problem (Γ). That of (iv) results from finite machine precision and is a concern when the surface is over discretized or the unknown is expanded beyond machine limit. Thus, in an attempt to compare different methods to solve (3.1), we limit the boundary to the unit interval and specify a logarithmic kernel. In addition, we impose the condition that f has zero derivative at the endpoints. The kernel and condition on f were chosen since these characteristics are present in most problems which exhibit axial symmetry.

We implement and compare two numerical methods to solve Symm's integral equation

$$g(y) = \int_0^1 \ln|x-y| f(x)\,dx, \qquad 0 \le y \le 1, \tag{3.4}$$

on the unit interval. In the specific example used here we have

$$g(y) = \frac{1}{15}\,y^3\left(3\,y^2 - 15\,y + 20\right)\ln(y) + \frac{1}{15}\,(1-y)\left(3\,y^4 - 12\,y^3 + 8\,y^2 + 8\,y + 8\right)\ln(1-y) + \frac{9}{10}\,y^3 - \frac{1}{5}\,y^4 - \frac{9}{10}\,y^2 - \frac{23}{60}\,y - \frac{211}{900}.$$

The exact solution is

$$f(x) = x^2(x-2)^2, \qquad 0 \le x \le 1. \tag{3.5}$$

The first method, in section 3.2, uses a spectral approach where the unknown is represented as a finite sum of orthogonal functions. That is,

$$f(x) = \sum_{n=0}^{N} \alpha_n \varphi_n(x), \qquad 0 \le x \le 1, \tag{3.6}$$

where $\{\varphi_n\}$ is the orthogonal basis of some inner product space, in which f is a member, and

$$\alpha_n = \frac{\langle f, \varphi_n \rangle}{\langle \varphi_n, \varphi_n \rangle}, \qquad n = 0, 1, \ldots \tag{3.7}$$

are the Euler equations that define the coefficients of the expansion. The method calculates these coefficients and the resulting approximation is thus furnished. There are a few advantages and disadvantages with this approach:

- the approximation is valid for the entire domain as no discretization is employed.
- the properties of the solution are derived from those of the basis functions $\{\varphi_n\}$. For orthogonal polynomials, and in particular trigonometric polynomials, this would mean that the solution is everywhere analytic. This may not be expedient if one knows that the solution should exhibit some non-analytic behaviour. Usually, this property is desirable and non-analytic functions can be catered for by expanding in a weighted series

  $$f(x) = w(x) \sum_{n=0}^{N} \alpha_n \varphi_n(x).$$

  In this case the basis, $\{\varphi_n\}$, would be orthogonal with respect to the weight w. A Maple routine to calculate the orthonormal polynomials, up to any order, given any integrable positive weight, on any domain is given in Appendix E.
- as shown earlier, the higher order coefficients are usually difficult to obtain. It is a well known property of orthogonal polynomials that they possess linear roots (Atkinson 1989, p. 213) and (Abramowitz and Stegun 1965, p. 787). Hence, for large n, these functions are highly oscillatory and so inverting first kind equations can become difficult. In addition, there are complications in the numerical integration of integrands that are highly oscillatory and singular. For this reason the use of Shanks' transform to increase the accuracy of low order approximations is explored in sub-section 3.2.2.

We remark that the Galerkin method (Delves and Mohamed 1985) is another expansion approach. It involves expanding the unknown in an orthogonal series, as in (3.6) and substituting into the integral equation. A linear system is formed by applying the inner product of each basis function with the resulting equation. It has been used with much success in solving integral equations of the second kind with regular kernels. The method will not be considered here since (3.4) is an equation of the first kind with a singular kernel. The Galerkin method has been augmented to account for these features, but as a result is much more difficult to implement and requires two dimensional weighted cubature formulae. It will not be pursued further here.

In section 3.3 the second method explored involves discretizing the boundary into n elements

$$0 = x_0 < x_1 < \cdots < x_{n-1} < x_n = 1$$

and approximating f using some interpolatory function in each interval. Previous methods for Stokes flow have used constant and linear approximations (Youngren and Acrivos 1975; Leal and Lee 1982; Lee and Leal 1982; Geller *et al.* 1986; Chi and Leal 1989; Pozrikidis 1989; Tjahjadi *et al.* 1992; Pozrikidis 1997). Pozrikidis (1990b, 1990a) uses quadratic elements. This contrasts with the methods in chapters 4 and 5 where, for rigid body problems, Hermite polynomials of arbitrary order are used and for free surfaces both the surface and the unknown velocity are approximated using cubic splines.

To gauge the effect of a cubic approximation, section 3.3 investigates the use of the Hermite cubic interpolation polynomials. That is,

$$f(x) \approx f(x_{i-1})H_1\left(\frac{x-x_{i-1}}{x_i-x_{i-1}}\right) + (x_i-x_{i-1})f'(x_{i-1})H_2\left(\frac{x-x_{i-1}}{x_i-x_{i-1}}\right) + f(x_i)H_3\left(\frac{x-x_{i-1}}{x_i-x_{i-1}}\right) + (x_i-x_{i-1})f'(x_i)H_4\left(\frac{x-x_{i-1}}{x_i-x_{i-1}}\right), \quad (3.8)$$

for $x_{i-1} \le x \le x_i$, $i = 1, 2, \ldots, n$. The properties of Hermite interpolation polynomials, as well as the notation used, is outlined in Appendix B. These interpolation polynomials have compact support and thus the behaviour of f in any one region has little effect on the solution in other regions. The solution thus obtained is continuous and differentiable.

Unlike the spectral approach, where analyticity is guaranteed for any order approximation, we would require higher order Hermite polynomials for more smoothness. It is interesting to note that although cubic polynomials are employed, second derivative continuity is not guaranteed. This would only be the case with the cubic spline, which has global support. Unfortunately, a cubic spline method is difficult to implement since the nodal behaviour of f would be required *a priori*. Such a method would be attractive since half the storage and number of integrations would be needed when compared to the Hermite approximation. A method, based on a conjunction of Fortran and Maple, has been developed by the author (Roumeliotis 1991), but currently only works for small n and has the drawback of not being adaptable to dynamic grids.

Examination of (3.8) reveals that for large n, the approximation breaks down. We investigate how large n can be as well as explore the role of the collocation points in determining the accuracy of the solution.

For a detailed exposition of boundary integral equation methods we refer to the work of Lucas (1993).

3.2 Spectral Method

Boundary integral problems that involve closed boundaries must have periodic solutions. This inherent periodicity has been previously exploited to construct a solution scheme that employs the trapezoidal rule (Fenton 1994) which is extremely accurate for periodic functions. Here we choose to exploit the periodicity only to the extent that the unknown is expanded in a Fourier series.

Thus, we present a numerical method for solving a linear Fredholm integral equation of the first kind of the form

$$g(y) = \int_0^L K(x,y) f(x)\, dx, \qquad 0 \le y \le L, \tag{3.9}$$

where it will be assumed that the function f is even and periodic

$$f(x) = f(-x), \qquad -L \le x \le L. \tag{3.10}$$

This condition enables f to be expanded into a Fourier cosine series

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos \frac{n\pi x}{L}, \tag{3.11}$$

where

$$a_n = \frac{2}{L} \int_0^L f(x) \cos \frac{n\pi x}{L}\, dx. \tag{3.12}$$

Equation (3.10) is imposed merely as a matter of convenience. Without it, equation (3.11) would be changed so that $f(x)$ is expanded into its full Fourier series. We make the obvious assumption that (3.11) converges almost everywhere to $f(x)$. In order to evaluate the Fourier coefficients (3.12), the function must be integrable. This condition can be relaxed somewhat and other Euler equations have been defined for more general classes of functions. We refer to Kubota (1973) and the references therein for the specific details.

In what follows we will refer to the Fourier cosine series (3.11) as the Fourier series.

Substituting equation (3.11) into equation (3.9) gives

$$g(y) = \frac{a_0}{2} B_0(y) + \sum_{n=1}^{\infty} a_n B_n(y), \tag{3.13}$$

where

$$B_n(y) = \int_0^L K(x,y)\cos\frac{n\pi x}{L}\,dx. \tag{3.14}$$

According to the well known Riemann-Lebesgue theorem, if $K(\cdot, y)$ and f are absolutely integrable, then

$$B_n(y) \to 0 \quad \text{and} \quad a_n \to 0 \qquad \text{as} \qquad n \to \infty. \tag{3.15}$$

Thus the high frequency terms may be neglected from (3.11), and we can approximate $f(x)$ by

$$f_N^*(x) = \frac{a_0^*}{2} + \sum_{n=1}^{N} a_n^* \cos\frac{n\pi x}{L}, \tag{3.16}$$

for some $a_0^*, a_1^*, \ldots, a_N^*$, $N > 0$. Substituting (3.16) into (3.9), we obtain

$$g(y) = \frac{a_0^*}{2}B_0(y) + \sum_{n=1}^{N} a_n^* B_n(y). \tag{3.17}$$

To obtain the Fourier coefficients, $N+1$ collocation points

$$0 \le y_0 < y_1 < \cdots < y_{N-1} < y_N \le L, \tag{3.18}$$

are used to produce the $N+1$ equations

$$g(y_j) = \frac{a_0^*}{2}A_{0,j} + \sum_{n=1}^{N} a_n^* A_{n,j}, \qquad \text{where} \qquad A_{n,j} = B_n(y_j), \qquad j = 0, 1, \ldots, N. \tag{3.19}$$

The corresponding matrix equation is

$$\mathcal{A}\mathbf{a} = \mathbf{g}, \tag{3.20}$$

where

$$\mathcal{A} = \begin{bmatrix} \frac{A_{0,0}}{2} & A_{1,0} & A_{2,0} & \ldots & A_{N,0} \\ \frac{A_{0,1}}{2} & A_{1,1} & A_{2,1} & \ldots & A_{N,1} \\ \vdots & & & & \\ \frac{A_{0,N}}{2} & A_{1,N} & A_{2,N} & \ldots & A_{N,N} \end{bmatrix}, \quad \mathbf{a} = \begin{bmatrix} a_0^* \\ a_1^* \\ \vdots \\ a_N^* \end{bmatrix} \quad \text{and} \quad \mathbf{g} = \begin{bmatrix} g(y_0) \\ g(y_1) \\ \vdots \\ g(y_N) \end{bmatrix}. \tag{3.21}$$

Solving (3.20) provides $a_0^*, a_1^*, \ldots, a_N^*$, which on substitution into (3.16) yields an approximation to $f(x)$.

We mention briefly that the integrals in $A_{n,j}$ were performed by first subdividing at the zeros of the integrand and then using a combination of Gauss-Legendre and Gauss-Log quadrature. That

is

$$\begin{aligned} A_{n,j} &= \int_0^L K(x, y_j) \cos \frac{n\pi x}{L}\, dx \\ &= \int_0^1 \log|x - y_j| \cos n\pi x\, dx \\ &= \int_0^{y_{j-1}} \log(y_j - x) \cos n\pi x\, dx + \int_{y_{j-1}}^{y_j} \log(y_j - x) \cos n\pi x\, dx \\ &\quad + \int_{y_j}^{y_{j+1}} \log(x - y_j) \cos n\pi x\, dx + \int_{y_{j+1}}^1 \log(x - y_j) \cos n\pi x\, dx. \end{aligned} \tag{3.22}$$

The first and last integrands in (3.22) are not singular and the integrals are evaluated using a Gauss-Legendre rule over each zero of the integrand, the middle two integrals exhibit a logarithmic singularity and they are evaluated by a combination of Gauss-Log and Gauss-Legendre rules over the zero's of the respective integrands. In this way, the logarithmic singularity and the high frequency nature of the integrands are both accounted for. Another approach would be to employ the automatic quadrature routines in QUADPACK (Piessens *et al.* 1983). QUADPACK is a collection of state-of-the-art integration routines written in Fortran. They are extremely robust and are capable of evaluating quite complicated integrals to high precision. One possible drawback in their use is that the suite necessarily compromises speed for reliability and so may not be highly suited to algorithms which call integration routines repeatedly. For example, boundary element methods which assemble matrices whose individual elements are themselves integrals. In this case, as in this dissertation, it would be more appropriate to construct tailor-made quadrature rules. QUADPACK may be obtained from the internet site `http://www.netlib.org/quadpack/`.

3.2.1 Results

The Fourier series approximation performs reasonably for small N. Figure 3.1 compares the exact solution with the approximation for $N = 2$. The error is around 2%, which is quite acceptable for this low order expansion. As expected most of the error occurs at the boundary points $x = 0$ and $x = 1$. This is typical of spectral expansions in that the endpoints have the slowest convergence rate and is due to the fact that at these points the coefficients, a_n^*, are summed with the greatest magnitude.

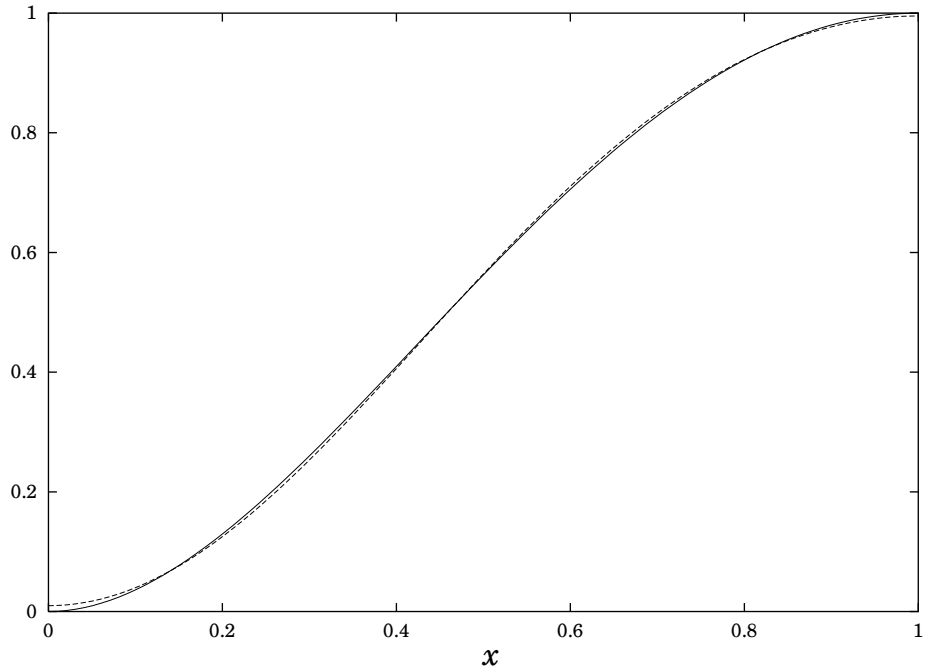


Figure 3.1: Plot of $f(x) = x^2(x-2)^2$ — and the numerical approximation $f_2^*(x)$ (3.16) - - -.

In Figures 3.2 and 3.3 we show the error in the Fourier approximation for $N = 4, 8, 16, 32, 64$ and 128. The most obvious characteristic of each graph is the increasing frequency; and in fact the number of peaks and troughs in each graph in Figure 3.2 corresponds exactly with $N + 1$. This suggests that for each N, the dominant component of the error is the last few terms of the sum.

As N doubles from $N = 4$ to $N = 32$, the error decreases by an order of magnitude (Figure 3.2). The error then begins to increase, and this is shown in Figure 3.3 for $N = 64$ and 128. Figure 3.3(b) shows that the error is largest at $x = 1$, and this coupled with the high frequency of the graph indicates a breakdown in the evaluation of the last few coefficients.

Inspection of the last few terms in each Fourier expansion reveals this breakdown. Table 3.1 shows the relative error in each Fourier coefficient for $N = 4, 8$ and 16. The error increases with n except for the last coefficient. The breakdown in the last few terms is quite large where the relative error is 28% for $N = 8$ and 53% for $N = 16$. It is clear that these terms are the cause of the large deviations shown in Figure 3.3 near $x = 1$. Figure 3.4 shows the relative error in the Fourier coefficients for $N = 4$ up to $N = 128$. The large increase in errors near $n = N$ for $N = 64$ and $N = 128$ prevent the series from realizing its previous convergence rate.

Another feature revealed in Figure 3.4 is that doubling N up to 32 increases the accuracy of

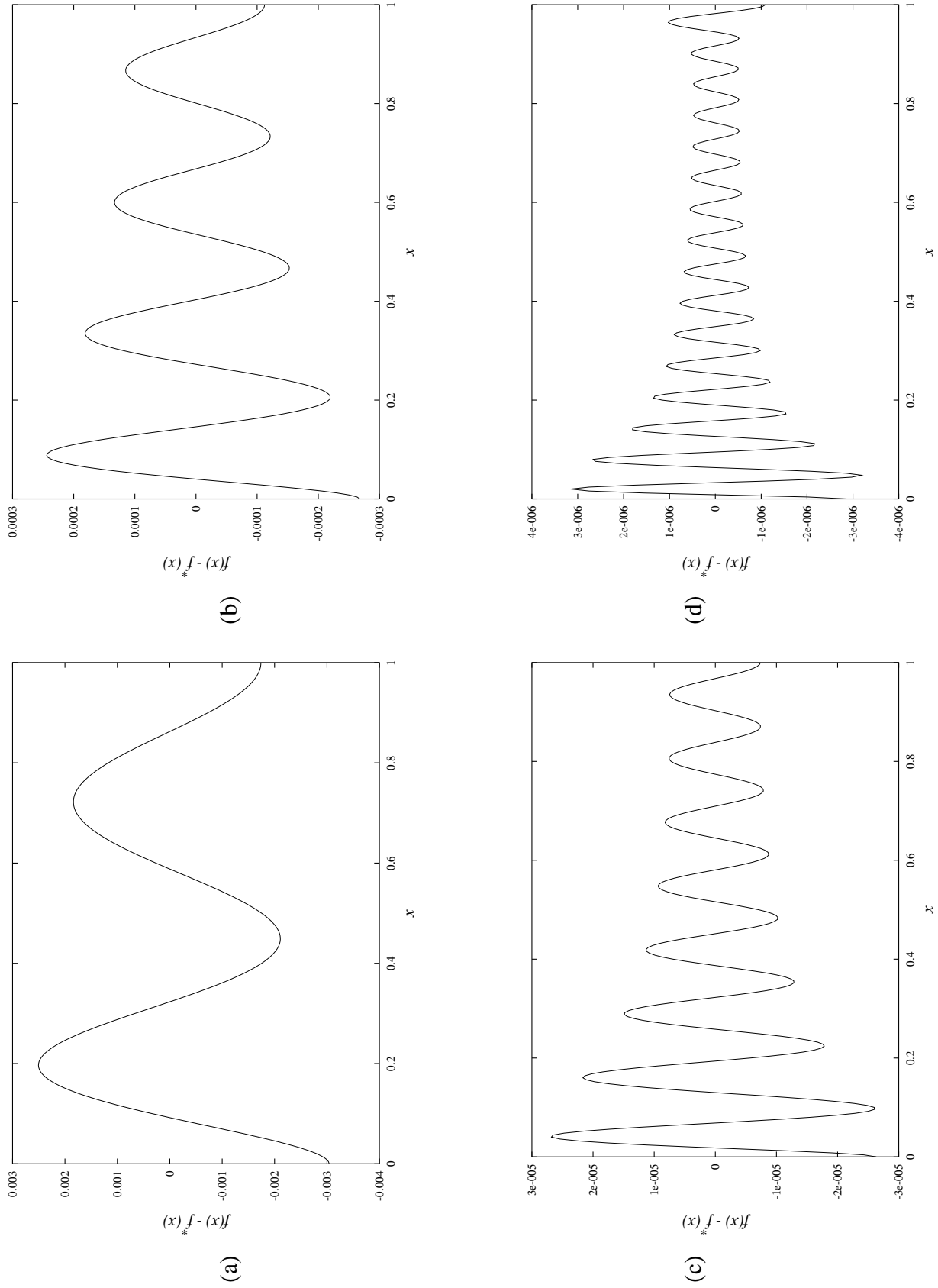


Figure 3.2: $f(x) - f_N^*(x)$ for (a) $N = 4$, (b) $N = 8$, (c) $N = 16$ and (d) $N = 32$.

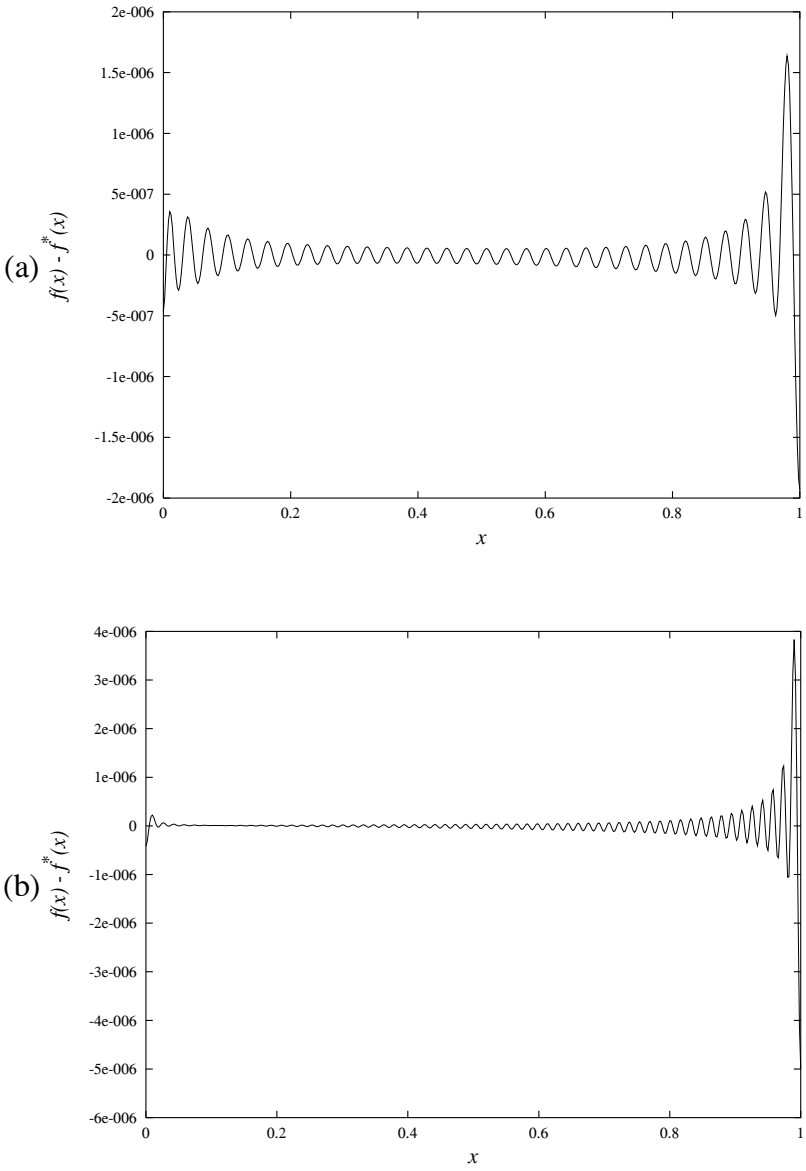


Figure 3.3: $f(x) - f_N^*(x)$ *for (a)* $N = 64$ *and (b)* $N = 128$.

n	a_n	$\left\lvert\frac{a_n - a_n^*}{a_n}\right\rvert$		
		$N = 4$	$N = 8$	$N = 16$
0	1.0666666666666667	0.2396E-04	0.1126E-05	0.5179E-07
1	-0.4927671482248482	0.7770E-04	0.2725E-05	0.1090E-06
2	-0.0307979467640530	0.4306E-02	0.8200E-04	0.3437E-05
3	-0.0060835450398129	0.7569E-01	0.6883E-03	0.1582E-04
4	-0.0019248716727533	0.1443E-02	0.4154E-02	0.8204E-04
5	-0.0007884274371598		0.1882E-01	0.2402E-03
6	-0.0003802215649883		0.7640E-01	0.6903E-03
7	-0.0002052341308725		0.2796E+00	0.1753E-02
8	-0.0001203044795471		0.3760E-02	0.3973E-02
9	-0.0000751054943187			0.9088E-02
10	-0.0000492767148225			0.1846E-01
11	-0.0000336566592599			0.3886E-01
12	-0.0000237638478118			0.7467E-01
13	-0.0000172531475867			0.1482E+00
14	-0.0000128271331795			0.2771E+00
15	-0.0000097336720637			0.5308E+00
16	-0.0000075190299717			0.7605E-02

Table 3.1: Error in the coefficients of the Fourier series for $N = 4$, $N = 8$ and $N = 16$.

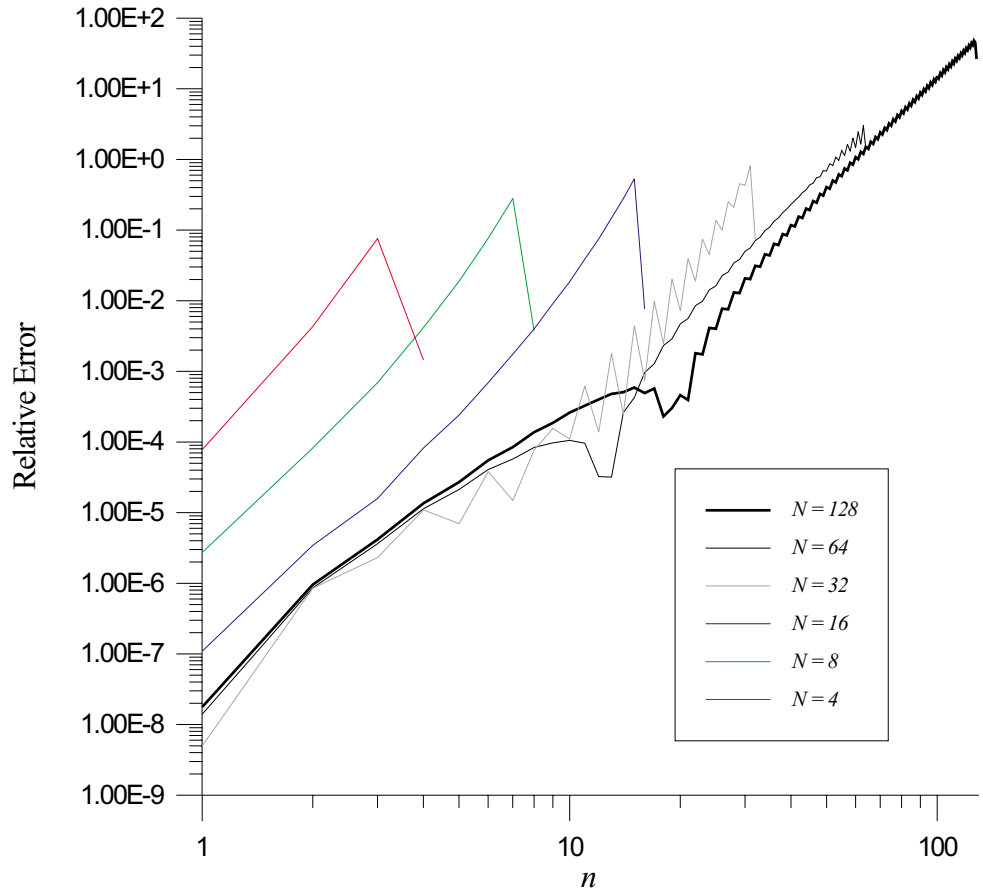


Figure 3.4: Relative error in Fourier coefficients for $N = 4, 8, 16, 32, 64, 128$.

the coefficients by an order of magnitude, but thereafter the error remains relatively static ($N = 64, 128$). In other words, with this method the uppermost accuracy is attained at approximately $N = 32$ and increasing N beyond this will not dramatically decrease the error and may even increase it. This is consistent with Figures 3.2 and 3.3.

An obvious course of action would be to calculate a large number, N, of coefficients but employ a limited number of them, $M \leq N$, in the expansion

$$f^*_{N,M} = \frac{a^*_0}{2} + \sum_{n=1}^{M} a^*_n \cos(n\pi x). \tag{3.23}$$

This would ensure that the first coefficients are obtained with high accuracy, but the last (inaccurate) few do not corrupt the series. The results are presented in Figure 3.5 where the coefficients have been calculated for $N = 64$. Excluding the last 8 coefficients (Figure 3.5(b)) slightly decreases the maximum error by a factor of 1.35. Excluding the last 16 coefficients (Figure 3.5(c)) decreases the maximum error by roughly the same amount, while omitting half the coefficients ($M = 32$, Figure 3.5(d)) more than doubles the error. It can be seen that decreasing M changes

the location of the maximum error from $x = 1$ to $x = 0$.

The calculation is repeated for $N = 128$ and the results are shown in Figure 3.6. The behaviour is qualitatively similar to $N = 64$. Excluding the last 48 coefficients (Figure 3.6(b)) increases the accuracy by more than a factor of 3, excluding the last 72 coefficients (Figure 3.6(c)) produces a result that is over 5.5 times more accurate, while omitting the last 80 coefficients (Figure 3.6(d)) is roughly 2.7 times more accurate. We note that in Figure 3.5(d) the error has increased while in Figure 3.6(d) the error still decreases. It would seem that with $N = 64, M = 32$ we have reached the obvious limit of leaving too few terms in the expansion.

To improve the expansion method, the Shanks transform is explored in the next section. This non-linear transformation has been shown to improve convergence rates for many sequences and it will be used here in an attempt to improve accuracy and avoid the high frequency terms.

3.2.2 Shanks' transform

The Fourier series (3.11) can be considered as a sequence of partial sums for each x

$$\{A_n\} = \left\{ \frac{a_0}{2}, \frac{a_0}{2} + a_1 \cos \pi x, \frac{a_0}{2} + a_1 \cos \pi x + a_2 \cos 2\pi x, \dots \right\}. \tag{3.24}$$

We attempt to speed the converge of (3.24) for each x via application of Shanks' transform.

Shanks' transform (Shanks 1955) is a set of non-linear transformations that maps one sequence to another. Given a sequence $\{A_n\}$, $n = 0, 1, 2, \dots$ and the forward difference $\Delta A_n = A_{n+1} - A_n$, then the k^{th} transform of $\{A_n\}$ defines a new sequence $\{B_{k,n}\}$, $n = k, k+1, \dots$ given by

$$B_{k,n} = \frac{\begin{vmatrix} A_{n-k} & \cdots & A_{n-1} & A_n \\ \Delta A_{n-k} & \cdots & \Delta A_{n-1} & \Delta A_n \\ \Delta A_{n-k+1} & \cdots & \Delta A_n & \Delta A_{n+1} \\ \vdots & & \vdots & \vdots \\ \Delta A_{n-1} & \cdots & & \Delta A_{n+k-1} \end{vmatrix}}{\begin{vmatrix} 1 & \cdots & 1 & 1 \\ \Delta A_{n-k} & \cdots & \Delta A_{n-1} & \Delta A_n \\ \Delta A_{n-k+1} & \cdots & \Delta A_n & \Delta A_{n+1} \\ \vdots & & \vdots & \vdots \\ \Delta A_{n-1} & \cdots & & \Delta A_{n+k-1} \end{vmatrix}} \tag{3.25}$$

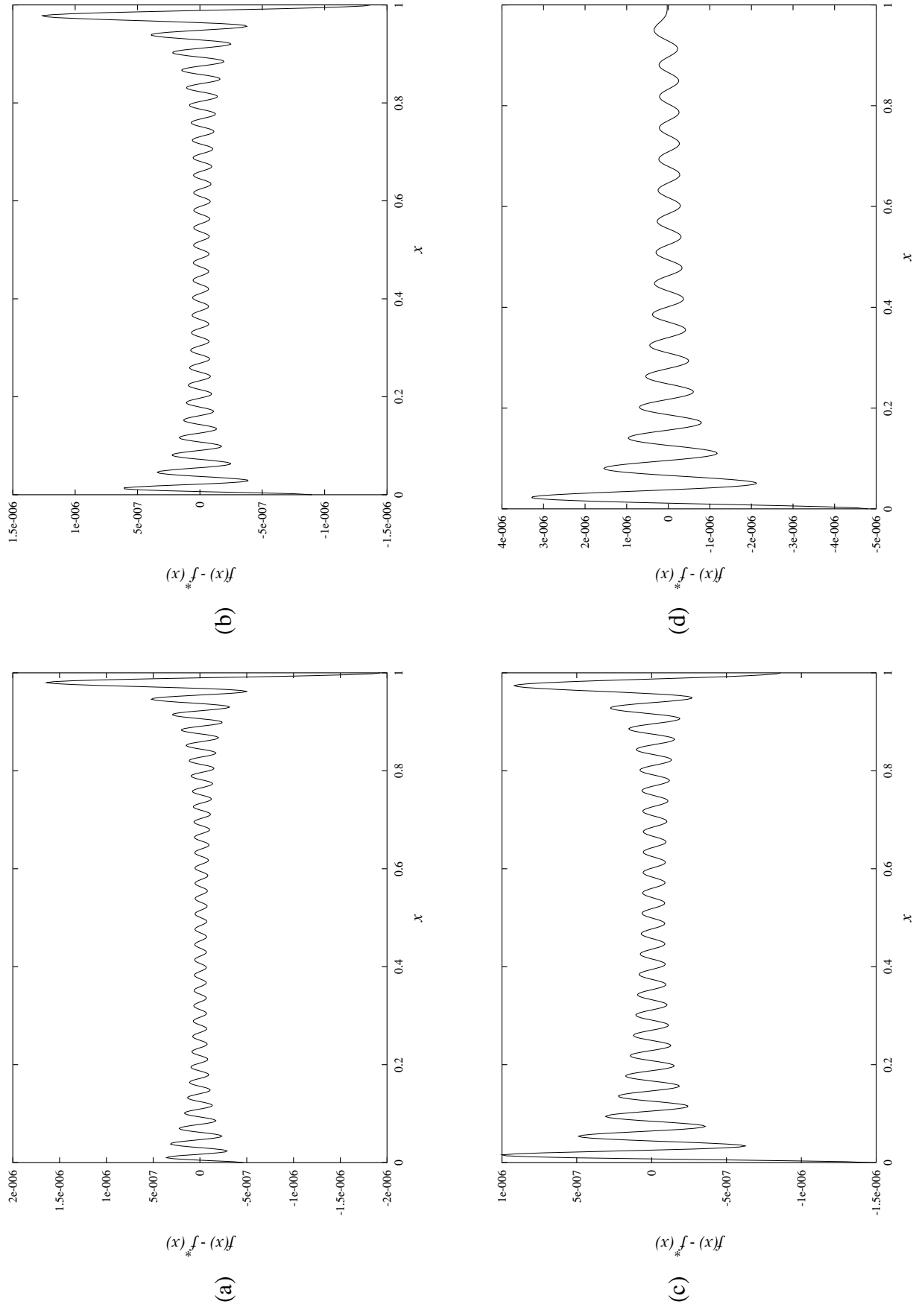


Figure 3.5: $f(x) - f^*_{64,M}(x)$ *for (a)* $M = 64$*, (b)* $M = 56$*, (c)* $M = 48$ *and (d)* $M = 32$*.*

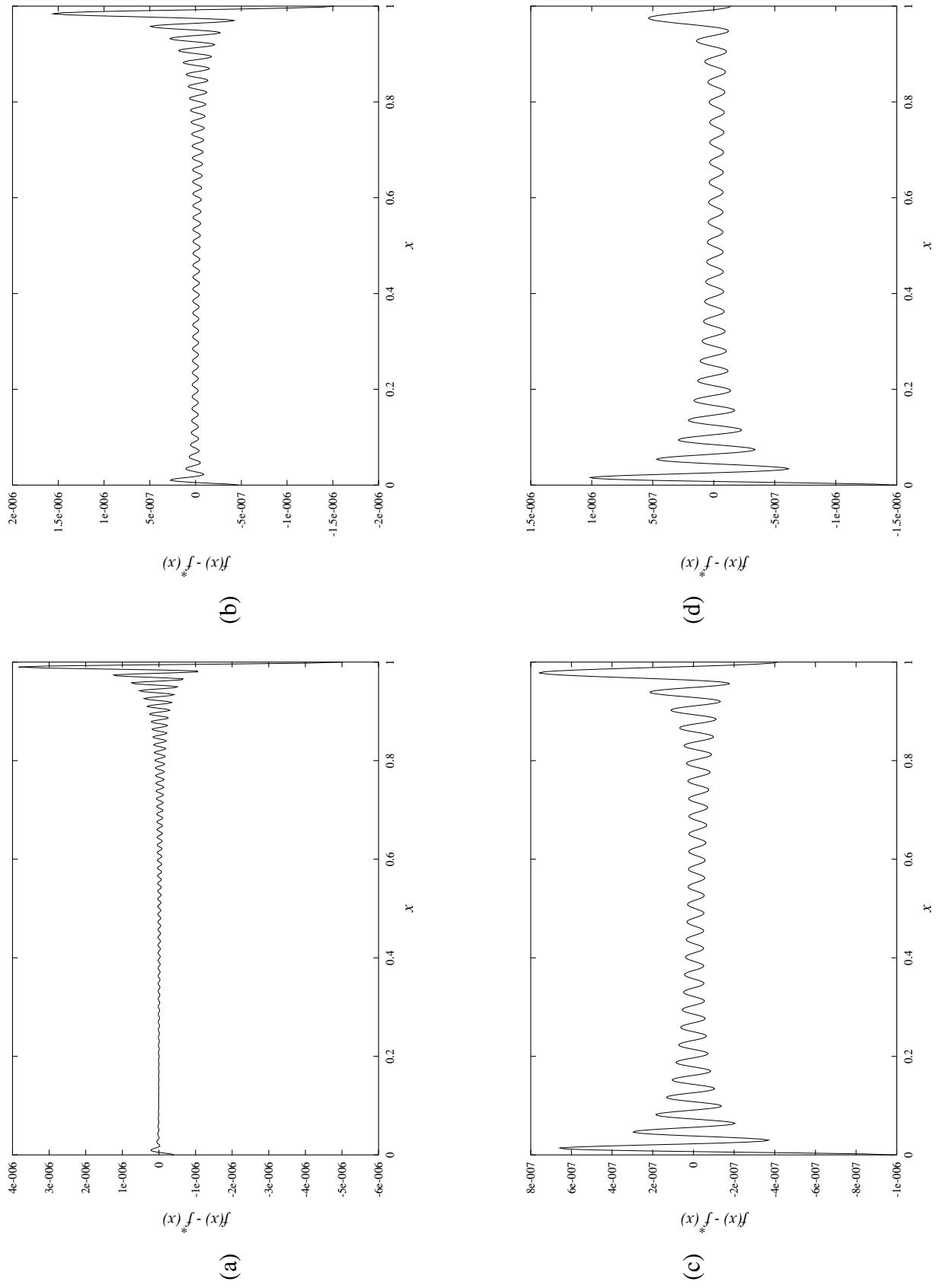


Figure 3.6: $f(x) - f^*_{128,M}(x)$ *for (a)* $M = 128$*, (b)* $M = 80$*, (c)* $M = 56$ *and (d)* $M = 48$.

whenever the denominator does not vanish. Shanks (1955) has shown that (3.25) can dramatically speed up slowly convergent sequences and even evaluate, in an *analytic continuation* sense, divergent sequences. The motivation behind the general transform is in many ways similar to the derivation of Aitken's formula. Here an estimate is made on the rate of convergence of the original sequence, and hence the error, thus providing an opportunity to enhance convergence. The sequence itself is considered a function that can be approximated by a few dominant terms of its spectrum (called *transients* by Shanks)

$$A_n \approx B + \sum_{i=1}^{k} a_i q_i^n. \tag{3.26}$$

Identifying q in terms of A and k leads to the Shanks transform (3.25) with the ultimate goal being the faster evaluation of the limit B. Inspection of (3.26) reveals that if the sequence is nearly geometric then $k = 1$ is the appropriate transformation. More complicated transformations will apply depending on how many geometric progressions are required to approximate the original sequence. We note that the form of (3.26) is quite general as most convergent (and divergent) sequences can be expressed in this way using simple techniques like decomposition into partial fractions.

Since the Shanks transform of a sequence is itself a sequence, it is of course possible to generate families of sequences. Taking this into consideration, the transform can be defined as an operator

$$B_{k,n} = e_k(A_n). \tag{3.27}$$

The strength of the transform comes from this very idea of embedding sequences to greatly enhance convergence rate. We explore two such possibilities, the iterated 1st order transform and the iterated diagonal transform. We refer to the original paper of Shanks (1955) for the theoretical details behind (3.25) and the exploration of more elaborate transformations.

The 1st order transform has a simple form and is readily recognisable as Aitken's extrapolation formula (Atkinson 1989, p. 85). Substituting $k = 1$ into (3.25) gives

$$e_1(A_n) = \frac{A_{n+1}A_{n-1} - A_n^2}{A_{n+1} + A_{n-1} - 2A_n}, \qquad n = 1, 2, \ldots.$$

Using (3.27) allows us to write the *iterated 1st order transform*. The first sequence is $B_n = e_1(A_n)$, which can itself be transformed producing a repeated set of sequences viz. $C_n =$

$e_1(B_n),\ D_n = e_1(C_n), \ldots$. In tabular form we have

$$\begin{array}{llllll} A_0 & & & & & \\ A_1 & B_1 & & & & \\ A_2 & B_2 & C_2 & & & \\ A_3 & B_3 & C_3 & D_3 & & \\ A_4 & B_4 & C_4 & \vdots & \ddots & \\ A_5 & B_5 & \vdots & & & \\ A_6 & \vdots & & & & \\ \vdots & & & & & \end{array} \tag{3.28}$$

The first term of the k^{th} iterated sequence would require $2k+1$ elements of the original sequence. It has been shown for many sequences that the top diagonal in (3.28) can converge much faster than the first column.

To include the possibility that (3.24) has other than geometric convergence we also define the *diagonal transform* $B_{n,n} = e_n(A_n)$

$$\begin{array}{llllllll} & & & & A_0 & & & \\ & & A_0 & & A_1 & & & \\ A_0 & & A_1 & & A_2 & & & \\ A_1 & B_{1,1} & A_2 & B_{2,2} & A_3 & B_{3,3} & \cdots & \\ A_2 & & A_3 & & A_4 & & & \\ & & A_4 & & A_5 & & & \\ & & & & A_6 & & & \end{array} \tag{3.29}$$

To evaluate $\{B_{n,n}\}$ would require $2n+1$ terms of the original sequence. Clearly, this itself can be iterated using (3.29). The first iterate would appear as

$$\begin{array}{llllllll} & & & & A_0 & & & \\ & & A_0 & & B_{1,1} & & & \\ A_0 & & B_{1,1} & & B_{2,2} & & & \\ B_{1,1} & C_{1,1} & B_{2,2} & C_{2,2} & B_{3,3} & C_{3,3} & \cdots & \\ B_{2,2} & & B_{3,3} & & B_{4,4} & & & \\ & & B_{4,4} & & B_{5,5} & & & \\ & & & & B_{6,6} & & & \end{array}$$

Results are shown in Tables 3.2 and 3.3 where $f(0)$ and $f(1)$ are evaluated using the Fourier series for various N. The partial sums (3.24) are transformed using (3.28) and (3.29). As we've

	Absolute Error in $f(0)$					
N	Fourier Series	Diag.	Diag. (1)	Diag. (2)	Diag. (3)	1st Shanks
4	0.176E-02	0.778E-03				0.456E-03
8	0.266E-03	0.431E-04	0.196E-04			0.302E-05
12	0.838E-04	0.627E-05	0.133E-05			0.253E-06
16	0.365E-04	0.147E-05	0.153E-06	0.715E-07		0.619E-06
20	0.190E-04	0.280E-06	0.163E-06	0.163E-06		0.126E-05
24	0.112E-04	0.172E-06	0.778E-07	0.968E-07		0.188E-06
28	0.709E-05	0.192E-06	0.177E-06	0.342E-07		0.805E-07
32	0.478E-05	0.186E-06	0.180E-06	0.715E-07	0.842E-07	0.619E-07

Table 3.2: Error in $f(0)$ for number of terms in Fourier series and different extrapolation schemes.

seen, the Fourier series is slowest to converge at these end-points and they have been chosen to gauge the effect of Shanks' transform.

For each row in Table 3.2, we present the errors in evaluating $f(0)$ using the Fourier series (column 2), the diagonal transform (3.29) (column 3) and the iterated first order transform (3.28) (column 7). In columns 4 to 6, successive diagonal transforms are applied using the previous column as the starting sequence. We see that the transforms perform extremely well and are orders of magnitude more accurate than the Fourier series. In fact the 1st order iterated transform for $N = 8$ is more accurate than the Fourier series for $N = 32$.

These results are repeated in Table 3.3 where $f(1)$ is being approximated. The results here are even more accurate than in Table 3.2 and may be due to the fact that the sequence is alternating and not monotonic. In this case the iterated first order transform is three orders of magnitude more accurate for $N = 8$ than the Fourier series for $N = 32$. The diagonal transform seems to perform well after repeated use (columns 4-6) as in the iterated 1st order transform.

When applying the transforms for each $x \in [0, 1]$ the results are quite different. Figures 3.7, 3.8, 3.9 and 3.10 show the performance of the Fourier series and the Shanks transforms for $N = 4, 8, 16$ and 32 respectively. The iterated 1st order transform performs rather poorly (Figures 3.7(a), 3.8(a), 3.9(a)). For most values of x the Fourier series is more accurate and the error considerably more regular. There are a number of sharp peaks with the transform error and they increase in frequency as N increases. The results are better for the diagonal transform (Figures 3.7(c), 3.8(c), 3.9(d), 3.10(d)). For most of the domain the diagonal transform is orders

N	Absolute Error in $f(1)$					
	Fourier Series	Diag.	Diag. (1)	Diag. (2)	Diag. (3)	1st Shanks
4	0.539E-03	0.543E-04				0.197E-04
8	0.457E-04	0.327E-07	0.290E-08			0.527E-09
12	0.993E-05	0.294E-10	0.575E-11			0.000E+00
16	0.329E-05	0.261E-08	0.634E-11	0.000E+00		0.000E+00
20	0.139E-05	0.207E-05	0.222E-11	0.000E+00		0.000E+00
24	0.681E-06	0.870E-04	0.131E-11	0.225E-10		0.000E+00
28	0.372E-06	0.156E-03	0.832E-12	0.181E-11		0.000E+00
32	0.220E-06	0.122E-03	0.374E-08	0.227E-10	0.000E+00	0.000E+00

Table 3.3: Error in $f(1)$ for number of terms in Fourier series and different extrapolation schemes.

of magnitude more accurate, but some of the sharp spikes are orders of magnitude worse. The exception is for $N = 32$ (Figure 3.10(d)) where the diagonal transform has greatly deteriorated the original sequence. The peaks increase in frequency as N increases.

