Numerical Modelling of Flows Involving Submerged Bodies and Free Surfaces

Mathew Bernard Robert Topper

Doctor of Philosophy

The University of Edinburgh

2010

To my loving parents,

Joan and Michael

Abstract

Kinetic energy extraction devices for ocean and river flows are often located in the vicinity of the fluid free surface. This differs from wind turbines where the atmosphere may be considered to extend to infinity for the purposes of numerical modelling. As most kinetic energy extraction devices are based on lifting surfaces, a numerical model is sought which can model both lifting and free surface flows. One such model is the boundary element method which has been successfully applied to free surface problems and to lifting flows as well as the combined problem.

This study seeks to develop a high order boundary element method that is capable of modelling unsteady lifting and free surface flows in three dimensions. Although high order formulations of boundary element methods are common for free surface problems, providing improved accuracy and computational time, their usage for lifting flows is less frequent. This may be due to the hypersingular boundary integral equation (HBIE) which must be solved in order to find the velocity of the vortex wakes behind lifting surfaces.

In previous lifting flow studies using high order boundary element methods the wake velocities have been determined at the element centres and then interpolated to the collocation points. Not until the paper of Gray *et al.* (2004b) has a method been available for the direct solution of the HBIEs at the edges of three dimensional high order elements with C^0 continuous interfaces. The solution employs a technique known as the Galerkin boundary element method.

This study shows, for the first time, that the Galerkin boundary element method is applicable to the solution of the HBIE on the vortex wake of a lifting body. The application of the technique is then demonstrated as part of the numerical model developed herein. The model is based on the high order boundary element method developed by Xü (1992) for non-linear free surface flows. This formulation is extended to include steady uniform flow throughout the computational domain as well as the presence of lifting and non-lifting bodies. Several verification cases are implemented to test the accuracy of the model.

Acknowledgements

I have a great number of people to thank for their time and support during the production of this work. Foremost among them is my colleague and friend, Dr. David Forehand of the University of Edinburgh, who was also my original principal supervisor at the Robert Gordon University, and ghost supervisor following my move. I owe much to his intelligence, persistence and attention to detail. A huge debt of gratitude is also owed to Professor Ian Bryden who gave me the opportunity to start my research at the Robert Gordon University and then took the role of principal supervisor after our defection to the University of Edinburgh. Ian, thank you for your patience and your belief in my abilities. I would also like to thank my second supervisor, Professor David Ingram of the University of Edinburgh.

Many many thank yous are offered to Professor Leonard Gray of the Oak Ridge National Laboratory, Tennessee. If research is built on the shoulders of giants then Len is my giant. I am indebted to you for your time, good advice and, importantly, allowing me the use of your code, without which a lot more additional time would have been required.

I could not have completed this work without the love and support of my family. I'm sorry for being so distant. I have made a great many friends on this journey and I would like to thank them for taking it with me. Thank you to all my friends who I met in Aberdeen, particularly Paula Saikko, Greg Cook, Henry Jeffrey and