The other plots show the performance of the iterated diagonal transform. At times they do not enhance the diagonal result (Figures 3.7(b), 3.8(b), 3.9(b), 3.9(c)), at others performance is markedly better (Figures 3.10(b), 3.10(c)). Overall, the irregularity of the Shanks errors leads one to conclude that the transform is not a reliable method for improving performance. For much of the domain of f the performance can be quite spectacular, but at others it is disastrous. Table 3.4 shows the maximum and average relative errors for the transforms and original Fourier series. In every case the maximum error is worse than the Fourier series and is due to the sharp peaks discussed previously. As expected, the average errors are somewhat better. For moderate values of N, the diagonal transform increases convergence of the Fourier series. Iterating does not improve performance. The iterated 1st order transform degrades performance. These results can be partly explained by simple analysis of (3.25). Substituting the sequence of partial Fourier sums (3.24) into (3.25) for $k = 1$ gives

$$B_{1,n} = \frac{\begin{vmatrix} A_{n-1} & A_n \\ \Delta A_{n-1} & \Delta A_n \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ \Delta A_{n-1} & \Delta A_n \end{vmatrix}} = \frac{A_{n-1}\Delta A_n - A_n \Delta A_{n-1}}{\Delta A_n - \Delta A_{n-1}} = \frac{A_{n-1}\Delta A_n - A_n \Delta A_{n-1}}{a_n \cos n\pi x - a_{n-1}\cos(n-1)\pi x}. \tag{3.30}$$

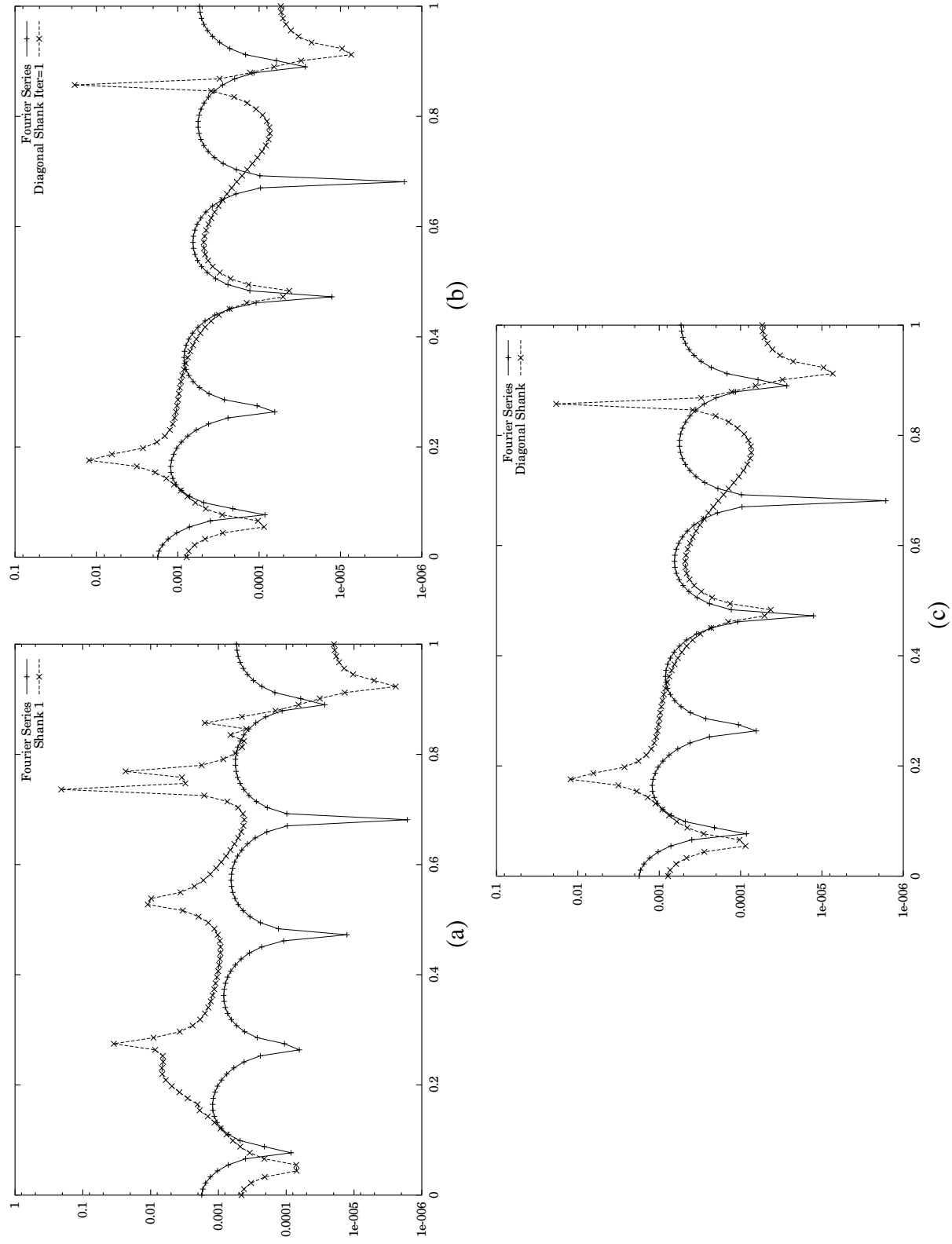


Figure 3.7: Comparison of the error of the Fourier series $f_4^*(x)$ with the (a) Shanks transform of degree 1, (b) first iteration of the diagonal Shanks transform and (c) diagonal Shanks transform.

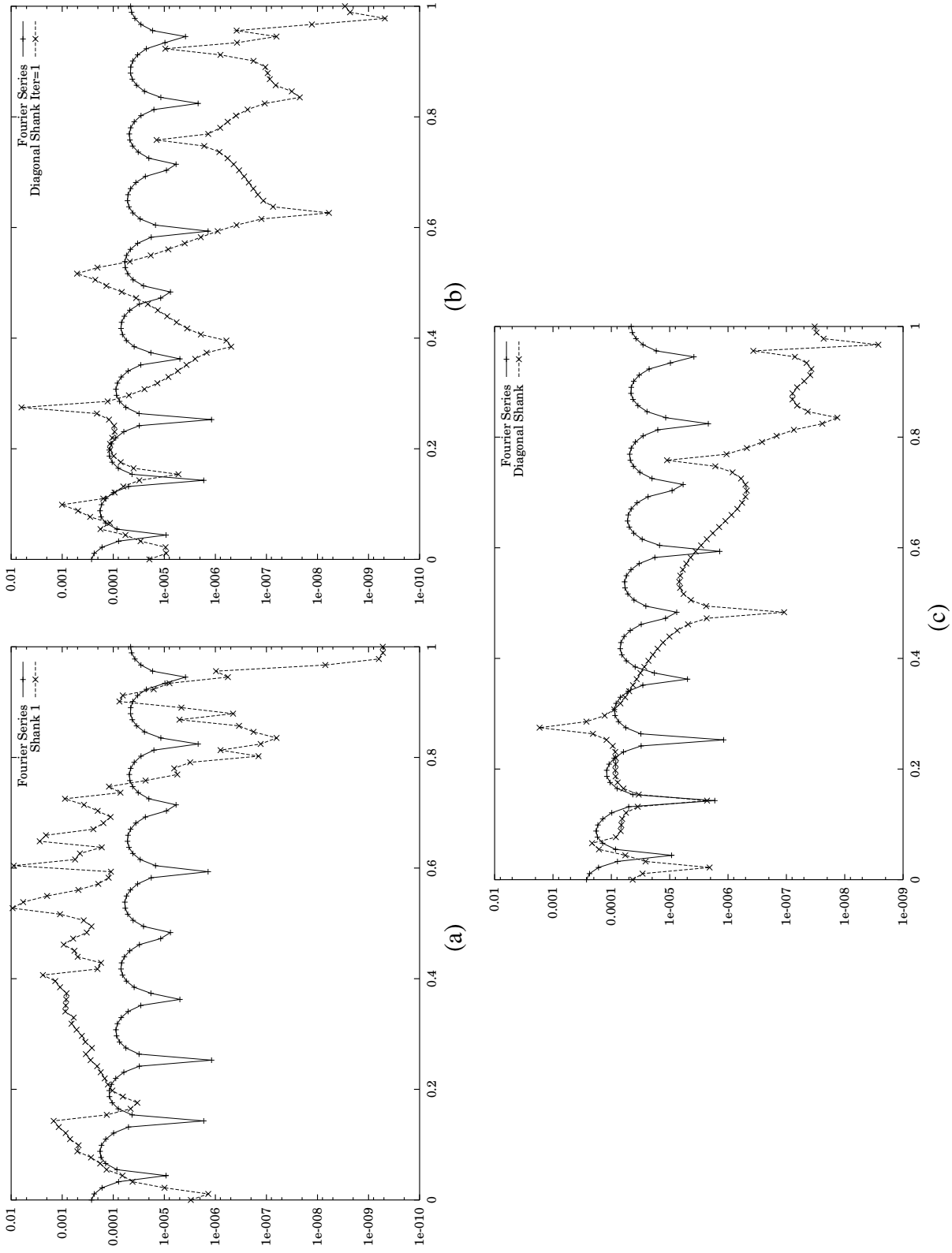


Figure 3.8: Comparison of the error of the Fourier series $f_8^*(x)$ with the (a) Shanks transform of degree 1, (b) first iteration of the diagonal Shanks transform and (c) diagonal Shanks transform.

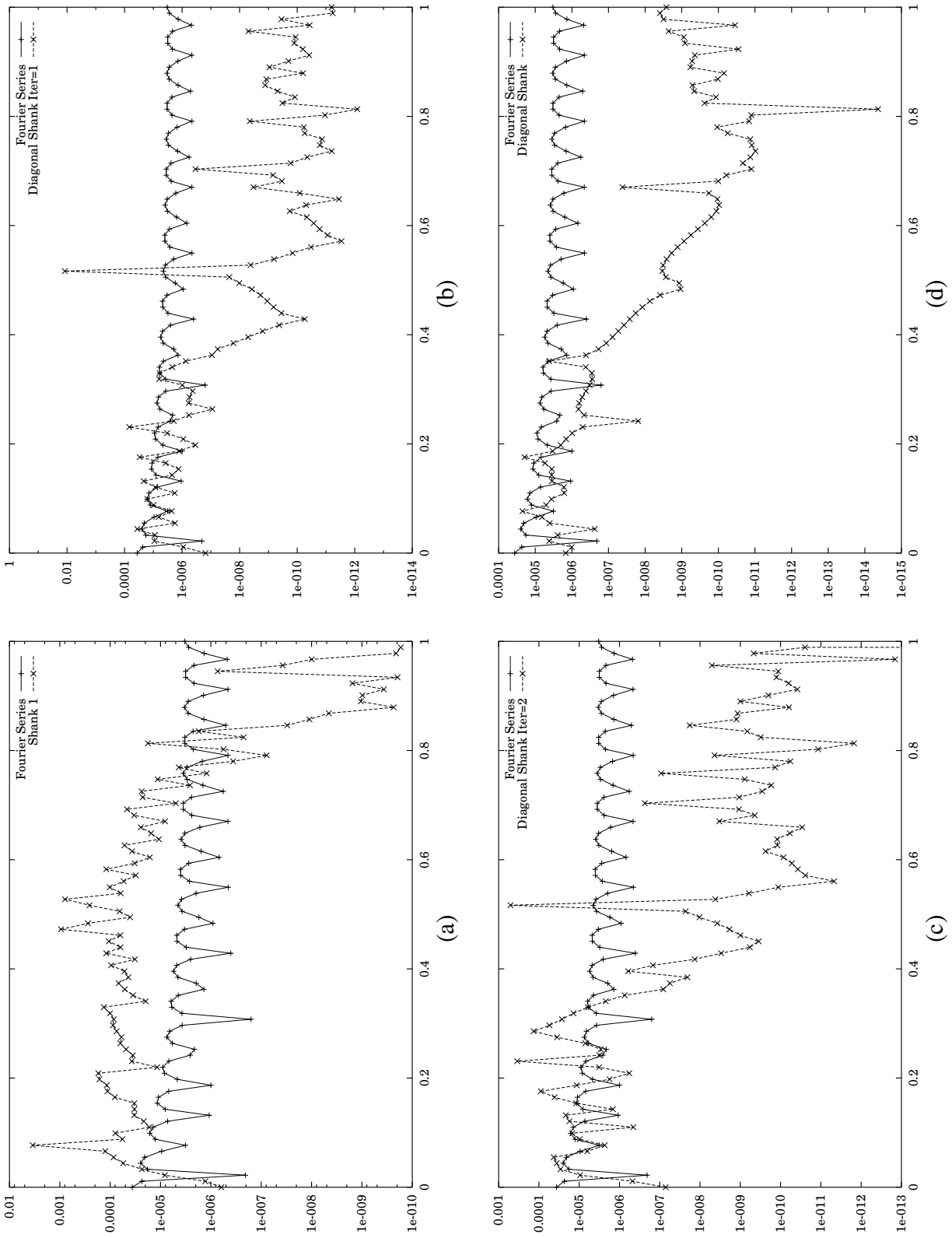


Figure 3.9: Comparison of the error of the Fourier series $f_{16}^{*}(x)$ with the (a) Shanks transform of degree 1, (b) first iteration of the diagonal Shanks transform, (c) second iteration of the diagonal Shanks transform and (d) diagonal Shanks transform.

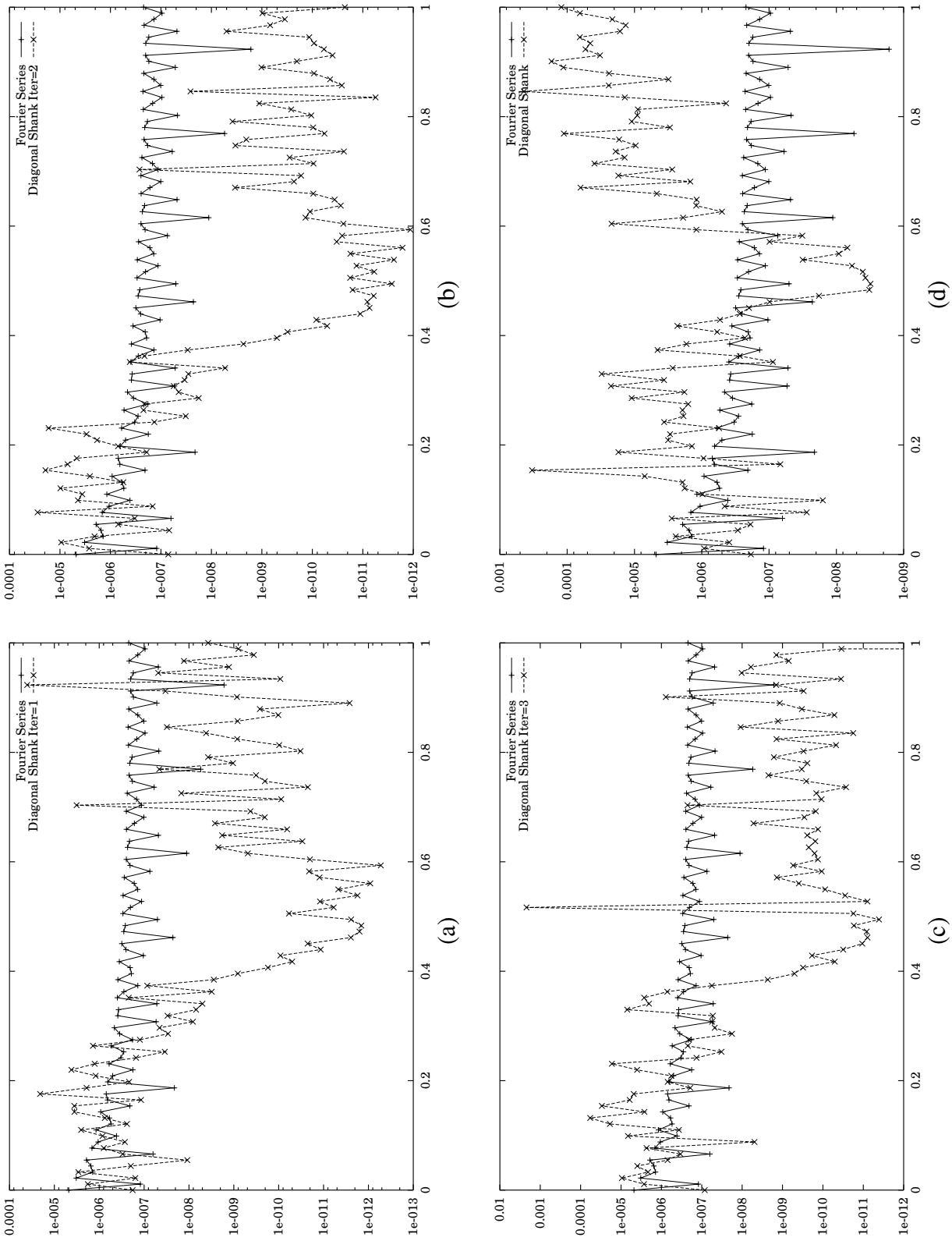


Figure 3.10: Comparison of the error of the Fourier series $f_{32}^{*}(x)$ with the (a) first iteration of the diagonal Shanks transform, (b) second iteration of the diagonal Shanks transform, (c) third iteration of the diagonal Shanks transform and (d) diagonal Shanks transform.

N	Maximum Error					
	Diag.	Diag. (1)	Diag. (2)	Diag. (3)	1st Shanks	Fourier Series
4	0.185E-01	0.185E-01			0.208E+00	0.176E-02
8	0.170E-02	0.629E-02			0.936E-02	0.266E-03
12	0.677E-04	0.504E-03	0.504E-03		0.222E-02	0.838E-04
16	0.223E-04	0.119E-01	0.508E-03		0.345E-02	0.365E-04
20	0.356E-04	0.386E-04	0.605E-02		0.179E-02	0.190E-04
24	0.159E-03	0.716E-04	0.411E-02	0.411E-02	0.525E-03	0.112E-04
28	0.414E-03	0.431E-04	0.290E-04	0.290E-04	0.587E-03	0.709E-05
32	0.440E-03	0.404E-04	0.278E-04	0.219E-02	0.729E-03	0.478E-05
	Average Error					
4	0.897E-03	0.897E-03			0.460E-02	0.544E-03
8	0.502E-04	0.125E-03			0.642E-03	0.562E-04
12	0.501E-05	0.190E-04	0.190E-04		0.199E-03	0.138E-04
16	0.117E-05	0.133E-03	0.163E-04		0.101E-03	0.494E-05
20	0.776E-06	0.146E-05	0.695E-04		0.565E-04	0.225E-05
24	0.769E-05	0.119E-05	0.468E-04	0.468E-04	0.193E-04	0.116E-05
28	0.160E-04	0.931E-06	0.983E-06	0.983E-06	0.143E-04	0.647E-06
32	0.223E-04	0.102E-05	0.130E-05	0.258E-04	0.132E-04	0.395E-06

Table 3.4: Maximum error and the average error in approximating $f(x)$ using the Fourier series and different extrapolation schemes.

The denominator of equation (3.30) will have approximately n zeros. Let us assume that the denominator is zero. Then the numerator of (3.30) becomes

$$A_{n-1}\Delta A_n - A_n\Delta A_{n-1} = \Delta A_n(A_{n-1} - A_n) = -\left(\Delta A_n\right)^2 = -a_n^2\cos^2 n\pi x \neq 0.$$

Thus the numerator does not vanish and the poles of the transformation are not removable. That is, as N increases the transformation becomes unbounded at a linearly increasing number of x values. This explains the peaks and their increasing number in Figures 3.7-3.10. Inspection of the denominator in (3.30) shows that this singular behaviour does not occur at the endpoints, $x = 0$ and $x = 1$, explaining the spectacular success of Shanks' transform at these points.

In general, since analytic orthogonal expansions are highly oscillatory, more analysis is required to identify a mechanism that would allow a robust application of the Shanks transform. One natural approach would be to transform only those x values that are not singular. Identification of the poles of (3.30) should not be too difficult for a Fourier expansion. The entire function can then be recaptured via some interpolation. Of course, roots of some orthogonal polynomials (other than trigonometric or Chebyshev) may be difficult to find.

3.3 Hermite polynomial interpolant

In the last section, we solved the integral equation by approximating the unknown using a trigonometric series and then determined the coefficients. Here, we discretize the region and interpolate in each interval. To this end, we consider the integral equation

$$g(y) = \int_a^b K(x,y)f(x)\,dx, \qquad a \leq y \leq b, \tag{3.31}$$

and define a grid

$$a = x_0 < x_1 < \cdots < x_{n-1} < x_n = b. \tag{3.32}$$

Thus, we may write (3.31) as

$$g(y) = \sum_{j=1}^{n} S_j \int_0^1 K(S_j t + x_{j-1}, y)F_j(t)\,dt, \tag{3.33}$$

where we have mapped the integral over each sub-division to the unit interval and $F_j(t) = f(S_j t + x_{j-1})$. The length of each interval is $S_j = x_j - x_{j-1}$, for $j = 1, 2, \ldots, n$. Using cubic Hermite interpolation polynomials, we can write

$$F_j(t) = F_j(0)H_1(t) + \dot{F}_j(0)H_2(t) + F_j(1)H_3(t) + \dot{F}_j(1)H_4(t) \quad \text{for} \quad j = 1, 2, \ldots n. \tag{3.34}$$

In Appendix B we outline the theory of Hermite interpolation polynomials as well as provide a Maple routine to calculate them for any order.

If the nodal behaviour of F is known, then substitution into (3.34) will provide the interpolant as an approximate solution. We first proceed by appealing to the continuity and differentiability of f to identify a few conditions at each internal node

$$f(x_j) = F_j(1) = F_{j+1}(0), \quad f'(x_j) = \frac{1}{S_j}\dot{F}_j(1) = \frac{1}{S_{j+1}}\dot{F}_{j+1}(0), \qquad \text{for} \quad j = 1, 2, \ldots, n-1. \tag{3.35}$$

Using (3.35) to substitute for each $F_j(1)$ and $\dot{F}_j(1)$ in (3.34) and re-arranging gives

$$\begin{aligned} g(y) = F_1 B_{1,1}(y) + \dot{F}_1 B_{1,2}(y) + \sum_{j=2}^{n} F_j\big(B_{j,1}(y) + B_{j-1,3}(y)\big) \\ + \dot{F}_j\left(B_{j,2}(y) + \frac{S_{j-1}}{S_j}B_{j-1,4}(y)\right) + F_{n+1}B_{n,3}(y) + \dot{F}_{n+1}B_{n,4}(y), \end{aligned} \tag{3.36}$$

where we have used the notation

$$F_j = F_j(0), \qquad \dot{F}_j = \dot{F}_j(0), \qquad F_{n+1} = F_n(1), \qquad \dot{F}_{n+1} = \dot{F}_n(1),$$

and

$$B_{j,k}(y) = S_j \int_0^1 K(s_j t + x_{j-1}, y) H_k(t)\, dt,$$

for $j = 1, \ldots, n$ and $k = 1, 2, 3, 4$.

Since $\dot{F}_1 = \dot{F}_{n+1} = 0$, we have the $2n$ unknowns

$$F_1, F_2, \dot{F}_2, \ldots, F_n, \dot{F}_n, F_{n+1}. \tag{3.37}$$

To formulate a linear system we evaluate (3.36) at $2n$ collocation points

$$a \le y_1 < y_2 < \cdots < y_{2n-1} < y_{2n} \le b, \tag{3.38}$$

thus obtaining the linear equations

$$\begin{aligned} g(y_I) = F_1 A_{1,1,I} + \sum_{j=2}^{n} F_j\big(A_{j,1,I} + A_{j-1,3,I}\big) \\ + \dot{F}_j\left(A_{j,2,I} + \frac{S_{j-1}}{S_j}A_{j-1,4,I}\right) + F_{n+1}A_{n,3,I}, \qquad \text{for} \quad I = 1, 2, \ldots, 2n \end{aligned} \tag{3.39}$$

where $A_{j,k,I} = B_{j,k}(y_I)$. Inverting (3.39) to obtain (3.37) will provide, on substitution into (3.34), a cubic approximation that is continuous and differentiable.

3.3.1 Collocation

It might first appear that the collocation points (3.38) can be selected without regard to the distribution of the grid (3.32). This being the case, one might choose to distribute the $2n$ collocation points uniformly over $[a, b]$, therefore ensuring that the points are as far away from each other as possible. The expected result being that the collocation equations are *the most* linearly independent (i.e. the system is assured of being well-conditioned). This is certainly not the case. We require an approximation to be provided for each interval j, thus evaluation should be performed in all intervals. In addition, if the linear system in (3.39) is assembled in the order depicted in (3.37) and we collocate in every interval then the singular integrals will lie on the diagonal, thus producing a matrix that is diagonally dominant. Collocating arbitrarily will certainly not produce a diagonally dominant matrix. On this basis, the collocation points should be distributed equally across the grid. That is, for cubic Hermite interpolation we should have two collocation points in each subdivision.

The question then arises: How should the collocation points be chosen for each interval? Clearly, collocation points should be unique, thus the node points are not candidates. Scaling each interval to (0,1), we collocate twice, $0 < y_1, y_2 < 1$. Figure 3.11 shows a surface and contour plot of the reciprocal condition number of (3.39) as a function of scaled collocation points for $n = 64$. It is clear from Figure 3.11 that the condition number of the linear system is highly dependent on the position of the collocation points. The contour plot shows that the surface is symmetric about the line $y_1 = y_2$ and two distinct shallow regions are clearly identifiable. Region A occupies the areas $y_1 \in (0, 1/2),\ y_2 \in (0, 1/2 - y1)$ and $y_1 \in (1/2, 1),\ y_2 \in (3/2 - y_1, 1)$. In these regions the collocation points are on the same side of the midpoint and the reciprocal condition number is of the order of 10^{-7}. The middle strip, region B, places the collocation points on opposite sides of the midpoint. The reciprocal condition number here is of order 10^{-4}. Clearly, it would be disastrous to place the points near $y_2 = y_1$ and a good strategy for collocation would be to place the points almost anywhere on the line $y_2 = 1 - y_1$. According to this measure, the optimal points lie near 0.3 and 0.7.

To obtain a stable system, the distribution of collocation points must be considered as a function of both polynomial interpolation order and kernel singularity. Much work has been done in the 1980's in this direction where a convergence theory for piecewise constant and linear interpolants was developed (Voronin and Cecoho 1974; Prössdorf and Schmidt 1981; Wendland 1981; Schmidt 1983; Prößdorf and Rathsfeld 1984; Schmidt 1984; Arnold and Wendland 1985; Niessner and Ribaut 1985;

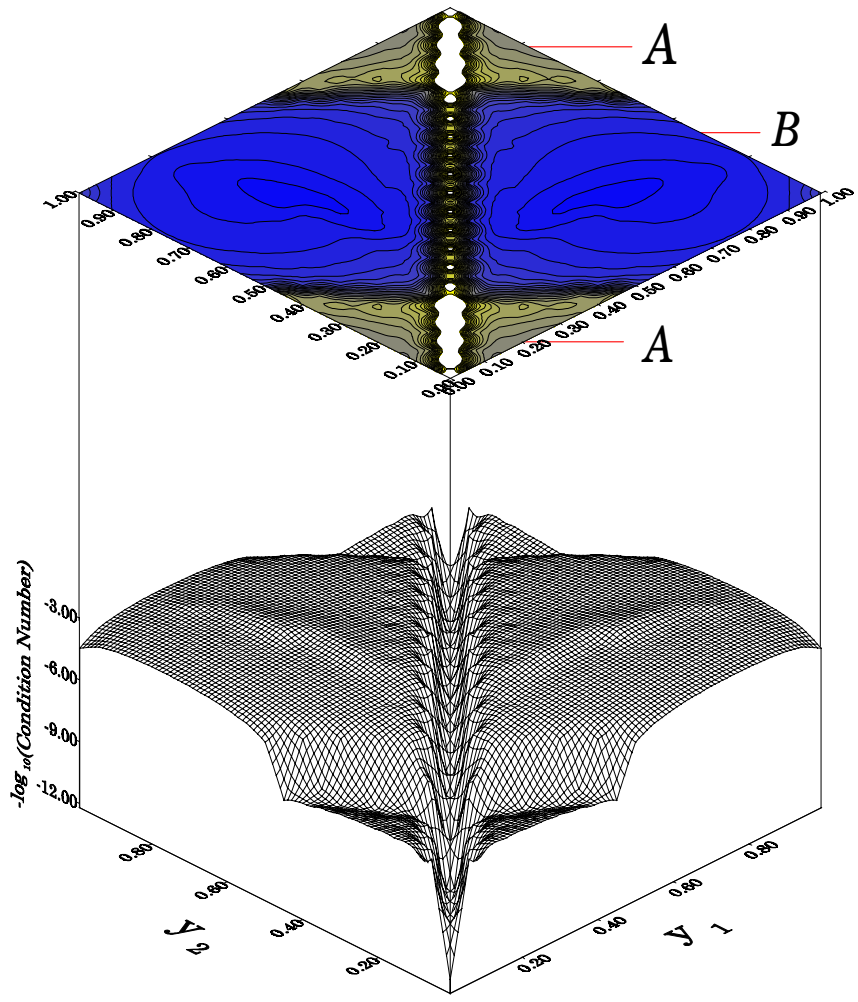


Figure 3.11: Reciprocal condition number of the linear system in (3.39) for $n = 64$ as a function of the collocation points scaled to (0,1).

Saranen and Wendland 1985; Schmidt 1985; Schmidt 1986; Niessner 1987; Prössdorf and Rathsfeld 1987). For an excellent review see Arnold and Wendland (1983). Convergence of the numerical solution is guaranteed if one collocates evenly between the node points (Arnold and Wendland 1983; Arnold and Wendland 1985; Niessner and Ribaut 1985; Saranen and Wendland 1985; Schmidt 1985), though not necessarily to the solution (Arnold and Wendland 1983; Collatz 1966, pp. 260-262). Recently, McLean and Prößdorf (1996) extended this theory to include Hermite cubics. They show that with a logarithmic kernel, collocating symmetrically in each interval gives a theoretical convergence rate of $O(h^5)$ and collocating at the special points $0.2451188417393386, .754881158261$ induces a *super-convergence* of $O(h^7)$. These rates have been measured using the norm of some special Sobolev space. They do, however, identify that an *optimal* collocation regime does exist. Unfortunately, their results apply for a closed boundary with smooth data.

In an effort to identify optimal collocation points, we develop a theory that writes down an explicit expression for the approximate solution of first kind integral equations. The only assumptions made are that the kernel is positive and integrable and the unknown possesses a derivative that is bounded. The method is based on the Peano kernel approach that is outlined in appendix C. Many of the formal details of the method are presented in that appendix and we will only reproduce the required particulars here. In that appendix a kernel, equation (C.8), was carefully developed and employed to derive an interior point rule to approximate weighted integrals whose functions have bounded second derivatives.

In equation (3.31) we will assume $K(\cdot, y) : [a, b] \to (0, \infty)$ to be integrable and positive, that is $K(\cdot, y) \in L_1(a, b)$ and $K(x, y) \geq 0,\ \forall (x, y) \in [a, b] \times [a, b]$. In addition, we assume that $f : [a, b] \to \mathbb{R}$ has bounded first derivative and we approximate it using the constant functional

$$f(x) \approx \begin{cases} f(a), & a \leq x \leq \xi, \\ f(b), & \xi < x \leq b. \end{cases} \tag{3.40}$$

We seek to write down an explicit formula for $f(a)$ and $f(b)$ in terms of g and K. Hence the kernel (C.8) requires modification to produce a trapezoidal-like quadrature rule for functions with bounded derivatives. The Peano kernel is

$$p(x, y, \xi) = \int_{\xi}^{x} K(z, y)\, dz, \qquad a \leq x, \xi \leq b. \tag{3.41}$$

Substituting (3.41) into $\int_a^b p(x,y,\xi)f'(x)\,dx$ and integrating by parts gives, upon using (3.31),

$$\int_a^b p(x,y,\xi)f'(x)\,dx = \left(\int_a^\xi K(z,y)\,dz\right) f(a) + \left(\int_\xi^b K(z,y)\,dz\right) f(b) - g(y). \tag{3.42}$$

Taking the modulus of both sides of (3.42) and applying a Hölder type inequality (similar to the Cauchy-Schwartz-Buniakowsky inequality) provides the result

$$\left| g(y) - f(a)\int_a^\xi K(z,y)\,dz - f(b)\int_\xi^b K(z,y)\,dz \right| \leq \|f'\|_\infty I(\xi,y), \tag{3.43}$$

$$\leq \|f'\|_\infty \|K(\cdot,y)\|_1 \left(\frac{b-a}{2} + \left|\xi - \frac{a+b}{2}\right|\right) \tag{3.44}$$

where the bound $I(\xi,y)$ has been simplified by reversing the order of integration to

$$I(\xi,y) = \int_a^\xi (z-a)K(z,y)\,dz + \int_\xi^b (b-z)K(z,y)\,dz. \tag{3.45}$$

Equation (3.44) was obtained by taking an upper bound in (3.45) and using the well known result

$$\max\{a,b\} = \frac{1}{2}\big(a+b+|a-b|\big).$$

By differentiating and appealing to the properties of convex functions, it is a simple matter to show that I is minimised at the midpoint of the interval $\xi = \frac{a+b}{2}$ (see appendix C for the details). The result is independent of the kernel, K, and is due to the rule sampling at both end-points.

Thus, with this class of function, the optimal approximation to g is

$$g(y) \approx f(a)\int_a^{\frac{a+b}{2}} K(z,y)\,dz + f(b)\int_{\frac{a+b}{2}}^b K(z,y)\,dz. \tag{3.46}$$

We note that we are justified in asserting the approximation (3.46) since the error is bounded by (3.44). This equation clearly shows that the approximation is valid if the interval size $b-a$ is small.

Evaluating (3.46) at two distinct (collocation) points will produce two linear equations in $f(a)$

and $f(b)$ that can be solved to provide an approximation for f. The result is

$$f(a) = \frac{1}{D(y_1, y_2)} \left(g(y_1) \int_{\frac{a+b}{2}}^{b} K(z, y_2)\, dz - g(y_2) \int_{\frac{a+b}{2}}^{b} K(z, y_1)\, dz \right), \tag{3.47}$$

$$f(b) = \frac{1}{D(y_1, y_2)} \left(g(y_2) \int_{a}^{\frac{a+b}{2}} K(z, y_1)\, dz - g(y_1) \int_{a}^{\frac{a+b}{2}} K(z, y_2)\, dz \right), \tag{3.48}$$

where the determinant D is

$$D(y_1, y_2) = \int_{a}^{\frac{a+b}{2}} K(z, y_1)\, dz \int_{\frac{a+b}{2}}^{b} K(z, y_2)\, dz - \int_{\frac{a+b}{2}}^{b} K(z, y_1)\, dz \int_{a}^{\frac{a+b}{2}} K(z, y_2)\, dz. \tag{3.49}$$

Thus equation (3.40) with (3.47) and (3.48) gives us an explicit approximation to the integral equation. We make the point that the method is quite general and is applicable to a wide class of functions in that differentiability is the only assumption made on f. Even this condition may be relaxed and a similar result, for example, may be obtained for functions with bounded variation (Cerone and Dragomir 1999) or mappings that are merely integrable (Cerone *et al.* 1999a). Other norms have been explored (Cerone *et al.* 1999b; Cerone *et al.* 1999c; Cerone *et al.* 1998; Cerone *et al.* 1999d; Dragomir *et al.* 2000), and in combination with weighted mappings (Roumeliotis *et al.* Dragomir *et al.* 1999; Cerone *et al.* 1999).

We do not make the assertion that (3.40) is an accurate approximation to the unknown; indeed one would expect poor results since the entire integral in (3.31) has been approximated by a single weighted trapezoidal-like rule. A composite rule, like that developed in appendix C, would be the natural next step, as well as a formal proof of the convergence of the resulting expansion.

It is important to note that the method outlined in (3.40) to (3.49) is Nyström like in that the integral equation is evaluated via a quadrature rule. The drawback of the traditional Nyström method is its reliance on Gauss quadrature to perform the integration. While Gauss rules are computationally extremely efficient, their lack of error bounds makes theoretical work difficult. Newton-Cotes rules have an abundance of error results, but their poor performance, especially for singular integrands, makes their use prohibitive. The theory outlined here and in appendix C provides a basis for the formulation of weighted Newton-Cotes type rules. They have the advantage of the availability of error bounds in a variety of norms, whilst being computationally more efficient than the traditional trapezoidal, Simpson's, etc. rules. We do not proceed with developing a *composite* interpolation expansion, since the purpose of this section is to identify general collocation points.

Inspection of (3.49) shows that D is anti-symmetric in y_1 and y_2. That is,

$$D(y_1, y_2) = -D(y_2, y_1).$$

This suggests that the collocation points should be chosen symmetrically about the line $y_2 = y_1$

$$y_2 = C - y_1, \qquad \text{for some} \quad 0 < C < 1. \tag{3.50}$$

One way of identifying *good* collocation points is to ensure that (3.47) and (3.48) are well-defined. Thus we seek the points y_1, y_2 that maximises D. We note that the choice of y_1, y_2 is dependent on K and not g. This is a desirable property since the collocation points should be distributed according to the kernel and the shape of the boundary, not the data.

For Symm's equation, we have $K(x, y) = \ln|x - y|,\ a = 0,\ b = 1$. With the aid of Maple, the determinant is simple to evaluate (but cumbersome to write down) and D becomes

$$D(y_1, y_2) = \begin{cases} D_{11}(y_1, y_2), & 0 < y_1, y_2 < 1/2, \\ D_{12}(y_1, y_2), & 0 < y_1, 1 - y_2 < 1/2, \\ D_{21}(y_1, y_2), & 0 < 1 - y_1, y_2 < 1/2, \\ D_{22}(y_1, y_2), & 1/2 < y_1, y_2 < 1, \end{cases} \tag{3.51}$$

where D_{11}, D_{12}, D_{21} and D_{22} are defined respectively as

$$\begin{aligned} D_{11}(y_1, y_2) = &\Big(\ln(y_1)y_1 - 1/2\ \ln(2) + \ln(2)y_1 + 1/2\ \ln(1 - 2\,y_1) - \ln(1 - 2\,y_1)y_1 - 1/2\Big) \\ &\Big(\ln(1 - y_2) - \ln(1 - y_2)y_2 - 1/2 + 1/2\ \ln(2) - \ln(2)y_2 - 1/2\ \ln(1 - 2\,y_2) \\ &+ \ln(1 - 2\,y_2)y_2\Big) - \Big(\ln(1 - y_1) - \ln(1 - y_1)y_1 - 1/2 + 1/2\ \ln(2) - \ln(2)y_1 \\ &- 1/2\ \ln(1 - 2\,y_1) + \ln(1 - 2\,y_1)y_1\Big)\Big(\ln(y_2)y_2 - 1/2\ \ln(2) + \ln(2)y_2 \\ &+ 1/2\ \ln(1 - 2\,y_2) - \ln(1 - 2\,y_2)y_2 - 1/2\Big), \end{aligned} \tag{3.52}$$

$$\begin{aligned} D_{12}(y_1, y_2) = &\Big(\ln(y_1)y_1 - 1/2\ \ln(2) + \ln(2)y_1 + 1/2\ \ln(1 - 2\,y_1) - \ln(1 - 2\,y_1)y_1 - 1/2\Big) \\ &\Big(-\ln(2)y_2 + 1/2\ \ln(2) + \ln(2\,y_2 - 1)y_2 - 1/2\ \ln(2\,y_2 - 1) - 1/2 + \ln(1 - y_2) \\ &- \ln(1 - y_2)y_2\Big) - \Big(\ln(1 - y_1) - \ln(1 - y_1)y_1 - 1/2 + 1/2\ \ln(2) - \ln(2)y_1 \\ &- 1/2\ \ln(1 - 2\,y_1) + \ln(1 - 2\,y_1)y_1\Big)\Big(\ln(2)y_2 - 1/2\ \ln(2) - \ln(2\,y_2 - 1)y_2 \\ &+ 1/2\ \ln(2\,y_2 - 1) - 1/2 + \ln(y_2)y_2\Big), \end{aligned} \tag{3.53}$$

$$D_{21}(y_1,y_2) = \Big(\ln(2)y_1 - 1/2\ \ln(2) - \ln(2\,y_1-1)y_1 + 1/2\ \ln(2\,y_1-1) - 1/2 + \ln(y_1)y_1\Big)$$
$$\Big(-\ln(2)y_2 + 1/2\ \ln(2) + \ln(2\,y_2-1)y_2 - 1/2\ \ln(2\,y_2-1) - 1/2 + \ln(1-y_2)$$
$$-\ln(1-y_2)y_2\Big) - \Big(-\ln(2)y_1 + 1/2\ \ln(2) + \ln(2\,y_1-1)y_1 - 1/2\ \ln(2\,y_1-1)$$
$$-1/2 + \ln(1-y_1) - \ln(1-y_1)y_1\Big)\Big(\ln(2)y_2 - 1/2\ \ln(2) - \ln(2\,y_2-1)y_2$$
$$+1/2\ \ln(2\,y_2-1) - 1/2 + \ln(y_2)y_2\Big), \quad (3.54)$$

$$D_{22}(y_1,y_2) = \Big(\ln(2)y_1 - 1/2\ \ln(2) - \ln(2\,y_1-1)y_1 + 1/2\ \ln(2\,y_1-1) - 1/2 + \ln(y_1)y_1\Big)$$
$$\Big(-\ln(2)y_2 + 1/2\ \ln(2) + \ln(2\,y_2-1)y_2 - 1/2\ \ln(2\,y_2-1) - 1/2 + \ln(1-y_2)$$
$$-\ln(1-y_2)y_2\Big) - \Big(-\ln(2)y_1 + 1/2\ \ln(2) + \ln(2\,y_1-1)y_1 - 1/2\ \ln(2\,y_1-1)$$
$$-1/2 + \ln(1-y_1) - \ln(1-y_1)y_1\Big)\Big(\ln(2)y_2 - 1/2\ \ln(2) - \ln(2\,y_2-1)y_2$$
$$+1/2\ \ln(2\,y_2-1) - 1/2 + \ln(y_2)y_2\Big). \quad (3.55)$$

Figure 3.12 shows a surface and contour plot of (3.51). The optimal location of the collocation points are

$$.2148458281482123 \qquad \text{and} \qquad .7851541718517877, \quad (3.56)$$

at which D from (3.51) is minimal. The results are in general agreement with the work of McLean and Prößdorf (1996). They showed that with cubic Hermite interpolation, the (scaled) collocation are symmetric about 0.5. It is of some comfort that the optimal points given in their work are close to those identified here. However, McLean and Prößdorf assumed a closed 2D curve with smooth data; features absent in the integral equation solved here. In addition, our work is only valid for small $b-a$.

In Figure 3.13, we compare the theoretical analysis here, with the numerical inversion of (3.39). In this figure, (a) is the contour plot of Figure 3.12, (b) the reciprocal condition number of the system in (3.39) for $n = 2$ and (c) for $n = 64$. We can see that all three surfaces share the same general features: the plots are symmetric about $y_1 = y_2$ and two maxima appear near the collocation points $0.25, 0.75$. When n increases from 2 to 64 the surface evolves and forms two shallow regions, previously labelled A in Figure 3.11.

This *a posteriori* check confirms the validity of the theory presented here.

(a)

(b)

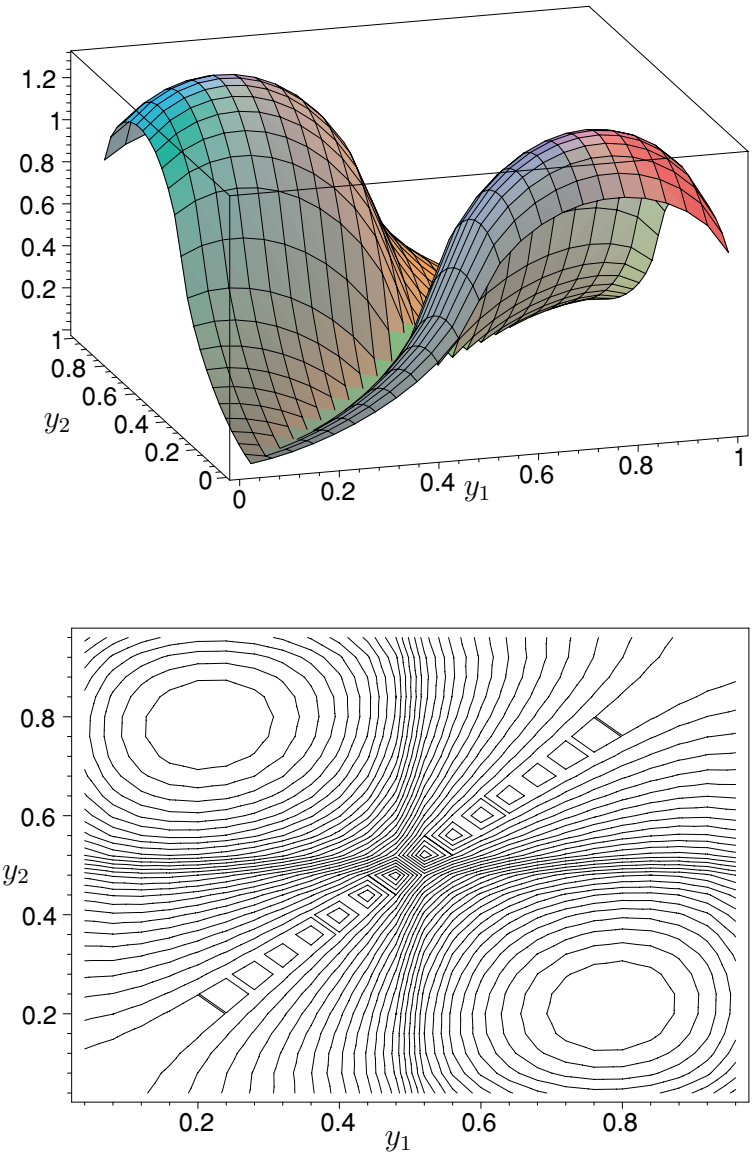


Figure 3.12: (a) Surface and (b) contour plots of absolute value of the determinant (3.51) showing optimal location of collocation points y_1 and y_2.

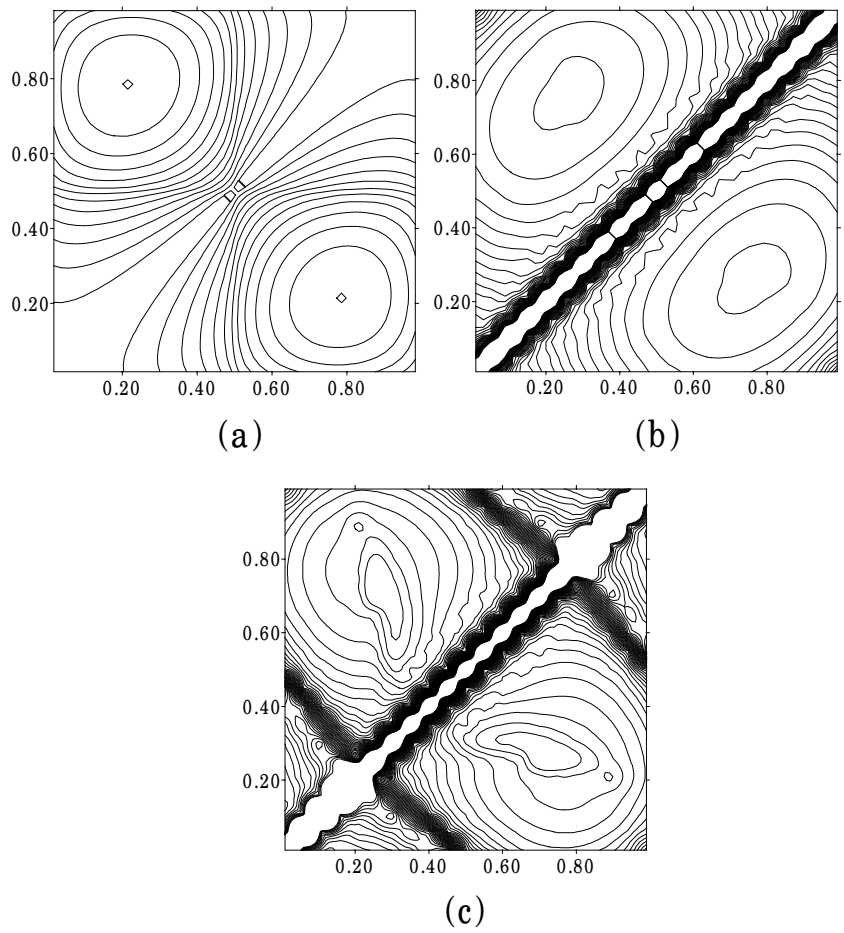


Figure 3.13: Comparison of the contour plots of (a) the determinant (3.51) (b) the reciprocal condition number of the system (3.39) when $n = 2$ and (c) the reciprocal condition number of the system (3.39) when $n = 64$ as a function of scaled collocation points.

3.3.2 Numerical results

Results for different grid sizes are presented in Table 3.5. The maximum error in the function and derivative are tabulated for each node point. In all cases, the scaled collocation points were chosen to be 0.25 and 0.75.

The data in Table 3.5 clearly shows that the interpolation-collocation method performs extremely well. The numerical solution converges to the exact solution as the number of grid points increases. A curious feature is that the error in the derivative is smaller than in the function value. This may be due to the fact that the exact solution (3.5) is a polynomial.

Various norms for the absolute error in the function are shown in Table 3.6. We can see that the maximum error (column 3) decreases approximately $1\frac{1}{2}$ orders whenever the number of segments

		$n = 4$		$n = 8$		$n = 16$	
i	x_i	Error in Function	Error in Derivative	Error in Function	Error in Derivative	Error in Function	Error in Derivative
0	0.0000	0.147E-03	0.000E+00	0.896E-05	0.000E+00	0.551E-06	0.000E+00
1	0.0625					0.511E-06	0.421E-06
2	0.1250			0.823E-05	0.359E-05	0.513E-06	0.514E-07
3	0.1875					0.513E-06	0.242E-08
4	0.2500	0.133E-03	0.312E-04	0.825E-05	0.423E-06	0.512E-06	0.265E-08
5	0.3125					0.512E-06	0.226E-08
6	0.3750			0.825E-05	0.264E-07	0.512E-06	0.143E-08
7	0.4375					0.512E-06	0.680E-09
8	0.5000	0.133E-03	0.198E-13	0.825E-05	0.235E-13	0.512E-06	0.110E-12
9	0.5625					0.512E-06	0.680E-09
10	0.6250			0.825E-05	0.264E-07	0.512E-06	0.143E-08
11	0.6875					0.512E-06	0.226E-08
12	0.7500	0.133E-03	0.312E-04	0.825E-05	0.423E-06	0.512E-06	0.265E-08
13	0.8125					0.513E-06	0.242E-08
14	0.8750			0.823E-05	0.359E-05	0.513E-06	0.514E-07
15	0.9375					0.511E-06	0.421E-06
16	1.0000	0.147E-03	0.000E+00	0.896E-05	0.000E+00	0.551E-06	0.000E+00

Table 3.5: Absolute error in function and derivative values at the node points for $n = 4, 8$ and 16.

doubles. There is little difference in the L_1 and L_2 norms and this indicates that the error is distributed in a reasonably uniform manner across the entire domain.

A fair comparison of the expansion and interpolation methods can be made on the basis of size of their respective linear systems. A Fourier expansion up to N results in an $(N+1)\times(N+1)$ system (3.21). This contrasts with the Hermite interpolation, where a grid with n intervals produces a $2n \times 2n$ linear system. Hence we will compare the values in the last column in Table 3.6 for a given n with those in the last column of Table 3.4 for $N = 2n$.

It is obvious that the Hermite interpolation far outperforms the Fourier series at all values of n. The convergence rate is greater and, unlike the Fourier series, does not stop when $n =$

n	$\|f - f^*\|_1$	$\|f - f^*\|_2$	$\|f - f^*\|_\infty$
4	0.76532E-04	0.85599E-04	0.14682E-03
8	0.48003E-05	0.53344E-05	0.89576E-05
16	0.30053E-06	0.33311E-06	0.55057E-06
32	0.18799E-07	0.20814E-07	0.34012E-07
64	0.11754E-08	0.13007E-08	0.21086E-08
128	0.73477E-10	0.81294E-10	0.13103E-09

Table 3.6: Interpolation errors measured via various norms for different values of n.

$32, 64, 128$. Even with the method of reducing the number of coefficients in the Fourier expansion, the Hermite interpolation is superior.

3.4 Conclusion

The Fourier series method successfully solved the integral equation, but the results were disappointing. The error deteriorated after an expansion of 32 terms and this is due to the great difficulty of finding high frequency contributions to the unknown in first kind equations. Measures taken to address this point were somewhat successful and a more detailed investigation of the expansion method is warranted. This is especially true if one notes that the traction of an axisymmetric particle is inherently periodic. The expansion, coupled with a robust Shanks-type transform, may prove fruitful.

However, it is clear in this case that interpolation-collocation outperforms expansion-collocation. To reinforce this notion we compare the linear system produced by the Fourier series (3.21) with that of the Hermite interpolation scheme. Figure 3.14 shows a grey-scale map of the matrix entries of (a) equation (3.21) and (b) equation (3.39). Darker areas represent numbers that are larger in magnitude.

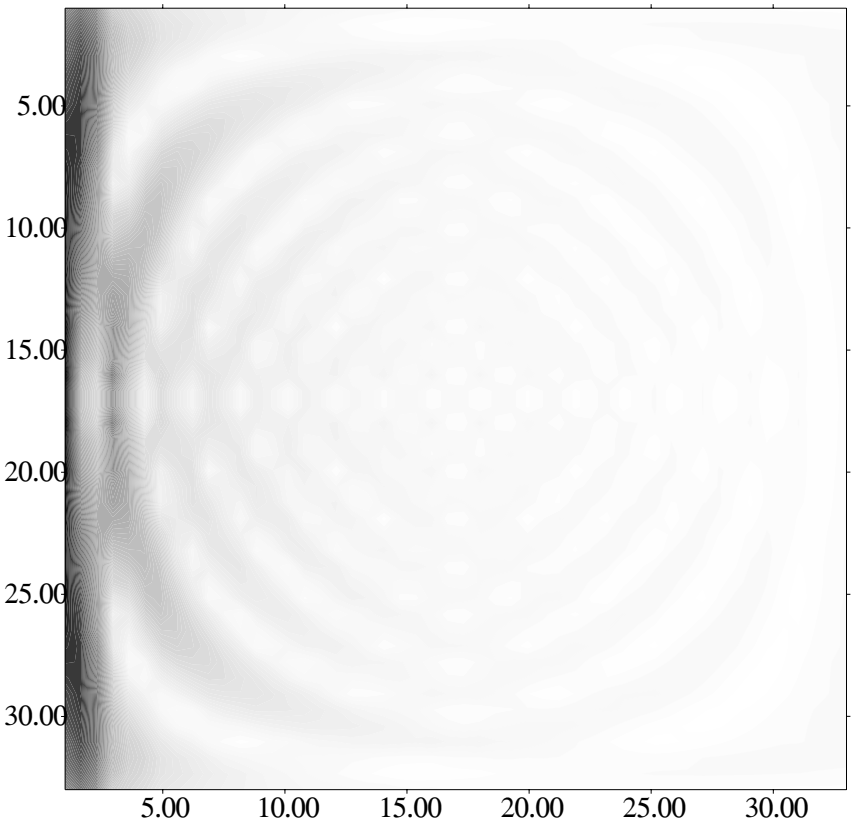
As expected, the interpolation method produces a diagonally dominant system. In contrast, the expansion method does not. This is because the frequency of the integrand (3.14) increases in accordance with the column. Thus higher relative values will appear in column 1 and they will

tend to decrease in magnitude with each successive column.

Based on the results of this chapter, the Hermite-collocation method will be used to solve the coupled integral equations (2.18) and (2.19) in the following chapter.

In appendix C, we outline the method of obtaining *weighted* integral inequalities. The method is able to deal with quite general weight functions and may be extended to bounds involving higher derivatives or other norms. Applications in the areas of numerical integration (Cerone *et al.* 1999; Dragomir *et al.* 1999; Roumeliotis *et al.* 1999), and probability theory (Barnett *et al.* 2000a; Barnett *et al.* 2
have benefited from their attention. It is hoped that the work in this chapter will instigate some activity in approximation theory relating to integral equations.

(a)



(b)

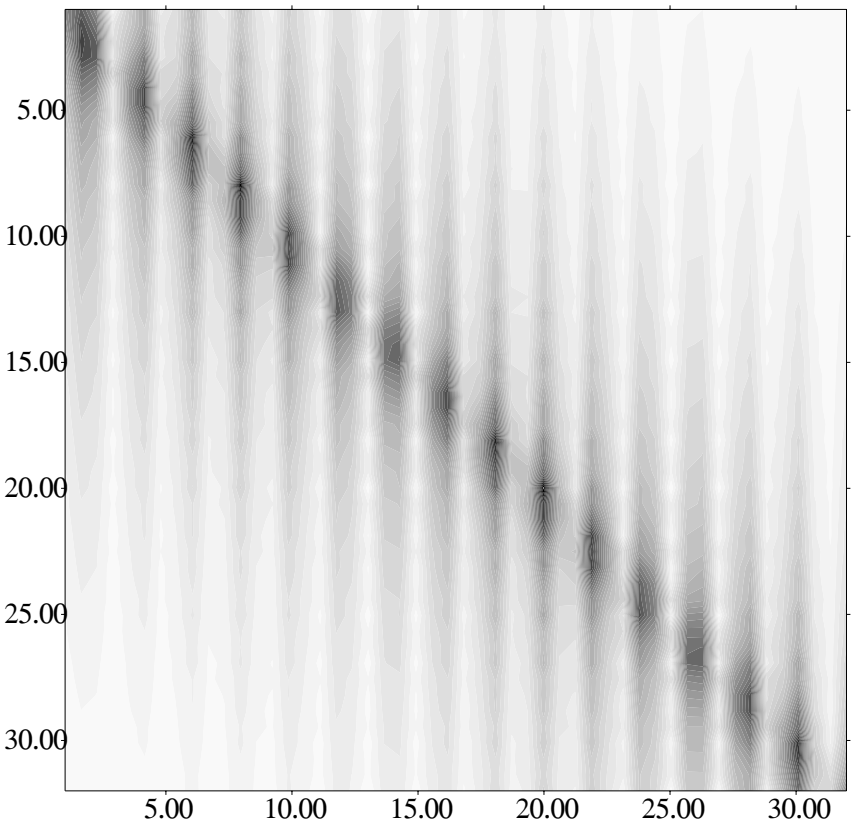


Figure 3.14: Grey-scale map showing the relative magnitude of the matrix entries for (a) equation (3.21) and (b) equation (3.39). Darker areas indicate larger values

Chapter 4

Numerical Solution of Rigid Body Problems

4.1 Introduction

In this chapter we present a method inspired by the results of the previous chapter to solve the axisymmetric rigid body resistance problem via numerical inversion of the coupled integral equations (2.18) and (2.19).

We are by no means the first to attempt a boundary integral solution to the Stokes equations. Beginning with the ground-breaking work of Youngren and Acrivos (1975), there have been a myriad of excellent papers dealing with rigid body motion, free surfaces, drops, interaction between drops and rigid bodies and others. See Kim and Karrila (1991) and Pozrikidis (1992). The distinction with the work presented here will be on insight into the numerical implementation, with a view to increased stability and accuracy, rather than insight into physical problems of interest.

The Stokes equations are linear and the boundary conditions simple, however, we will show that the traction is highly dependent on surface curvature. This is not unexpected, since the the stress of a free surface is directly proportional to its mean curvature, see equation (2.27). We will also investigate the effect of interpolation order and grid distribution on the error of the numerical solution. It will be shown that constant and linear representation, the method of choice even in recent publications, is insufficient for highly deformed surfaces, as is a large mesh size.

Methods to find an *efficient* grid and reducing the curvature dependence of the traction will be presented.

In summary, we aim to show that the traditional methods of increasing grid resolution or interpolation are prescriptions for failure. The remedies we propose involve curvature dependent methods that reduce the requirement for large grids and high interpolation orders. This will lead to the possibility of multi-complex particle interactions being investigated. In the micro-fluidic domain, these are the physical problems of interest.

The algorithm will be presented for a one particle configuration and the generalisation to multi-particles is trivial. The algorithm has been coded to deal with an arbitrary number of particles.

4.2 Numerical method

We can see from equation (2.18)–(2.19) that for a solid particle in an infinite fluid the boundary integral equation can be written as

$$V_p(\xi^*) = \sum_{q=1}^{2} \int_0^{\Xi} \mathcal{K}_{p,q}(\xi, \xi^*) f_q(\xi)\, d\xi, \quad \text{for} \quad 0 \leq \xi^* \leq \Xi,\ p = 1, 2, \tag{4.1}$$

where ξ is the arclength parameter describing the meridional curve C of $\mathcal{S}$ and Ξ is the total length of this curve. The numerical solution of equation (4.1) rests broadly on two features. One is the construction of arbitrary degree piecewise-polynomials over a non-uniform grid as an approximation to the unknown functions f_q. The other consists of introducing these approximations into equation (4.1) and by collocation at a sufficient number of points, set up a system of equations to determine the nodal behaviour of f_q. There are two main advantages to this design. The first is that the grid can be easily adapted to suit irregular particles (where discretization would depend on the particle geometry) or bubbles (where the grid may be recalculated at each time step). Secondly, with a high enough degree polynomial interpolant we can estimate any order derivative of the unknown functions f_q.

4.2.1 Discretization

First, we subdivide the surface (range of integration) into n segments

$$V_p(\xi^*) = \sum_{q=1}^{2} \sum_{j=1}^{n} \int_{\xi_{j-1}}^{\xi_j} \mathcal{K}_{p,q}(\xi, \xi^*) f_q(\xi)\, d\xi. \tag{4.2}$$

Note that the node points have an arbitrary distribution, except for the ordering

$$0 = \xi_0 < \xi_1 < \cdots < \xi_i < \cdots < \xi_{n-1} < \xi_n = \Xi, \tag{4.3}$$

for $i = 0, 1, \ldots, n$. Next, in order to simplify the integrals in (4.2), we employ the following linear transformations

$$t = \frac{\xi - \xi_{j-1}}{\xi_j - \xi_{j-1}} \qquad \text{with} \qquad 0 \le t \le 1, \qquad \text{for} \qquad j = 1, \ldots, n, \tag{4.4}$$

which maps each integral onto the unit interval. This enables equation (4.2) to be re-written as

$$V_p(\xi^*) = \sum_{q=1}^{2} \sum_{j=1}^{n} S_j \int_{0}^{1} \mathcal{K}_{p,q}\left(S_j t + \xi_{j-1}, \xi^*\right) \bar{F}_{q,j}(t)\, dt, \tag{4.5}$$

where

$$S_j = \xi_j - \xi_{j-1} \tag{4.6}$$

and

$$\bar{F}_{q,j}(t) = f_q\left(S_j t + \xi_{j-1}\right). \tag{4.7}$$

Physically, the constants S_j denote the individual segment lengths of the discretization (4.3). The discretization of the particle surface is shown in Figure 4.1.

4.2.2 Interpolation

We employ Hermite interpolation polynomials of order N to interpolate the first $N-1$ derivatives of each $\bar{F}_{q,j}$. The Hermite polynomials, $H_k(N;t)$, (Davis 1963; Gontscharoff 1954; Atkinson 1989) are of degree $2N-1$ with the defining properties

$$H_k^{(m)}(N;0) = \delta_{m+1,k} \qquad \text{and} \qquad H_k^{(m)}(N;1) = \delta_{m+N+1,k}, \tag{4.8}$$

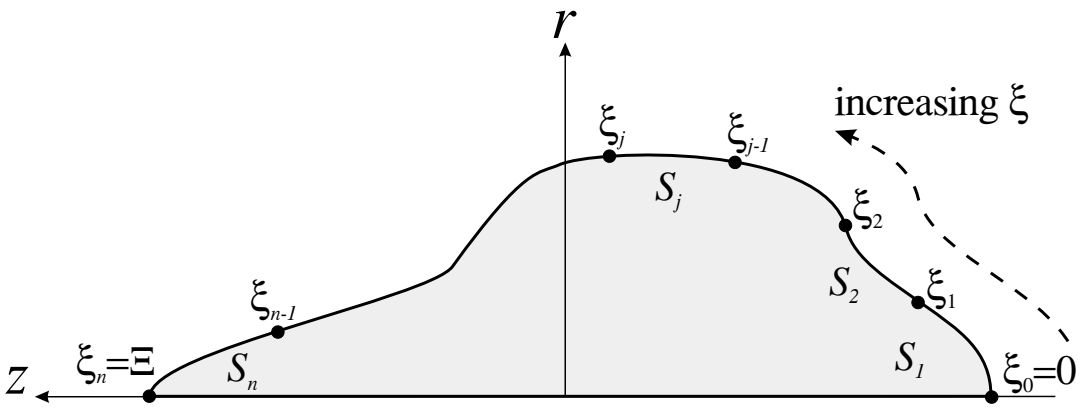


Figure 4.1: Discretization, arclength and segment arclengths of an axially symmetric particle surface.

for $m = 0, 1, \ldots, N-1$ and $k = 1, 2, \ldots, 2N$. The bracketed super-scripts denote the order of the derivative with respect to the scaled parameter t. It is a simple matter to calculate these polynomials using a computer algebra package such as Maple (Char *et al*. 1991). For convenience we have tabulated the polynomials for $N = 2, 4, 6$ (Table B.1) and attached a Maple program (Algorithm B.1) to generate them for higher values of N in appendix B.

If we let $\bar{F}^{*}_{q,j}(t)$ be the polynomial approximation to $\bar{F}_{q,j}(t)$, then equation (4.8) furnishes the proper form of the interpolant

$$\bar{F}^{*}_{q,j}(t) = \sum_{m=0}^{N-1} \bar{F}^{(m)}_{q,j}(0) H_{m+1}(N;t) + \bar{F}^{(m)}_{q,j}(1) H_{m+N+1}(N;t), \tag{4.9}$$

yielding the desired result at the node points

$$\bar{F}^{(m)}_{q,j}(0) = \bar{F}^{*(m)}_{q,j}(0) \qquad \text{and} \qquad \bar{F}^{(m)}_{q,j}(1) = \bar{F}^{*(m)}_{q,j}(1). \tag{4.10}$$

If one assumes that the traction is analytic at the internal node points $(i = 1, 2, \ldots, n-1)$ then equation (4.9) can be simplified as follows. From equation (4.7)

$$\bar{F}^{(m)}_{q,j}(0) = (S_j)^m \frac{d^m f}{d\xi^m}(\xi_{j-1}) \tag{4.11}$$

and

$$\bar{F}^{(m)}_{q,j}(1) = (S_j)^m \frac{d^m f}{d\xi^m}(\xi_j)\,. \tag{4.12}$$

Substituting j for $j-1$ in equation (4.11) and equating with (4.12) leads us to the following

relations

$$\bar{F}^{(m)}_{q,j}(1) = \left(\frac{S_j}{S_{j+1}}\right)^m \bar{F}^{(m)}_{q,j+1}(0) \qquad \text{if} \qquad j = 1, 2, \ldots, n-1. \tag{4.13}$$

Since we are only concerned with the traction at the nodes, we define

$$F^{(m)}_{q,i} = \begin{cases} \bar{F}^{(m)}_{q,i+1}(0) & \text{if} \quad i = 0, 1, \ldots, n-1, \\ \\ \bar{F}^{(m)}_{q,n}(1) & \text{if} \quad i = n. \end{cases} \tag{4.14}$$

Making use of (4.13) and (4.14), the interpolant has the following form

$$\bar{F}^{*}_{q,j}(t) = \begin{cases} \displaystyle\sum_{m=0}^{N-1} F^{(m)}_{q,j-1} H_{m+1}(N;t) + \left(\frac{S_j}{S_{j+1}}\right)^m F^{(m)}_{q,j} H_{m+N+1}(N;t) & \text{if} \quad j = 1, 2, \ldots, n-1, \\ \displaystyle\sum_{m=0}^{N-1} F^{(m)}_{q,n-1} H_{m+1}(N;t) + F^{(m)}_{q,n} H_{m+N+1}(N;t) & \text{if} \quad j = n. \end{cases} \tag{4.15}$$

Substituting the approximation (4.15) for the scaled traction into the integral equation (4.5) and collecting in terms of $F^{(m)}_{q,i}$ produces

$$V_p(\xi^*) = \sum_{q=1}^{2} \sum_{m=0}^{N-1} \left\{ F^{(m)}_{q,0} B_{p,q,1,m+1}(\xi^*) + \sum_{i=1}^{n-1} F^{(m)}_{q,i} D_{p,q,i,m}(\xi^*) + F^{(m)}_{q,n} B_{p,q,n,m+N+1}(\xi^*) \right\}, \tag{4.16}$$

where

$$B_{p,q,j,k}(\xi^*) = S_j \int_0^1 \mathcal{K}_{p,q}(S_j t + \xi_{j-1}, \xi^*) H_k(N;t)\, dt \tag{4.17}$$

and

$$D_{p,q,j,m}(\xi^*) = B_{p,q,j+1,m+1}(\xi^*) + \left(\frac{S_j}{S_{j+1}}\right)^m B_{p,q,j,m+N+1}(\xi^*) \qquad \text{if} \qquad j = 1, 2, \ldots, n-1. \tag{4.18}$$

We can further reduce the number of unknowns by exploiting symmetries at the end nodes.

4.2.3 End node considerations

Assuming that the region is simply connected leads to the required set of boundary conditions for the tractions.

If we extend the domain of the arclength to outside $(0, \Xi)$ then symmetry requires the axial component of traction f_1 to be an even function and the radial component f_2 to be an odd function of ξ about $\xi = 0$ and $\xi = \Xi$.

Choosing N to be even gives the following boundary conditions for the scaled traction

$$\left.\begin{aligned} &F_{1,0}^{(1)} = F_{1,0}^{(3)} = \cdots = F_{1,0}^{(N-1)} = 0, \qquad F_{2,0} = F_{2,0}^{(2)} = \cdots = F_{2,0}^{(N-2)} = 0, \\ &\qquad\qquad\qquad\qquad \text{and} \\ &F_{1,n}^{(1)} = F_{1,n}^{(3)} = \cdots = F_{1,n}^{(N-1)} = 0, \qquad F_{2,n} = F_{2,n}^{(2)} = \cdots = F_{2,n}^{(N-2)} = 0. \end{aligned}\right\} \tag{4.19}$$

Thus the problem is transformed into finding the $2nN$ unknowns

$$\begin{array}{cc} \overbrace{\qquad\qquad\qquad\qquad}^{z \text{ component}} & \overbrace{\qquad\qquad\qquad\qquad}^{r \text{ component}} \\ \textit{bottom of particle} & \textit{bottom of particle} \\ F_{1,0}, F_{1,0}^{(2)}, F_{1,0}^{(4)}, \ldots, F_{1,0}^{(N-2)}, & F_{2,0}^{(1)}, F_{2,0}^{(3)}, F_{2,0}^{(5)}, \ldots, F_{2,0}^{(N-1)}, \\ \textit{internal node points} & \textit{internal node points} \\ F_{1,1}, F_{1,1}^{(1)}, F_{1,1}^{(2)}, \ldots, F_{1,1}^{(N-1)}, & F_{2,1}, F_{2,1}^{(1)}, F_{2,1}^{(2)}, \ldots, F_{2,1}^{(N-1)}, \\ F_{1,2}, F_{1,2}^{(1)}, F_{1,2}^{(2)}, \ldots, F_{1,2}^{(N-1)}, & F_{2,2}, F_{2,2}^{(1)}, F_{2,2}^{(2)}, \ldots, F_{2,2}^{(N-1)}, \\ \vdots & \vdots \\ F_{1,n-1}, F_{1,n-1}^{(1)}, F_{1,n-1}^{(2)}, \ldots, F_{1,n-1}^{(N-1)}, & F_{2,n-1}, F_{2,n-1}^{(1)}, F_{2,n-1}^{(2)}, \ldots, F_{2,n-1}^{(N-1)}, \\ \textit{top of particle} & \textit{top of particle} \\ F_{1,n}, F_{1,n}^{(2)}, F_{1,n}^{(4)}, \ldots, F_{1,n}^{(N-2)}, & F_{2,n}^{(1)}, F_{2,n}^{(3)}, F_{2,n}^{(5)}, \ldots, F_{2,n}^{(N-1)}. \end{array} \tag{4.20}$$

from the 2 equations (4.16).

For completeness, we will also consider multiply connected regions. Surfaces that are embedded in a multiply connected region, for example a torus, require continuity conditions on the end nodes rather than the above symmetry conditions.

These requirements lead to the following end node conditions

$$F_{q,0} = F_{q,n}, \quad F_{q,0}^{(1)} = F_{q,n}^{(1)}, \ \ldots, \quad F_{q,0}^{(N-1)} = F_{q,n}^{(N-1)}, \qquad \text{for } q = 1, 2. \tag{4.21}$$

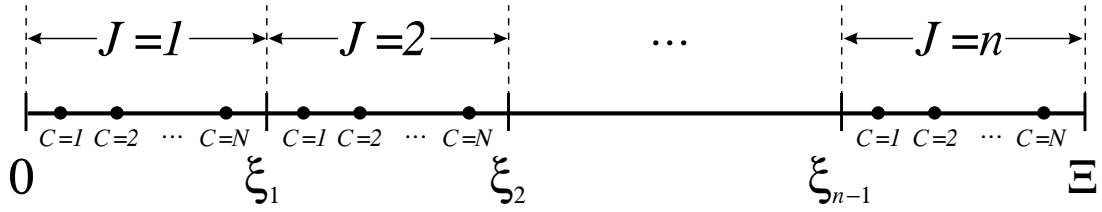


Figure 4.2: The collocation scheme used in the numerical solution of the integral equation (4.1).

4.2.4 Collocation

To form a linear system, we evaluate (4.16) at $2nN$ collocation points. As shown in chapter 3, collocating evenly in all intervals is crucial. The results of that chapter also showed that one should collocate symmetrically about the midpoint of the interval. With $N = 2$, anywhere in the regions $(0.2, 0.3)$ and $(0.7, 0.8)$ were acceptable. Here, the points, $\xi_{J,C}$, follow the ordering

$$\xi_{J-1} < \xi_{J,1} < \xi_{J,2} < \cdots < \xi_{J,N} < \xi_J, \qquad J = 1, 2, \ldots n, \tag{4.22}$$

and are given by

$$\xi_{J,C} = S_J \left(\frac{1}{2N} + \frac{C-1}{N} \right) + \xi_{J-1}, \qquad J = 1, \ldots, n, \quad C = 1, \ldots, N.$$

see Figure 4.2 for a diagrammatic representation of the collocation scheme.

With equation (4.20), the discrete form of equation (4.1) can be written as

$$V_{p,J,C} = \sum_{m=0}^{N-1} \left\{ F^{(m)}_{q^\dagger,0} A_{p,q^\dagger,1,m+1,J,C} + \sum_{q=1}^{2} \sum_{i=1}^{n-1} F^{(m)}_{q,i} E_{p,q,i,m,J,C} + F^{(m)}_{q^\dagger,n} A_{p,q^\dagger,n,m+N+1,J,C} \right\}, \tag{4.23}$$

where

$$A_{p,q,j,k,J,C} = B_{p,q,j,k}(\xi^*_{J,C}), \qquad E_{p,q,j,m,J,C} = D_{p,q,j,m}(\xi^*_{J,C}), \qquad V_{p,J,C} = V_p(\xi^*_{J,C}), \tag{4.24}$$

and $q^\dagger = \frac{1}{2}\left(3 - (-1)^m\right)$.

Equations (4.23) can be recast in matrix form as

$$\mathcal{A} \times \mathcal{F} = \mathcal{V}. \tag{4.25}$$

The layout of the solution array $\mathcal{F}$ and the velocity array $\mathcal{V}$ are shown in Figures 4.3 and 4.4 respectively. Thus, for given values of p, q, J, C the matrix of coefficients has the form given in Figure 4.5.

$$
\mathcal{F} = \left[\begin{array}{l}
\left.\begin{array}{l}
F_{1,0} \\ F_{1,0}^{(2)} \\ F_{1,0}^{(4)} \\ \vdots \\ F_{1,0}^{(N-2)}
\end{array}\right\} q=1 \\
\left.\begin{array}{l}
F_{2,0}^{(1)} \\ F_{2,0}^{(3)} \\ \vdots \\ F_{2,0}^{(N-1)}
\end{array}\right\} q=2
\end{array}\right\} i=0
\left[\begin{array}{l}
\left.\begin{array}{l}
F_{1,1} \\ F_{1,1}^{(1)} \\ F_{1,1}^{(2)} \\ \vdots \\ F_{1,1}^{(N-1)}
\end{array}\right\} q=1 \\
\left.\begin{array}{l}
F_{2,1} \\ F_{2,1}^{(1)} \\ F_{2,1}^{(2)} \\ \vdots \\ F_{2,1}^{(N-1)}
\end{array}\right\} q=2
\end{array}\right\} i=1
\left[\begin{array}{l}
\vdots
\end{array}\right.
\left[\begin{array}{l}
\left.\begin{array}{l}
F_{1,n-1} \\ F_{1,n-1}^{(1)} \\ F_{1,n-1}^{(2)} \\ \vdots \\ F_{1,n-1}^{(N-1)}
\end{array}\right\} q=1 \\
\left.\begin{array}{l}
F_{2,n-1} \\ F_{2,n-1}^{(1)} \\ F_{2,n-1}^{(2)} \\ \vdots \\ F_{2,n-1}^{(N-1)}
\end{array}\right\} q=2
\end{array}\right\} i=n-1
\left[\begin{array}{l}
\left.\begin{array}{l}
F_{1,n} \\ F_{1,n}^{(2)} \\ F_{1,n}^{(4)} \\ \vdots \\ F_{1,n}^{(N-2)}
\end{array}\right\} q=1 \\
\left.\begin{array}{l}
F_{2,n}^{(1)} \\ F_{2,n}^{(3)} \\ \vdots \\ F_{2,n}^{(N-1)}
\end{array}\right\} q=2
\end{array}\right\} i=n
$$

Figure 4.3: The form of the solution array (traction at the node points) of the matrix equation (4.25).

$$
\mathcal{V} = \left[\begin{array}{ll}
V_{1,1,1} & p=1 \\
V_{2,1,1} & p=2 \\
V_{1,1,2} & p=1 \\
V_{2,1,2} & p=2 \\
\vdots & \\
V_{1,1,N} & p=1 \\
V_{2,1,N} & p=2 \\
V_{1,2,1} & p=1 \\
V_{2,2,1} & p=2 \\
V_{1,2,2} & p=1 \\
V_{2,2,2} & p=2 \\
\vdots & \\
V_{1,2,N} & p=1 \\
V_{2,2,N} & p=2 \\
\vdots & \\
V_{1,n,1} & p=1 \\
V_{2,n,1} & p=2 \\
V_{1,n,2} & p=1 \\
V_{2,n,2} & p=2 \\
\vdots & \\
V_{1,n,N} & p=1 \\
V_{2,n,N} & p=2
\end{array}\right]
\begin{array}{l}
\left.\begin{array}{l} \\ \\ \end{array}\right\} C=1 \\
\left.\begin{array}{l} \\ \\ \end{array}\right\} C=2 \\
\\
\left.\begin{array}{l} \\ \\ \end{array}\right\} C=N \\
\left.\begin{array}{l} \\ \\ \end{array}\right\} C=1 \\
\left.\begin{array}{l} \\ \\ \end{array}\right\} C=2 \\
\\
\left.\begin{array}{l} \\ \\ \end{array}\right\} C=N \\
\\
\left.\begin{array}{l} \\ \\ \end{array}\right\} C=1 \\
\left.\begin{array}{l} \\ \\ \end{array}\right\} C=2 \\
\\
\left.\begin{array}{l} \\ \\ \end{array}\right\} C=N
\end{array}
\begin{array}{l}
\left.\begin{array}{l} \\ \end{array}\right\} J=1 \\
\left.\begin{array}{l} \\ \end{array}\right\} J=2 \\
\left.\begin{array}{l} \\ \end{array}\right\} J=n
\end{array}
$$

Figure 4.4: The form of the velocity array (velocity at the collocation points) of the matrix equation (4.25).

The most crucial part of the algorithm is the efficient and accurate calculation of the elements in $\mathcal{A}$. When the collocation point does not lie on the line segment being integrated, the integrand is analytic and the integral can be evaluated using a 16 point Gauss-Legendre quadrature rule. However, when the collocation point does lie on the line segment being integrated, the integrand has a logarithmic singularity and therefore must be evaluated using the appropriate combination of Gauss-Legendre and Gauss-Log quadrature rules (Stroud and Secrest 1966). The reader is referred to Taib (1985) for a detailed description of the numerical integration.

Equation (4.25) can now be solved using readily available software.

$$
\begin{array}{lll}
i = 0 & & \\
\quad (\mathcal{A})_{\alpha,\beta} & = A_{p,q,1,q,J,C} & \\
\quad (\mathcal{A})_{\alpha,\beta+1} & = A_{p,q,1,q+2,J,C} & \alpha = 2N(J-1) + 2(C-1) + p \\
\quad (\mathcal{A})_{\alpha,\beta+2} & = A_{p,q,1,q+4,J,C} & \beta = \frac{N}{2}(q-1) + 1 \\
\quad \vdots & \vdots & \\
\quad (\mathcal{A})_{\alpha,\beta+N/2+1} & = A_{p,q,1,q+N-2,J,C} & \\
1 \leq i \leq N-1 & & \\
\quad (\mathcal{A})_{\alpha,\beta} & = E_{p,q,i,0,J,C} & \\
\quad (\mathcal{A})_{\alpha,\beta+1} & = E_{p,q,i,1,J,C} & \alpha = 2N(J-1) + 2(C-1) + p \\
\quad (\mathcal{A})_{\alpha,\beta+2} & = E_{p,q,i,2,J,C} & \beta = 2N(i-1) + qN + 1 \\
\quad \vdots & \vdots & \\
\quad (\mathcal{A})_{\alpha,\beta+N-1} & = E_{p,q,i,N-1,J,C} & \\
i = N & & \\
\quad (\mathcal{A})_{\alpha,\beta} & = A_{p,q,N,N+q,J,C} & \\
\quad (\mathcal{A})_{\alpha,\beta+1} & = A_{p,q,N,N+q+2,J,C} & \alpha = 2N(J-1) + 2(C-1) + p \\
\quad (\mathcal{A})_{\alpha,\beta+2} & = A_{p,q,N,N+q+4,J,C} & \beta = 2Nn - N + \frac{N}{2}(q-1) + 1 \\
\quad \vdots & \vdots & \\
\quad (\mathcal{A})_{\alpha,\beta+N/2+1} & = A_{p,q,N,2N+q-2,J,C} &
\end{array}
$$

Figure 4.5: The form of the matrix of coefficients $(\mathcal{A})_{\alpha,\beta}$ of the matrix equation (4.25).

4.3 Translating spheroid

In this section the algorithm, represented by the formation and inversion of (4.25), is tested by calculating the traction of a single spheroid translating uniformly in a Stokes fluid field. The spheroid is parameterised with respect to arclength and different discretization schemes (4.3) and interpolation orders, N, are investigated. Improvements in the discretization, based on curvature, are also presented and examined.

Youngren and Acrivos (1975) were the first to solve this problem. They used constant elements for the traction and Simpson's rule to evaluate the integrals over each interval. The singular nature of the integrands were only accounted for when they became unbounded. In this case, the integrand was expanded in a small series about the singular point and the integral evaluated analytically for a small region. Their results were acceptable for moderate eccentricities and they reported no ill-conditioning in the first kind equation. It is envisaged that higher eccentricities,

or more general particle shapes would have severely challenged their method.

In more recent studies, Gauss-Legendre quadrature and linear interpolation have been employed. Curiously, the scheme of analytic evaluation about a small cut has been retained. This not only adds to the complexity of the method, but discounts the possibility of using the correct Gaussian quadrature rules; that is, a linear combination of Gauss-Legendre and Gauss-Log, to handle the singular integral. We refer to section D.2 of appendix D for a more detailed discussion of this point.

We choose to parameterise with respect to arclength as this becomes more important for complicated geometries and especially for deformable surfaces such as the drop interactions considered in chapter 5. The advantages of arclength parameterisation are outlined below and on page 105.

Hermite interpolation order will be investigated as will node point distribution. Both are effected by the eccentricity of the spheroid, or more generally the curvature of the particle. To counter these effects two curvature based methods are presented. The first aims to reduce the number of node points by placing them to uniformly approximate the unknown. The second, introduced in section 4.4, is designed to reduce the large gradients in the unknown.

The spheroid is a suitable test surface, since it's traction has a simple analytic expression (see equations (4.38) and (4.39)). In addition, the surface itself can possess high curvatures given large or small aspect ratios. Figure 4.6 shows a schematic of a spheroidal body. In cylindrical coordinates, the surface is parameterised by

$$z = a\sin\theta, \quad r = b\cos\theta, \quad \text{for} \quad -\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}. \tag{4.26}$$

We define the aspect ratio λ and eccentricity e to be respectively

$$\lambda = \frac{b}{a}, \qquad e = \begin{cases} \dfrac{\sqrt{a^2-b^2}}{a} = \sqrt{1-\lambda^2}, & \lambda \leq 1, \\ \dfrac{\sqrt{b^2-a^2}}{b} = \dfrac{\sqrt{\lambda^2-1}}{\lambda}, & \lambda > 1. \end{cases} \tag{4.27}$$

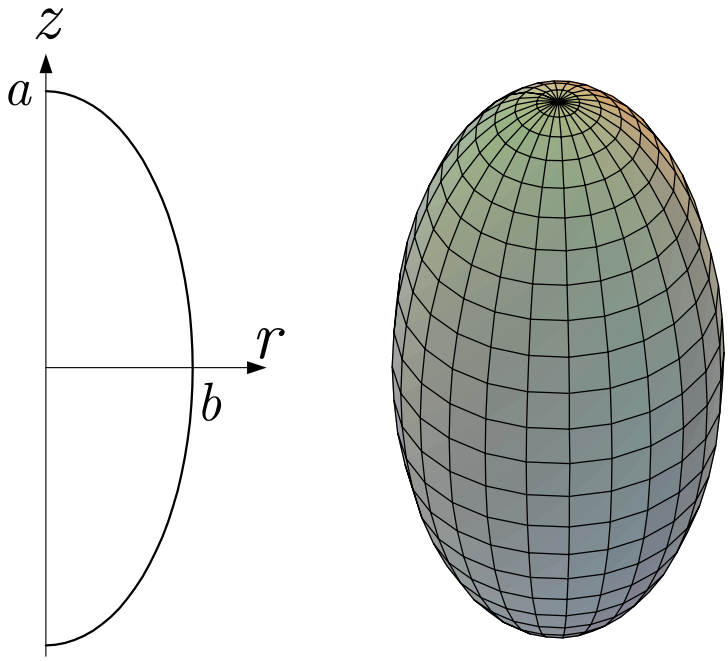


Figure 4.6: Geometry of a spheroidal body using a cylindrical coordinate system. The surface is parameterised by $z = a\sin\theta, r = b\cos\theta, -\pi/2 \leq \theta \leq \pi/2$.

The arclength ξ is given by

$$\xi = \int_{t_0}^{t_1} \sqrt{\left(\frac{dr}{dt}\right)^2 + \left(\frac{dz}{dt}\right)^2}\, dt \tag{4.28}$$

$$= \int_{-\pi/2}^{\theta} \sqrt{a^2\cos^2 t + b^2 \sin^2 t}\, dt = \begin{cases} bE\left(\theta + \frac{\pi}{2}; e\right), & \theta \leq 0,\ \lambda > 1, \\ aE(e) - E(-\theta; e), & \theta \leq 0,\ \lambda \leq 1, \\ b\left[2E(e) - E\left(\frac{\pi}{2} - \theta; e\right)\right], & \theta > 0,\ \lambda > 1, \\ a\left[E(e) + E(\theta; e)\right], & \theta > 0,\ \lambda \leq 1, \end{cases} \tag{4.29}$$

where $E(\cdot)$ and $E(\cdot;\cdot)$ are respectively the complete and incomplete elliptic integrals of the second kind (Abramowitz and Stegun 1965, p. 589). We note that there are some advantages to parameterising with respect to arclength, for example the parameterisation is unique in that the mapping is bijective, the Jacobian is of unit size and the normal vector is defined in a consistent manner. It is a relatively simple matter to evaluate (4.28) for a given general curve. For a curve defined via piecewise functions, as in a spline, then a method devised by Kucera (1992) can be used. This is further described in chapter 5 where drop deformation and interaction is investigated.

The volume of the particle is constrained to that of a unit sphere. Thus, we have

$$a = \frac{1}{\left(1-e^2\right)^{1/3}}, \quad b = \left(1-e^2\right)^{1/6}, \qquad \lambda \leq 1, \tag{4.30}$$

$$a = \left(1-e^2\right)^{1/3}, \quad b = \frac{1}{\left(1-e^2\right)^{1/6}}, \qquad \lambda > 1. \tag{4.31}$$

4.3.1 Discretization

In sub-sections 4.2.1 and 4.2.2 we use some discretization (4.3) and Hermite interpolation (4.15) to evaluate the unknown traction. This involves choosing node points so that the traction is reasonably well approximated in each sub-domain. In practice, for polynomial interpolation, this means using enough node points so that the function is almost linear in the relevant domain. This will always be true if the unknown is analytic and the domain is small. Another strategy commonly used is to increase the order of the interpolating functions, so that less node points are required.

Given a fixed number of node points, it is desirable that they are positioned so that they cluster around large rates of change. In this way, the function is *almost linear* in the sub-domain and the interpolation will generally be improved. Since the traction is unknown, we will use the particle shape to guide our clustering regime. The justification for this is that the traction, by its very nature, is highly shape dependent (force per unit length on the surface). Large values of the surface curvature correspond to large tractions. In fact for a spheroid, the traction is directly proportional to the cube root of its curvature.

Let $(r(t), z(t))$ be a space-curve, C, that represents the projection of a body in the polar plane. We will investigate the effect of discretizing C uniformly with respect to an arbitrary parameter t. To this end, define

$$\zeta(t,h) = \sqrt{\left(r(t+h)-r(t)\right)^2 + \left(z(t+h)-z(t)\right)^2}$$

to be the Euclidean distance between node points. A diagram is shown in Figure 4.7.

For small h, ζ will approximate the arclength. We now define the *density* ρ to be the reciprocal of ζ

$$\begin{aligned} \rho(t,h) &:= \frac{1}{\zeta(t,h)} \\ &\sim \frac{1}{h\sqrt{\left(\frac{dr}{dt}\right)^2 + \left(\frac{dz}{dt}\right)^2}} + O(1), \quad h \to 0. \end{aligned} \tag{4.32}$$

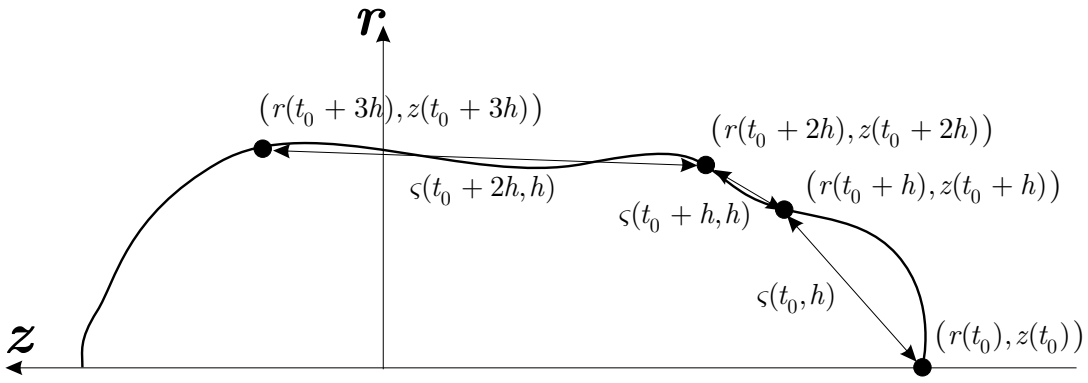


Figure 4.7: Discretizing a space-curve uniformly with respect to the parameter t. ζ is the Euclidean distance.

The density (4.32) provides a quantitative description of the clustering for any general parameter t. Relatively large values of ρ represents high clustering measured via the arclength. Low values of ρ signifies node points spaced further apart. Four simple methods for discretizing the spheroid (4.26) are investigated. They are, uniformly with respect to

1. the axial (z) coordinate,
2. the radial (r) coordinate,
3. θ as defined in (4.26),
4. the arclength ξ (4.29).

Uniform z discretization

The parametric equations are

$$z = z, \quad r = \lambda\sqrt{a^2 - z^2}, \qquad -a \leq z \leq a.$$

With (4.32) the density for uniform z discretization is

$$\rho(z, h) \sim \frac{1}{h}\sqrt{\frac{a^2 - z^2}{\left(\lambda^2 - 1\right)z^2 + a^2}} + O(1), \quad -a \leq z \leq a. \tag{4.33}$$

In order to compare each different scheme, we make use of (4.26) to write the density in terms of θ. Thus (4.33) becomes

$$\rho(z(\theta), h) = \frac{\cos\theta}{h\sqrt{1 - (1 - \lambda^2)\sin^2\theta}}, \quad -\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}. \tag{4.34}$$

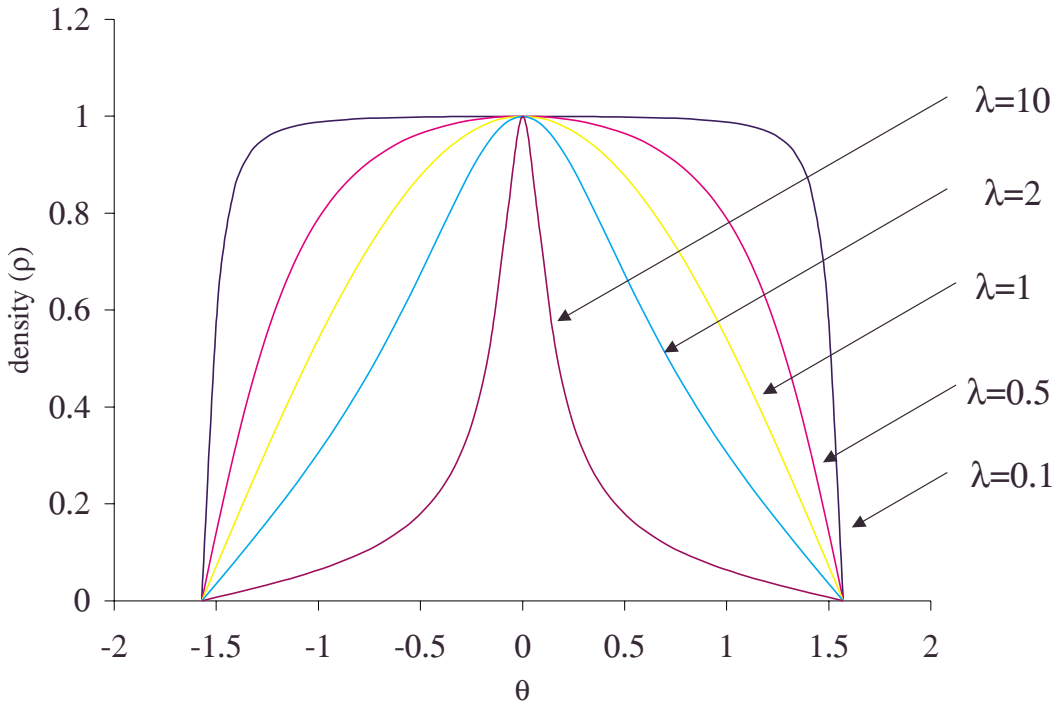


Figure 4.8: Plots of density (equation (4.34)) for various values of the aspect ratio λ.

Plots of the density (equation (4.34)) are shown in Figure 4.8. For all values of λ, the graph increases monotonically for $-\pi/2 \leq \theta \leq 0$ and decreases for $0 \leq \theta \leq \pi/2$. For very prolate spheroids, the density is almost uniformly 1, for very oblate spheroids the density tends to zero everywhere except for a strong peak at $\theta = 0$. This is as expected since as $\lambda \to 0$, the spheroid tends to the z axis, and as $\lambda \to \infty$ the spheroid tends to the r axis. The peak near $\theta = 0$ indicates strong clustering near the radial axis and node points are sparse near the z axis. This would be a desirable feature for oblate spheroids ($\lambda > 1$) but not prolate ($0 < \lambda < 1$).

Uniform r discretization

The parametric equations are

$$z = \pm\frac{1}{\lambda}\sqrt{b^2 - r^2}, \quad r = r, \qquad 0 \leq r \leq b,$$

and the density for uniform r discretization is

$$\rho(r,h) \sim \frac{\lambda}{h}\sqrt{\frac{b^2 - r^2}{(1-\lambda^2)r^2 + \lambda^2 b^2}} + O(1), \quad 0 \leq r \leq b. \tag{4.35}$$

In terms of θ, using (4.26), we have

$$\rho(r(\theta),h) = \frac{\lambda|\sin\theta|}{h\sqrt{1-(1-\lambda^2)\sin^2\theta}}, \quad -\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}. \tag{4.36}$$

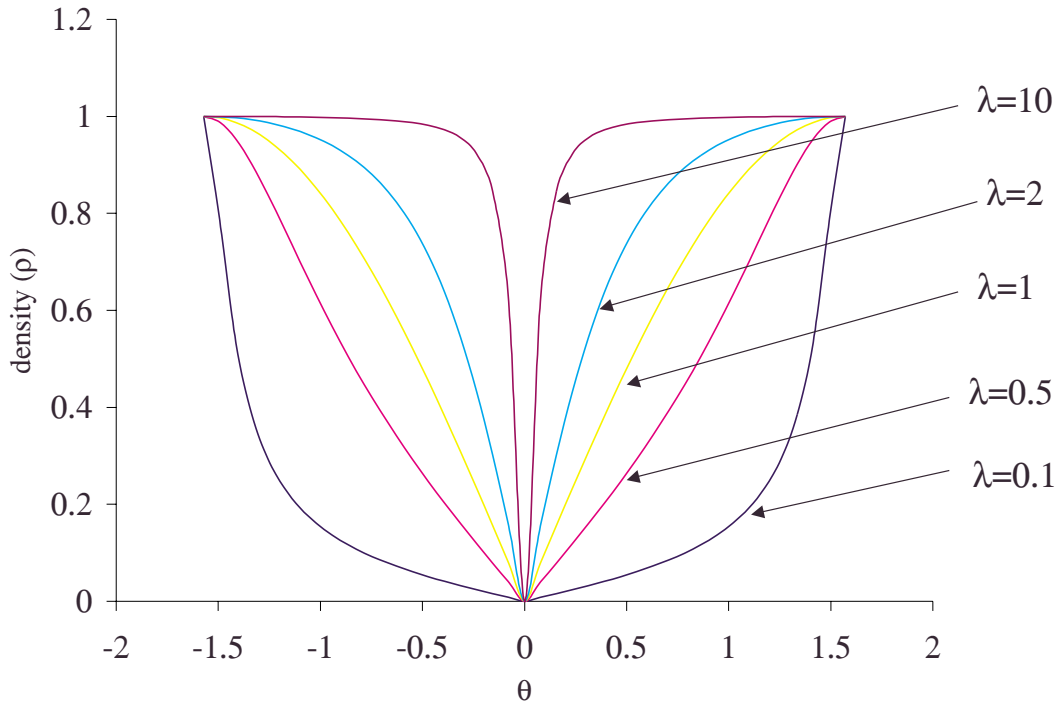


Figure 4.9: Plots of the density (equation (4.36)) for various values of the aspect ratio λ.

Plots of the density (equation (4.36)) are shown in Figure 4.9 for various values of λ. Figure 4.9 appears as an inverted version of Figure 4.8. That is, this scheme features clustering near the z axis; a desirable property for prolate spheroids but not oblate.

Uniform θ discretization

The parametric equations are given in (4.26). The density function is

$$\rho(\theta, h) = \frac{1}{ah\sqrt{1-(1-\lambda^2)\sin^2\theta}}, \quad -\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}, \tag{4.37}$$

and is shown in Figure 4.10. This discretization scheme exhibits the desirable features of the previous two. That is, for prolate spheroids clustering occurs near the symmetry axis (z), and for oblate spheroids the clustering is near the radial axis. We note that for $\lambda = 1$, the density is uniform. This is because at this aspect ratio the particle is a sphere and θ acts as the arclength.

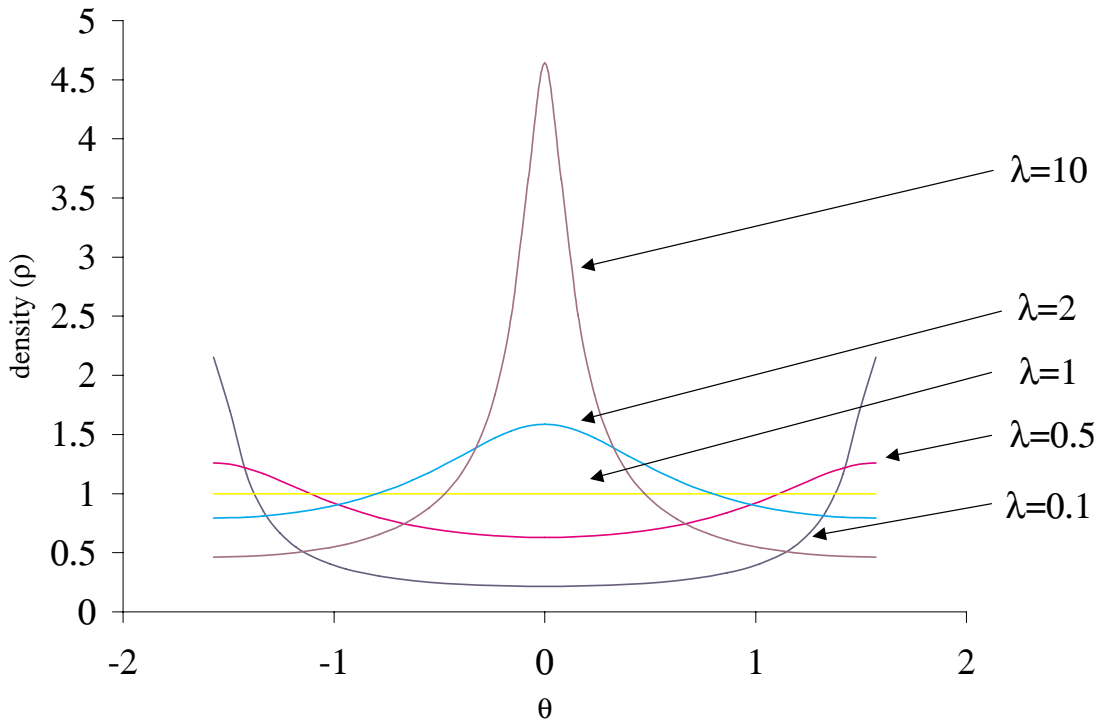


Figure 4.10: Plot of the density (equation (4.37))f for various values of the aspect ratio λ.

Uniform arclength ξ discretization

If the space-curve C is parameterised with respect to arclength then

$$\left(\frac{dr}{d\xi}\right)^2 + \left(\frac{dz}{d\xi}\right)^2 = 1, \qquad \text{for all } \xi.$$

Thus $\rho(\xi, h) = 1/h$ regardless of λ. Obviously, this is also true for any general particle and no obvious clustering would occur. In this sense, this would be the worst discretization scheme of the four.

The obvious scheme to employ is θ discretization. It has the advantage of being trivial to implement while exhibiting the beneficial clustering for prolate and oblate shapes. In sub-section 4.3.3, we will introduce a more general discretization scheme to cater for arbitrary shapes based on an arclength parameterisation and a curvature dependent rule. This method far outperforms all those listed here but has the disadvantage of requiring a much longer compute time to pre-process the node point distribution.

4.3.2 Error in the traction of a prolate spheroid

The errors in calculating the z component of traction for a spheroid with unit volume translating at a constant speed $V = 1$ are presented. The exact solution for the spheroid with unit volume is (Oberbeck 1876)

$$f_1 = 2V \frac{e^3}{(1-e^2)^{1/6}\left(\frac{1+e^2}{2}\ln(\frac{1+e}{1-e}) - e\right)} \frac{1}{\sqrt{1-e^2\sin^2\theta}}, \quad f_2 = 0, \qquad -\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}, \quad (4.38)$$

for $\lambda \leq 1$ and

$$f_1 = -2V \frac{e^{1/3}\delta^{4/3}}{((1-\delta^2)\tan^{-1}(\delta) - \delta)\sqrt{1+\delta^2\sin^2\theta}}, \quad f_2 = 0, \qquad -\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}, \qquad (4.39)$$

for $\lambda > 1$; where f_1 is axial (z) component of traction, f_2 the radial (r) component of traction and $\delta = \sqrt{e^2/(1-e^2)}$.

Plots of the traction for a prolate spheroid are shown in Figure 4.11. Each plot possesses a large gradient near $\theta = \pm\pi/2$ that correspond to points of high curvature. The gradient of the traction increases as the eccentricity. Figure 4.12 shows the first two derivatives of the traction for $e = 0.9$. Inspection of equation (4.38), as well as Figure 4.12, indicates that each successive derivative increases in magnitude near $\theta = \pm\pi/2$. The traction is everywhere analytic for $0 \leq e < 1$, but the observations above suggest that interpolation schemes that attempt to calculate the higher order derivatives may not succeed.

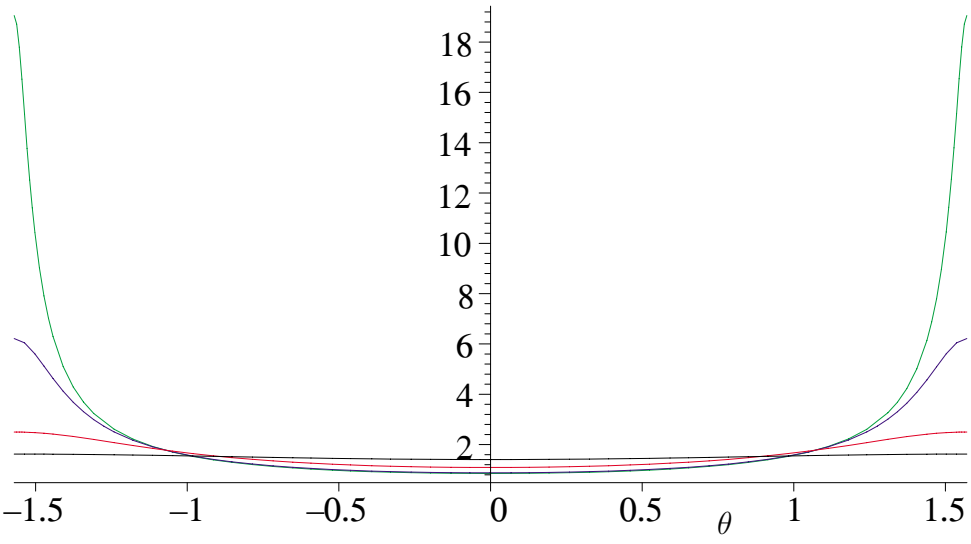


Figure 4.11: Plot of the traction for a prolate spheroid of unit volume with $V = 1$. — ($e = 0.5$), — ($e = 0.9$), — ($e = 0.99$), — ($e = 0.999$).

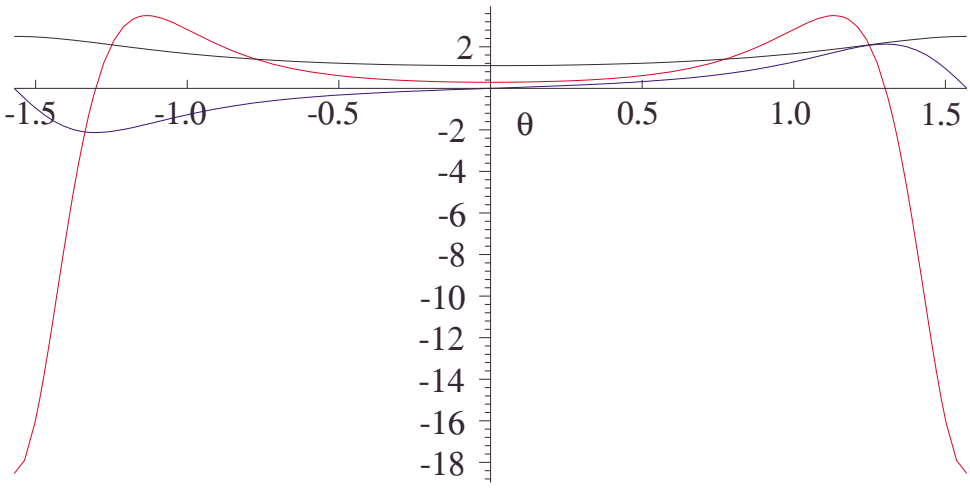


Figure 4.12: Plot of the derivatives of the traction, $df_1/d\xi$ as a function of θ for a prolate spheroid of unit volume with $V = 1$ and $e = 0.9$. — (function value), — (first derivative), — (second derivative).

Figure 4.13 shows the maximum and average relative error in computing f_1 with cubic Hermite interpolation ($N = 2$) for different grid sizes (n). Numerical values of the error are tabulated in Table 4.1. Unless otherwise stated, the grid is chosen uniformly with respect to θ. It is clear that the algorithm performs quite well for almost all values of n. Up to $n = 64$ both errors decrease by approximately an order of magnitude as the step size n doubles. From $n = 128$ onward, the error begins to oscillate but the algorithm does not breakdown.

As shown, first kind equations are intrinsically ill-conditioned. Collocation-interpolation methods manifest this ill-conditioning in one of two ways; they both involve reducing grid size. As n becomes large, the segment lengths S_j (equation (4.6)) become small. Hence every element in the coefficient matrix $\mathcal{A}$ tends to zero. This is easily seen by inspection of equations (4.17), (4.18), (4.24) and Figure 4.5. Thus, with finite precision there exists an upper limit for n in which $\mathcal{A}$ (see Figure 4.5) can be represented with sufficient accuracy. It would be unreasonable, though, to suggest this as the cause of the erratic error behaviour for $n = 128$–1024 since all calculations have been performed with double precision (16 digit) accuracy. Another reason for the breakdown involves the numerical inversion of $\mathcal{A}$. As each interval becomes small the collocation points $\xi_{J,C}$ will tend to cluster. Recall that each row of $\mathcal{A}$ is simply the discrete boundary integral equation (4.23) evaluated at a collocation point. Hence, successive rows will tend to be similar, leading to an ill-conditioned linear system.

Figure 4.14 shows the maximum relative error for different interpolation orders and number of grid points. For both $N = 4$ and $N = 6$, the error begins to oscillate after $n = 8$. This could

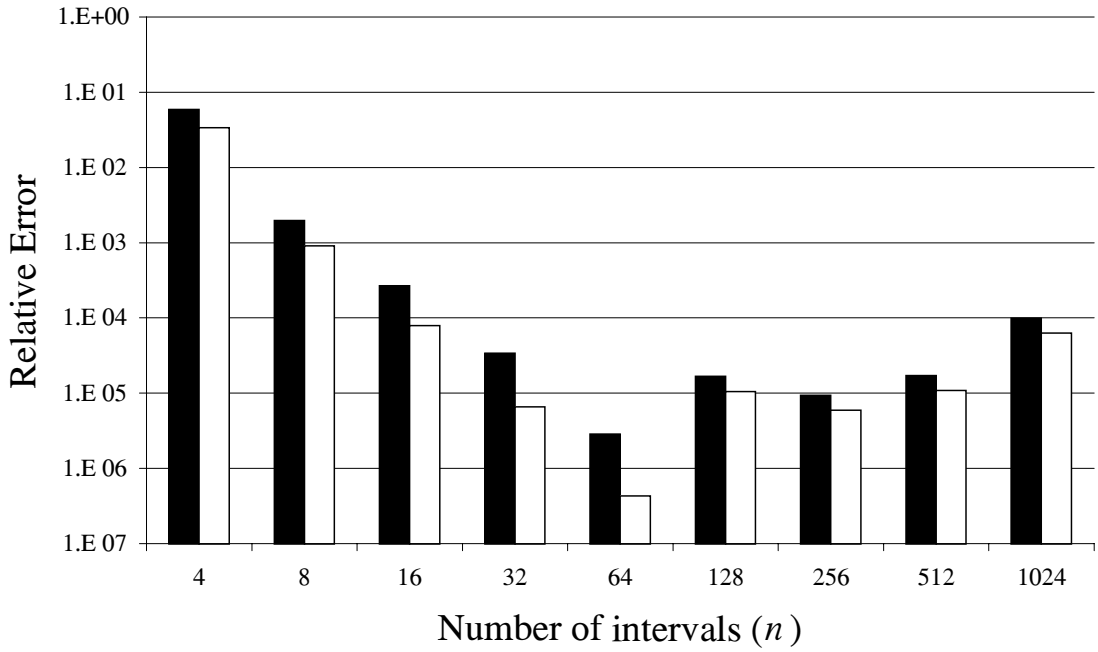


Figure 4.13: Relative error versus number of elements in calculating the axial component of traction of a spheroid, $e = 0.9, N = 2$. Maximum error (■), average error (□).

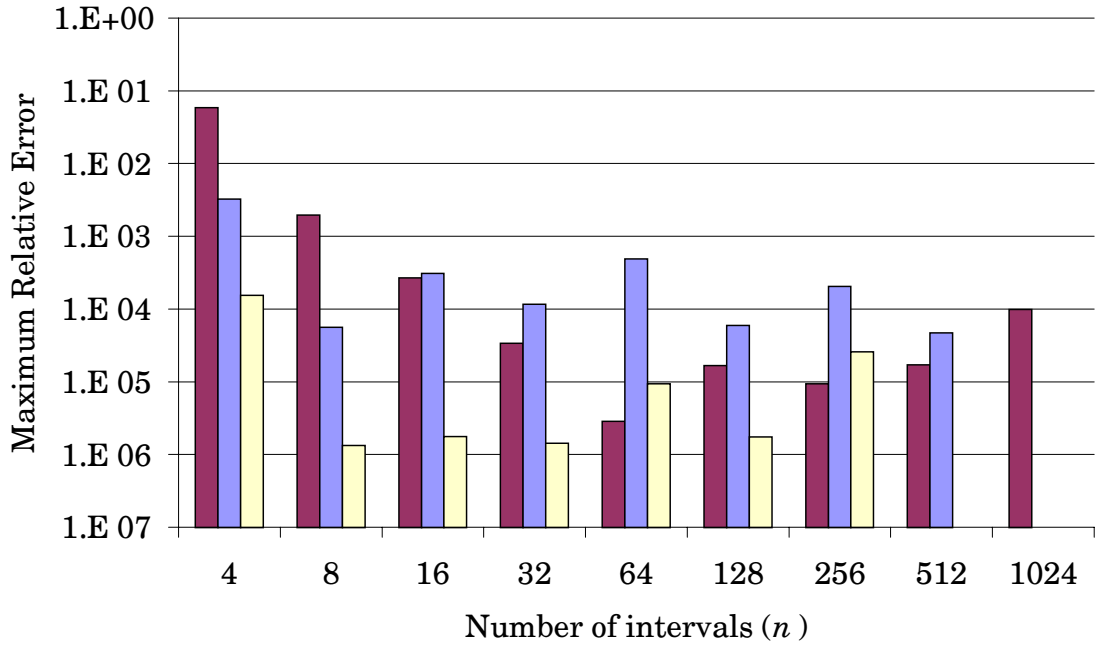


Figure 4.14: Maximum relative error versus number of elements in calculating the axial component of traction of a spheroid ($e = 0.9$) for different interpolation orders. $N = 2$ (■), $N = 4$ (■), $N = 6$ (■).

be due to the difficulty in approximating the higher derivative of the traction for this spheroid. When $N = 4$, the first three derivatives are calculated, for $N = 6$, the first five derivatives are calculated.

An indicator of numerical difficulties for linear systems is the condition number. Inspecting

n	Hermite interpolation order N					
	2		4		6	
	Max err.	Avg. err.	Max err.	Avg. err.	Max err.	Avg. err.
4	5.88E-02	3.38E-02	3.25E-03	2.27E-03	1.55E-04	1.26E-04
8	1.97E-03	9.11E-04	5.58E-05	1.83E-05	1.34E-06	6.87E-07
16	2.69E-04	7.87E-05	3.10E-04	2.03E-04	1.77E-06	1.16E-06
32	3.38E-05	6.60E-06	1.17E-04	7.55E-05	1.43E-06	9.22E-07
64	2.86E-06	4.29E-07	4.91E-04	3.15E-04	9.35E-06	6.00E-06
128	1.67E-05	1.05E-05	5.97E-05	3.82E-05	1.74E-06	1.11E-06
256	9.36E-06	5.96E-06	2.06E-04	1.31E-04	2.59E-05	1.65E-05
512	1.71E-05	1.09E-05	4.72E-05	3.01E-05		
1024	9.91E-05	6.31E-05				

Table 4.1: Maximum and average relative error in calculating f_1 for a prolate spheroid ($e = 0.9$) for various values of n and N.

the reciprocal condition number versus number of segments, Figure 4.15, shows that the linear system can become ill-conditioned for large n and large N. The reciprocal condition number is a parameter returned from the LINPACK (Dongarra *et al.* 1979) sub-routine `DGECO`. This popular Fortran suite is highly regarded for its robust and efficient matrix manipulation routines.

We can see that with $N = 2$, the reciprocal condition number falls to and then below 10^{-13} after $n = 64$. At this value, it would be expected that a double precision inversion (or Gauss elimination) algorithm would be subject to significant round off. As seen in Figure 4.13, after $n = 64$ the error no longer converges, but begins to oscillate.

This phenomenon occurs earlier with $N = 4$. Inspection of Figure 4.14 reveals this oscillating behaviour from $n = 16$. Figure 4.15 shows that at this value the reciprocal condition number is near 10^{-14}.

The most accurate result is obtained with $n = 8, N = 6$ - a mere 96×96 linear system. Despite this, it seems as if this level of interpolation induces an ill-conditioning in the linear system for all values of n. From Figure 4.15 the reciprocal condition number is of order 10^{-13} or smaller. This is not unexpected since the problem of higher interpolation orders can be explained by considering the exact solution for the traction.

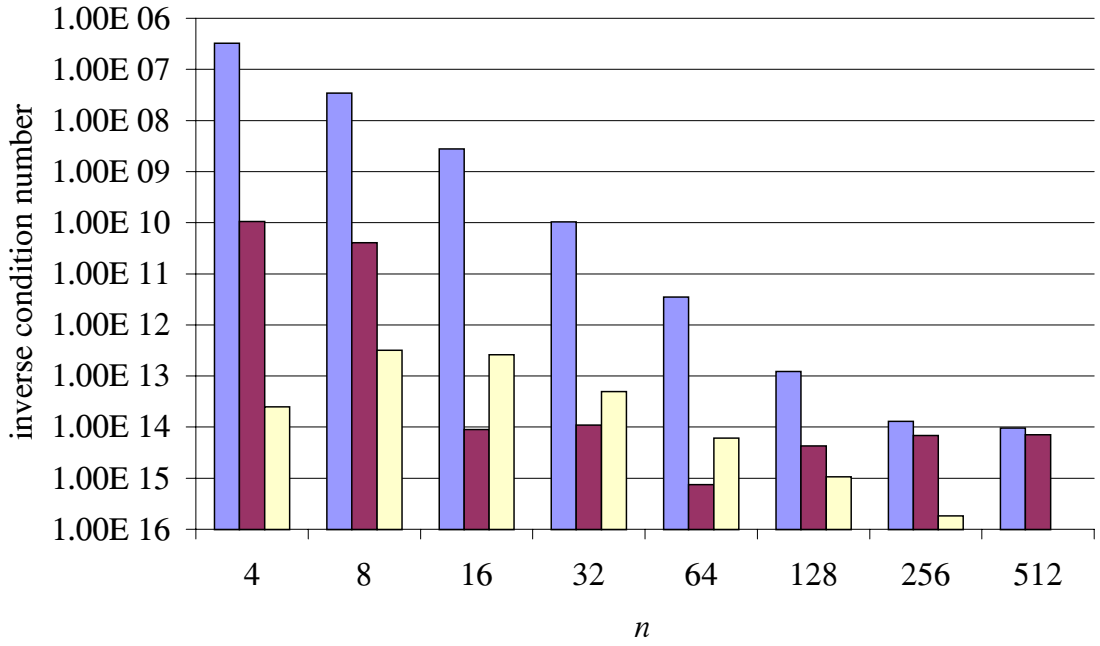


Figure 4.15: Reciprocal condition number of the matrix (4.25) for a prolate spheroid ($e = 0.9$) for different grid sizes (n) and interpolation orders (N). $N = 2$ ( ), $N = 4$ ( ), $N = 6$ ( ).

In the prolate case we have

$$f_1 = \frac{k}{\sqrt{a^2 - e^2 z^2}}, \qquad -a \leq z \leq a,$$

for some constant k and $0 \leq e < 1$. Each successive derivative of f_1 increases in magnitude. It has been shown in classical numerical analysis texts that high order interpolation rules fail to converge when used with such functions (see for example (Atkinson 1989, p. 158)). This argument extends to general surfaces. The curvature for a general curve parameterised by $\boldsymbol{r}(t)$ is

$$\kappa(t) = \frac{|\boldsymbol{r}' \times \boldsymbol{r}''|}{|\boldsymbol{r}'|^3}. \tag{4.40}$$

For many curves, each successive derivative of κ will increase. If one has any confidence in the claim that traction is highly dependent on curvature, then the answer to interpolation lies in employing low order polynomials on a denser grid.

The difference between the average error and maximum error in Figure 4.13 and Table 4.1 is due to the steep gradient of the traction at the endpoints $\xi = 0, \Xi$. In Figure 4.16 we plot the relative error of the traction at each node point for $n = 32$ and $n = 128$. For $n = 32$, we can see that the error at the endpoints (regions of high curvature) are an order of magnitude larger than those elsewhere. This difference is even more pronounced for $n = 128$ where the variation is approximately two orders of magnitude. Figure 4.17 shows that the difference between average

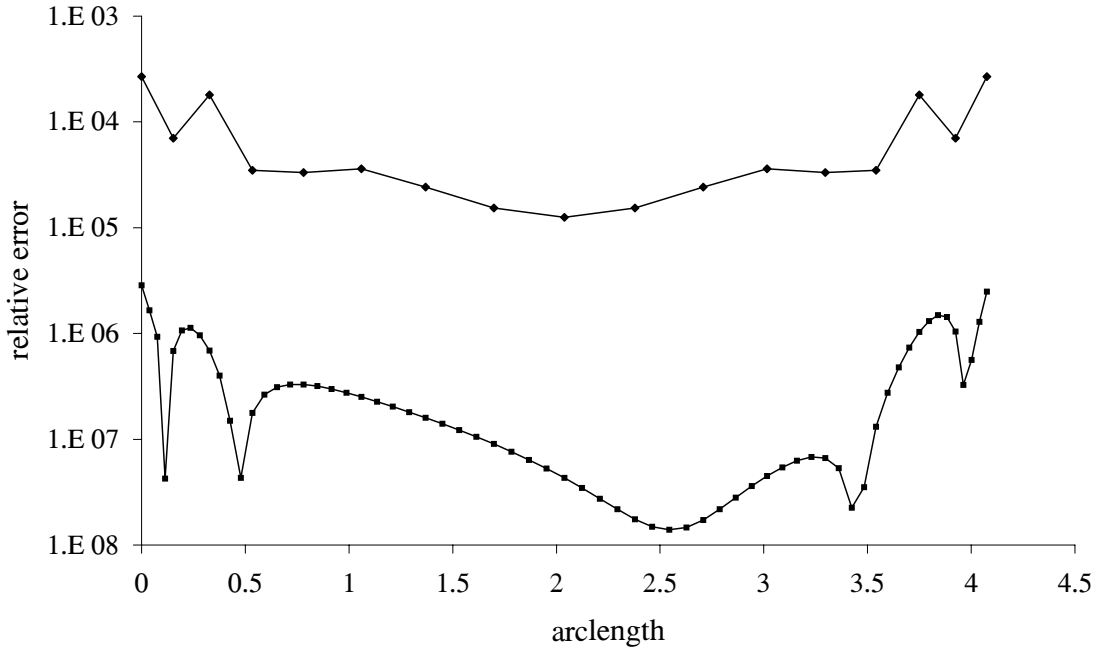


Figure 4.16: Comparison of relative error at each node point in the axial component of the traction for a spheroid ($e = 0.9$), $n = 32$ (♦), $n = 128$ (■). Interpolation order $N = 2$.

and maximum error generally increases in line with eccentricity. It is obvious that some clustering needs to occur in regions of high curvature in order to make the most efficient use of the node points available. The overriding principle in choosing an *a priori* grid is that the unknown is represented equally as well in all intervals. We attempt to address this point in the following sub-section.

4.3.3 Curvature discretization

The difference in the average and maximum error in Figure 4.17 and the relatively large variation in the error curves of Figure 4.16 suggests that improvements can be made in the distribution of the node points $\{\xi_i\}$.

We do not seek to improve the error at every node. Rather, the basis for an optimal node point distribution is that the unknown should be approximated uniformly across the entire grid. That is, error curves like that shown in Figure 4.16 should be relatively constant.

Recall the one dimensional integral equation (3.31), reproduced here with different parameters

$$g(\xi^*) = \int_0^{\Xi} K(\xi, \xi^*) f(\xi)\, d\xi, \qquad 0 \leq \xi^* \leq \Xi. \tag{4.41}$$

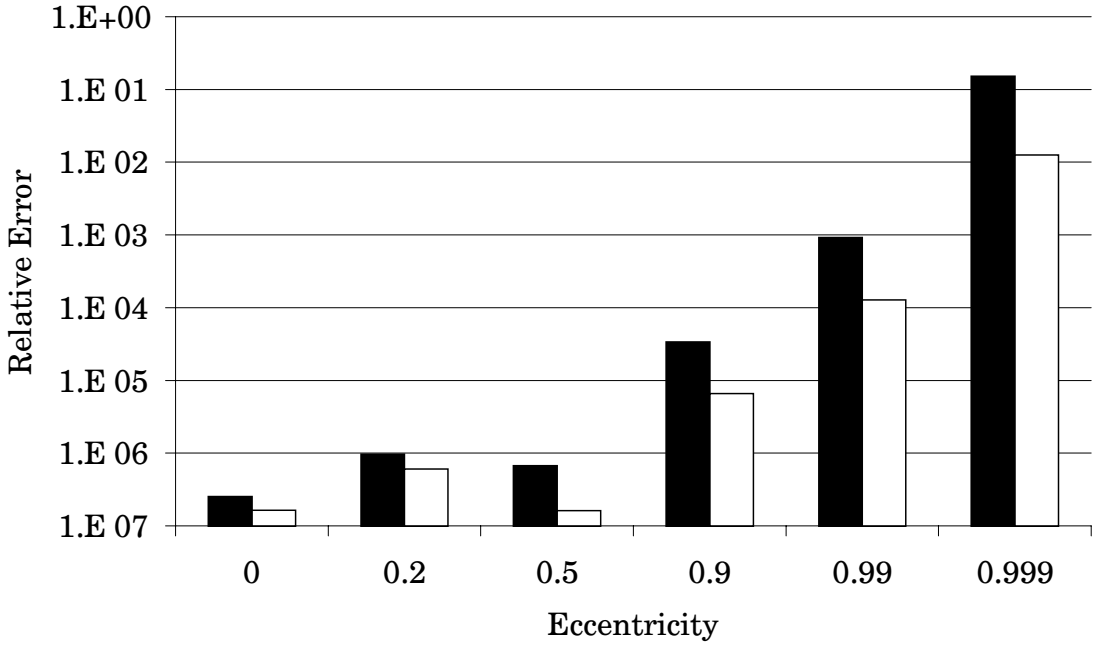


Figure 4.17: Relative error versus eccentricity in calculating the axial component of traction of a spheroid, $n = 32, N = 2$. *Maximum error (■), average error (□).*

Equation (4.41) may be considered as a parameterisation of the general boundary integral equation (3.1) along a curve $\Gamma(\xi) : \mathbb{R} \to \mathbb{R}^2$. Given ξ_0, f and f^*, we seek to find ξ_1 according to

$$\int_{\xi_0}^{\xi_1} K(\xi, \xi^*) \, |f(\xi) - f^*(\xi)| \, d\xi = \epsilon. \tag{4.42}$$

The function f^* is an approximation of f using some interpolation scheme and ϵ is the prescribed error in the interpolation. Once ξ_1 is found, (4.42) can be iterated by replacing ξ_1 with ξ_2 and ξ_0 with ξ_1. Since f is unknown, we surmise that κ is similar to f and hence can be used in its place. Thus, given a node point ξ_{i-1}, the next node ξ_i is the root of

$$F(\xi_i; \xi^*) = \int_{\xi_{i-1}}^{\xi_i} K(\xi, \xi^*) \, |\kappa(\xi) - \kappa^*(\xi)| \, d\xi - \epsilon = 0, \quad i = 1, 2, \ldots, \tag{4.43}$$

where ϵ is some small number, κ is the curvature of the curve Γ and κ^* the Hermite interpolant of κ. The number of nodes n is not given but is returned as part of the solution. The parameter ξ^* is free to be chosen in the domain $[0, \Xi]$.

We solve F using the modified Muller method (Blatt 1975). As in the bisection method, this algorithm is stable and requires a region in which the root lies, an obvious interval is $\xi_{i-1} < \xi_i < \Xi$. This root finding algorithm is employed since it has a faster rate of convergence than that of the bisection method.

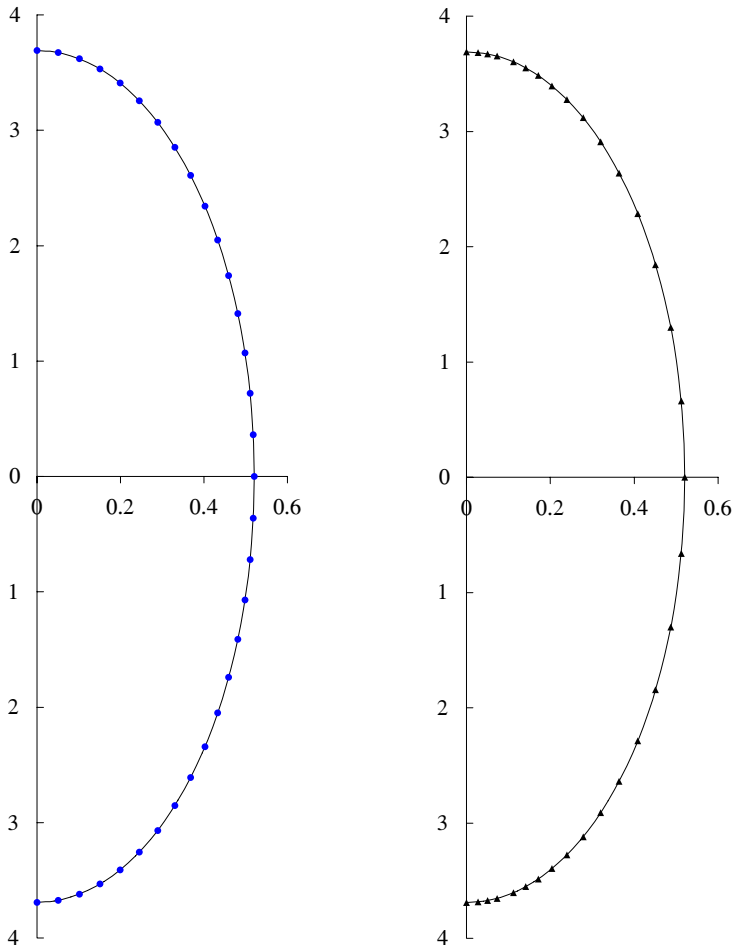


Figure 4.18: Spheroid ($e = 0.99$) showing uniform (●) and curvature (▲) discretization. $n = 32$.

Figure 4.18 compares a uniform-θ discretization with the curvature method described here for a prolate spheroid with $e = 0.99$. The node points are chosen so that the error in approximating the curvature with Hermite interpolation is uniform with $\epsilon = 10^{-5}$ in each interval. The clustering of nodes near high curvature is obvious.

In Figure 4.19 we compare the effect of using the two discretization schemes in solving the integral equation for an $\epsilon = 0.99$ prolate spheroid. With a uniform θ grid, the relative error in the traction spans approximately three orders of magnitude. In contrast, using the curvature grid results in an error distribution that spans $1\frac{1}{2}$ orders of magnitude and, more importantly, a maximum error that is reduced by over a half.

It should be noted that the evaluation of the grid (4.43) is considerably more time consuming than that for the θ parameter, but the distribution thus furnished may be stored and used, when required, without any extra effort.

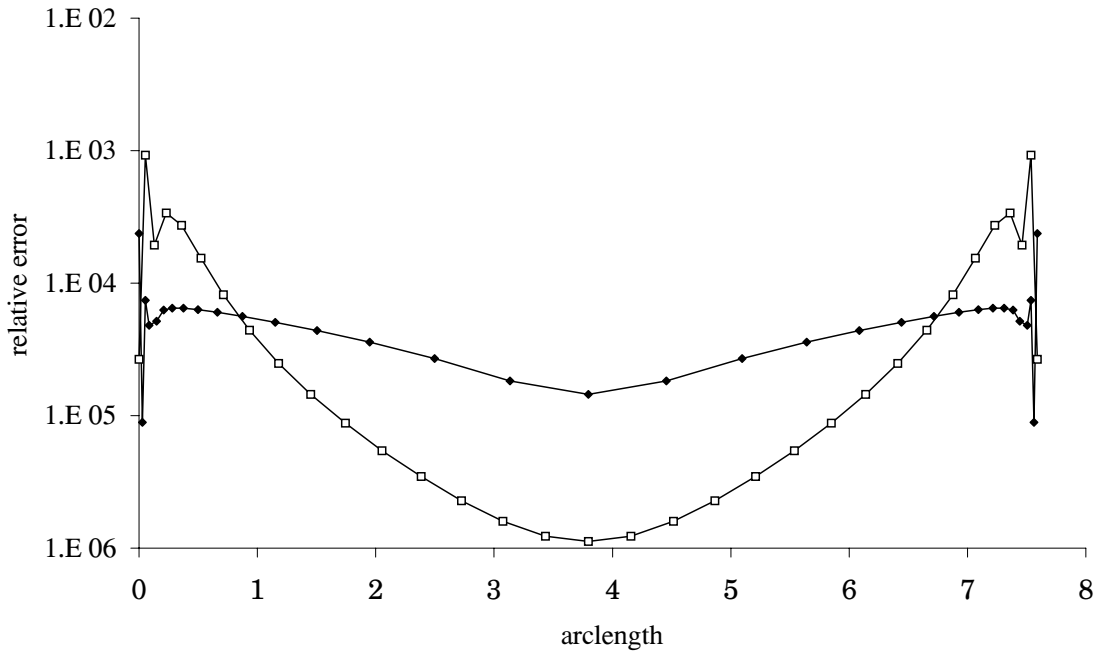


Figure 4.19: Relative error at each node point in the axial component of the traction for a spheroid ($e = 0.99, n = 32$) using different discretization schemes. Uniform (□), curvature (⧫).

In the previous section we showed that increasing the number of intervals is preferable to increasing interpolation order. The difficulty here is that the linear system $\mathcal{A}$ has size $2nN$. Storing $4n^2N^2$ double precision digits is very memory intensive. The ubiquitous *virtual memory* that is a feature of practically all 32-bit operating systems is of little help. For matrix inversion to succeed, the entire matrix must be stored in RAM. Hence this curvature based discretization scheme can be used to increase grid size (i.e. reduce the number of grid points) without adversely affecting the accuracy of the solution.

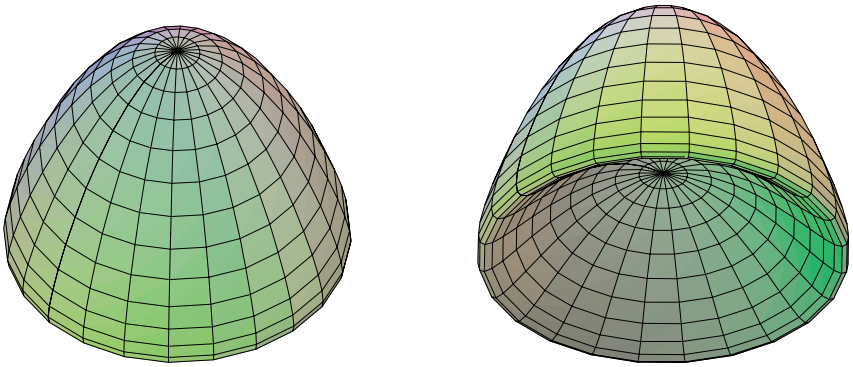
The ideas presented thus far are put to the test in the following section where a more complex surface is considered.

4.4 Difficult geometries

We consider the surface parameterised by

$$z = a\sin\theta - c\cos 2\theta, \quad r = b\cos\theta, \qquad -\pi/2 \le \theta \le \pi/2. \tag{4.44}$$

For lack of a better name, the surface (4.44) will be called surface A and a schematic is shown in Figure 4.20. In addition, its curvature is plotted in Figure 4.21. The diagrams clearly show that the curvature exhibits a very large peak and we expect its traction to preserve this property.



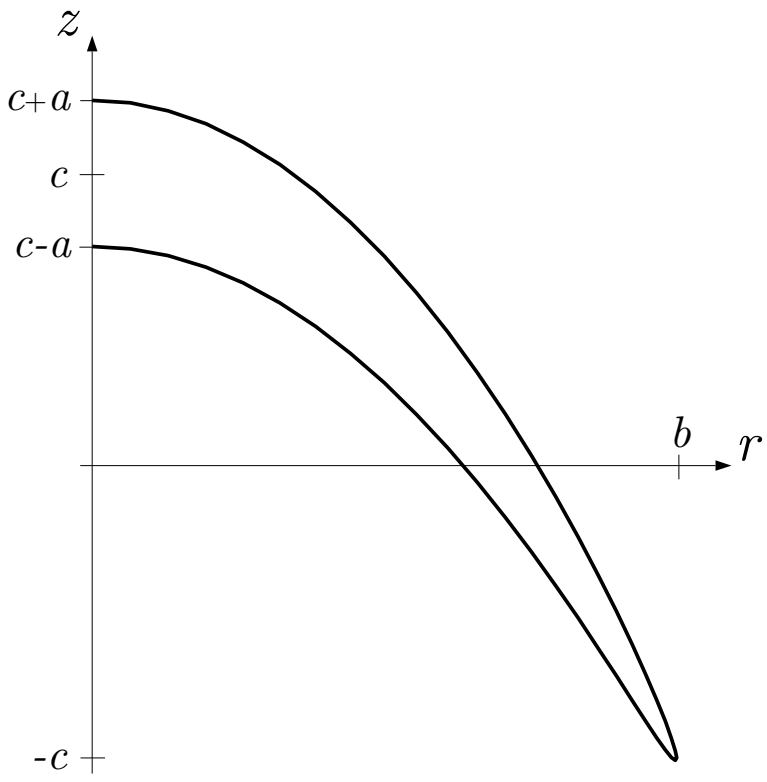


Figure 4.20: Schematic of test surface A. Parametric equations are $z = a\sin\theta - c\cos 2\theta, r = b\cos\theta, -\pi/2 \leq \theta \leq \pi/2$.

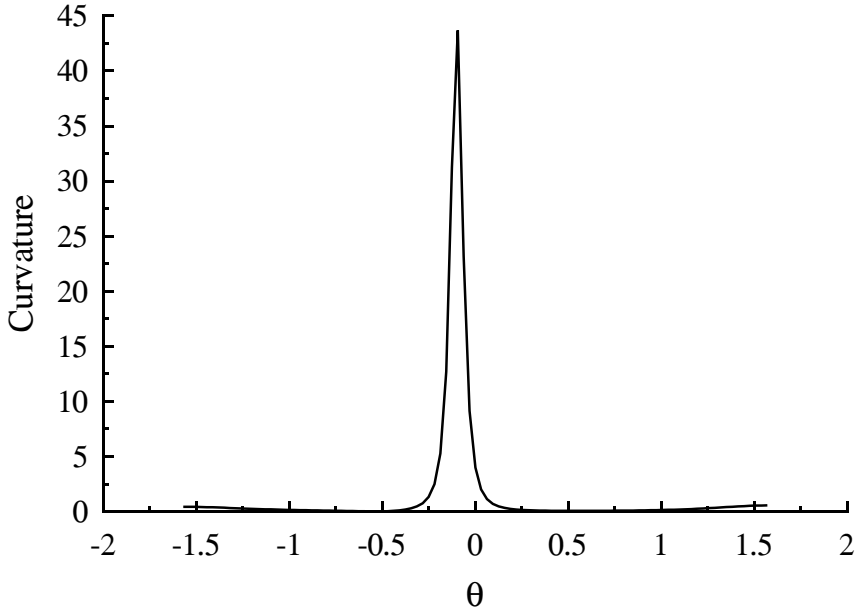


Figure 4.21: Curvature of surface A with $a = 1$, $b = 4$, $c = 2$.

As mentioned, we will employ the ideas of the previous section in an attempt to accurately calculate the traction of surface A. We know of no analytic solution, thus to begin Figure 4.22 shows the calculated traction for increasing number of node points. The points have been chosen uniformly with respect to the parameter θ in (4.44), the velocity is 1.

In Figure 4.22, we show the calculated axial component of traction, f_1 as a function of arclength for $n = 32, 128, 256$. In all the plots, f_1 exhibits a large peak that corresponds in location to that in Figure 4.21. A disconcerting feature in the first two plots ($n = 32, 128$) is that the traction is negative for a portion of the domain. Clearly, this should be rejected on physical grounds. For $n = 256$, the traction takes on more realistic features. Another feature is the size of the peak. For $n = 32$, the peak is 56% smaller when compared with $n = 256$ and for $n = 128$ the peak is 24% smaller.

To obtain some estimate of the error, the traction can be re-substituted into equation (4.23) to produce an approximation to the velocity at each node point. This global error may indicate some breakdown in the algorithm. Upon performing this error check, we report that for $n = 32$ the error in the velocity is of order 10^{-3} at all node points with variation in the third digit. For $n = 128$ the error is of order 10^{-4} with variation in the fourth digit. Finally, for $n = 256$ the error is of order 10^{-5} with variation in the fourth or fifth digits. These results are a clear indication of the instability of first kind equations and suggest that techniques that improve results for small n are of great value. This is further clarified in section 4.5. We shall use the traction shown for

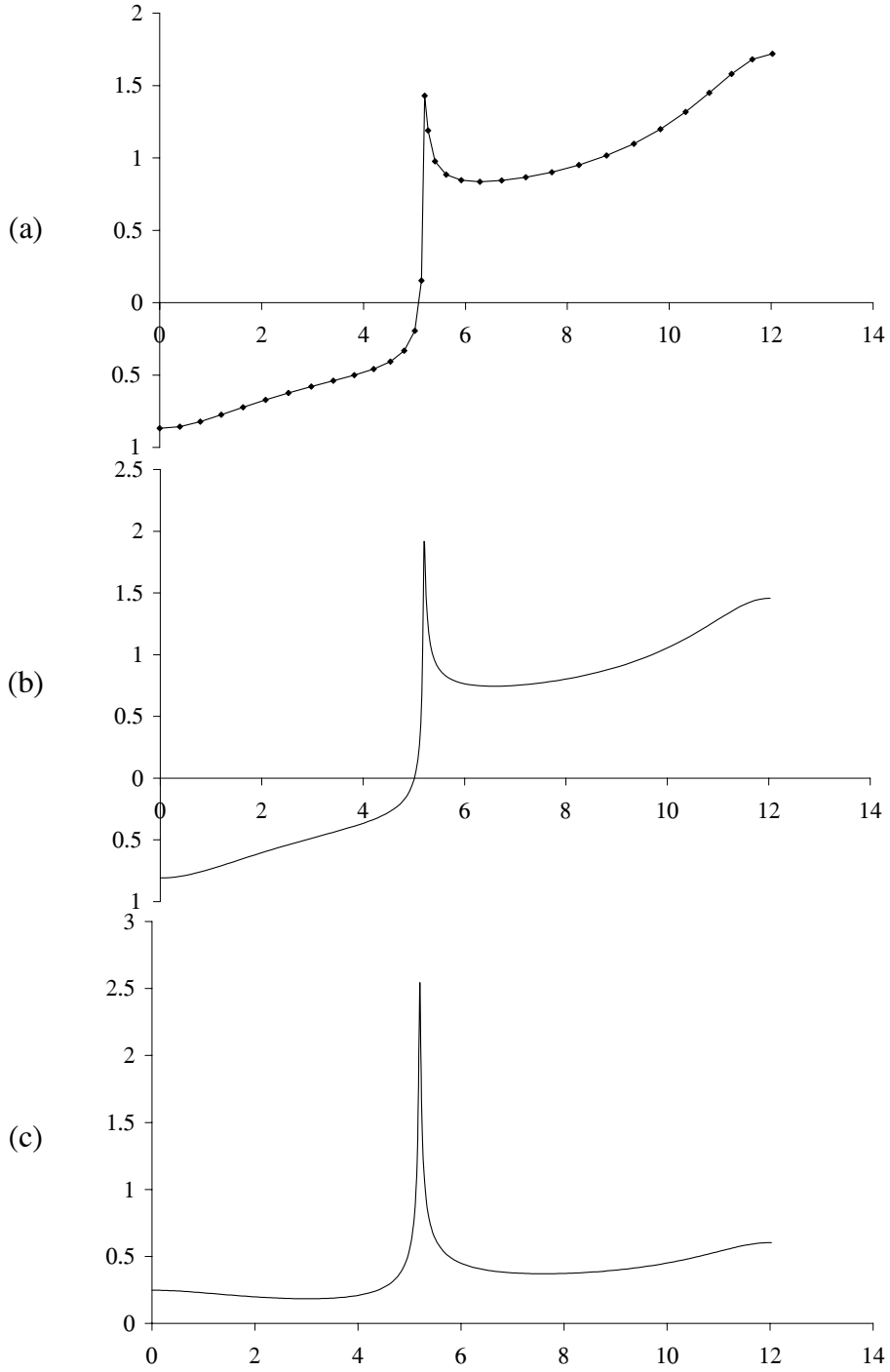


Figure 4.22: Calculated traction for surface A as a function of arclength. Node points are chosen uniformly with respect to θ as defined in (4.44). (a) $n = 32$ (node points are shown), (b) $n = 128$, (c) $n = 256$.

$n = 256$ in Figure 4.22 as the benchmark to measure the success or failure of the results to be presented.

The reduced height in the peak of Figure 4.22(a) and its node point distribution shows that there is a lack of nodes near the peak. That is, in this region the traction is not adequately represented.

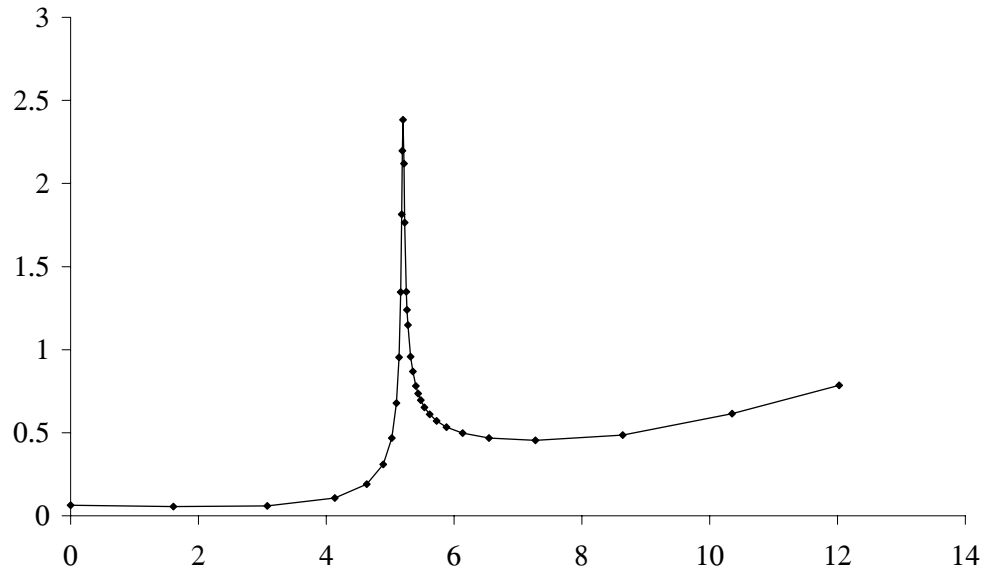


Figure 4.23: Calculated traction versus arclength for surface A when the node points are calculated using the curvature method of section 4.3.3. n=32.

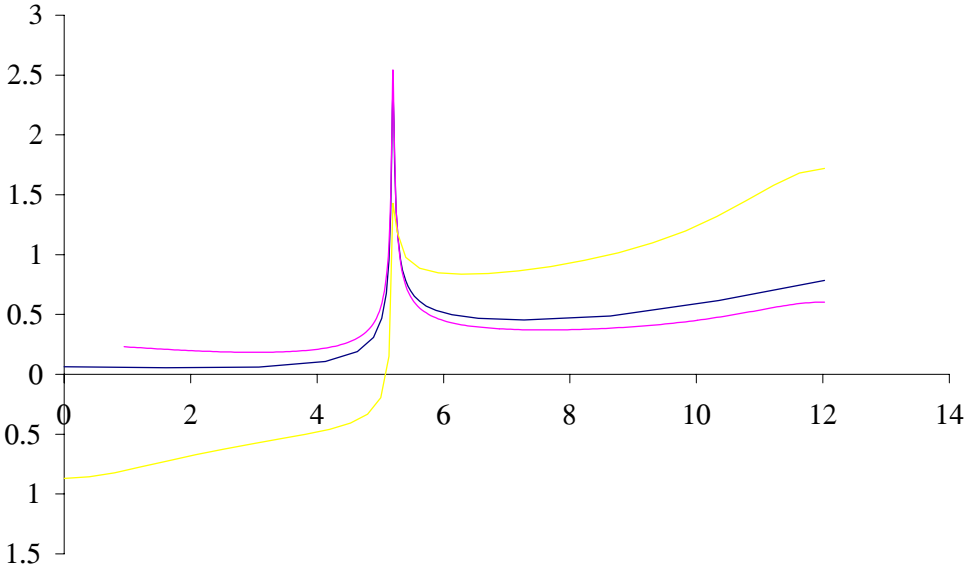


Figure 4.24: Comparison of different discretization schemes for calculating the traction of surface A. — (uniform θ, $n = 32$), — (curvature, $n = 32$), — (uniform θ, $n = 256$.)

In Figure 4.23, the traction has been calculated using the curvature discretization scheme of subsection 4.3.3; 33 node points (i.e. $n = 32$ segments) have been used. It is clear that this method, which chooses points according to adequate representation of the curvature, has clustered nodes about the peak. The height of the peak here is only 6% smaller than in Figure 4.22(c). In addition, the traction is not negative. A comparison of Figures 4.22(a), 4.22(c) and 4.23 is shown in Figure 4.24.

Results are also presented when the number of segments is increased to $n = 128$. Even at this

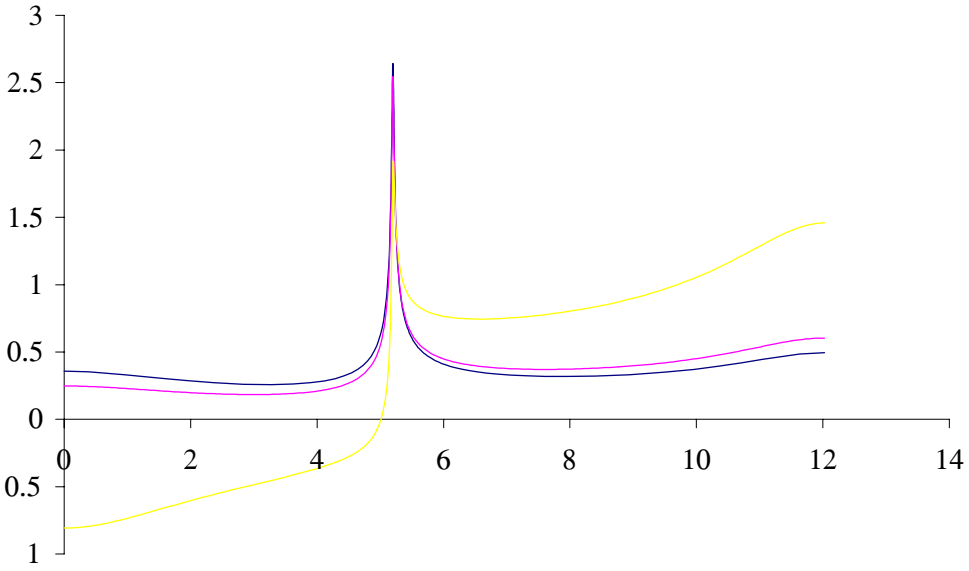


Figure 4.25: Comparison of different discretization schemes for calculating the traction of surface A. — *(uniform* θ*,* $n = 128$*),* — *(curvature,* $n = 128$*),* — *(uniform* θ*,* $n = 256$*.)*

higher resolution, Figure 4.22(b) shows that there is little change in the traction from $n = 32$. In Figure 4.25 we compare the resultant traction using θ and curvature grids with Figure 4.22(c). The increase in accuracy using the curvature grid is obvious. The traction is never negative and closely resembles Figure 4.22(c), also the peak is 4% higher and may indicate a more accurate solution. Figures 4.24 and 4.25 show the clear improvement of the curvature discretization over uniform θ. The drawback of the curvature method is the need to calculate the distribution via a robust root finding scheme. This involves evaluating the non-singular[1] integral (4.43) many times over. In fact orders of magnitude more than the resulting size of the grid. Indeed this process may, on occasion, take longer than solving the integral equation. In the next section we introduce another method that can be used with any grid distribution and also improves calculation of the traction at large grid sizes (small n).

4.4.1 Weighted kernel

The notion that the curvature is analogous to the traction can be further exploited. Recall the integral equation (4.41)

$$g(\xi^*) = \int_0^{\Xi} K(\xi, \xi^*) f(\xi)\, d\xi, \qquad 0 \leq \xi^* \leq \Xi.$$

[1] Judicious choice of ξ^* will always avoid the singularity in K.

If f is known and appended to the kernel K then the integral equation becomes

$$g(\xi^*) = \int_0^{\Xi} \big(K(\xi, \xi^*) f(\xi)\big)\, 1\, d\xi, \qquad 0 \le \xi^* \le \Xi,$$

and solving will produce the constant solution 1. Since f is unknown, we use the curvature in its stead. Specifically, the following *weighted* integral equation will be solved

$$g(\xi^*) = \int_0^{\Xi} K(\xi, \xi^*) w(\xi) f_w(\xi)\, d\xi, \tag{4.45}$$

where $w = \kappa^{1/3}$. This form of the weight was chosen since $f_1 \propto \kappa^{1/3}$ for a spheroid.

The idea in (4.45) being that f_w is a simpler function than f and hence an algorithm will produce a more accurate solution. Once f_w is found, the traction is easily furnished via

$$f = w f_w. \tag{4.46}$$

Figure 4.26 shows the weighted traction f_w as well as the traction f_1 for an $n = 32$ uniform θ grid. The weighted traction is monotonic and uniform almost everywhere and the sharp peaks of the traction have been eliminated. There is a slight *kink* in the curve near $\xi = 4.5$ and this is due to change in concavity of surface A. At this point the curvature is zero, thus affecting the weighted integral equation and hence the weighted solution. The traction seems to be unaffected by this flaw, but is still negative for much of its domain. This said, f_w has a much simpler structure than f_1 and there is an obvious improvement in the traction over the solution presented in Figure 4.22(b). The peak in the traction here is only 12% smaller and is consequently an improvement over $n = 32$ or $n = 128$ using the un-weighted integral equation.

In Figure 4.27 we compare three different schemes with Figure 4.22(c). They are, choosing 33 node points (i.e. $n = 32$ segments): *(i)* using the curvature *(ii)* according to θ with the weighted integral and *(iii)* using the curvature with the weighted integral. Method *(i)* is clearly superior to *(ii)* and *(iii)*. There is little difference in *(ii)* and *(iii)* since discretization now plays a smaller role. That is, there is no requirement to cluster around the peak of f_1 since the solution obtained is f_w.

The results with $n = 128$ are similar to those reported above. Figure 4.28 shows the weighted solution f_w as well as the traction f_1 for a uniform θ grid. The *kink* in the weighted solution is now much more pronounced and this has started to affect the traction. The solution is improved when compared to that in Figure 4.22(b) but is clearly inferior to the solution using the curvature grid.

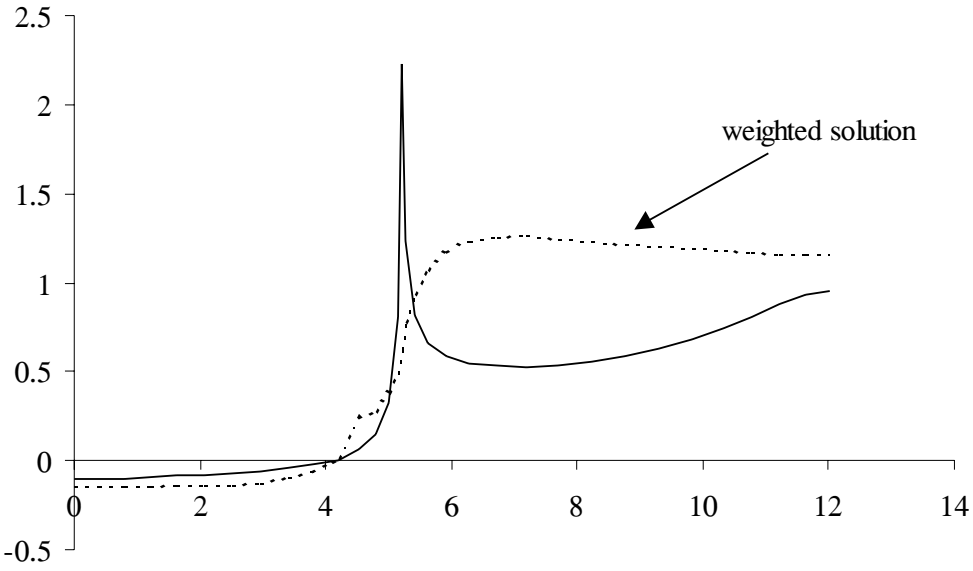


Figure 4.26: Results using the weighted integral equation. The traction f_1 (—) and the weighted traction f_w (- - -). Nodes are distributed according to θ. $n = 32$.

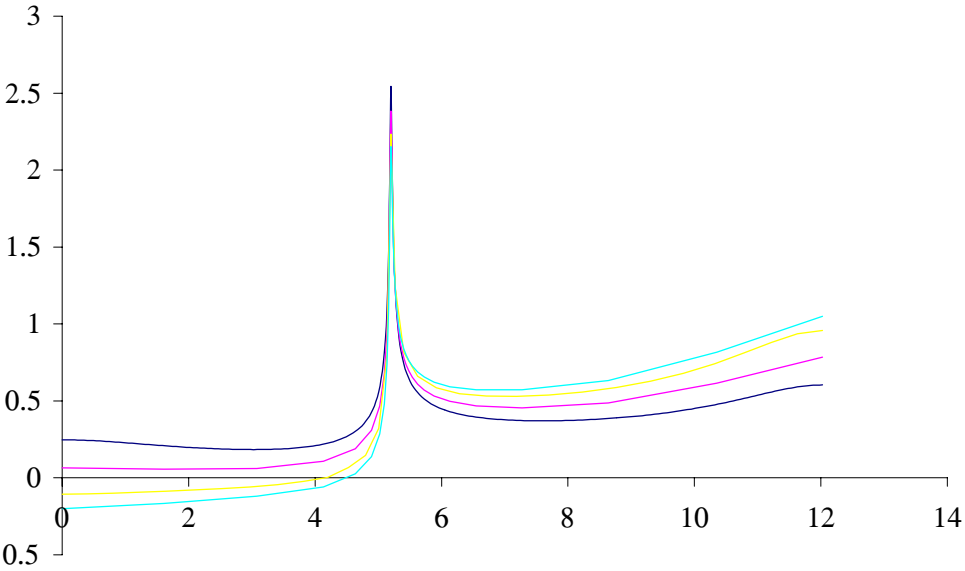


Figure 4.27: Comparison of different methods for calculating the traction of surface A. — (curvature, $n = 32$), — (weight with uniform θ, $n = 32$), — (weight with curvature grid, $n = 32$), — (uniform θ, $n = 256$.)

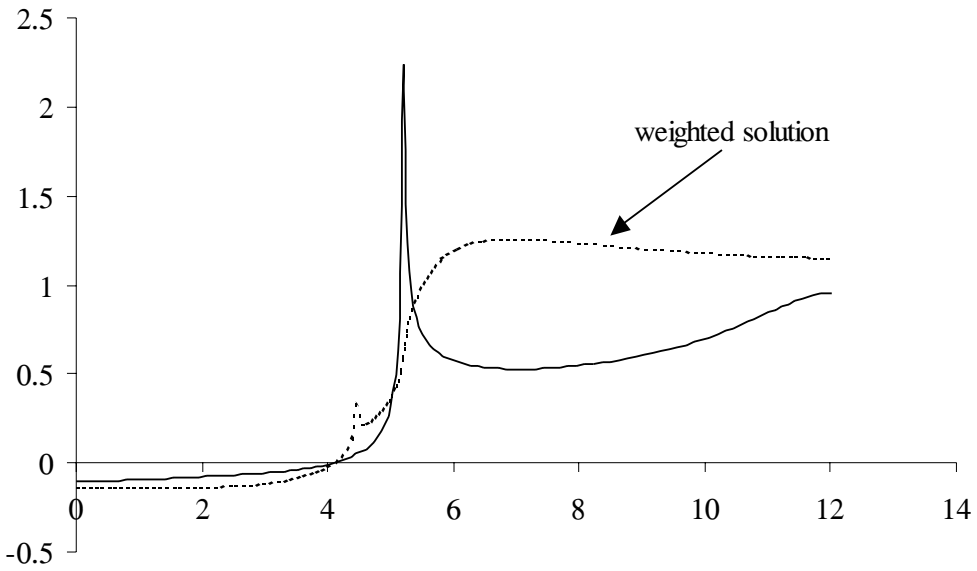


Figure 4.28: Calculated traction when using the weighted integral (—) and the weighted traction (- - -). Nodes are distributed according to θ. $n = 128$.

4.5 Conclusion

The algorithm performs reasonably well and produces quite acceptable results for simple geometries. Low order interpolation seems to be preferable to higher order schemes.

The instability of first kind equations revealed in section 4.4 show that the techniques developed here not only have a role in reducing error for large grid sizes (small n), but may indeed prevent the production of spurious results. To further clarify this point, we have shown that a small number of intervals can produce large errors and/or invalid results. We have also shown that a large number of intervals can lead to breakdown due to finite machine precision and/or finite computer memory[2]. Thus we have presented a scheme that may be able to be used as a tool to study the interaction of multi-complex particle systems using a small number of segments and efficient use of computer resources.

Finally, we note that the methods presented in this chapter are quite general and apply to any first kind boundary integral equation whose boundary has a known expression.

[2]A linear system with $n = 1024, N = 2$ involves over 16,777,216 double precision digits or approximately 128Mb of storage.

Chapter 5

Free Surface Problems

5.1 Introduction

The motion and interaction of liquid drops in fluids arises in many applications, in particular, the chemical engineering and the petroleum industries. Many of these areas involve low Reynolds' number flows where the drops or bubbles are small or the fluids are highly viscous. In such flows the fluid inertia may be neglected. The development of an understanding of how drops break up and coalesce is of primary importance in these areas.

Although real processes rarely involve axisymmetric motion of drops, experimental studies sometimes constrain the motion in this way. Numerical computation is also a powerful tool for such investigations. The Boundary Integral Method is particularly effective for droplet or bubble problems in an infinite domain, since it only requires discretization of the drop surfaces, and for axisymmetric problems, this reduces to calculating line integrals.

The Boundary Integral Method has been usefully applied to a number of problems involving the motion of single bubbles, bubble interactions with solid boundaries, free surfaces and interfaces, and interactions between two drops. Previous studies, using the Boundary Integral Method, include Rallison and Acrivos (1978), who investigated the stability of a drop in an extensional flow and Geller *et al.* (1986), who developed code for a bubble or drop moving through an interface. Both these studies used a crude approach for approximating the boundary integrals by representing the boundary elements and approximating the unknowns by piecewise constants which results in a matrix equation. Later studies include that of Pozrikidis (1990b), Manga and Stone (1993) and Zinchenko *et al.* (1997), who focused on drop-drop interactions, (a comprehensive review is

given in Stone (1994)) and solved the resulting integral equations using an iterative technique, which is much faster and uses less storage than solving a matrix equation (Pozrikidis 1990b).

Problems sometimes arise which can restrict the accuracy of such calculations, for instance numerical difficulties arise when two drops begin to coalesce or when the curvature is large. The challenge is to develop an algorithm which can handle these situations. One main difficulty is in calculating the surface tension forces with sufficient accuracy since this relies on the precise calculation of the curvature of the drop surface. Zinchenko *et al.* (1997) dealt with the troublesome mean curvature term by eliminating it from the Boundary Integral equations. The price paid for this was that the kernels of the integral equations became more singular.

In this chapter we propose a relatively simple, but effective, technique for parameterising the drop surfaces with respect to arclength using cubic splines. This enables a piecewise linear representation for the surface tension force. We also improve on previously developed codes by evaluating the singular integrals using appropriate Gaussian quadrature formulae. In general, for a given number of function evaluations, this is more accurate than evaluating the integrand about a singular point using some series expansion (Atkinson 1989, p. 270). It is also more elegant than the standard approach of subtracting the non-analytic part since the nature of the singularity is incorporated in the Gauss quadrature rule (see appendix D, section D.2).

The procedure outlined below is capable of simulating multi-drop buoyancy driven interactions. We shall focus firstly on single drop problems, using the code to investigate the stability of a rising drop. We then look at a two drop interaction problem and investigate how close to coalescence of the drops we can get and subsequently examine a three drop simulation and investigate how initial separation can effect drop deformation. Finally, we examine a single toroidal drop deformation and compare with the experimental results reported in Kojima *et al.* (1984).

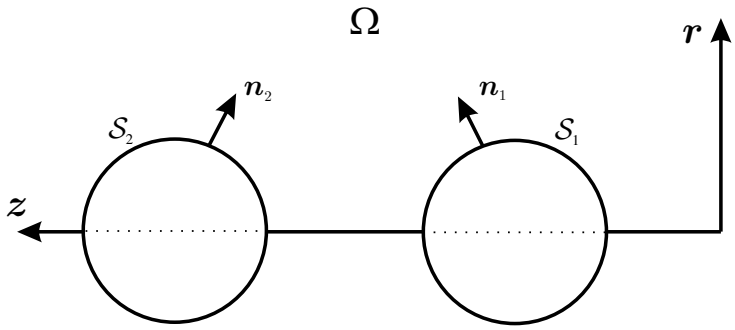


Figure 5.1: Schematic of two liquid drops showing geometry and notation.

5.2 Equations

The integral representation of the governing equations for drops in Stokes flow is described in chapter 2. The second kind equation is of Fredholm type, the unknown being the surface velocity

$$u_k(\boldsymbol{x}) = \frac{2}{1+\lambda}\int_{\mathcal{S}} G_{ik}(\boldsymbol{x}-\boldsymbol{y})\Delta f_i(\boldsymbol{y})\,dS(\boldsymbol{y}) \\ - 2\left(\frac{1-\lambda}{1+\lambda}\right)\int_{\mathcal{S}} \Sigma_{ijk}(\boldsymbol{x}-\boldsymbol{y})u_i(\boldsymbol{y})n_j(\boldsymbol{y})\,dS(\boldsymbol{y}) \quad (5.1)$$

where $\mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \ldots$ is the collection of drop surfaces and $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{S}$. (see Figure 5.1). The integrals in (5.1) are known as the single-layer and double-layer potentials (Odqvist 1930). The single layer term has a $1/r$ type singularity which reduces to an integrable log type in an axisymmetric regime. We are not so fortunate with the double-layer term, the Greens function Σ_{ijk} exhibits a singularity of strength $1/r^2$ and reduces to a $\log(r)/r$ singularity for an axisymmetric problem. This singularity is obviously non-integrable and the double-layer term exists in a Cauchy principal value sense. To circumvent the need to calculate special quadrature rules, we can use the divergence theorem to re-write the double-layer term as (see equation A.13)

$$\int_{\mathcal{S}} \Sigma_{ijk}(\boldsymbol{x}-\boldsymbol{y})u_i(\boldsymbol{y})n_j(\boldsymbol{y})\,dS(\boldsymbol{y}) \\ = -\int_{\mathcal{S}} \Sigma_{ijk}(\boldsymbol{x}-\boldsymbol{y})\big(u_i(\boldsymbol{x})-u_i(\boldsymbol{y})\big)n_j(\boldsymbol{y})\,dS(\boldsymbol{y}) + \frac{1}{2}u_k(\boldsymbol{x}). \quad (5.2)$$

Hence, a regularized form of the boundary integral equation is

$$u_k(\boldsymbol{x}) = \int_{\mathcal{S}} G_{ik}(\boldsymbol{x}-\boldsymbol{y})\Delta f_i(\boldsymbol{y})\,dS(\boldsymbol{y}) \\ + (1-\lambda)\int_{\mathcal{S}} \Sigma_{ijk}(\boldsymbol{x}-\boldsymbol{y})\big(u_i(\boldsymbol{x})-u_i(\boldsymbol{y})\big)n_j(\boldsymbol{y})\,dS(\boldsymbol{y}). \quad (5.3)$$

The first step in solving (5.3) is to exploit the axial symmetry and analytically integrate the azimuthal component. This results in line integrals whose kernels can be expressed as linear combinations of elliptic integrals of the first and second kind

$$u_\beta(\xi_0) = \int_C \tilde{G}_{\alpha\beta}(\xi,\xi_0)\Delta f_\alpha(\xi)\, dl(\xi) + (1-\lambda)\int_C \tilde{\Sigma}_{\alpha\beta}(\xi,\xi_0)\big(u_\alpha(\xi_0) - u_\alpha(\xi)\big)\, dl(\xi), \quad (5.4)$$

where $C = C_1 \cup C_2 \cup \ldots$ is the collection of meridional curves parameterised with ξ and the subscripts α, β take on the values 1 and 2 to denote the axial (z) or radial (r) component respectively. Also, $\tilde{G}_{\alpha\beta}$ and $\tilde{\Sigma}_{\alpha\beta}$ are G_{ik} and $\Sigma_{ijk}n_j$ integrated around the axis of symmetry. A detailed explanation of this may be found in Taib (1985) whilst the axisymmetric kernels are documented in Pozrikidis (1992).

5.3 Numerical method

Given a description of the drop shapes equation (5.4) is a set of $2m$ Fredholm integral equations of the second kind, where m is the number of drops. These are solved iteratively to furnish the velocity distribution of each drop. The surfaces can then be deformed by using the kinematic condition that states that a fluid element on the drop surface will stay on the surface

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{u}. \quad (5.5)$$

The numerical implementation of this procedure hinges on three factors: (i) accurate representation of the curves in the meridional plane, (ii) accurate integration of the singular kernels and (iii) choice of time step and method to solve (5.5).

5.3.1 Representation and discretization

The calculation of the dynamic condition (2.27) involves second order derivatives, for this reason we employ cubic splines to represent the curves, C. To this end, we divide each drop surface into n intervals. The subscript i will be used to designate marker points with $i = 0$ and $i = n$ being nodes on the axis of symmetry, while j will denote the line segment (i.e. $j = 1, \ldots, n$), and $\ell = 1, \ldots, m$ the particular drop. The clamped spline interpolant can be written parametrically as

$$\begin{aligned} r(\xi) &= a_{r_{j,\ell}} + b_{r_{j,\ell}}(\xi_{j,\ell} - \xi) + c_{r_{j,\ell}}(\xi_{j,\ell} - \xi)^2 + d_{r_{j,\ell}}(\xi_{j,\ell} - \xi)^3 \\ z(\xi) &= a_{z_{j,\ell}} + b_{z_{j,\ell}}(\xi_{j,\ell} - \xi) + c_{z_{j,\ell}}(\xi_{j,\ell} - \xi)^2 + d_{z_{j,\ell}}(\xi_{j,\ell} - \xi)^3 \end{aligned} \quad (5.6)$$

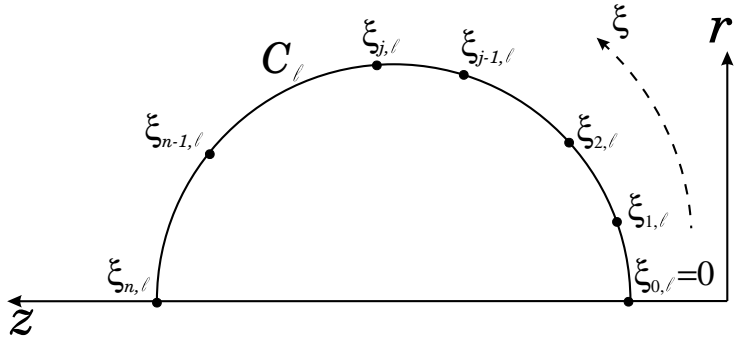


Figure 5.2: Discretization of a single drop surface.

for $\xi_{j-1,\ell} \leq \xi \leq \xi_{j,\ell}$, with the clamped conditions

$$\frac{dr}{d\xi} = +1, \quad \text{and} \quad \frac{dz}{d\xi} = 0, \qquad \text{on} \quad \xi = \xi_{0,\ell}, \tag{5.7}$$

$$\frac{dr}{d\xi} = -1, \quad \text{and} \quad \frac{dz}{d\xi} = 0, \qquad \text{on} \quad \xi = \xi_{n,\ell}, \tag{5.8}$$

where the $\xi_{j,\ell}$'s are the progressive arclengths and ξ is the arclength parameter (see Figure 5.2). The constants $a_{r_{j,\ell}}, b_{r_{j,\ell}}, c_{r_{j,\ell}}, d_{r_{j,\ell}}$ and $a_{z_{j,\ell}}, b_{z_{j,\ell}}, c_{z_{j,\ell}}, d_{z_{j,\ell}}$ are the usual spline coefficients (DeBoor 1979). The arclength parameterisation is used to overcome difficulties reported by earlier investigators (Rallison and Acrivos 1978; Geller *et al*. 1986; Pozrikidis 1990b). With arclength as a parameter the normal vector is defined in a consistent manner regardless of curve shape and the parametric derivatives are never large. In fact they are always bounded by unity since for any space curve $|\boldsymbol{r}'(\xi)| = 1$. Other advantages of using arclength are that the end-derivative conditions (5.7)-(5.8) become simple to implement and that the interpolant is guaranteed to be a bijective function of the parameter. In order to calculate the parametric splines, accurate approximations to the individual segment lengths are required. This can be achieved by starting off with the Euclidean distance between the marker points as the initial approximation to the segment lengths, using (5.6) and (5.7) and performing the segment length integrations

$$T_{j,\ell} = \int_{\xi_{j-1,\ell}}^{\xi_{j,\ell}} \sqrt{\left(\frac{dr}{d\xi}\right)^2 + \left(\frac{dz}{d\xi}\right)^2}\, d\xi, \qquad \text{for} \quad \begin{matrix} j &= 1, \ldots, n \\ \ell &= 1, \ldots, m, \end{matrix} \quad \text{and} \tag{5.9}$$

numerically using a Gauss-Legendre rule. Updating the progressive arclengths is then accomplished by the following relation

$$\xi_{0,\ell} = 0, \quad \text{and} \quad \xi_{j,\ell} = \xi_{j-1,\ell} + T_{j,\ell}. \tag{5.10}$$

Repeating the process defined by (5.6) to (5.10) should, in principle, yield accurate approximations to the $\xi_{j,\ell}$'s.

When the above procedure is used to calculate arclengths on a unit sphere which has been subdivided into sixteen equal intervals, the relative error in the interval arclengths is 4.1×10^{-7} uniformly after three iterations using a three-point Gauss-Legendre rule.

With the arclengths calculated we can construct cubic spline interpolants for the unknown velocities

$$\begin{aligned} u_r(\xi) &= a_{ur_{j,\ell}} + b_{ur_{j,\ell}}(\xi_{j,\ell} - \xi) + c_{ur_{j,\ell}}(\xi_{j,\ell} - \xi)^2 + d_{ur_{j,\ell}}(\xi_{j,\ell} - \xi)^3 \\ u_z(\xi) &= a_{uz_{j,\ell}} + b_{uz_{j,\ell}}(\xi_{j,\ell} - \xi) + c_{uz_{j,\ell}}(\xi_{j,\ell} - \xi)^2 + d_{uz_{j,\ell}}(\xi_{j,\ell} - \xi)^3 \end{aligned} \tag{5.11}$$

with $\xi_{j-1,\ell} \leq \xi \leq \xi_{j,\ell}$. By considering the behaviour of these functions on the symmetry axis, it is clear that the radial velocity must be an odd function and the axial velocity an even function of the arclength parameter. Hence the end conditions for the splines (5.11) are

$$\begin{aligned} \frac{d^2u_r}{d\xi^2} = 0, \quad \text{and} \quad \frac{du_z}{d\xi} = 0, \qquad \text{on} \quad \xi = \xi_{0,\ell}, \\ \frac{d^2u_r}{d\xi^2} = 0, \quad \text{and} \quad \frac{du_z}{d\xi} = 0, \qquad \text{on} \quad \xi = \xi_{n,\ell}. \end{aligned} \tag{5.12}$$

5.3.2 Integration

Now that we have defined a grid, we can evaluate the integral equation (5.4) at the marker points

$$u_\beta(\xi_{I,L}) = \sum_{\ell=1}^{m}\sum_{j=1}^{n}\left\{ \int_{\xi_{j-1,\ell}}^{\xi_{j,\ell}} \tilde{G}_{\alpha\beta}(\xi, \xi_{I,L})\Delta f_\alpha(\xi)\, d\xi \right. \\ \left. +(1-\lambda_\ell) \int_{\xi_{j-1,\ell}}^{\xi_{j,\ell}} \tilde{\Sigma}_{\alpha\beta}(\xi, \xi_{I,L})\big(u_\alpha(\xi_{I,L}) - u_\alpha(\xi)\big)\, d\xi \right\}, \tag{5.13}$$

for $I = 0, \ldots, n$ and $L = 1, \ldots, m$. When $j = I, I+1$ and $\ell = L$ the integrals in (5.13) exhibit a log strength singularity. We note that at the singular point the first integrand is unbounded whilst the second is bounded but non-analytic. Usually, the numerical calculation of (5.13) is completed in three steps: (i) define a small region around the singular point in the first integral and evaluate analytically using some series expansion of the integrand (ii) evaluate the remainder of the first integral — which should now be regular — using a standard Gauss-Legendre rule (iii) calculate the second integral using Gauss-Legendre quadrature (Geller *et al.* 1986; Pozrikidis 1990b; Stone and Leal 1989). This is cumbersome and can lead to inaccuracies. The integrals in (5.13) can each be decomposed into a regular and a singular part. In this form, they are easily and

accurately evaluated using Gauss-Legendre and Gauss-Log rules (Stroud and Secrest 1966) on each of these respective parts. Details regarding these aspects may be found in Taib (1985).

5.3.3 Time step

The integral equation is now easily solved by guessing an initial velocity at the marker points, splining using (5.11) and (5.12) to obtain a velocity distribution and evaluating the right hand side of (5.13). In this way a new velocity distribution is calculated and we continue this until the velocity has converged at all the marker points. Equation (5.5) is then solved via an explicit fourth order Runge-Kutta scheme to evolve each surface. We employ a variable time step of the form

$$\Delta t = \frac{\Delta x_{\max}}{\max_i |\boldsymbol{u}(\xi_i)|} \tag{5.14}$$

where $\Delta x_{\max}$ is some constant and $\max_i$ denotes the maximum value over all node points. This has the effect of constraining the maximum deformation of each surface to the chosen constant $\Delta x_{\max}$.

5.4 Numerical accuracy

To test the accuracy of the quadrature and spline routines we calculate

$$\int_{\text{unit sphere}} G_{11}(\boldsymbol{x}-\boldsymbol{y})\, dS(\boldsymbol{y}) = -\frac{16}{3}\pi. \tag{5.15}$$

This result is easily obtained by considering the uniform translation of a solid unit sphere in an infinite medium. Table 5.1 shows the relative error in calculating (5.15) for various values of quadrature points and number of intervals (n). We can see that doubling the number of intervals increases the accuracy by approximately an order of magnitude, while increasing the number of Gauss points has little effect. With the simulations shown in the next section we found that using an 8 point Gauss-Legendre rule for the regular integrals, an 8 point Gauss-Log rule for the singular integrals and an initial distribution of 33 marker points ($n = 32$) gave good results.

The evolution of the drop surface is calculated by following the trajectories of the marker points in time. These points tend to migrate strongly to regions of high curvature. Thus we introduce a re-distribution mechanism, whereby a marker point is introduced at the bisector of a segment

	No. of Gauss quadrature points			
n	4	8	12	16
8	5.60E-5	5.60E-5	5.60E-5	5.60E-5
16	3.54E-6	3.53E-6	3.53E-6	3.53E-6
32	2.26E-7	2.23E-7	2.23E-7	2.23E-7
64	1.56E-8	1.40E-8	1.40E-8	1.40E-8
128	1.69E-9	8.79E-10	8.79E-10	8.79E-10

Table 5.1: Relative error in computing (5.15) given the number of Gauss quadrature points and the number of subintervals n to discretize the unit sphere.

whose length exceeds some maximum and a marker point is removed if a segment length is less than some minimum.

The numerical method can be tested by comparing with the Hadamard-Rybczynski solution for a spherical drop in an unbounded fluid (Batchelor 1967). In dimensionless form, the terminal velocity of the spherical drop is

$$u = \frac{2}{3}\frac{1+\lambda}{2+3\lambda}. \tag{5.16}$$

In Figure 5.3 we show the relative deviation of the numerical method from (5.16) for different values of the time-step parameter $\Delta x_{\max}$. All of the iso-curves follow the same general pattern. Considering $\Delta x_{\max}$=0.05 we can see that the numerical solution gradually loses accuracy as it progresses in time, this behaviour is well understood in the context of Euler and Runge-Kutta schemes. At $t = t^* \approx 24$, the simulation breaks down and this is reflected in the steep gradient of the curve at this point. By inspecting the drop surfaces near the trailing edge the formation of a numerical instability develops where the drop surface takes on a wave-like pattern with wavelength equal to the distance between two successive marker points. The end point of each iso-curve indicates the time at which the numerical algorithm (as opposed to the accuracy of the simulation) breaks down. At this time the relative error increased by several orders of magnitude. We can see from the other curves in Figure 5.3 that this instability can be significantly delayed by taking smaller time steps. Similar phenomenon were observed with non-zero inverse Bond number where decreasing Γ increased the value of t^*, the point at which the simulation breaks down (see Figure 5.4). On physical grounds, we might have expected that increasing Γ (increasing surface tension) would enhance stability. However, this refers to hydrodynamic stability whereas Figure 5.4 refers to numerical stability. That numerical stability increases with decreasing Γ could be explained by recognising that decreasing Γ diminishes the contribution of

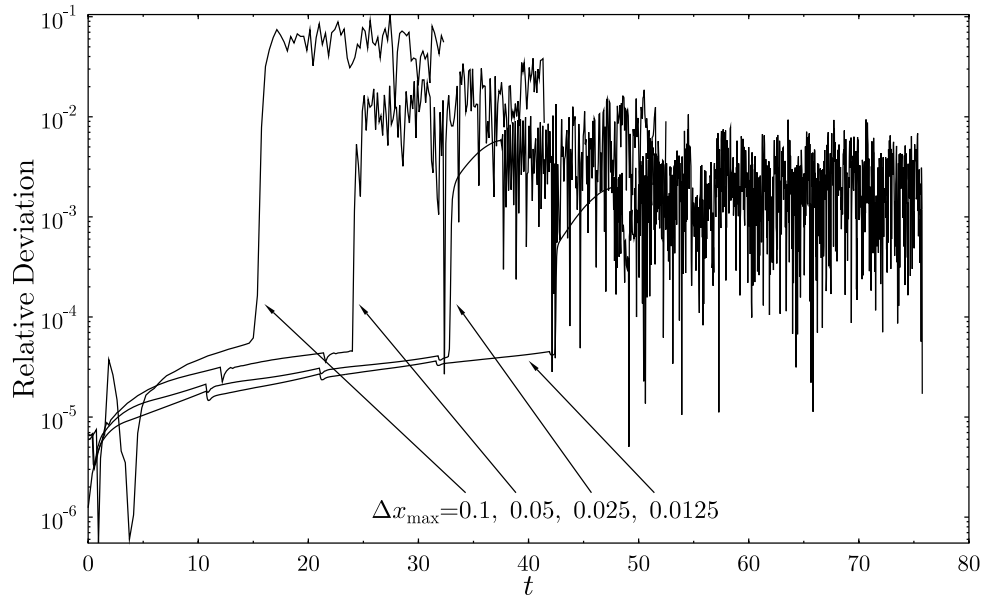


Figure 5.3: Relative error of the numerical method from the Hadamard-Rybczynski result for various values of the time-step parameter Δx_{max} with $n = 32$, $\lambda = 1$ and $\Gamma = 0.5$.

the curvature term in the boundary condition (2.27). We anticipate that using higher order splines would result in a more accurate calculation of this curvature term.

Finally, we can compare these results with those of Koh and Leal (1989). They report accuracies of 0.05% with the Hadamard-Rybczynski solution for $n = 50$. Manga and Stone (1993) report deviation of less than 1% . The results given here are more accurate when $t < t^*$.

5.5 Results for two-drop simulations

Results are shown for two different two-drop simulations that contrast in terms of their parameter space. In the first simulation (called S-I) the parameters are chosen to test the robustness of the numerical method, while the second simulation (S-II) is motivated by photographs taken of an experiment reported in Manga and Stone (1993). The results in Figure 5.5 show the deformation of the drops in a reference frame relative to the front of the leading drop. As an indicator of cumulative error (in time) we monitor the volume of each drop. In S-I the relative change in volume is less than 1% for $t < 16$ and near 10% for $t > 16$. In S-II the relative volume change is less than 10^{-6} for the span of the simulation. This compares favourably with other recent work. For example, for a one drop simulation (Pozrikidis 1990b) reports volume changes of less than 0.2% for $\lambda = 1$ and less than 1% for $\lambda \neq 1$. With their two drop simulations,

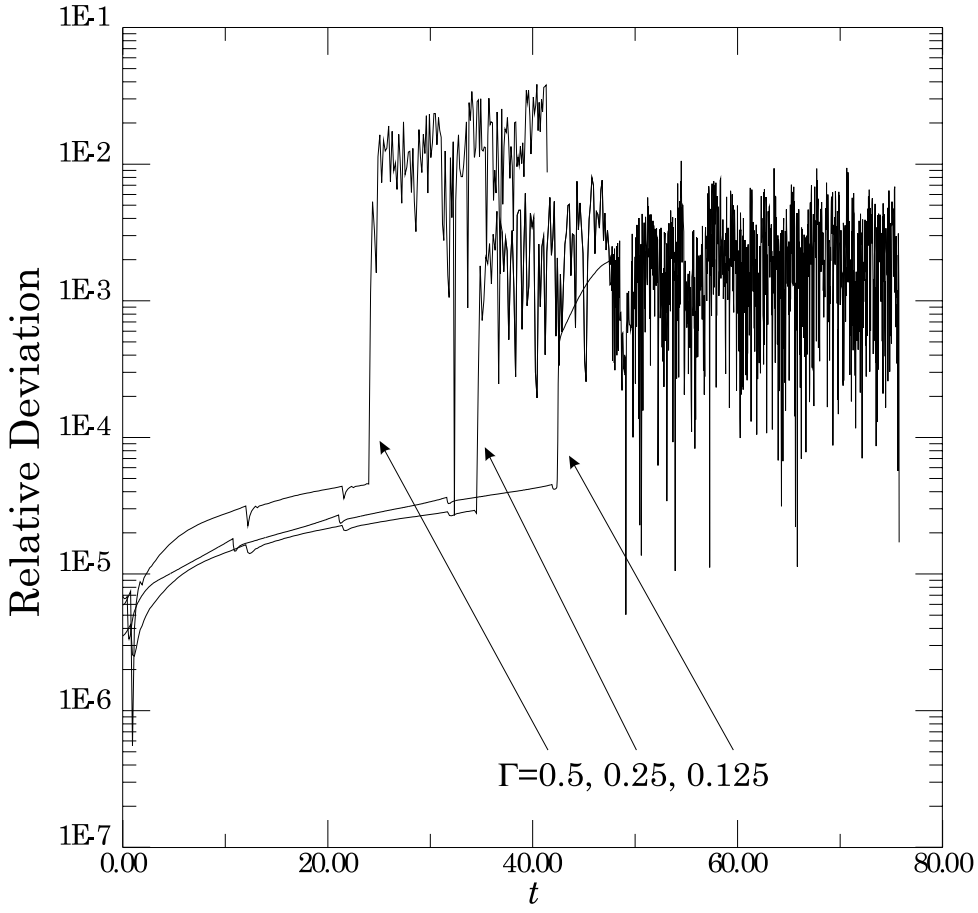


Figure 5.4: Relative error of the numerical method from the Hadamard-Rybczynski result for various values of Γ *with* $n = 32$, $\lambda = 1$ *and* $\Delta x_{max} = 0.05$.

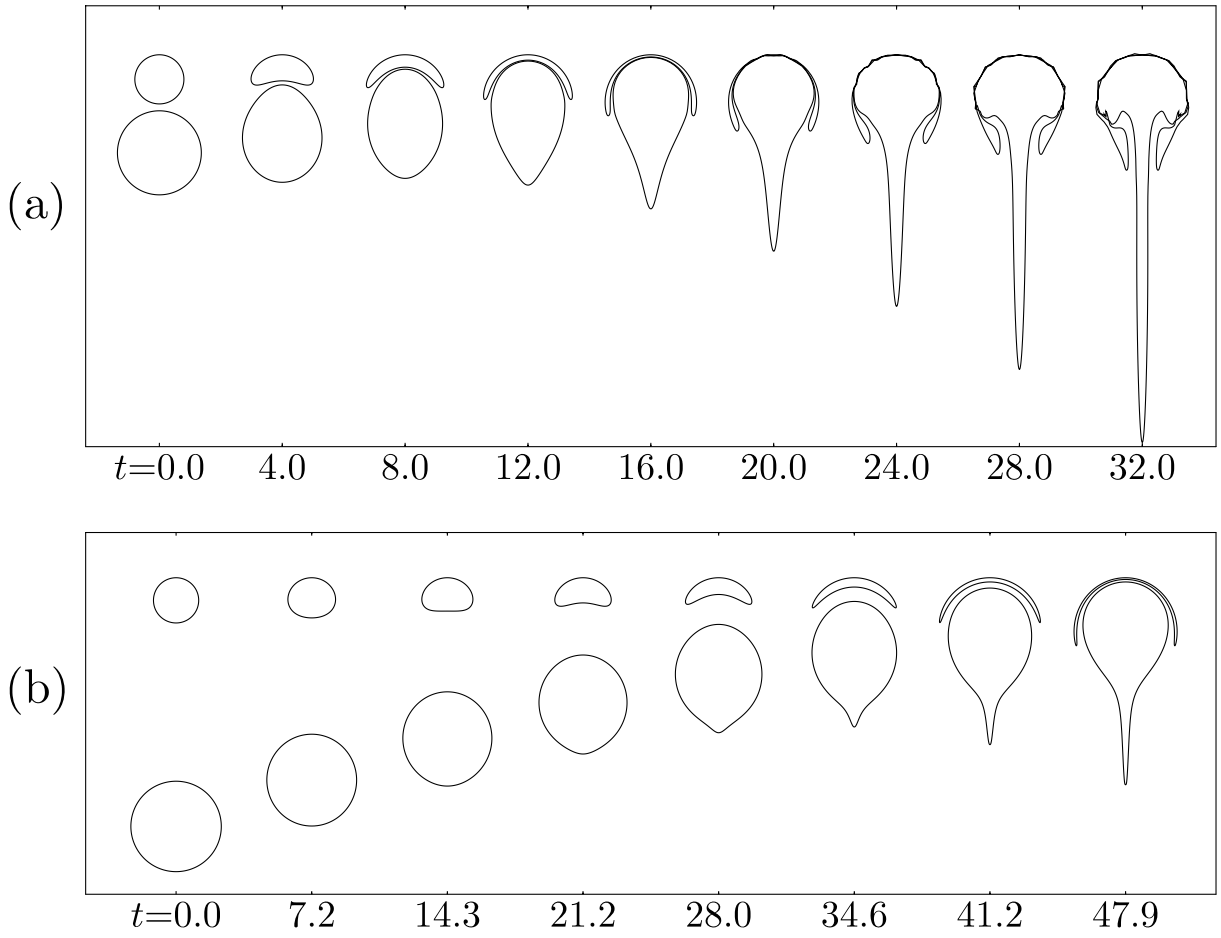


Figure 5.5: Two-drop simulations. (a) For the lower drop $\lambda = 1.1, \Gamma = 0.0$ *and volume ratio=5, for the upper drop* $\lambda = 1.1, \Gamma = 0.05$ *and volume ratio=1. Initial drop separation is 3. (b) For the lower drop* $\lambda = 1.1, \Gamma = 0.01$ *and volume ratio=1, for the upper drop* $\lambda = 1.1, \Gamma = 0.003$ *and volume ratio=0.125. Initial drop separation is 5.*

Manga and Stone (1993) report volume changes of less than 10%. In S-I we choose $\Gamma = 0$ for the lower drop and $\Gamma = 0.05$ for the upper drop since small Bond numbers lead to highly deformable surfaces. To test the algorithm in the absence of a dynamic time step we specify a static $\Delta t = 0.1$ and ensure that deformation proceeds quickly by giving the lower drop a volume five times larger than the upper drop (volumes quoted are relative to the unit sphere). To exaggerate the already strong interaction the initial separation between drop centres is small (3 units). The simulation is shown in Figure 5.5(a). At time $t = 8$ a region of rapid drainage has formed along the centreline between the two drops. This is indicative of interactions with small initial separations. As a result the upper drop quickly moulds to the surface of the lower drop ($t = 16$). The lower drop deformation is comparable to that of a single prolate drop (Pozrikidis 1990b). The drop takes on an initial prolate shape ($t = 4.0$ and $t = 8.0$) but as it rises most of the mass begins to collect near the front forming a spherical head leaving a long elongated tail ($t = 24, 28, 32$). At time

$t = 28$ ambient fluid near the base of the tail induces an internal spike in the lower drop ($t = 32$).

Numerical instabilities appear near the front of the drops when $t = t^* \approx 20$. Careful examination of this region reveals that the marker points of the trailing drop are moving into the interior of the leading drop. Figure 5.6 shows a magnified view of this region for $t = 28$. It is immediately clear that the drop surfaces are merging as the leading drop coats the surface of the trailing drop. In reality the two drops will coalesce and the leading drop will evolve into a multiply connected toroidal surface. This phenomenon has been reported in other numerical solutions (Pozrikidis 1990b) and validated experimentally (Kojima *et al.* 1984). After this time the degradation of the simulation is clearly evident. A dynamic time-step (with an appropriately chosen $\Delta x_{\max}$) would have been required to examine the interaction of the later stages in any detail. This simulation took approximately 9 minutes on a Pentium-90 personal computer. With such a large time step the algorithm performs well and clearly shows the general trend of the drop interaction. In S-II, shown in Figure 5.5(b), the parameters are chosen to match the experiment

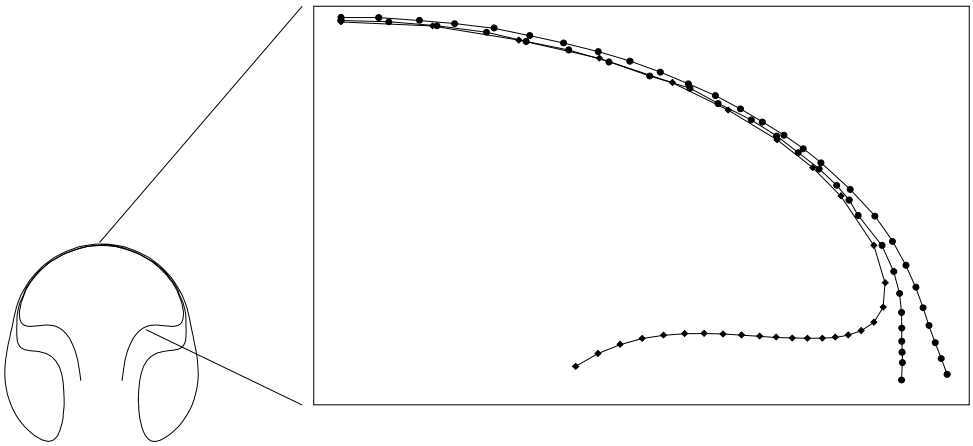


Figure 5.6: Two-drop simulation. A magnified view of the top surface showing the migration of the marker points into the drop interior.

in Manga and Stone (1993) (their Figure 1) where two air bubbles are observed to interact in an ambient fluid of corn syrup. In order to ensure that the numerical simulation is stable (t^* is large) we choose $\Delta x_{\max} = 0.01$. The separation between the drop centres is 5 units. This is a key difference between S-I and S-II. In this simulation as the trailing drop approaches the leading drop a region of uniform drainage occurs $t = 28.0$. In the presence of the lower larger drop, the upper drop forms a spherical cap and begins to encapsulate it ($t = 34.6$). The overall drop dynamics appear to be in good agreement with experimental results. This simulation was run on a DEC Alpha and took 45 hours to execute. The very small time step and the node point re-distribution mechanism have both contributed to this larger execution time. The number of

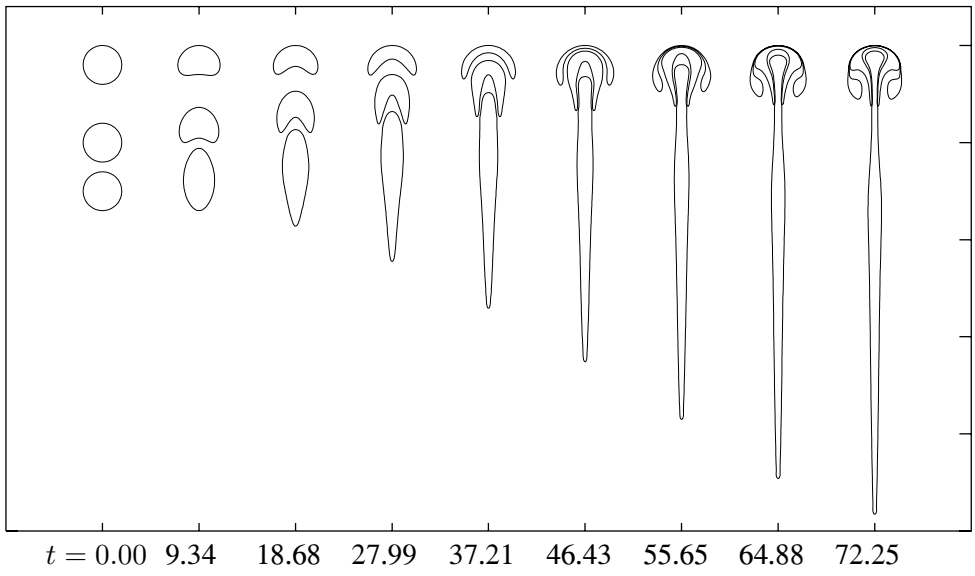


Figure 5.7: Three-drop simulation. For each drop $\lambda = 1, \Gamma = 0$ and volume ratio=1. Initial drop separation is 2.5 for the lower drops and 4 for the upper drops.

segments increased from an initial $n = 32$ at $t = 0$ to $n = 195$ for the leading drop and $n = 200$ for the trailing drop.

5.6 Results for three-drop simulations

Results for three drop simulations are presented. In each case the surfaces are initially spherical of unit radius and we set $\Gamma = 0.05$, $\lambda = 1$ and investigate the effect of different initial separations of the drops on the type of drop deformation. We define Δr_1 as the initial separation of the lower drop from the middle drop, and Δr_2 as the initial separation of the upper drop from the middle drop. We investigate three cases: (i) $\Delta r_1 = 2.5$, $\Delta r_2 = 4.0$, see Figure 5.7, (ii) $\Delta r_1 = 4.0$, $\Delta r_2 = 2.5$, see Figure 5.8, and (iii) $\Delta r_1 = 4.0$, $\Delta r_2 = 4.0$, see Figure 5.9.

In each of the three cases the top drop becomes oblate and the bottom drop becomes prolate. In Figure 5.7, where initially the bottom two drops are closer together, the middle drop becomes oblate. Both upper drops appear to wrap themselves around the bottom drop as the drops move closer together under buoyancy. In Figure 5.8, where initially the upper two drops are closer together, the middle drop becomes prolate. The two upper drops appear to eventually coalesce and form a toroidal surface as the separation of these and the bottom drop becomes greater. In Figure 5.9, where the drops are initially equally spaced, the behaviour has characteristics similar to both Figure 5.7 and Figure 5.8. It firstly becomes oblate and forms an inverted spike but unlike

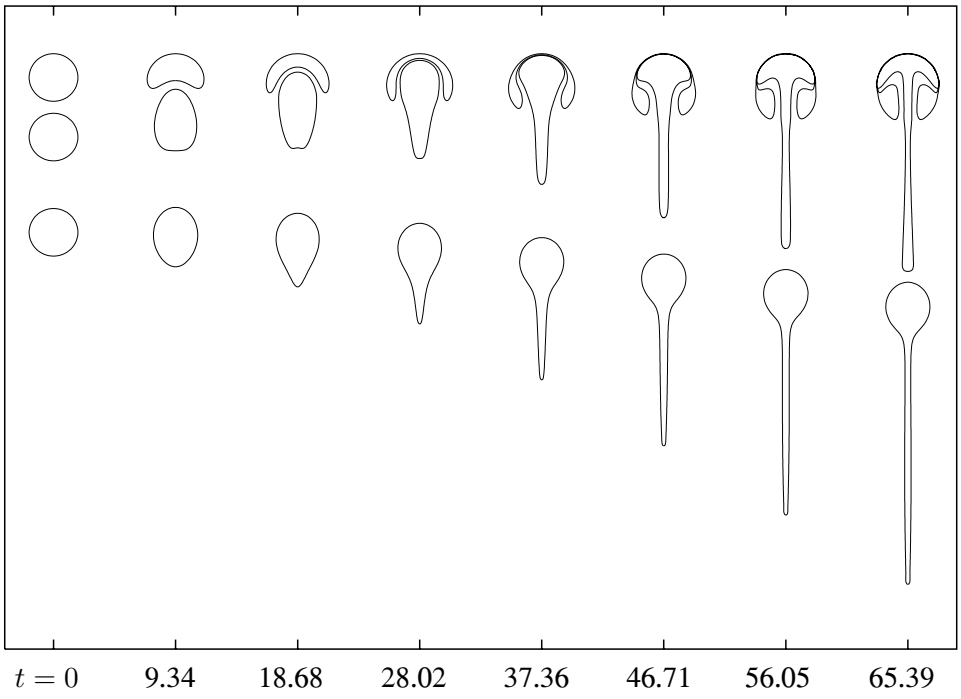


Figure 5.8: Three drop simulation. For each drop $\lambda = 1, \Gamma = 0$ and volume ratio=1. Initial drop separation is 4 for the lower drops and 2.5 for the upper drops.

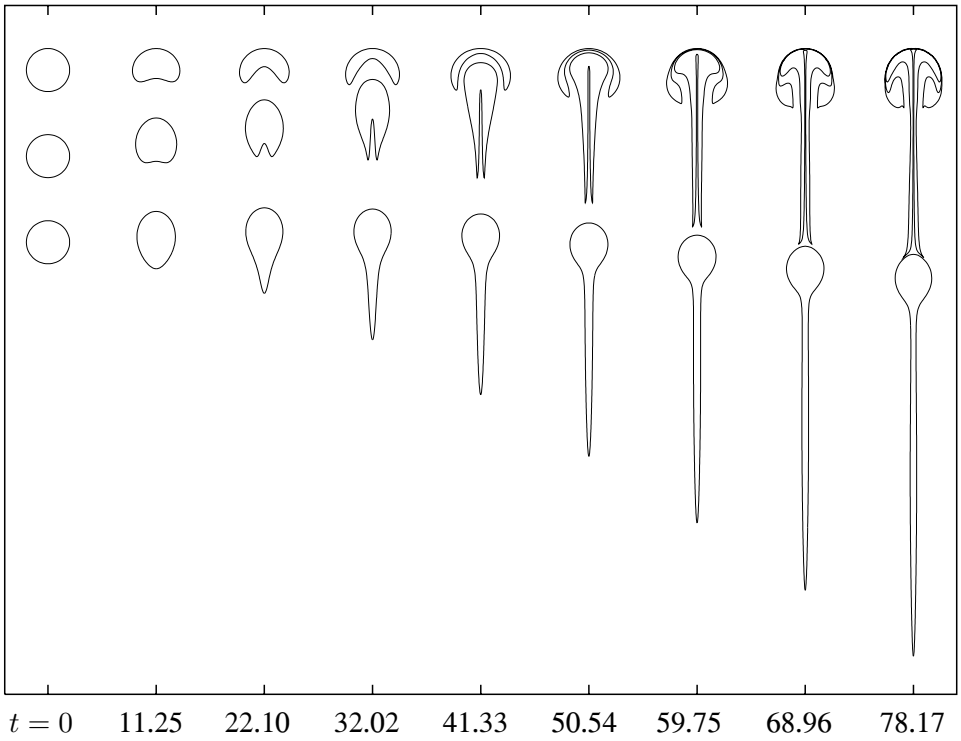


Figure 5.9: Three drop simulation. For each drop $\lambda = 1, \Gamma = 0$ and volume ratio=1. Initial drop separation is 4 for the lower drops and 4 for the upper drops.

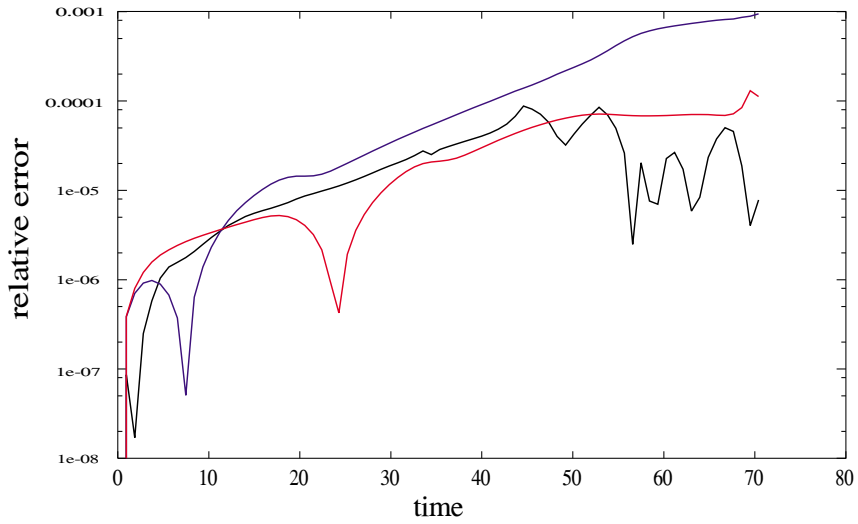


Figure 5.10: Three-drop simulations — an accuracy check. Relative deviation from initial volume of each drop. Trailing drop (—), middle drop (—), leading drop (—).

the previous oblate deformation the spike becomes elongated, producing an overall prolate shape.

As an indicator of the global error in these computations we can calculate the relative change in volume of each of the drops. In Figure 5.10 we show the relative changes for each of the three drops corresponding to Figure 5.7. This shows global errors are well within acceptable bounds except right at the very final stages of the simulation $t \simeq 70$ seconds. An interesting point about Figure 5.7 is the deformation of the bottom trailing drop, which becomes prolate but doesn't form a spherical head (as it would in a two-drop simulation). In order to investigate the effect of the presence of the leading drop on the trailing drop, in Figure 5.11 we compare the later stages of the simulation with a similar simulation where the leading drop is absent. The bottom two drops in Figure 5.11(a) act together to have a similar effect on the top drop as the single bottom drop in Figure 5.11(b).

5.7 Toroidal drops

Results for a single toroidal drop deformation are briefly presented. The parameters used to define each simulation are dependent on the geometry: α, λ_1, λ_2, and fluid properties: λ, Γ. Here α is the volume ratio of the drop with a unit sphere, λ_1 is the aspect ratio that includes the inner radius and λ_2 is the aspect ratio of the drop cross-section, see Figure 5.12. The parameter α is an important control variable since drop volume has a direct influence on the time scale

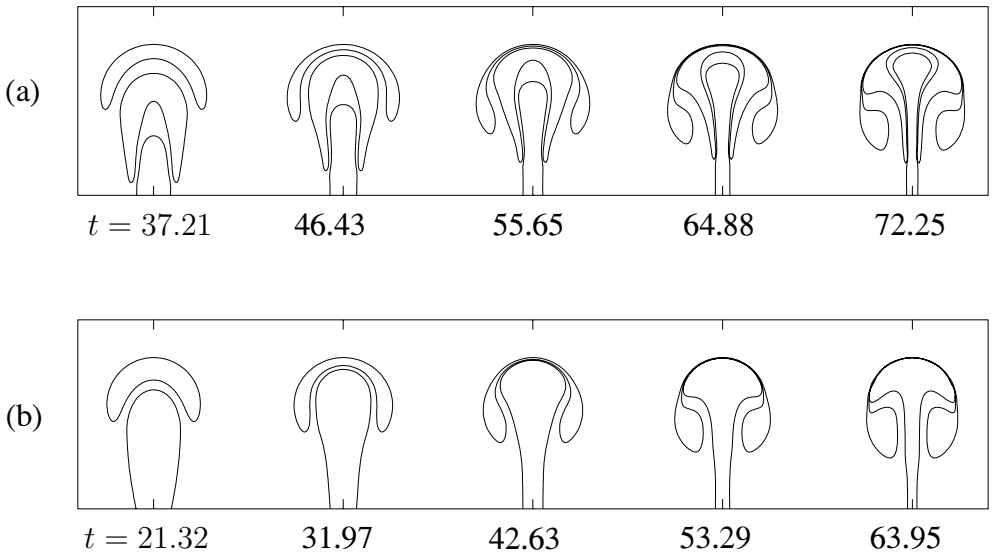


Figure 5.11: Comparison of (a) the later stages of the simulation in Fig. 5.7 with (b) two-drop simulation when the leading drop is absent.

of a simulation. That is, the larger the volume, the greater the buoyancy force and hence the shorter the simulation time span. Different simulations can be compared at similar times of drop evolution by restricting α to be the same for all experiments. The parameters λ_1 and λ_2 are varied to produce different initial spheroidal torus configurations as these will induce quite different deformations.

To prescribe the volume, we equate the toroidal volume to given multiples, α, of a unit sphere

$$abc = \frac{2\alpha}{3\pi}. \tag{5.17}$$

The equations for the aspect ratios are

$$\lambda_1 = \frac{c}{a} \qquad \text{and} \qquad \lambda_2 = \frac{b}{a}. \tag{5.18}$$

Solving (5.17) and (5.18) simultaneously produces

$$a = \sqrt[3]{\left(\frac{2\alpha}{3\pi\lambda_1\lambda_2)}\right)}, \qquad b = a\lambda_2, \quad \text{and} \quad c = a\lambda_1 \tag{5.19}$$

The conditions that a, b, c be real and positive and that $c > b$ produces the constraint

$$\lambda_1 > \lambda_2. \tag{5.20}$$

In Figure 5.13 we show the evolution of a buoyant toroidal drop. The initial configuration is $\lambda_1 = 4$ and $\lambda_2 = 1$. That is, the cross-section is circular and the inner radius is large producing a slender torus.

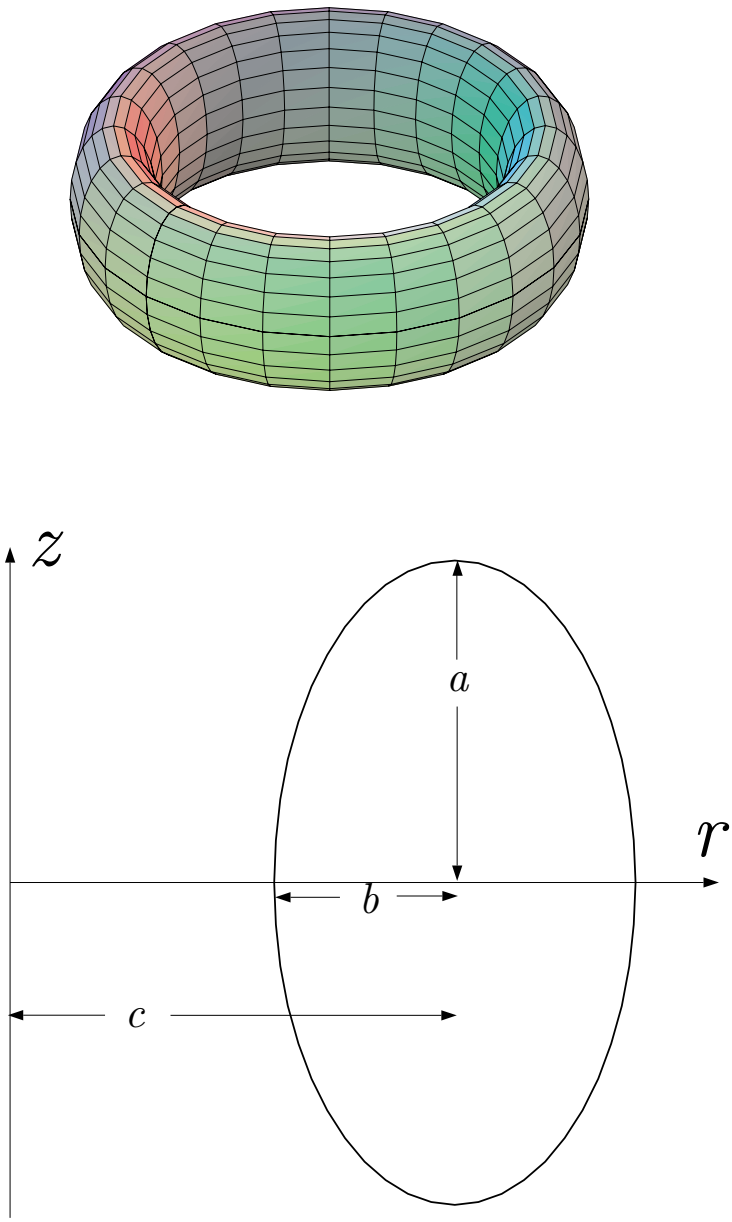


Figure 5.12: Geometry of a toroidal surface. Parametric equations are $r = b\cos\theta + c$, $z = a\sin\theta$, $0 \leq \theta \leq 2\pi$. *Aspect ratios are* $\lambda_1 = c/a$ *and* $\lambda_2 = b/a$.

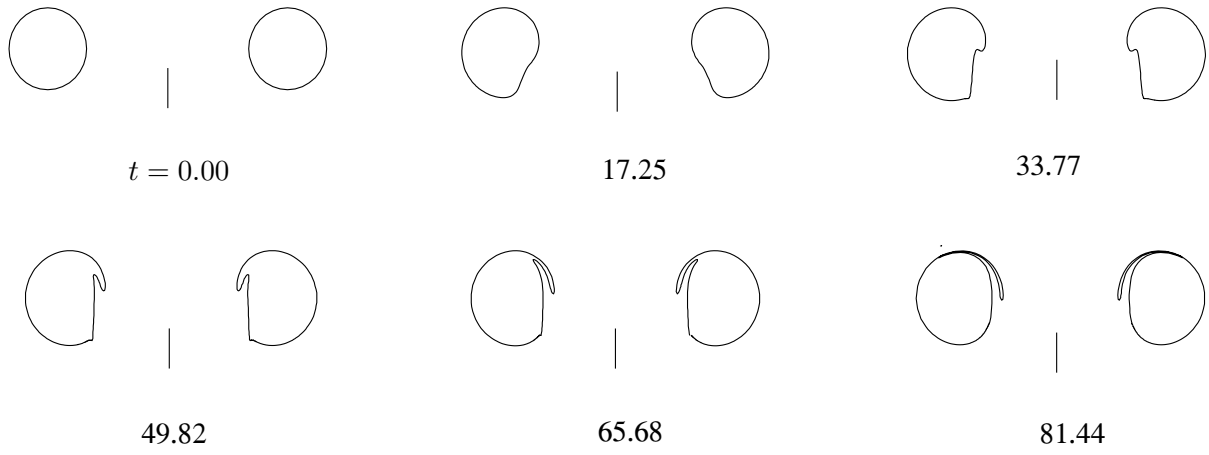


Figure 5.13: Deformation of a single toroidal drop. Volume 1, $\Gamma = 0.02$, $\lambda = 1$, $\lambda_1 = 4$ *and* $\lambda_2 = 1$.

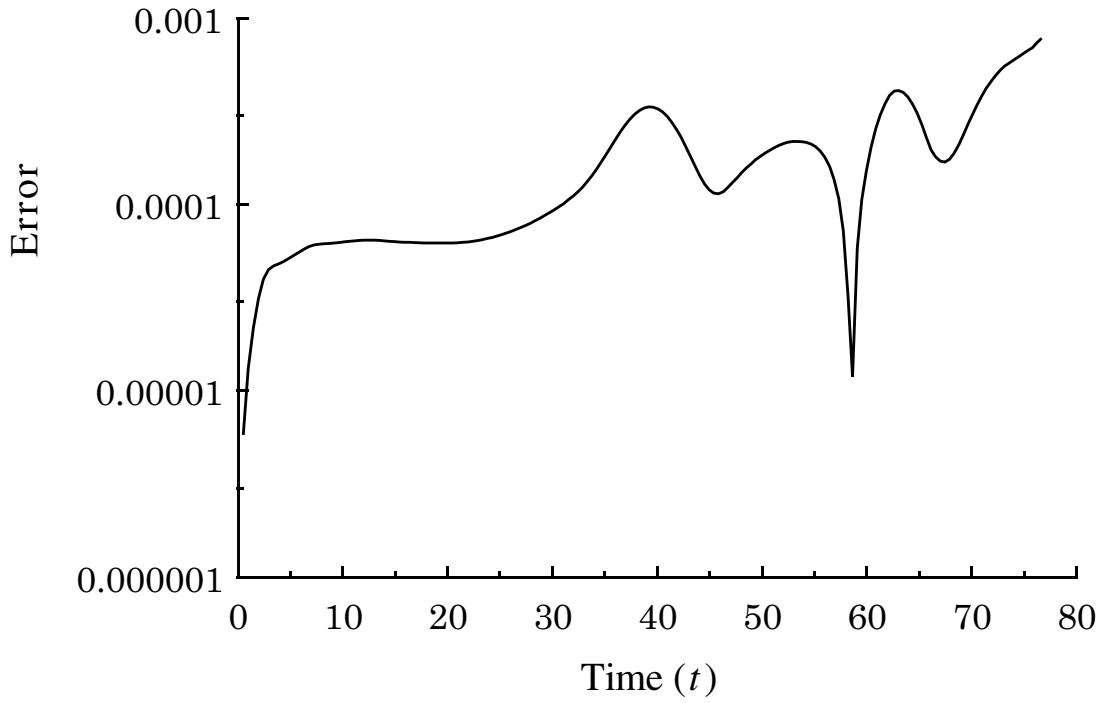


Figure 5.14: Deviation in volume for simulation shown in Figure 5.13.

As the simulation progresses, the outer surface remains circular while the inner region deforms and filamentation is evident near the top of the inner region. In addition, the inner radius does not change significantly. This is in accord with the linear analysis of Kojima *et al.* (1984) where they show that for a slender circular torus, the inner radius should not expand.

The deviation from initial volume is shown in Figure 5.14. This error increases with time, but remains at acceptable levels throughout the simulation.

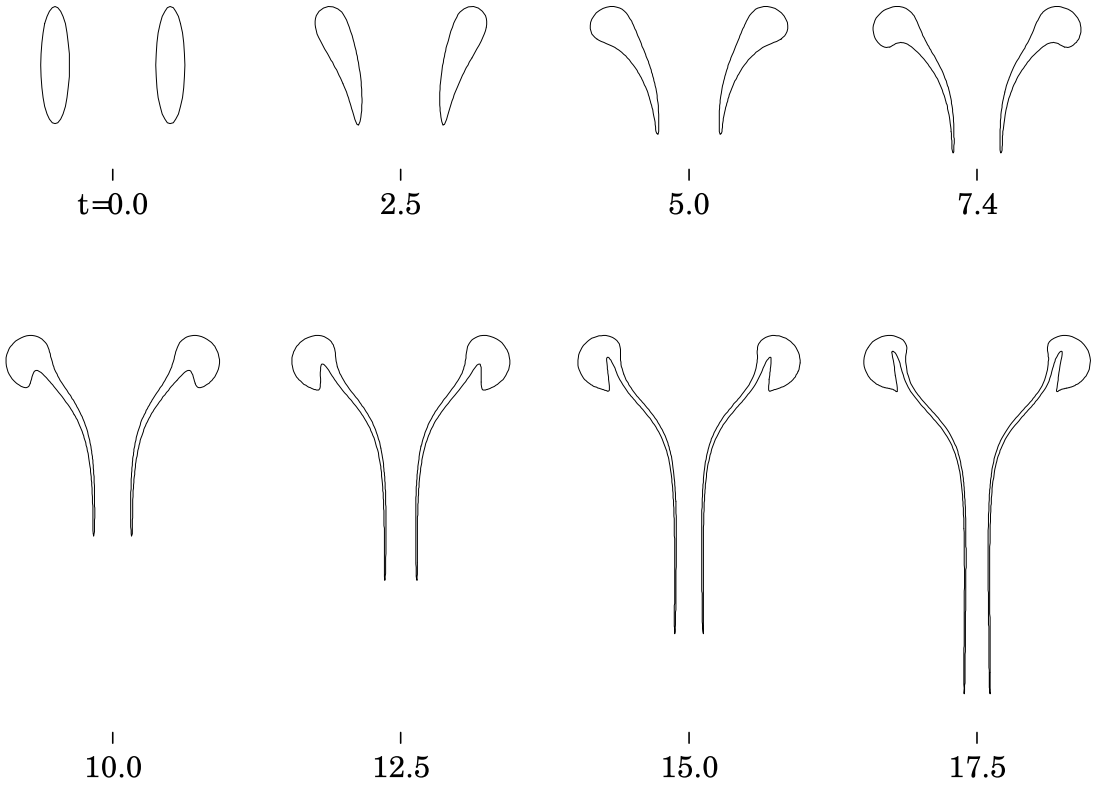


Figure 5.15: Deformation of a single toroidal drop. Volume=4, $\Gamma = 0.02$, $\lambda = 1$, $\lambda_1 = 1.2$, $\lambda_2 = 0.2$

In Figure 5.15, the initial parameters are changed so that the cross section takes on a distinct prolate shape. The overall torus is no longer slender and the parameters are $\lambda_1 = 1.2, \lambda_2 = 0.2$. The evolution here is similar to that of a single prolate spheroid (Pozrikidis 1990b, Figure $2(a-i)$). At the leading edge, a circular region is formed, as does a long tail filament. Also, an internal spike is present near the trailing edge of the circular head. We report that the inner radius increases in size in contrast to the prediction of Kojima *et al.* (1984). However, their analysis was restricted to slender toroidal drops with a circular cross-section.

As a final remark, we point out that the deformation in Figure 5.15 proceeds at a much faster pace that in Figure 5.13. The obvious reason being that the drop volume in Figure 5.15 is 4 times larger than in Figure 5.13.

5.8 Conclusions

The behaviour of the three-drop interactions can be explained, in part, from the results for two-drop interactions. The drops which are closer together can act like a single drop in their effect

on the other drop. In two-drop dynamics, the leading drop usually takes on an oblate shape and coats the trailing drop (if surface tension is low enough). The trailing drop takes on an initial prolate shape, then forms an external spike (protrusion) as most of its mass is concentrated into a spherical head, as in Figure 5.5. This is pointed out in Manga and Stone (1993) and explained using a simple point force analogy. In the three cases considered in Figures 5.7, 5.8 and 5.9, the first and last drops broadly exhibit this behaviour. The major effect of separation is the resulting deformation of the middle drop. It feels the presence of the flow field disturbance both ahead and behind.

We have developed a boundary integral algorithm that can simulate the axisymmetric interaction of many viscous drops under a constant buoyancy force. The main strengths of this method are the accurate representation of drop surfaces, modification of the singular kernels to enable accurate evaluation of the integrals using appropriate Gauss quadrature rules and a simple adaptive time-step to constrain the amount of surface deformation. These numerical investigations suggest we are able to carry out drop simulations with greater accuracy and efficiency.

Chapter 6

Concluding Remarks and Future Directions

The main aim of this work was to present highly accurate and reliable algorithms to study particle interactions in Stokes flow. Below, we briefly summarise the significant results and indicate directions for possible future work.

The boundary integral equations are the most appropriate form of the governing equations when considering creeping flows in external domains with boundaries that may have complicated and/or dynamic shapes. To obtain meaningful results for complex multi-particle systems one needs to carefully consider the solution method which will involve, at least, some representation for the unknown and an algorithm to perform the singular quadrature. This can be further complicated by the dimension of the solution (two for axi-symmetric problems) and the nature of the boundaries.

In order to isolate the main issues involved with the numerical inversion of boundary integral equations we considered the ideal case of solving Symm's integral equation in chapter 3. We showed that an interpolation-collocation scheme performed extremely well and an optimal grid as well as optimal collocation points were identified. We should, of course, provide a caveat for the use of the word *optimal*. The results identified in this chapter are optimal in the sense that the unknown was approximated by a constant functional over one interval. Even so, we showed that this *crude* approximation can provide meaningful insights into how one should discretize and collocate. The method was expanded and generalised somewhat in appendix C to show how one can construct an n point composite quadrature rule to account for general singular integrands

where only limited knowledge of the singularity is required (first two moments), whilst identifying an accurate grid. This contrasts with Newton-Cotes rules where no provision is made for the singular integrand and with Gauss quadrature where an n point rule requires knowledge of the first $2n$ weighted moments and no sense of *composite grid* exists. We hasten to add that Gauss rules are of course more accurate than the ones developed here, they are used in the simulations of chapters 4 and 5, but we propose the use of these Newton-Cotes like rules for theoretical analysis of integral equations because of their ease of use in obtaining an abundance of error results in a variety of norms and still accounting for the singular nature of the integrands.

Two obvious improvements can be made with the Peano kernel approximation (3.41). One is to increase the number of intervals, thus adapting the calculus to a piecewise functional defined over some arbitrary grid. This was done in section C.6 of appendix C, but the main difficulty would be in obtaining, and subsequently manipulating, the expression for the determinant of the composite system. We refer to the cumbersome expression (3.51) as evidence of this. Another approach would be to increase the order of the approximation. For example, one could use the Peano kernel

$$p_n(t,x) := \frac{1}{(n-1)!}\int_t^x (t-u)^{n-1}w(u)\,du, \qquad a \le t,x \le b,\ n \ge 1,$$

to obtain an integral inequality in norms of $f^{(n)}$. Integrating by parts once, solving the resulting recurrence relation and taking bounds gives

$$\left|\int_a^b f(t)w(t)\,dt - \sum_{k=0}^{n-1}(-1)^{n-k}\left(f^{(k)}(a)p_{k+1}(a,x) - f^{(k)}(b)p_{k+1}(b,x)\right)\right| \le \|f^{(n)}\|_\infty \int_a^b |p_n(t,x)|\,dt, \tag{6.1}$$

an approximation to the weighted integral in terms of the first $n-1$ derivatives of f at the endpoints $t = a, b$. This is ideally suited to Hermite interpolation since, as in equation (4.9),

$$f(t) \approx \sum_{k=0}^{n-1} f^{(k)}(a)H_{k+1}(t) + f^{(k)}(b)H_{k+1+n}(t). \tag{6.2}$$

Making use of (6.1) and (6.2) should, in principle, lead to error results for arbitrary order Hermite interpolation. The main idea in chapter 3 and appendix C is that such simple analytic techniques may be employed to provide insight into how numerical schemes will impact the resulting solution. Future work, as identified here, is being actively pursued.

The results of chapter 3 inspired the algorithm presented in chapter 4. An arbitrary order Hermite interpolant over a general grid was employed to construct a linear system which, when solved,

returned the behaviour of the particle traction at the grid points. The complete elliptic integrals were approximated with a polynomial-logarithmic expansion developed in appendix D. As a result, a combination of Gauss-Legendre and Gauss-Log quadrature rules were employed to evaluate the integrals. We showed that this method worked well for simple spheroidal geometries, but became unstable for more complicated particle shapes (surface A). In an effort to increase stability, as well as accuracy, two curvature based methods were presented.

The first method was based on choosing an *a priori* grid so that the curvature was represented uniformly across the entire domain. This scheme produced a five-fold increase in accuracy for the calculated traction of a highly prolate spheroid and prevented the instability reported earlier with the surface A. The second method involved modification of the integrand. The idea being that the traction f_1 may be decomposed as

$$f_1 = F(\kappa) f_w, \tag{6.3}$$

where F is some function of the curvature chosen so that f_w is independent of the curvature. In this way, almost any trivial grid can be used. We chose $F(\kappa) = \kappa^{1/3}$ since this would return a constant solution $f_w = c$ for the spheroid. With surface A, the returned solution f_w lacked the very large derivatives of the traction f_1, giving an *a posteriori* confirmation to the validity of the decomposition (6.3). In addition, it was clear that the traction, $f_1 = \kappa^{1/3} f_w$ was much more accurate when compared with the un-weighted solution f_1. Unfortunately, this method was not as accurate as the curvature grid scheme and it produced spurious results at inflection points of the particle.

Properly implemented, the weighted kernel approach should produce stable and accurate results without requiring the calculation of special grids. For this reason, it warrants further investigation in an effort to improve the performance and eliminate the current flaw at inflection points.

The results presented in chapter 4 provide the basis for the numerical investigation of multi-particle interactions. For example, a simple three sphere buoyant interaction is shown in Figures 6.1 and 6.2. We can see that if the middle sphere begins it's motion closer to the trailing sphere then it will move, in a relative sense, from the trailing sphere to the leading sphere. This behaviour was first predicted by Happel and Brenner (1965). In contrast, attempting to solve a similar problem with three surface A's, instead of three spheres, will almost certainly fail unless care is taken with the interpolation, integration and grid distribution. It is clear that the methods presented here can be used to study similar phenomena for a larger number of particles involving complex geometries.

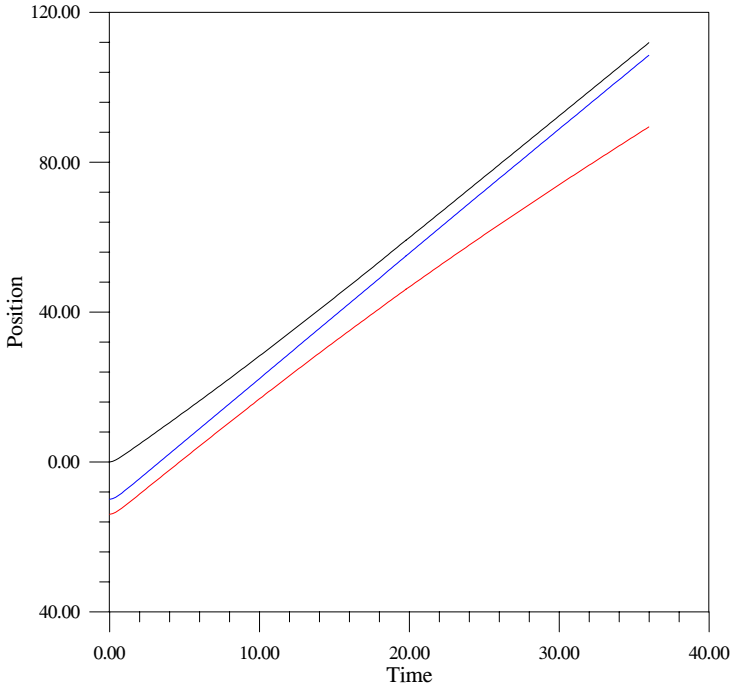


Figure 6.1: Buoyant three sphere simulation (trailing sphere) —, (middle sphere) —, (leading sphere) —. Position versus time

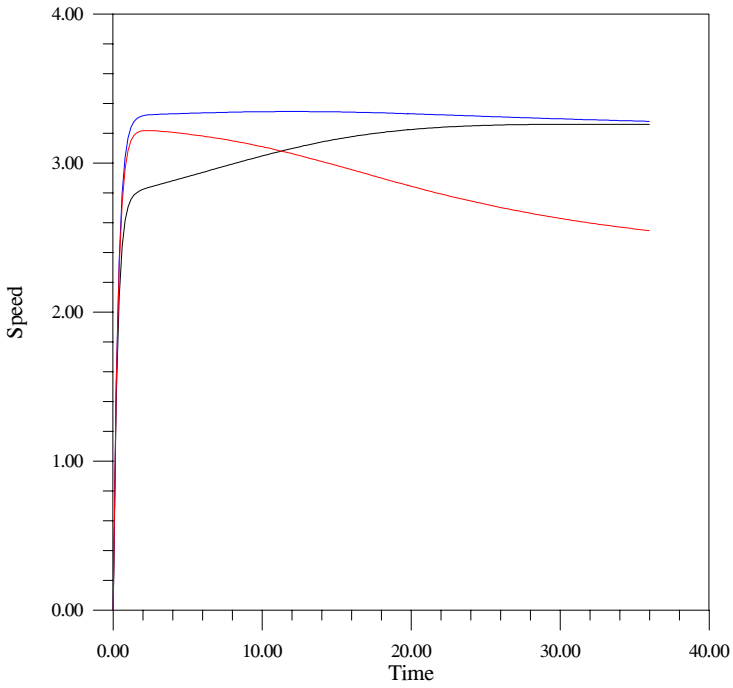


Figure 6.2: Buoyant three sphere simulation (trailing sphere) —, (middle sphere) —, (leading sphere) —. Velocity versus time

In chapter 5, studies of liquid drop interactions were presented. The numerical integration, including the polynomial approximation to the complete elliptic integrals, followed that of chapter 4. The unknown surfaces and velocities were represented using clamped cubic splines parameterised with arc-length. One and two drop simulations were presented and compared with previously reported results and we showed that the numerical method is more accurate in terms of volume changes.

Numerical simulations were also presented for three drop interactions and a single toroidal drop deformation. We reported that a three drop interaction can be explained, in part, in terms of two drop dynamics. The evolution of the toroidal drop was compared with the experimental results of Kojima *et al*. (1984). They report that slender, circular toroidal drops increase their inner radius under buoyant motion. Their linear analysis suggests that the creeping flow approximation is deficient in this regard and that small inertia terms are required to reproduce their experimental results. We confirmed this and reported that the numerical simulation for a slender circular torus admits no solution involving increase in the inner radius (Figure 5.13). In addition, we also reported that changing the initial toroidal configuration to a prolate cross-section with no overall slenderness did lead to a drop evolution whose inner radius increased (Figure 5.15).

It is clear that the algorithms of chapters 4 and 5 are capable of simulating complex particle behaviour in Stokes flow. Even though most physical systems rarely exhibit axial symmetry, some insight may be gained by restricting the flow in this way. One example is the brief report on the observation of two interacting toroidal drops. Kojima *et al*. (1984) report that toroidal drops seem to interact via a *leap frog* mechanism. That is, their leading drop expands allowing the contracting trailing drop to pass through its inner radius . We report that this behaviour has been reproduced numerically, see Figure 6.3, confirming the validity of the Stokes approximation as well as the numerical method given here.

In conclusion, the analysis undertaken in this dissertation provides a sound basis for the theoretical investigation of Fredholm integral equations, for implementing general numerical methods based on boundary integral equations where a high degree of accuracy is required and, not least, the study of complex particle motion in Stokes flows.

Figure 6.3: Interaction of two toroidal drops. $\Gamma = 0.1, \lambda = 1.1$

Appendix A

Boundary Integral Formulation

For completeness we present here the derivation of the boundary integral equations (2.9) for a particle immersed in an unbounded and otherwise stationary fluid in creeping flow. For the derivation of the boundary integral equation for particles in an unbounded fluid with an external flow see Kim and Karrila (1991, p. 25).

The Green's function for Stokes flow in an infinite medium is given by the solution to

$$\frac{\partial Q_k}{\partial x_i}(\boldsymbol{x}) - \frac{\partial^2 G_{ik}}{\partial x_j \partial x_j} = \delta_{ik}\delta(\boldsymbol{x}) \tag{A.1}$$

and

$$\frac{\partial G_{ik}}{\partial x_i}(\boldsymbol{x}) = 0, \tag{A.2}$$

with the boundary conditions

$$|\boldsymbol{G}| \to 0 \qquad \text{and} \qquad |\boldsymbol{Q}| \to 0 \qquad \text{as} \qquad |\boldsymbol{x}| \to \infty, \tag{A.3}$$

where the usual summation convention has been adopted.

Physically, $G_{ik}(\boldsymbol{x})$ represents the velocity of the fluid at $\boldsymbol{x}$ in the i direction due to a Stokeslet (point force) in the k direction located at the origin. The associated pressure field is $Q_k(\boldsymbol{x})$. We will denote the stress tensor of this system by $\boldsymbol{\Sigma}$, it is given by

$$\Sigma_{ijk} = -Q_k\delta_{ij} + \frac{\partial G_{ik}}{\partial x_j} + \frac{\partial G_{jk}}{\partial x_i}, \tag{A.4}$$

and the stress tensor for the system (2.6)-(2.7) by $\boldsymbol{\tau}$, where

$$\tau_{ij} = -p\delta_{ij} + \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}. \tag{A.5}$$

Taking the divergence of the products $G_{ik}(\boldsymbol{y}-\boldsymbol{x})\tau_{ij}(\boldsymbol{y})$ and $u_i(\boldsymbol{y})\Sigma_{ijk}(\boldsymbol{y}-\boldsymbol{x})$ gives

$$G_{ik}(\boldsymbol{y}-\boldsymbol{x})\frac{\partial \tau_{ij}}{\partial y_j}(\boldsymbol{y}) = \frac{\partial}{\partial y_j}\Big(G_{ik}(\boldsymbol{y}-\boldsymbol{x})\tau_{ij}(\boldsymbol{y})\Big) - \frac{\partial G_{ik}}{\partial y_j}(\boldsymbol{y}-\boldsymbol{x})\tau_{ij}(\boldsymbol{y}), \tag{A.6}$$

and

$$u_i(\boldsymbol{y})\frac{\partial \Sigma_{ijk}}{\partial y_j}(\boldsymbol{y}-\boldsymbol{x}) = \frac{\partial}{\partial y_j}\Big(u_i(\boldsymbol{y})\Sigma_{ijk}(\boldsymbol{y}-\boldsymbol{x})\Big) - \frac{\partial u_i}{\partial y_j}(\boldsymbol{y})\Sigma_{ijk}(\boldsymbol{y}-\boldsymbol{x}), \tag{A.7}$$

respectively.

Subtracting (A.7) from (A.6), integrating over the whole fluid domain (Ω) and applying the divergence theorem gives

$$\begin{aligned}\int_{\Omega}\left\{G_{ik}(\boldsymbol{y}-\boldsymbol{x})\frac{\partial \tau_{ij}}{\partial y_j}(\boldsymbol{y}) - u_i(\boldsymbol{y})\frac{\partial \Sigma_{ijk}}{\partial y_j}(\boldsymbol{y}-\boldsymbol{x})\right\} dV(\boldsymbol{y}) = \\ -\int_{\mathcal{S}}\Big\{G_{ik}(\boldsymbol{y}-\boldsymbol{x})\tau_{ij}(\boldsymbol{y}) - u_i(\boldsymbol{y})\Sigma_{ijk}(\boldsymbol{y}-\boldsymbol{x})\Big\}n_j(\boldsymbol{y})\,dS(\boldsymbol{y}) - \\ \int_{\Omega}\left\{\frac{\partial G_{ik}}{\partial y_j}(\boldsymbol{y}-\boldsymbol{x})\tau_{ij}(\boldsymbol{y}) - \frac{\partial u_i}{\partial y_j}(\boldsymbol{y})\Sigma_{ijk}(\boldsymbol{y}-\boldsymbol{x})\right\} dV(\boldsymbol{y}).\end{aligned} \tag{A.8}$$

The parameter $\mathcal{S}$ represents the particle surface and $\boldsymbol{n}$ the unit outward normal to $\mathcal{S}$. We note that there are no integrals over the surface of the infinite fluid domain since the Stokeslet is $O(1/|\boldsymbol{y}-\boldsymbol{x}|)$ and its associated stress tensor decays like $O(1/|\boldsymbol{y}-\boldsymbol{x}|^2)$.

The incompressibility condition (2.7) and (A.2) with the stress tensors (A.5) and (A.4) causes the integrand of the last term in equation (A.8) to vanish. Furthermore, we know that the governing equations (2.6) and (A.1) can be reformulated as

$$\frac{\partial \tau_{ij}}{\partial y_j}(\boldsymbol{y}) = 0 \qquad \text{and} \qquad \frac{\partial \Sigma_{ijk}}{\partial y_j}(\boldsymbol{y}-\boldsymbol{x}) = -\delta_{ik}\delta(\boldsymbol{y}-\boldsymbol{x}), \tag{A.9}$$

respectively, which upon substitution into (A.8) yields

$$c(\boldsymbol{x})u_k(\boldsymbol{x}) = -\int_{\mathcal{S}} G_{ik}(\boldsymbol{y}-\boldsymbol{x})\tau_{ij}(\boldsymbol{y})n_j(\boldsymbol{y})\,dS(\boldsymbol{y}) + \int_{\mathcal{S}}\Sigma_{ijk}(\boldsymbol{y}-\boldsymbol{x})u_i(\boldsymbol{y})n_j(\boldsymbol{y})\,dS(\boldsymbol{y}), \tag{A.10}$$

where

$$c(\boldsymbol{x}) = \begin{cases} 1, & \boldsymbol{x}\in\Omega, \\ \frac{1}{2}, & \boldsymbol{x}\in\mathcal{S}. \end{cases} \tag{A.11}$$

Since $\boldsymbol{G}$ is even and $\boldsymbol{\Sigma}$ is odd, we may re-write (A.10) as

$$c(\boldsymbol{x})u_k(\boldsymbol{x}) = -\int_{\mathcal{S}} G_{ik}(\boldsymbol{x}-\boldsymbol{y})f_i(\boldsymbol{y})\,dS(\boldsymbol{y}) - \int_{\mathcal{S}}\Sigma_{ijk}(\boldsymbol{x}-\boldsymbol{y})u_i(\boldsymbol{y})n_j(\boldsymbol{y})\,dS(\boldsymbol{y}), \tag{A.12}$$

where $\boldsymbol{f} = \boldsymbol{\tau} \cdot \boldsymbol{n}$ is the traction.

The surface $\mathcal{S}$ is assumed to be everywhere smooth (to be precise $\mathcal{S}$ is assumed to be a Lyapunov surface). At any non-analytic point[1] of $\mathcal{S}$, c can be calculated from the solid-angle at that point (Seybert *et al*. 1985).

If the particle is rigid then the double-layer term may be simplified. In this case, the velocity is constant over the surface and so may be taken out of the integral, then re-applying the divergence theorem so that the region of interest is in the particle interior gives

$$\int_{\mathcal{S}} \Sigma_{ijk}(\boldsymbol{x} - \boldsymbol{y}) u_i(\boldsymbol{y}) n_j(\boldsymbol{y})\, dS(\boldsymbol{y}) = -(1 - c(\boldsymbol{x})) u_k(\boldsymbol{x}). \tag{A.13}$$

Substituting (A.13) into (A.12) we arrive at the equation

$$u_k(\boldsymbol{x}) = -\int_{\mathcal{S}} G_{ik}(\boldsymbol{x} - \boldsymbol{y}) f_i(\boldsymbol{y})\, dS(\boldsymbol{y}), \tag{A.14}$$

where $\boldsymbol{f} = \boldsymbol{\tau} \cdot \boldsymbol{n}$.

If $\boldsymbol{x} \in \mathcal{S}$ then (A.14) is a three-dimensional Fredholm integral equation of the first kind in the surface traction $\boldsymbol{f}$.

[1] By non-analytic point, $P \in \mathcal{S}$, we mean that any parameterisation of a space-curve embedded in $\mathcal{S}$ that contains P has no Taylor expansion about P.

Appendix B

Hermite Interpolation Polynomials

We seek a polynomial $F^*(t)$ that interpolates a function $F(t)$ on the unit interval. In addition, we require the first $N-1$ derivatives of $F^*(t)$ to interpolate the first $N-1$ derivatives of $F(t)$. Therefore we impose the conditions

$$F^{*^{(m)}}(0) = F^{(m)}(0) \quad \text{and} \quad F^{*^{(m)}}(1) = F^{(m)}(1), \quad \text{for} \quad m = 0, 1, \ldots, N-1, \tag{B.1}$$

and write the interpolant in the form

$$F^*(t) = \sum_{m=0}^{N-1} F^{(m)}(0) H_{m+1}(N;t) + F^{(m)}(1) H_{m+N+1}(N;t), \tag{B.2}$$

where $H_k(N;t)$, $k = 1, 2, \ldots, 2N$ are the Hermite interpolation polynomials to be calculated. By substituting (B.2) into (B.1) it is easy to see that these polynomials satisfy

$$H_k^{(m)}(N;0) = \delta_{m+1,k} \qquad \text{and} \qquad H_k^{(m)}(N;1) = \delta_{m+N+1,k}. \tag{B.3}$$

It is now a simple matter of utilising these $2N$ conditions to calculate the coefficients of each polynomial $H_k(N;t)$. Algorithm B.1 lists a Maple (Char *et al*. 1991) program that generates the Hermite interpolation polynomials for any given order and Figure B.1 shows some sample output. Table B.1 tabulates the polynomials for $N = 2, 4$ and 6.

```
      herm := proc( x , n )
#
          local i , j , eq , oldeq , poly , c , H , soln:
#
#     Written by:     John Roumeliotis and Adam Kucera
#
#     Written on:     10 / 04 / 1992.
#     Last update:    02 / 03 / 2000.
#
#     Given parameters x and n, this routine returns the
#     Hermite interpolation polynomials of order n in the
#     parameter x.
#
#  Parameter Checking
          if not( type( x , symbol ) and type( n , integer ) ) then
              ERROR( `Arguments of wrong type` )
          elif n <= 0 then
              ERROR( `The order (n) must be positive` )
          fi:
#  Set up Hermite polynomial, derivatives and set each
#  condition to zero (eqn B3)
          poly[0] := sum( c[i]*x^i , i=0..2*n-1 ):
          eq[1] := subs( x=0 , poly[0] ) =0:
          eq[n+1] := subs( x=1 , poly[0] ) =0:
          for i from 1 to n-1 do
              poly[ i ] := diff( poly[ i-1 ] , x ):
              eq[i+1] := subs( x=0 , poly[i] ) =0:
              eq[i+1+n] := subs( x=1 , poly[i] ) =0
          od: i := 'i':
#  Evaluate each Hermite
          for i from 1 to 2*n do
#  Implement the proper condition (eqn B3)
              oldeq := eq[i]:
              if i > n then
                  eq[i] := subs( x=1 , poly[i-n-1] ) = 1
              else
                  eq[i] := subs( x=0 , poly[i-1] ) = 1
              fi:
#  Solve equations for coefficients (c[i])
              soln := solve( {eq[j]$j=1..2*n} , {c[j]$j=0..2*n-1} ):
              H[i] := factor( subs( soln , poly[0] ) ):
              eq[i] := oldeq
          od: i := 'i':
          [ H[i]$ i=1..2*n ]
      end;
```

Algorithm B.1: Maple program to calculate Hermite interpolation polynomials of any order.

N	k	Hermite interpolation polynomials $H_k(N;t)$	
2	1, 2	$(2\,t+1)\,(1-t)^2$	$t\,(1-t)^2$
	3, 4	$-t^2\,(-3+2\,t)$	$-t^2\,(1-t)$
4	1, 2	$(20\,t^3+10\,t^2+4\,t+1)\,(1-t)^4$	$t\,(10\,t^2+4\,t+1)\,(1-t)^4$
	3, 4	$\frac{1}{2}t^2\,(4\,t+1)\,(1-t)^4$	$\frac{1}{6}t^3\,(1-t)^4$
	5, 6	$-t^4\,(-35+84\,t-70\,t^2+20\,t^3)$	$-t^4\,(1-t)\,(10\,t^2-24\,t+15)$
	7, 8	$-\frac{1}{2}t^4\,(4\,t-5)\,(1-t)^2$	$-\frac{1}{6}t^4\,(1-t)^3$
6	1	$(252\,t^5+126\,t^4+56\,t^3+21\,t^2+6\,t+1)\,(1-t)^6$	
	2	$t\,(126\,t^4+56\,t^3+21\,t^2+6\,t+1)\,(1-t)^6$	
	3	$\frac{1}{2}t^2\,(56\,t^3+21\,t^2+6\,t+1)\,(1-t)^6$	
	4	$\frac{1}{6}t^3\,(21\,t^2+6\,t+1)\,(1-t)^6$	
	5	$\frac{1}{24}t^4\,(6\,t+1)\,(1-t)^6$	
	6	$\frac{1}{120}t^5\,(1-t)^6$	
	7	$-t^6\,(-462+1980\,t-3465\,t^2+3080\,t^3-1386\,t^4+252\,t^5)$	
	8	$-t^6\,(1-t)\,(126\,t^4-560\,t^3+945\,t^2-720\,t+210)$	
	9	$-\frac{1}{2}t^6\,(56\,t^3-189\,t^2+216\,t-84)\,(1-t)^2$	
	10	$-\frac{1}{6}t^6\,(21\,t^2-48\,t+28)\,(1-t)^3$	
	11	$-\frac{1}{24}t^6\,(6\,t-7)\,(1-t)^4$	
	12	$-\frac{1}{120}t^6\,(1-t)^5$	

Table B.1: Hermite interpolation polynomials, $H_k(N;t)$, for $k = 1, \ldots, 2N$, $N = 2, 4, 6$.

```
    |\^/|     Maple V Release 5.1 (Single User)
._|\|   |/|_. Copyright (c) 1981-1998 by Waterloo Maple Inc. All rights
 \  MAPLE  /  reserved. Maple and Maple V are registered trademarks of
 <____ ____>  Waterloo Maple Inc.
      |       Type ? for help.
> read `herm1.mpl`;
> herm(t,2);
                           2            2    2               2
        [(2 t + 1) (-1 + t) , t (-1 + t) , -t  (-3 + 2 t), t  (-1 + t)]

> done
```

Figure B.1: Sample Maple output from algorithm B.1

Appendix C

Some Integral Inequalities

C.1 Introduction

In this appendix a weighted (or product) quadrature rule is developed and an *a priori* error bound is obtained for functions with bounded second derivative. The method is based on Ostrowski's integral inequality, and as such is amenable to the production of error bounds for a variety of norms. This has been done for the usual Newton-Cotes type rules (Cerone *et al.* 1999b; Cerone *et al.* 1999a; Cerone *et al.* 1999c; Cerone *et al.* 1999d) and two-dimensional (cubature) rules (Dragomir *et al.* 1999; Hanna *et al.* 1999). In addition, weighted rules for differentiable functions using a number of norms have recently been established (Cerone *et al.* 1999) as well as for functions of Hölder type (Dragomir *et al.* 1999).

As well as developing a weighted integral inequality, an application in numerical integration is presented. However, the method may be employed in other areas such as approximation theory (integral equations being but one example) and probability distributions. In this particular case, the weight function w can be considered as some probability density function (pdf).

C.2 Ostrowski's inequality

In 1938, Ostrowski (see for example Mitrinović *et al.* (1994, p. 468)) proved the following bound between a function and its average:

THEOREM C.1. *Let $f : I \subseteq \mathbb{R} \to \mathbb{R}$ be a differentiable mapping in I^o (I^o is the interior of I),*

and let $a, b \in I^o$ with $a < b$. If $f' : (a, b) \to \mathbb{R}$ is bounded on (a, b), i.e., $\|f'\|_\infty := \sup_{t\in(a,b)} |f'(t)| < \infty$, then we have the inequality:

$$\left| \frac{1}{b-a} \int_a^b f(t)\,dt - f(x) \right| \leq \left[\frac{1}{4} + \frac{\left(x - \frac{a+b}{2}\right)^2}{(b-a)^2} \right] (b-a)\|f'\|_\infty \tag{C.1}$$

for all $x \in (a, b)$.

The constant $\frac{1}{4}$ is sharp in the sense that it cannot be replaced by a smaller one.

Recently, a similar result for twice differentiable mappings (Cerone *et al*. 1999b) was proved.

THEOREM C.2. *Let $f : [a, b] \to \mathbb{R}$ be a twice differentiable mapping such that $f'' : (a, b) \to \mathbb{R}$ is bounded on (a,b), i.e. $\|f''\|_\infty := \sup_{t\in(a,b)} |f''(t)| < \infty$. Then we have the inequality*

$$\left| \frac{1}{b-a} \int_a^b f(t)\,dt - f(x) + \left(x - \frac{a+b}{2} \right) f'(x) \right| \leq \left[\frac{1}{24} + \frac{\left(x - \frac{a+b}{2}\right)^2}{2(b-a)^2} \right] (b-a)^2\|f''\|_\infty \tag{C.2}$$

for all $x \in [a, b]$.

In this appendix, this result is extended by developing an Ostrowski-type inequality for weighted integrals. Applications to special weight functions and numerical integration are investigated.

C.3 Preliminaries

In the next section we will construct weighted (or product) integral inequalities. The weight function (or density) is assumed to be non-negative and integrable over its entire domain. We begin by defining a few generic quantitative measures of the weight.

Definition C.3. Let $w : (a, b) \to [0, \infty)$ be an integrable function, i.e. $\int_a^b w(t)\,dt < \infty$, then we define

$$m_i(a, b) = \int_a^b t^i w(t)\,dt, \qquad i = 0, 1, \ldots \tag{C.3}$$

as the i^{th} *moment* of w.

Definition C.4. We define the *mean* of the interval $[a, b]$ with respect to the density w as

$$\mu(a,b) = \frac{m_1(a,b)}{m_0(a,b)} \tag{C.4}$$

and the *variance* by

$$\sigma^2(a,b) = \frac{m_2(a,b)}{m_0(a,b)} - \mu^2(a,b). \tag{C.5}$$

C.4 The results

C.4.1 1-point inequality

THEOREM C.5. *Let $f, w : (a,b) \to \mathbb{R}$ be two mappings on (a,b) with the following properties:*

1. $\sup_{t\in(a,b)} |f''(t)| < \infty$,
2. $w(t) \geq 0,\ \forall t \in (a,b)$,
3. $\int_a^b w(t)\,dt < \infty$,

then the following inequalities hold

$$\left|\frac{1}{m_0(a,b)}\int_a^b w(t)f(t)\,dt - f(x) + \big(x-\mu(a,b)\big)f'(x)\right|$$

$$\leq \frac{\|f''\|_\infty}{2}\Big[\big(x-\mu(a,b)\big)^2 + \sigma^2(a,b)\Big] \tag{C.6}$$

$$\leq \frac{\|f''\|_\infty}{2}\left(\left|x - \frac{a+b}{2}\right| + \frac{b-a}{2}\right)^2 \tag{C.7}$$

for all $x \in [a,b]$.

Proof. Define the mapping $K(\cdot,\cdot) : [a,b]^2 \to \mathbb{R}$ by

$$K(x,t) := \begin{cases} \int_a^t (t-u)w(u)\,du & a \leq t \leq x, \\ \int_b^t (t-u)w(u)\,du & x < t \leq b. \end{cases} \tag{C.8}$$

Integrating by parts gives

$$\begin{aligned}\int_a^b K(x,t)f''(t)\,dt &= \int_a^x\int_a^t (t-u)w(u)f''(t)\,dudt + \int_x^b\int_b^t (t-u)w(u)f''(t)\,dudt \\ &= f'(x)\int_a^b (x-u)w(u)\,du - \\ &\quad \int_a^x\int_a^t (t-u)w(u)f'(t)\,dudt - \int_x^b\int_b^t (t-u)w(u)f'(t)\,dudt \\ &= \int_a^b w(t)f(t)\,dt + f'(x)\int_a^b (x-u)w(u)\,du - f(x)\int_a^b w(u)\,du\end{aligned}$$

providing the identity

$$\int_a^b K(x,t)f''(t)\,dt = \int_a^b w(t)f(t)\,dt - m_0(a,b)f(x) + m_0(a,b)\big(x-\mu(a,b)\big)f'(x) \tag{C.9}$$

that is valid for all $x \in [a,b]$.

Now taking the modulus of (C.9) and using the well established properties of definite integrals we have

$$\begin{aligned}&\left|\int_a^b w(t)f(t)\,dt - m_0(a,b)f(x) + m_0(a,b)\big(x-\mu(a,b)\big)f'(x)\right| \\ &\qquad = \left|\int_a^b K(x,t)f''(t)\,dt\right| \\ &\qquad \le \|f''\|_\infty \int_a^b |K(x,t)|\,dt \\ &\qquad = \|f''\|_\infty \left[\int_a^x\int_a^t (t-u)w(u)\,dudt + \int_x^b\int_b^t (t-u)w(u)\,dudt\right] \\ &\qquad = \frac{\|f''\|_\infty}{2}\int_a^b (x-t)^2 w(t)\,dt.\end{aligned} \tag{C.10}$$

The last line was computed by reversing the order of integration and evaluating the inner integrals. To obtain the desired result (C.6) we expand the quadratic and complete the square

$$\int_a^b (x-t)^2 w(t)\,dt = m_0(a,b)\Big[\big(x-\mu(a,b)\big)^2 + \sigma^2(a,b)\Big]. \tag{C.11}$$

To obtain (C.7) note that

$$\begin{aligned}\int_a^b (x-t)^2 w(t)\,dt &\le \sup_{t\in[a,b]} (x-t)^2 m_o(a,b)\\ &= \max\{(x-a)^2,(x-b)^2\} m_0(a,b)\\ &= \frac{1}{2}\left((x-a)^2+(x-b)^2+\left|(x-a)^2-(x-b)^2\right|\right) m_0(a,b)\\ &= \left(\left|x-\frac{a+b}{2}\right|+\frac{b-a}{2}\right)^2 m_0(a,b) \qquad \text{(C.12)}\end{aligned}$$

which upon substitution into (C.10) furnishes the result. ■

Note that the inequality (C.6) is valid even for unbounded w or interval $[a,b]$. This is not the case with (C.2), hence the advantage of (C.6) is self evident.

COROLLARY C.6. *The following inequality holds*

$$\left(x-\mu(a,b)\right)^2+\sigma^2(a,b) \le \left(\left|x-\frac{a+b}{2}\right|+\frac{b-a}{2}\right)^2 \qquad \text{(C.13)}$$

for all $x\in[a,b]$.

Proof. The result is immediately obvious when the identity (C.11) is substituted into (C.12). ■

The inequality (C.6) produces a tighter bound than that of (C.7). We note that the latter inequality is independent of the weight w and so may be of use when comparing integrals with differing weight functions.

Remark C.7. Substituting $x=\mu(a,b)$ and $x=\frac{a+b}{2}$ into (C.13) and adding produces the following inequality for the variance in terms of the mean of a density

$$\sigma^2(a,b) \le \frac{b-a}{2}\left(\frac{b-a}{2}+\left|\mu(a,b)-\frac{a+b}{2}\right|\right). \qquad \text{(C.14)}$$

A tighter bound is obtained by using (C.13) only once. Substituting $x=\frac{a+b}{2}$ gives

$$\begin{aligned}\sigma^2(a,b) &\le \frac{(b-a)^2}{4}-\left(\frac{a+b}{2}-\mu\right)^2\\ &= (\mu-a)(b-\mu) \qquad \text{(C.15)}\\ &\le \frac{(b-a)^2}{4}. \qquad \text{(C.16)}\end{aligned}$$

COROLLARY C.8. *The inequality (C.6) is minimised at $x = \mu(a,b)$ producing the generalised* mean-point *inequality*

$$\left| \frac{1}{m_0(a,b)} \int_a^b w(t)f(t)\,dt - f(\mu(a,b)) \right| \leq \|f''\|_\infty \frac{\sigma^2(a,b)}{2} \tag{C.17}$$

$$\leq \|f''\|_\infty \frac{(\mu-a)(b-\mu)}{2}. \tag{C.18}$$

Proof. Substituting $\mu(a,b)$ for x in (C.6) produces the desired result. We remark that the mean point $x = \mu(a,b)$ not only minimises the bound of the inequality (C.6), but also causes the derivative term to vanish. The second bound is immediately obtained from (C.15). ■

The optimal point (C.4) can be interpreted in many ways. In a physical context, $\mu(a,b)$ represents the centre of mass of a one dimensional rod with mass density w. Equivalently, this point can be viewed as that which minimises the error variance for the probability density w (see Barnett *et al*. (1995) for an application). Finally (C.4) is also the Gauss node point for a one-point rule (Stroud and Secrest 1966). The bound in (C.17) is directly proportional to the variance of the density w. So that the tightest bound is achieved by sampling at the mean point of the interval (a,b), while its value is given by the variance. The bound in terms of the mean μ, equation (C.18), may be of value if the variance is unavailable.

C.4.2 2-point inequality

We develop a two point analogy of (C.6) where the result is extended to create an inequality with two independent parameters x_1 and x_2. This is mainly used in section C.6 to find an optimal grid for composite weighted-quadrature rules.

THEOREM C.9. *Let the conditions of Theorem C.5 hold, then we have the following 2-point inequality*

$$\begin{aligned}\Bigg| \int_a^b w(t)f(t)\,dt - m_0(a,\xi)f(x_1) + m_0(a,\xi)\big(x_1-\mu(a,\xi)\big)f'(x_1) \\ - m_0(\xi,b)f(x_2) + m_0(\xi,b)\big(x_2-\mu(\xi,b)\big)f'(x_2)\Bigg| \\ \leq \frac{\|f''\|_\infty}{2}\Big\{ m_0(a,\xi)\Big[\big(x_1-\mu(a,\xi)\big)^2 + \sigma^2(a,\xi)\Big] \\ + m_0(\xi,b)\Big[\big(x_2-\mu(\xi,b)\big)^2 + \sigma^2(\xi,b)\Big]\Big\} \end{aligned} \tag{C.19}$$

for all $a \le x_1 < \xi < x_2 \le b$.

Proof. We define the mapping $K(\cdot,\cdot,\cdot,\cdot) : [a,b]^4 \to \mathbb{R}$ by

$$K(x_1,x_2,\xi,t) := \begin{cases} \int_a^t (t-u)w(u)\,du & a \le t \le x_1 \\ \int_\xi^t (t-u)w(u)\,du & x_1 < t, \xi < x_2 \\ \int_b^t (t-u)w(u)\,du & x_2 \le t \le b \end{cases}$$

With this kernel, the proof is almost identical to that of Theorem C.5.

Integrating by parts produces the integral identity

$$\begin{aligned} &\int_a^b K(x_1,x_2,\xi,t)f''(t)\,dt \\ &\qquad = \int_a^b w(t)f(t)\,dt - m_0(a,\xi)f(x_1) + m_0(a,b)\big(x-\mu(a,\xi)\big)f'(x_1) \\ &\qquad\qquad - m_0(\xi,b)f(x_2) + m_0(\xi,b)\big(x-\mu(\xi,b)\big)f'(x_2). \end{aligned} \tag{C.20}$$

Re-arranging and taking bounds produces the result (C.19). ■

COROLLARY C.10. *The optimal location of the points* $x_1,\ x_2$ *and* ξ *satisfy*

$$x_1 = \mu(a,\xi), \qquad x_2 = \mu(\xi,b), \qquad \xi = \frac{\mu(a,\xi)+\mu(\xi,b)}{2} \tag{C.21}$$

Proof. By inspection of the right hand side of (C.19) it is obvious that choosing

$$x_1 = \mu(a,\xi) \qquad \text{and} \qquad x_2 = \mu(\xi,b) \tag{C.22}$$

minimises this quantity. To find the optimal value for ξ we write the expression in braces in (C.19) as

$$\begin{aligned} 2\int_a^b |K(x_1,x_2,\xi,t)|\,dt &= m_0(a,\xi)\Big[\big(x_1-\mu(a,\xi)\big)^2 + \sigma^2(a,\xi)\Big] \\ &\quad + m_0(\xi,b)\Big[\big(x_2-\mu(\xi,b)\big)^2 + \sigma^2(\xi,b)\Big] \\ &= \int_a^\xi (x_1-t)^2 w(t)\,dt + \int_\xi^b (x_2-t)^2 w(t)\,dt. \end{aligned} \tag{C.23}$$

Substituting (C.22) into the right hand side of (C.23) and differentiating with respect to ξ gives

$$\frac{d}{d\xi}\int_a^b |K(\mu(a,\xi),\mu(\xi,b),\xi,t)|\ dt = \big(\mu(\xi,b)-\mu(\xi,a)\big)\left(\xi - \frac{\mu(a,\xi)+\mu(\xi,b)}{2}\right)w(\xi).$$

Assuming $w(\xi) \neq 0$, then this equation possesses only one root. A minimum exists at this root since (C.23) is convex, and so the corollary is proved. ■

Equation (C.21) shows not only where we should sample within each subinterval (i.e. x_1 and x_2), but how the domain should be divided to make up these subintervals (ξ).

C.5 Some weighted integral inequalities

Integration with weight functions is used in countless mathematical problems. Two main areas are: (i) approximation theory and spectral analysis and (ii) statistical analysis and the theory of distributions.

In this section we evaluate the inequality (C.6) for the more popular weight functions. In each case (C.2) cannot be used since the weight w or the interval $(b-a)$ is unbounded. The optimal point (C.4) is easily identified.

C.5.1 Uniform (Legendre)

Substituting $w(t) = 1$ into (C.4) and (C.5) gives

$$\mu(a,b) = \frac{\int_a^b t\,dt}{\int_a^b dt} = \frac{a+b}{2} \tag{C.24}$$

and

$$\sigma^2(a,b) = \frac{\int_a^b t^2\,dt}{\int_a^b dt} - \left(\frac{a+b}{2}\right)^2 = \frac{(b-a)^2}{12}$$

respectively. Substituting into (C.6) produces (C.2). We note that the interval mean is simply the midpoint (C.24).

C.5.2 Logarithm

This weight is present in many physical problems; the main body of which exhibit some axial symmetry. Special logarithmic rules are used extensively in the Boundary Element Method

popularised by Brebbia (see for example Brebbia and Dominguez (1989)). Some applications of which include bubble cavitation (Blake and Gibson 1987) and viscous drop deformation (Rallison and Acriv and more recently by Roumeliotis and Fulford (2000)).

With $w(t) = \ln(1/t)$, $a = 0$, $b = 1$, (C.4) and (C.5) become

$$\mu(0,1) = \frac{\int_0^1 t \ln(1/t)\, dt}{\int_0^1 \ln(1/t)\, dt} = \frac{1}{4}$$

and

$$\sigma^2(0,1) = \frac{\int_0^1 t^2 \ln(1/t)\, dt}{\int_0^1 \ln(1/t)\, dt} - \left(\frac{1}{4}\right)^2 = \frac{7}{144}$$

respectively. Substituting into (C.6) gives

$$\left| \int_0^1 \ln(1/t) f(t)\, dt - f(x) + \left(x - \frac{1}{4}\right) f'(x) \right| \leq \frac{\|f''\|_\infty}{2} \left(\frac{7}{144} + \left(x - \frac{1}{4}\right)^2 \right).$$

The optimal point

$$x = \mu(0,1) = \frac{1}{4}$$

is closer to the origin than the midpoint (C.24) reflecting the strength of the log singularity.

C.5.3 Jacobi

Substituting $w(t) = 1/\sqrt{t}$, $a = 0$, $b = 1$ into (C.4) and (C.5) gives

$$\mu(0,1) = \frac{\int_0^1 \sqrt{t}\, dt}{\int_0^1 1/\sqrt{t}\, dt} = \frac{1}{3}$$

and

$$\sigma^2(0,1) = \frac{\int_0^1 t\sqrt{t}\, dt}{\int_0^1 1/\sqrt{t}\, dt} - \left(\frac{1}{3}\right)^2 = \frac{4}{45}$$

respectively. Hence, the inequality for a Jacobi weight is

$$\left| \frac{1}{2} \int_0^1 \frac{f(t)}{\sqrt{t}}\, dt - f(x) + \left(x - \frac{1}{3}\right) f'(x) \right| \leq \frac{\|f''\|_\infty}{2} \left(\frac{4}{45} + \left(x - \frac{1}{3}\right)^2 \right).$$

The optimal point

$$x = \mu(0,1) = \frac{1}{3}$$

is again shifted to the left of the mid-point due to the $t^{-1/2}$ singularity at the origin.

C.5.4 Chebyshev

The mean and variance for the Chebyshev weight $w(t) = 1/\sqrt{1-t^2}$, $a = -1, b = 1$ are

$$\mu(-1,1) = \frac{\int_{-1}^{1} t/\sqrt{1-t^2}\,dt}{\int_{-1}^{1} 1/\sqrt{1-t^2}\,dt} = 0$$

and

$$\sigma^2(-1,1) = \frac{\int_{-1}^{1} t^2\sqrt{1-t^2}\,dt}{\int_{-1}^{1} 1/\sqrt{1-t^2}\,dt} - 0^2 = \frac{1}{2}$$

respectively. Hence, the inequality corresponding to the Chebyshev weight is

$$\left|\frac{1}{\pi}\int_{-1}^{1}\frac{f(t)}{\sqrt{1-t^2}}\,dt - f(x) + xf'(x)\right| \leq \frac{\|f''\|_\infty}{2}\left(\frac{1}{2}+x^2\right).$$

The optimal point

$$x = \mu(-1,1) = 0$$

is at the mid-point of the interval reflecting the symmetry of the Chebyshev weight over its interval.

C.5.5 Laguerre

The conditions in Theorem C.5 are not violated if the integral domain is infinite. The Laguerre weight $w(t) = e^{-t}$ is defined for positive values, $t \in [0,\infty)$. The mean and variance of the Laguerre weight are

$$\mu(0,\infty) = \frac{\int_0^\infty te^{-t}\,dt}{\int_0^\infty e^{-t}\,dt} = 1$$

and

$$\sigma^2(0,\infty) = \frac{\int_0^\infty t^2e^{-t}\,dt}{\int_0^\infty e^{-t}\,dt} - 1^2 = 1$$

respectively.

The appropriate inequality is

$$\left|\int_0^\infty e^{-t}f(t)\,dt - f(x) + (x-1)f'(x)\right| \leq \frac{\|f''\|_\infty}{2}\left(1+(x-1)^2\right),$$

from which the optimal sample point of $x = 1$ may be deduced.

C.5.6 Hermite

Finally, the Hermite weight is $w(t) = e^{-t^2}$ defined over the entire real line. The mean and variance for this weight are

$$\mu(-\infty,\infty) = \frac{\int_{-\infty}^{\infty} te^{-t^2}\,dt}{\int_{-\infty}^{\infty} e^{-t^2}\,dt} = 0$$

and

$$\sigma^2(-\infty,\infty) = \frac{\int_{-\infty}^{\infty} t^2e^{-t^2}\,dt}{\int_{-\infty}^{\infty} e^{-t^2}\,dt} - 0^2 = \frac{1}{2}$$

respectively.

The inequality from Theorem C.5 with the Hermite weight function is thus

$$\left|\frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty} e^{-t^2}f(t)\,dt - f(x) + xf'(x)\right| \leq \frac{\|f''\|_\infty}{2}\left(\frac{1}{2}+x^2\right),$$

which results in an optimal sampling point of $x = 0$.

C.6 Application in numerical integration

We define a grid $I_n : a = \xi_0 < \xi_1 < \cdots < \xi_{n-1} < \xi_n = b$ on the interval [a,b], with $x_i \in [\xi_i, \xi_{i+1}]$ for $i = 0, 1, \ldots, n-1$. We have the following quadrature formula for weighted integrals:

THEOREM C.11. *Let the conditions in Theorem C.5 hold, then the following weighted quadrature rule holds*

$$\int_a^b w(t)f(t)\,dt = A(f,\boldsymbol{\xi},\boldsymbol{x}) + R(f,\boldsymbol{\xi},\boldsymbol{x}) \tag{C.25}$$

where

$$A(f,\boldsymbol{\xi},\boldsymbol{x}) = \sum_{i=0}^{n-1}\big(h_if(x_i) - h_i(x_i-\mu_i)f'(x_i)\big)$$

and

$$|R(f,\boldsymbol{\xi},\boldsymbol{x})| \leq \frac{\|f''\|_\infty}{2}\sum_{i=0}^{n-1}\left[(x_i-\mu_i)^2+\sigma_i^2\right]h_i. \tag{C.26}$$

The parameters h_i, μ_i and σ_i^2 are given by

$$h_i = m_0(\xi_i,\xi_{i+1}), \qquad \mu_i = \mu(\xi_i,\xi_{i+1}), \qquad \textit{and} \qquad \sigma_i^2 = \sigma^2(\xi_i,\xi_{i+1})$$

respectively.

Proof. Apply Theorem C.5 over the interval $[\xi_i, \xi_{i+1}]$ with $x = x_i$ to obtain

$$\left| \int_{\xi_i}^{\xi_{i+1}} w(t)f(t)\,dt - h_i f(x_i) + h_i(x_i - \mu_i)f'(x_i) \right| \leq \frac{\|f''\|_\infty}{2} h_i \left((x_i - \mu_i)^2 + \sigma_i^2 \right).$$

Summing over i from 0 to $n-1$ and using the triangle inequality produces the desired result. ■

COROLLARY C.12. *The optimal location of the points x_i, $i = 0, 1, 2, \ldots, n-1$, and grid distribution I_n satisfy*

$$x_i = \mu_i, \qquad i = 0, 1, \ldots, n-1 \qquad \textit{and} \tag{C.27}$$

$$\xi_i = \frac{\mu_{i-1} + \mu_i}{2}, \qquad i = 1, 2, \ldots, n-1, \tag{C.28}$$

producing the composite generalised mid-point rule for weighted integrals

$$\int_a^b w(t)f(t)\,dt = \sum_{i=0}^{n-1} h_i f(x_i) + R(f, \boldsymbol{\xi}, n) \tag{C.29}$$

where the remainder is bounded by

$$|R(f, \boldsymbol{\xi}, n)| \leq \frac{\|f''\|_\infty}{2} \sum_{i=0}^{n-1} h_i \sigma_i^2 \tag{C.30}$$

Proof. The proof follows that of Corollary C.10 where we observe that the minimum bound (C.26) will occur at $x_i = \mu_i$. Differentiating the right hand side of (C.26) we find

$$\frac{d}{d\xi_i} \sum_{i=0}^{n-1} \left[(x_i - \mu_i)^2 + \sigma_i^2 \right] h_i = 2w(\xi_i)(x_i - x_{i-1}) \left(\xi_i - \frac{x_{i-1} + x_i}{2} \right).$$

Inspection of the second derivative at the root reveals that the stationary point is a minimum and hence the result is proved. ■

C.7 Numerical results

In this section we illustrate the quadrature rule of section C.6 on the integral

$$\int_0^1 100t \ln(1/t) \cos(4\pi t)\,dt = -1.972189325199166 \tag{C.31}$$

The integral (C.31) is evaluated using three rules. They are:

n	Error (1)	Error (2)	Error (3)	Error ratio (3)	Bound ratio (3)
4	1.97(0)	2.38(0)	2.48(0)	–	–
8	3.41(-1)	2.93(-1)	2.35(-1)	10.56	3.90
16	8.63(-2)	5.68(-2)	2.62(-2)	8.97	3.95
32	2.37(-2)	1.31(-2)	4.34(-3)	6.04	3.97
64	6.58(-3)	3.20 (-3)	9.34(-4)	4.65	3.99
128	1.82(-3)	7.94(-4)	2.23(-4)	4.18	3.99
256	4.98(-4)	1.98(-4)	5.51(-5)	4.05	4.00

Table C.1: The error in evaluating (C.31) using different quadrature rules. The parameter n is the number of sample points.

(1) the composite mid-point rule, where the grid has a uniform step-size and the node is simply the mid-point of each sub-interval,

(2) the composite generalised mid-point rule (C.25). The grid, I_n, is uniform and the nodes are the mean point of each sub-interval (C.27),

(3) Equation (C.29) where the grid is distributed according to (C.28) and the nodes are the sub-interval means (C.27).

Table C.1 shows the numerical error of each method for increasing number of sample points.

For a uniform grid, we can see that changing the location of the sampling point from the midpoint [method (1)] to the mean point [method (2)] roughly doubles the accuracy. Changing the grid distribution as well as the node point [method (3)] from the composite mid-point rule [method (1)] increases the accuracy by approximately an order of magnitude. It is important to note that the nodes and weights for method (3) can be easily calculated numerically using an iterative scheme. For example on a Pentium-90 personal computer, with $n = 64$, calculating (C.27) and (C.28) took close to 37 seconds.

We remark that equations (C.27) and (C.28) are quite general in nature and only rely on the weight insofar as knowledge of the first two moments is required. This contrasts with Gaussian quadrature where for an n point rule, the first $2n$ moments are needed (or equivalently the $2n + 1$ coefficients of the continued fraction expansion (Rutishauser 1962a; Rutishauser 1962b)) to construct the appropriate orthogonal polynomial and then a root-finding procedure is called to find the abscissae (Atkinson 1989). We hasten to add that this procedure can be greatly simplified

for the more well known weight functions (Gautschi 1994).

The second last column of Table C.1 shows the ratio of the numerical errors for method (3) and the last column the ratio of the theoretical error bound (C.29)

$$\text{Bound ratio (3)} = \frac{|R(f, \boldsymbol{\xi}, n/2)|}{|R(f, \boldsymbol{\xi}, n)|}. \tag{C.32}$$

As n increases the numerical ratio approaches the theoretical one. The theoretical ratio is consistently close to 4. This value suggests an asymptotic form of the error bound

$$|R(f, \boldsymbol{\xi}, n)| \sim O\left(\frac{1}{n^2}\right)$$

for the log weight. This is consistent with mid-point type rules and we anticipate that developing other product rules, for example a generalised trapezoidal or Simpson's rule will yield more accurate results.

Appendix D

Complete Elliptic Integrals and some useful results

We define the complete elliptic integrals of the first, second and third kind to be

$$F(k) = \int_0^{\pi/2} \frac{d\theta}{\sqrt{1 - k\sin^2\theta}}, \tag{D.1}$$

$$E(k) = \int_0^{\pi/2} \sqrt{1 - k\sin^2\theta}\, d\theta, \tag{D.2}$$

$$\Pi(n,k) = \int_0^{\pi/2} \frac{d\theta}{(1 - n\sin^2\theta)\sqrt{1 - k\sin^2\theta}}, \tag{D.3}$$

respectively. The parameter k is known as the modulus and is restricted so that $0 \leq k \leq 1$. A useful result from (Abramowitz and Stegun 1965, equation 17.7.24) is

$$\Pi(k,k) = \frac{E(k)}{1-k}. \tag{D.4}$$

D.1 Axisymmetric integration

To derive the axi-symmetric integral equations used in chapters 4 and 5 one needs to evaluate integrals of the form

$$\int_0^{2\pi} \frac{\cos^n\theta}{(1 - k\cos^2\theta)^{m/2}}\, d\theta,$$

for $m = 1, 3, 5$ and $n = 0, 1, \ldots (m+1)/2$. These results are tabulated below.

With the aid of (D.1), (D.2), (D.3) and (D.4), the following results are easily proven

$$\int_0^{2\pi} \frac{d\theta}{\sqrt{1 - k\cos^2\theta/2}} = 4F(k), \tag{D.5}$$

$$\int_0^{2\pi} \frac{\cos\theta}{\sqrt{1 - k\cos^2\theta/2}}\, d\theta = \frac{4}{k}\Big((2-k)F(k) - 2E(k)\Big), \tag{D.6}$$

$$\int_0^{2\pi} \frac{d\theta}{\sqrt[3]{1 - k\cos^2\theta/2}} = \frac{4}{1-k}E(k), \tag{D.7}$$

$$\int_0^{2\pi} \frac{\cos\theta}{\sqrt[3]{1 - k\cos^2\theta/2}}\, d\theta = \frac{4}{k}\left(\frac{2-k}{1-k}E(k) - 2F(k)\right), \tag{D.8}$$

$$\int_0^{2\pi} \frac{\cos^2\theta}{\sqrt[3]{1 - k\cos^2\theta/2}}\, d\theta = \frac{4}{k^2}\left(\frac{k^2 - 8k + 8}{1-k}E(k) - 4(2-k)F(k)\right), \tag{D.9}$$

$$\int_0^{2\pi} \frac{d\theta}{\sqrt[5]{1 - k\cos^2\theta/2}} = \frac{4}{3(1-k)}\left(\frac{2}{1-k}(2-k)E(k) - F(k)\right), \tag{D.10}$$

$$\int_0^{2\pi} \frac{\cos\theta}{\sqrt[5]{1 - k\cos^2\theta/2}}\, d\theta = \frac{4}{3k(1-k)}\left(\frac{2}{1-k}\left(1 - k + k^2\right)E(k) - (2-k)F(k)\right), \tag{D.11}$$

$$\int_0^{2\pi} \frac{\cos^2\theta}{\sqrt[5]{1 - k\cos^2\theta/2}}\, d\theta = \frac{4}{3k^2(1-k)}\left(\left(8 - 8k - k^2\right)F(k) - \frac{2}{1-k}(2-k)\left(2 - 2k - k^2\right)E(k)\right), \tag{D.12}$$

$$\int_0^{2\pi} \frac{\cos^3\theta}{\sqrt[5]{1 - k\cos^2\theta/2}}\, d\theta = \frac{4}{3k^3(1-k)}\left(\frac{2}{1-k}\Big(k^4 + k^3 - 33k^2 + 64k - 32\Big)E(k) - (2-k)\left(k^2 + 32k - 32\right)F(k)\right). \tag{D.13}$$

D.2 Polynomial interpolation

Approximations to the complete elliptic integrals are required for evaluation of the integrals (2.18) and (2.19). There are two common methods, one is to employ the *arithmetic-geometric*

mean (AGM) (Abramowitz and Stegun 1965, p. 598) which produces a convergent sequence. The other is a polynomial approximation (Abramowitz and Stegun 1965, p. 591). Both methods are quite accurate, but are *not* interchangeable in their applicability to mathematical problems.

If elliptic integrals are to be used as integrands, then the polynomial approximation is the proper approximation to use. The reason is quite simple, a combination of the Gauss-Legendre and Gauss-Log rules can then be used. For some reason, this point has not been appreciated and previous methods to calculate the integrals have focused on the boundedness, and not the analytic nature of the integrands. Some earlier work report using Simpson's rule to evaluate integrals involving $E(k)$ (since this is bounded) and cutting out a small strip in the domain when the integrand is unbounded. This strip is then evaluated analytically using a small expansion (Youngren and Acrivos 1975; Lee and Leal 1982). This technique may have led to numerical difficulties reported in Lee and Leal (1982) and Leal and Lee (1982).

Even in more recent studies (Koh and Leal 1989; Stone and Leal 1989; Pozrikidis 1990b; Tjahjadi *et al.* 199 Pozrikidis 1997) a five or six point Gauss-Legendre rule is used with the expansion approach. We therefore illustrate the use of the polynomial approximation below and test its efficacy when compared to the AGM-expansion method. We will consider evaluation of the singular integral

$$\int_0^1 F(k)\,dk = 2. \tag{D.14}$$

AGM-Expansion

The integrand in (D.14) is unbounded at $k = 1$. Therefore we split the domain of integration into two parts

$$\int_0^1 F(k)\,dk = \int_0^{1-\epsilon} F(k)\,dk + \int_{1-\epsilon}^1 F(k)\,dk. \tag{D.15}$$

Substituting a linear series expansion for $F(k)$ about $k = 1$ (2.21) into (D.15) gives

$$\int_0^1 F(k)\,dk = \int_0^{1-\epsilon} F(k)\,dk + \frac{1}{16}\epsilon(8+\epsilon)\ln\left(\frac{16}{\epsilon}\right) + \frac{1}{32}\epsilon(16-3\epsilon) + O(\epsilon^3). \tag{D.16}$$

The integrand on the right-hand side is now analytic and so the integral may be evaluated using a Gauss-Legendre rule.

Table D.1 shows the computed right hand side of (D.16) using a 5 point Gauss-Legendre rule for various values of ϵ.

ϵ	$\int_0^1 F(k)\,dk$
0.2	2.36303
0.1	2.18806
0.05	2.09581
0.01	2.01587
0.001	1.99322
0.0001	1.99002
0.00001	1.98960

Table D.1: Evaluation of (D.16) using different values of ϵ and a 5-point Gauss-Legendre rule

As $\epsilon \to 0$, the results in Table D.1 steadily converge to 2. This is because the interval $(1-\epsilon, 1)$ is small, thus increasing the accuracy of the series expansion. The error starts to degenerate after $\epsilon = 0.001$, since the integral over $(0, 1-\epsilon)$ is becoming *more singular*. This illustrates the difficulty with series expansions, choosing the correct ϵ is crucial. These issues may be resolved by using the appropriate representation for F, one that takes account of its singular behaviour.

Polynomial approximation

The polynomial approximation for complete elliptic integrals has the form

$$F(1-k) = \sum_{i=0}^{n} a_i k^i - \sum_{i=0}^{n} b_i k^i \ln(k). \tag{D.17}$$

Abramowitz and Stegun (1965) lists values of $\{a_i, b_i\}$, with $n = 4$ that make the approximation accurate to order 2×10^{-8} for all values of k. This is quite suitable for single precision calculations.

Kucera and Briggs (1990) improved the approximation to order 10^{-13}. This was achieved by using a minimax technique coupled with a multiple precision Fortran package. The size of the polynomials increased to $n = 12$.

Here we present a simple method for obtaining polynomial approximations to the complete elliptic integrals with accuracy of order 10^{-16}. The size of the polynomials remains at $n = 12$.

Given an order n, there are $2n + 2$ coefficients to be calculated. One way of achieving this is to evaluate $F(k)$ at $2n + 2$ points and substituting into (D.17). The linear system thus formed may then be inverted to produce the desired coefficients. The evaluation points are split into two

```
restart;
# High precision!
Digits := 200;
n := 12;
#
# Set up the polynomial approximation
Ke := sum( a[i]*x^i ,i=0..n ) - ln(x) * sum(b[i]*x^i,i=0..n);
#
# Grid points
k1 := [i*.9/(n+1)$i=0..n];
k2 := [(1-.1^(i+1))$i=0..n];
#
# Linear System
Eqn1 := [ ( EllipticK(sqrt(k1[i+1]))=subs(x=1-k1[i+1],Ke) )$i=0..n ];
Eqn2 := [ ( EllipticK(sqrt(k2[i+1]))=subs(x=1-k2[i+1],Ke) )$i=0..n ];
#
# Solve
assign(fsolve({op(Eqn1),op(Eqn2)},{a[i]$i=0..n,b[i]$i=0..n},fulldigits));
```

Algorithm D.1: A Maple program to evaluate the polynomial approximation (D.17).

regions; $n+1$ distributed uniformly in $[0, 0.9]$ and $n+1$ distributed via an exponential clustering in $[0.9, 1]$. To clarify, the points k_i are

$$k_i = \frac{9i}{10(n+1)} \quad \text{and} \quad k_{i+n+1} = 1 - 10^{-i-1}, \qquad \text{for } i = 0, 1, \ldots, n.$$

The points were chosen in this way to ensure accuracy of the approximation near $k = 1$. A Maple program is given in Algorithm D.1 and the coefficients for $F(k)$ are listed in Table D.2. We report that very high accuracy ($\geq$ 120 digits) was required with the Maple routine, otherwise the solution returned was highly erratic. This would suggest that the linear system described above is ill-conditioned.

In Table D.3 we compare the polynomial approximation of Kucera and Briggs (1990) with that described here. We also list results for a simple Taylor polynomial approximation. It is clear that the polynomial approximation developed here is of sufficient accuracy for use with double precision calculations. When employed to calculate (D.14) using a 3-point Gauss-Legendre and 3-point Gauss-Log rule, the answer returned is 1.999827. With a 5-point Gauss-Legendre and 5-point Gauss-Log rule, the integral evaluates to 1.9999999602. Obviously, a 7-point rule is the optimal size for a polynomial of order 12, and when this is used the answer returned is 2.

i	a_i	b_i
0	0.1386294361119891D1	0.5000000000000000D0
1	0.9657359027997265D-1	0.1250000000000000D0
2	0.3088514453248462D-1	0.7031250000000000D-1
3	0.1493760036978099D-1	0.4882812500000000D-1
4	0.8766312198628351D-2	0.3738403320299965D-1
5	0.5754899916512118D-2	0.3028106526770420D-1
6	0.4068196489162360D-2	0.2544378896278751D-1
7	0.3167134481148402D-2	0.2189639358590440D-1
8	0.3859187350434518D-2	0.1859695172048566D-1
9	0.6972489272022876D-2	0.1326644642298081D-1
10	0.7000304984236619D-2	0.5721506651298451D-2
11	0.2355355762376631D-2	0.9874948865402975D-3
12	0.1617500382458659D-3	0.3519107157048046D-4

Table D.2: The coefficients of the polynomial approximation (D.17) using the Maple algorithm D.1.

k	Kucera & Briggs	Taylor expansion	Table D.2
0.00000000000	0.424E-15	0.283E-15	0.000E+00
0.10000000000	0.138E-15	0.138E-15	0.551E-15
0.20000000000	0.268E-15	0.268E-15	0.268E-15
0.30000000000	0.130E-15	0.000E+00	0.259E-15
0.40000000000	0.000E+00	0.125E-15	0.375E-15
0.50000000000	0.359E-15	0.120E-15	0.120E-15
0.60000000000	0.228E-15	0.114E-15	0.114E-15
0.70000000000	0.214E-15	0.000E+00	0.214E-15
0.80000000000	0.197E-15	0.393E-15	0.000E+00
0.90000000000	0.000E+00	0.293E-13	0.172E-15
0.99000000000	0.114E-13	0.532E-10	0.000E+00
0.99900000000	0.318E-12	0.484E-09	0.183E-15
0.99990000000	0.532E-12	0.111E-08	0.000E+00
0.99999000000	0.515E-12	0.164E-08	0.124E-15
0.99999900000	0.454E-12	0.204E-08	0.214E-15
0.99999990000	0.400E-12	0.234E-08	0.000E+00
0.99999999000	0.357E-12	0.258E-08	0.000E+00
0.99999999900	0.321E-12	0.277E-08	0.151E-15
0.99999999990	0.293E-12	0.293E-08	0.000E+00
0.99999999999	0.269E-12	0.306E-08	0.126E-15

Table D.3: The relative error for three polynomials expansions of $F(k)$. Column 2 uses the expansion in Kucera and Briggs (1990), column 3 is a simple Taylor expansion, column 4 shows the method described above.

Appendix E

Computing orthonormal polynomials using a computer algebra package

Define the inner product

$$\langle f, g \rangle_w := \int_a^b f(t)g(t)w(t)\,dt, \tag{E.1}$$

where w is some positive integrable function on [a,b]. That is, $w : [a, b] \to [0, \infty) \in L_1[a, b]$.

We seek the sequence of polynomials $\{\phi_i\}$, $i = 0, 1, \ldots$ that are orthonormal with respect to $\langle \cdot, \cdot \rangle_w$. Since these polynomials are linearly independent and form a basis for the inner product space, we can construct them using the well known *Gram-Schmidt* procedure (Atkinson 1989, p. 209). We can use the basis $\{1, x, x^2, \ldots\}$ as the generator.

Algorithm E.1 shows a Maple program that calculates a sequence of orthonormal polynomials to any order n given any arbitrary integrable weight w. Sample output, using the weight $w(t) = e^{-t}/\sqrt{t}$, $0 < t < \infty$ is presented in Figure E.1

Once the polynomials have been calculated, they can be used with orthogonal expansions, as in chapter 3, or for evaluating integrals with weight w to high precision (Stroud and Secrest 1966).

```
#
       Ortho := proc( n , w , x , a , b )
#
# This routine will calculate orthogonal polynomials
# up to order n with respect to the inner product
#
#     (f,g) = int( w(x)*f(x)*g(x) , x=a..b)
#
# using the Gram-Schmidt process.
#
          local v , i , j;
#
          v[1] := 1/sqrt( ip( 1,1,w,x,a,b ) );
          for i from 2 to n+1 do
              v[i] := x^(i-1) - sum( 'ip( x^(i-1),v[j],w,x,a,b )' * v[j] ,
                                     j=1..i-1 );
              v[i] := 1 / sqrt( ip( v[i],v[i],w,x,a,b ) ) * v[i]
          od; i := 'i';
#
          [ 'collect(expand(sort(v[i])),x)'$i=1..n+1 ]
      end;
#
#
      ip := proc( f1 , f2 , w , x , a , b )
#
#  The inner product with weight function w.
#  We can use 'evalf' if Maple is unable to integrate analytically.
#
          int( w * f1 * f2 , x=a..b )
      end;
```

Algorithm E.1: A Maple program to calculate the orthonormal polynomials for any degree given an arbitrary weight.

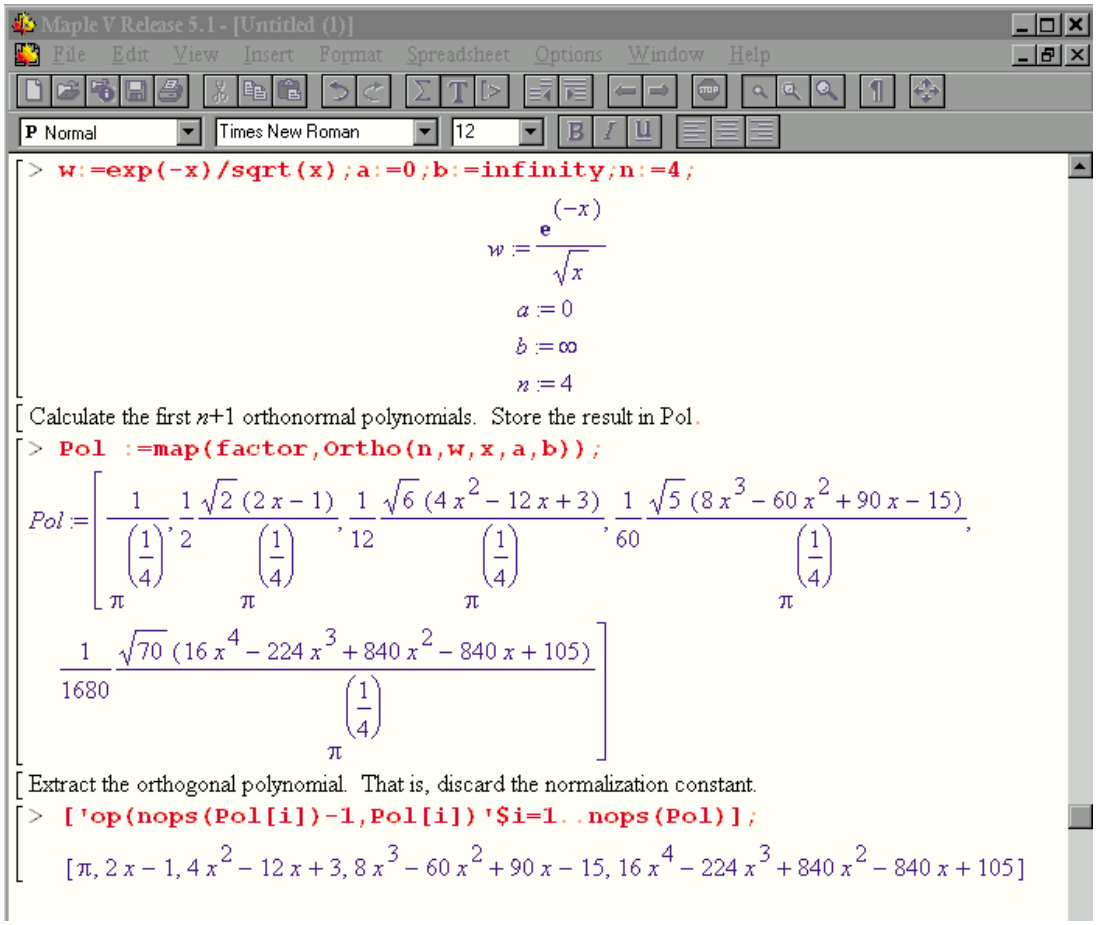


Figure E.1: Sample output from the Maple program E.1 showing the generation of the first five orthonormal polynomials using the weight $w(t) = e^{-t}/\sqrt{t}, 0 < t < \infty$.

Bibliography

Abramowitz, M. and I. A. Stegun (1965). *Handbook of Mathematical Functions*. New York: Dover.

Aderogba, K. and J. R. Blake (1978a). Action of a force near a planar surface between two semi-infinite immiscible liquids at very low Reynolds numbers. *Bull. Austral. Math. Soc.* **18**, 345–356.

Aderogba, K. and J. R. Blake (1978b). Action of a force near a planar surface between two semi-infinite immiscible liquids at very low Reynolds numbers. Addendum. *Bull. Austral. Math. Soc.* **19**, 309–318.

Arnold, D. N. and W. L. Wendland (1983). On the Asymptotic Convergence of Collocation Methods. *Math. Comput.* **41**(164), 349–381.

Arnold, D. N. and W. L. Wendland (1985). The Convergence of Spline Collocation for Strongly Elliptic Equations on Curves. *Numer. Math.* **1985**, 317–341.

Atkinson, K. E. (1989). *An Introduction to Numerical Analysis*. New York: John Wiley and Sons.

Barnett, N. S., P. Cerone, S. S. Dragomir and J. Roumeliotis (2000a). Some inequalities for the dispersion of a random variable whose pdf is defined on a finite interval. Accepted for publication in *J. Inequal. Pure Appl. Math.*.

Barnett, N. S., P. Cerone, S. S. Dragomir and J. Roumeliotis (2000b). Some inequalities for the expectation and variance of a random variable whose pdf is n-time differentiable. *J. Inequal. Pure Appl. Math.*, Article 11.

Barnett, N. S., I. S. Gomm and L. Armour (1995). Location of the optimal sampling point for the quality assessment of continuous streams. *Austral. J. Statist.* **37**(2), 145–152.

Barta, E. and N. Liron (1988). Slender body interactions for low Reynolds numbers. Part I: Body-wall interactions. *SIAM J. Appl. Math.* **48**(5), 992–1008.

Batchelor, G. K. (1967). *An Introduction to Fluid Dynamics*. London: Cambridge University Press.

Batchelor, G. K. (1970a). Slender-body theory for particles of arbitrary cross-section in Stokes flow. *J. Fluid Mech.* **44**, 419–440.

Batchelor, G. K. (1970b). The stress system in a suspension of force-free particles. *J. Fluid Mech.* **41**, 545–570.

Bettess, P. (1977). Infinite elements. *Int. J. Num. Meth. Eng.* **11**, 53–64.

Bettess, P. (1992). *Infinite Elements*. Sunderland, UK: Penshaw.

Blake, J. (1971a). A note on the image system for a Stokeslet in a no-slip boundary. *Proc. Camb. Phil. Soc.* **70**, 303–310.

Blake, J. (1972). A model for the micro-structure in ciliated organisms. *J. Fluid Mech.* **55**, 1–23.

Blake, J. R. (1971b). Infinite models for ciliary propulsion. *J. Fluid Mech.* **49**, 209–222.

Blake, J. R. (1974). Singularities of viscous flow. Part II: Applications to slender body theory. *J. Engng Math.* **8**(2), 113–124.

Blake, J. R. and A. T. Chwang (1974). Fundamental singularities of viscous flow. Part I: The image systems in the vicinity of a stationary no-slip boundary. *J. Engng Math.* **8**(1), 23–29.

Blake, J. R. and G. R. Fulford (1981). Force distribution on a slender body close to an interface. *Bull. Austral. Math. Soc.* **74**, 27–36.

Blake, J. R. and G. R. Fulford (1984). Mechanics of muco-ciliary transport. *PhysicoChemical Hydro.* **5**(5/6), 401–411.

Blake, J. R. and G. R. Fulford (1995). Hydrodynamics of filter feeding. *Soc. Exp. Biol.*, 183–197.

Blake, J. R. and D. C. Gibson (1987). Cavitation bubbles near boundaries. *Ann. Rev. Fluid Mech.* **19**, 99–123.

Blake, J. R., N. Liron and G. K. Aldis (1982). Flow patterns around micro-organisms and in ciliated ducts. *J. theor. Biol.* **98**, 127–141.

Blake, J. R., S. R. Otto and D. A. Blake (1998). Filter feeding, chaotic filtration, and a blinking Stokeslet. *Theor. Comp. Fluid Dyn.* **10**, 23–36.

Blake, J. R., P. Vann and H. Winet (1983). A model of ovum transport. *J. theor. Biol.* **102**, 145–166.

Blatt, J. M. (1975). A stable method of inverse interpolation. *Aus. Comp. J.* **7**(2), 51–57.

Brebbia, C. H. and J. Dominguez (1989). *Boundary elements: an introductory course.* Southampton: Computational Mechanics.

Brenner, H. (1961). The slow motion of a sphere through a viscous fluid towards a plane surface. *Chem. Eng. Sci.* **16**, 242–251.

Brenner, H. (1962). Effect of finite boundaries on the Stokes resistance of an arbitrary particle. *J. Fluid Mech.* **12**, 35–48.

Brenner, H. and J. Happel (1958). *J. Fluid Mech.* **4**, 195.

Brody, J. P. and P. Yager (1997). Diffusion based extraction in a microfabricated device. *Sensors & Actuators A* **58**(1), 13–18.

Brody, J. P., P. Yager, R. E. Goldstein and R. H. Austin (1996). Biotechnology at low Reynolds numbers. *Biophys. J.* **71**(6), 3430–3441.

Cannon, L. R. G. and K. B. Sewwll (1995). Craspedellinae baer, 1931 (Platyhelminthes: Temnocephalida) ectosymbionts from the branchial chamber of Australian crayfish (Crustacea: Parastacidae). *Memoirs of the Queensland Museum* **38**, 397–418.

Cerone, P. and S. S. Dragomir (1999). Three point quadrature rules involving, at most, a first derivative. Submitted to *SIAM J. Math. Anal.*.

Cerone, P., S. S. Dragomir and J. Roumeliotis (1998). An inequality of Ostrowski type for mappings whose second derivatives belong to $L_p(a, b)$ and applications. Accepted for publication in *The Journal of the Indian Mathematical Society*.

Cerone, P., S. S. Dragomir and J. Roumeliotis (1999d). An inequality of Ostrowski-Grüss type for twice differentiable mappings and applications in numerical integration. *KYUNGPOOK Math. J.* **39**(2), 333–341.

Cerone, P., S. S. Dragomir and J. Roumeliotis (1999a). An inequality of Ostrowski type for mappings whose second derivatives belong to $L_1(a,b)$ and applications. *Honam. Math. J.* **21**(1), 127–137.

Cerone, P., S. S. Dragomir and J. Roumeliotis (1999b). An inequality of Ostrowski type for mappings whose second derivatives are bounded with applications. *East Asian Math. J.* **15**(1), 1–9.

Cerone, P., S. S. Dragomir and J. Roumeliotis (1999c). Some Ostrowski type inequalities for n-time differentiable mappings and applications. *Demon. Math.* **32**(4), 697–712.

Cerone, P., J. Roumeliotis and G. Hanna (1999). On weighted three point quadrature rules. Accepted for publication in *Computational Techniques and Applications: CTAC99*.

Char, B. W., K. O. Geddes, G. H. Gonnet, B. L. Leong, M. B. Monagan and S. M. Watt (1991). *Maple V Library Reference Manual*. Heidelberg: Springer-Verlag.

Chi, B. K. and L. G. Leal (1989). A theoretical study of the motion of a viscous drop toward a fluid interface at low Reynolds number. *J. Fluid Mech.* **201**, 123–146.

Chwang, A. T. (1975). Hydromechanics of low-Reynolds-number flow. Part 3. Motion of a spheroidal particle in quadratic flows. *J. Fluid Mech.* **72**, 17–34.

Chwang, A. T. and T. Y. Wu (1971). A note on the helical movement of micro-organisms. *Proc. Roy. Soc. Lond. B* **178**, 327–346.

Chwang, A. T. and T. Y. Wu (1974). Hydromechanics of low-Reynolds-number flow. Part 1. Rotation of axisymmetric prolate bodies. *J. Fluid Mech.* **63**, 607–622.

Chwang, A. T. and T. Y. Wu (1975). Hydromechanics of low-Reynolds-number flow. Part 2. Singularity method for Stokes flows. *J. Fluid Mech.* **67**, 787–815.

Collatz, L. (1966). *The Numerical Treatment of Differential Equations*. Berlin: Springer-Verlag.

Cummins, J. M., S. K. Robson and G. W. Rouse (1988). The acrosome reaction in spermatozoa of the grey-headed flying fox (*Pteropus poliocephalus*: Chiroptera) exposes barbed subacrosomal material. *Gamete Research* **21**, 11–22.

Davis, P. J. (1963). *Interpolation and Approximation*. Massachusetts: Blaisdell Publishing Co.

de Mestre, N. J. (1974). *LOW-REYNOLDS-NUMBER FLOWS: The Application of Fundamental Singularities*. Ph. D. thesis, The University of New South Wales.

de Mestre, N. J. and W. B. Russel (1975). Low-Reynolds-number translation of a slender cylinder near a plane wall. *J. Engng Math.* **9**(2), 81–91.

DeBoor, C. (1979). *A practical guide to splines*. Springer–Verlag.

Delves, L. M. and J. L. Mohamed (1985). *Compuational methods for integral equations*. Cambridge University Press.

Dongarra, J. J., J. R. Bunch, C. B. Moler and G. W. Stewart (1979). *LINPACK User's Guide*. Philadelphia: SIAM.

Dorrepaal, J. M., M. E. O'Neill and K. B. Ranger (1984). Two-dimensional Stokes flows with cylinders and line singularities. *Mathematika* **31**, 65–75.

Dragomir, S. S., P. Cerone, N. Barnett and J. Roumeliotis (1999). An inequality of Ostrowski type for double integrals and applications for cubature formulae. Submitted to *Taiw. J. Math.*.

Dragomir, S. S., P. Cerone and J. Roumeliotis (2000). A new generalization of Ostrowski's integral inequality for mappings whose derivatives are bounded and applications in numerical integration and for special means. *Appl. Math. Lett.* **13**, 19–25.

Dragomir, S. S., P. Cerone, J. Roumeliotis and S. Wang (1999). A weighted version of Ostrowski inequality for mappings of Hölder type and applications in numerical analysis. *Bull. Math. Soc. Sc. Math. Roumanie* **90**, 301–314.

Edwardes, D. (1892). *Quart. J. Math.* **26**, 70.

Fenton, J. D. (1994). Exploiting boundary periodicity in the numerical solution of potential problems. Unpublished. Department of Mechanical Engineering. Monash University.

Forster, F. K., R. L. Bardell, M. A. Afromowitz, N. R. Sharma and A. Blanchard (1995). Design, fabrication and testing of fixed-valve micro-pumps. In D. C. Wiggert *et al.* (Eds.), *Proceedings of the ASME Fluids Enginerring Division*, Volume 234, pp. 39–44. ASME International Mechanical Engineering Conference and Exposition, San Fransisco, CA. Nov 12-17, 1995.

Fulford, G. R. (1984). *Slender body theory near an interface at very low Reynolds number*. Ph. D. thesis, The University of Wollongong.

Fulford, G. R. and J. R. Blake (1983). On the motion of a slender body near an interface between two immiscible liquids at very low Reynolds numbers. *J. Fluid Mech.* **127**, 203–217.

Fulford, G. R. and J. R. Blake (1986a). Force distribution along a slender body straddling an interface. *J. Austral. Math. Soc. Ser. B* **27**, 295–315.

Fulford, G. R. and J. R. Blake (1986b). Muco-ciliary Transport in the Lung. *J. theor. Biol.* **121**, 381–402.

Ganatos, P., S. Weinbaum and R. Pfeffer (1980). A strong interaction theory for the creeping motion of a sphere between plane parallel boundaries. Part 1. Perpendicular motion. *J. Fluid Mech.* **99**, 739–753.

Gautschi, W. (1994). Algorithm 726: ORTHPOL - a package of routines for generating orthogonal polynomials and Gauss-type quadrature rules. *ACM Trans. Math. Software* **20**, 21–62.

Gavze, E. (1990). A boundary integral equation solution of the Stokes flow due to the motion of an arbitrary body near a plane wall. *Int. J. Multiphase Flow* **16**, 529–543.

Geller, A. S., S. H. Lee and L. G. Leal (1986). The creeping motion of a spherical particle normal to a deformable interface. *J. Fluid Mech.* **169**, 27–69.

Gerdes, K. (2000). A review of infinite element methods for exterior Helmholtz problems. *J. Comp. Acoust.* **8**(1), 43–62.

Gontscharoff, V. L. (1954). *Theory of Interpolation and Approximation of Functions*. Moscow.

Gray, J. and G. J. Hancock (1955). The propulsion of sea-urchin spermatozoa. *J. Exp. Biol.* **32**, 802–814.

Haber, S., G. Hetsroni and A. Solan (1973). Low Reynolds number motion of two drops submerged in an unbounded arbitrary velocity field. *Intl. J. Multiphase Flow* **4**, 627–638.

Hancock, G. J. (1953). The self-propulsion of microscopic organisms through liquids. *Proc. Roy. Soc. A* **217**, 96–121.

Hanna, G., P. Cerone and J. Roumeliotis (1999). An Ostrowski type inequality in two dimensions using the three point rule. Accepted for publication in *Computational Techniques and Applications: CTAC99*.

Happel, J. and H. Brenner (1965). *Low Reynolds Number Hydrodynamics*. Englewood Cliffs, N.J.: Prentice-Hall.

Hasimoto, H. (1959). 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**, 317.

Hasimoto, H. and O. Sano (1980). Stokeslets and eddies in creeping flow. *Ann. Rev. Fluid Mech.* **12**, 335–363.

Huang, L. H. and A. T. Chwang (1986). Hydrodynamics of low-Reynolds-number flow. Part 6. Rotation of oblate bodies. *J. Eng. Math.* **20**, 307–322.

Ingber, M. S. and A. A. Mammoli (1999). A comparison of integral formulations for the analysis of low Reynolds number flows. *Eng. Anal. with Boundary Elements* **23**, 307–315.

Ingber, M. S. and L. A. Mondy (1993). Direct second kind boundary integral formulation for stokes flow problems. *Comp. Mech.* **11**, 11–27.

Jeffery, G. B. (1912). On a form of the solution of Laplace's equation suitable for problems relating to two spheres. *Proc. Roy. Soc. Lond. A* **87**, 109.

Karrila, S. J. and S. Kim (1989). Integral equations of the second kind for Stokes flow: direct solution for physical variables removal of inherent accuracy limitations. *Chem. Engng. Commun.* **82**, 123–161.

Katz, D. F., J. R. Blake and S. L. Paveri-Fontana (1975). On the movement of slender bodies near plane boundaries at low Reynolds number. *J. Fluid Mech.* **72**, 529–540.

Keller, J. B. and S. I. Rubinov (1976). Slender-body theory for slow viscous flow. *J. Fluid Mech.* **1976**, 705–714.

Kils, U. (1982, August). Swimming Behavior Swimming Performance and Energy Balance of Antarctic Krill: *Euphausia superba*. In S. Z. El-Sayed (Ed.), *BIOMASS Scientific Series No. 3*, College Station, Texas. SCAR, SCOR, IABO, ACMRR.

Kim, S. (1987). Stokes flow past three spheres: An analytic solution. *Phys. Fluids* **30**(8), 2309–2314.

Kim, S. and S. J. Karrila (1991). *Microhydrodynamics: Principles and Selected Applications*. Boston: Butterworth-Heinemann.

Koh, C. J. and L. G. Leal (1989). The stability of drop shapes for translation at zero Reynolds number through a quiescent fluid. *Phys. Fluids A* **1**(8), 1309–1313.

Kojima, M., E. J. Hinch and A. Acrivos (1984). The formation and expansion of a toroidal drop moving in a viscous fluid. *Phys. Fluids* **27**(1), 19–32.

Kubota, Y. (1973). An integral with basis and its application to trigonometric series. *Bull. Fac. Sci., Ibaraki Univ., Math.* **5**, 1–8.

Kucera, A. (1992). A boundary integral method applied to bubble cavitation. Department of Mathematics, Australian Defence Force Academy, Canberra 2600, Australia.

Kucera, A. and K. Briggs (1990). Accurate polynomial approximation for complete elliptic integrals using a multiple precision fortran package. Department of Mathematics, La Trobe University, Bundoora 3082, Australia.

Kynch, G. J. (1959). *J. Fluid Mech.* **5**, 193.

Ladyzhenskaya, O. A. (1963). *The Mathematical Theory of Viscous Incompressible Flow*. New York: Gordon and Breach.

Lamb, H. (1932). *Hydrodynamics* (6 ed.). Cambridge University Press.

Leal, L. G. (1980). Particle motions in a viscous fluid. *Ann. Rev. Fluid Mech.* **12**, 435–476.

Leal, L. G. and S. H. Lee (1982). Particle motion near a deformable fluid interface. *Adv. Colloid Int. Sci.* **17**, 61–81.

Lecoq, N., F. Feuillebois, N. Anthore, R. Anthore, F. Bostel and C. Petipas (1993). Precise measurement of particle-wall hydrodynamic interactions at low Reynolds number using laser interferometry. *Phys. Fluids A* **5**(1), 3–12.

Lee, S. H. and L. G. Leal (1980). Motion of a sphere in the presence of a plane interface. part 2. an exact solution in bipolar co-ordinates. *J. Fluid Mech.* **98**, 193–224.

Lee, S. H. and L. G. Leal (1982). Motion of a sphere in the presence of a deformable interface. II. A numerical study of the translation of a sphere normal to an interface. **87**(1), 81–106.

Lerner, L. and J. F. Harper (1991). Stokes flow past a pair of stagnant-cap bubbles. *J. Fluid Mech.* **232**, 167–190.

Liron, N. and E. Barta (1992). Motion of a rigid particle in Stokes flow: a new second-kind boundary-integral equation formulation. *J. Fluid Mech.* **238**, 579–598.

Liron, N. and J. R. Blake (1981). Existence of viscous eddies near boundaries. *J. Fluid Mech.* **107**, 109–129.

Liron, N. and S. Mochon (1976a). The discrete-cilia approach to propulsion of ciliated microorganisms. *J. Fluid Mech.* **75**, 593–607.

Liron, N. and S. Mochon (1976b). Stokes flow for a Stokeslet between two parallel flat plates. *J. Eng. Math.* **10**, 287–303.

Liron, N. and R. Shahar (1978). Stokes flow due to a Stokeslet in a pipe. *J. Fluid Mech.* **86**, 727–744.

Loewenberg, M. (1994). Axisymmetric unsteady Stokes flow past an oscillating finite-length cylinder. *J. Fluid Mech.* **265**, 265–288.

Lorentz, H. A. (1897). Ein allgemeiner Satz, die Bewegung einer reibenden Flüssigkeit betreffend, nebst einigen Andwendungen desselben (A general theorem concerning the motion of a viscous fluid and a few consequences derived from it). *Versl. Kon. Akad. Wetensch. Amsterdam* **5**, 168–175.

Lovalenti, P. M. and J. F. Brady (1993). The hydrodynamic force on a rigid particle undergoing arbitrary time-dependent motion at small Reynolds number. *J. Fluid Mech.* **256**, 561–605.

Lucas, S. K. (1993). *Some developments of the boundary integral equation method, and applications in free surface and elasticity problems*. Ph. D. thesis, The University of Sydney.

Manga, M. and H. A. Stone (1993). Buoyancy-driven interactions between deformable drops at low Reynolds numbers. *J. Fluid Mech.* **256**, 647–683.

McLean, W. and S. Prößdorf (1996). Boundary element collocation methods using splines with multiple knots. *Num. Math.* **74**, 419–451.

Meiners, J.-C. and S. R. Quake (1999). Direct measurement of hydrodynamic cross correlations between two particles in an external potential. *Phys. Rev. Lett.* **82**(10), 2211–2214.

Mitrinović, D. S., J. E. Pečarić and A. M. Fink (1994). *Inequalities for functions and their integrals and derivatives*. Dordrecht: Kluwer Academic.

Niessner, H. (1987). Significance of Kernel Singularities for the Numerical Solution of Fredholm Integral Equations. In C. A. Brebbia, W. L. Wendland, and G. Kuhn (Eds.), *Boundary Elements IX. Volume 1: Mathematical and Computational Aspects*, Berlin, pp. 213–227. Springer-Verlag.

Niessner, H. and M. Ribaut (1985). Condition of boundary integral equations arising from flow computations. *J. Comp. Appl. Math.* **12 & 13**, 491–503.

Oberbeck, H. A. (1876). *Crelles J.* **81**, 79.

Odqvist, F. K. G. (1930). Über die Randwertaufgaben der Hydrodynamik zäher Flüssigkeiten (On the boundary value problems in hydrodynamics of viscous fluids). *Math. Z.* **32**, 329–375.

Oseen, C. W. (1927). *Hydrodynamik*. Leipzig: Akademische Verlag.

Piessens, R., E. deDoncker-Kapenga, C. Uberhuber and D. Kahaner (1983). *Quadpack: a subroutine package for automatic integration*, Volume 1 of *Series in Computational Mathematics*. Berlin: Springer-Verlag.

Power, H. and B. F. de Power (1993). Second-kind integral equation formulation for the slow motion of a particle of arbitrary shape near a plane wall in a viscous fluid. *SIAM J. Appl. Math.* **53**, 60–70.

Power, H. and G. Miranda (1987). Second kind integral equation formulation of Stokes' flows past a particle of arbitrary shape. *SIAM J. Appl. Math.* **47**, 689–698.

Pozrikidis, C. (1988). The flow of a liquid film along a periodic wall. *J. Fluid Mech.* **188**, 275–300.

Pozrikidis, C. (1989). A study of linearized oscillatory flow past particles by the boundary-integral method. *J. Fluid Mech.* **202**, 17–41.

Pozrikidis, C. (1990a). The deformation of a liquid drop moving normal to a plane wall. *J. Fluid Mech.* **215**, 331–363.

Pozrikidis, C. (1990b). The instability of a moving viscous drop. *J. Fluid Mech.* **210**, 1–21.

Pozrikidis, C. (1992). *Boundary integral and singularity methods for linearized viscous flow*. Cambridge University Press.

Pozrikidis, C. (1997). Numerical studies of singularity formation at free surfaces and fluid interfaces in two-dimensional Stokes flow. *J. Fluid Mech.* **331**, 145–167.

Prößdorf, S. and A. Rathsfeld (1984). A spline collocation method for singular integral equations with piecewise continuous coefficients. *Integral Equations Oper. Theory* **7**, 536–560.

Prössdorf, S. and A. Rathsfeld (1987). On Quadrature Methods and Spline Approximation of Singular Integral Equations. In C. A. Brebbia, W. L. Wendland, and G. Kuhn (Eds.), *Boundary Elements IX. Volume 1: Mathematical and Computational Aspects*, Berlin, pp. 193–211. Springer-Verlag.

Prössdorf, S. and G. Schmidt (1981). A finite element collocation method for singular integral equations. *Math. Nachr.* **100**, 33–60.

Purcell, E. M. (1977). Life at low Reynolds number. *Am. J. Phys.* **45**, 3–11.

Rallison, J. M. and A. Acrivos (1978). A numerical study of the deformation and burst of a viscous drop in an extensional flow. *J. Fluid Mech.* **89**, 191–200.

Roumeliotis, J. (1991). An integral equation method using cubic splines. Department of Mathematics, Australian Defence Force Academy, Canberra 2600, Australia.

Roumeliotis, J., P. Cerone and S. S. Dragomir (1999). An Ostrowski type inequality for weighted mappings with bounded second derivatives. *J. KSIAM* **3**(2), 107–119.

Roumeliotis, J. and G. R. Fulford (2000). Droplet interactions in creeping flow. *Comp. and Fluids* **29**(4), 435–450.

Russel, W. B., E. J. Hinch, L. G. Leal and G. Tieffenbruck (1977). Rods falling near a vertical wall. *J. Fluid Mech.* **83**, 273–287.

Rutishauser, H. (1962a). Algorithm 125: WEIGHTCOEFF. *CACM* **5**(10), 510–511.

Rutishauser, H. (1962b). On a modification of the QD-algorithm with Graeffe-type convergence. In *Proceedings of the IFIPS Congress*, Munich.

Sano, O. and H. Hasimoto (1978). The effect of two plane walls on the motion of a small sphere in a viscous fluid. *J. Fluid Mech.* **87**, 673–694.

Saranen, J. and W. L. Wendland (1985). On the Asymptotic Convergence of Collocation Methods With Spline Functions of Even Degree. *Math. Comput.* **45**(171), 91–108.

Schmidt, G. (1983). On Spline Collocation for Singular Integral Equations. *Math. Nachr.* **111**, 177–196.

Schmidt, G. (1984). The Convergence of Galerkin and Collocation Methods with Splines for Pseudodifferential Equations on Closed Curves. *ZAA* **3**(4), 371–384.

Schmidt, G. (1985). On Spline Collocation Methods for Boundary Integral Equations in the Plane. *Math. Meth. in the Appl. Sci.* **7**, 74–89.

Schmidt, G. (1986). On ϵ-Collocation for Pseudodifferential Equations on a Closed Curve. *Math. Nachr.* **126**, 183–196.

Seybert, A. F., B. Soenarko, F. J. Rizzo and D. J. Shippy (1985). An advanced computational method for radiation and scattering of acoustic waves in three dimensions. *J. Acoust. Soc. Am.* **77**, 362–368.

Shail, R. (1983). On the slow motion of a solid submerged in a fluid with a surfactant surface film. *J. Engng. Math.* **17**, 239–256.

Shanks, D. (1955). Non-linear transformations of divergent and slowly convergent sequences. *J. Math. and Phys.* **34**, 1–42.

Sleigh, M. A., J. R. Blake and N. Liron (1988). State of Art: The Propulsion of Mucus by Cilia. *Am. Rev. respir. Dis.* **137**, 726–741.

Sonshine, R. M. and H. Brenner (1966). The Stokes translation of two or more particles along the axis of an infinitely long circular cylinder. *Appl. Sci. Res.* **16**, 425–454.

Steinberger, E. H., H. R. Pruppacher and M. Neiburger (1968). On the hydrodynamics of pairs of spheres falling along their line of centres in a viscous medium. *J. Fluid Mech.* **34**, 809–819.

Stewartson, K. and L. Howarth (1960). On the flow past a quarter infinite plate using Oseen's equations. *J. Fluid Mech.* **7**, 1–21.

Stimson, M. and G. B. Jeffery (1926). The motion of two spheres in a viscous fluid. *Proc. Roy. Soc. Lond. A* **111**, 110–116.

Stokes, G. G. (1851). On the effect of the internal friction of Fluids on the Motion of Pendulums. *Trans. Camb. Phil. Soc.* **9**, 8–10.

Stone, H. A. (1994). Dynamics of drop deformation and breakup in viscous fluids. *Ann. Rev. Fluid Mech.* **26**, 65–102.

Stone, H. A. and L. G. Leal (1989). Relaxation and breakup of an initially extended drop in an otherwise quiescent fluid. *J. Fluid Mech.* **198**, 399–427.

Stroud, A. H. and D. Secrest (1966). *Gaussian Quadrature Formulas*. Englewood Cliffs, N.J.: Prentice-Hall.

Taib, B. B. (1985). *Boundary integral methods applied to cavitation bubble dynamics*. Ph. D. thesis, The University of Wollongong.

Taylor, G. I. (1969). Motion of axisymmetric bodies in viscous fluids. In *Problems of Hydrodynamics amd Continuum Mechanics*, pp. 718–724. SIAM.

Tesla, N. (1920). Valvular conduit. *U.S. Patent No. 1,329,559*.

Tillett, J. P. K. (1970). Axial and transverse Stokes flow past slender axisymmetric bodies. *J. Fluid Mech.* **44**, 401–417.

Tjahjadi, M., H. A. Stone and J. M. Ottino (1992). Satellite and subsatellite formation in capillary breakup. *J. Fluid Mech.* **243**, 297–317.

Tuck, E. O. (1964). Some methods for flows past blunt slender bodies. *J. Fluid Mech.* **18**, 619–635.

Voronin, V. V. and V. A. Cecoho (1974). An interpolation method for solving an integral equation of the first kind with a logarithmic singularity. *Soviet Math. Dokl.* **15**, 949–952.

Wendland, W. L. (1981). On the asymptotic convergence of boundary integral methods. In C. A. Brebbia (Ed.), *Boundary Element Methods*, Berlin, pp. 412–430. Springer-Verlag.

Yang, S.-M. and L. G. Leal (1983). Particle motion in Stokes flow near a plane fluid-fluid interface. Part 1. Slender body in a quiescent fluid. *J. Fluid Mech.* **136**, 393–421.

Yang, S.-M. and L. G. Leal (1984). Particle motion in Stokes flow near a plane fluid-fluid interface. Part 2. Linear shear and axisymmetric straining flows. *J. Fluid Mech.* **149**, 275–304.

Youngren, G. K. and A. Acrivos (1975). Stokes flow past a particle of arbitrary shape: a numerical method of solution. *J. Fluid Mech.* **69**, 377–403.

Zinchenko, A. Z., M. A. Rother and R. H. Davis (1997). A novel boundary integral technique for viscous interaction of deformable drops. *Phys. Fluids A* **9**, 1493–1511.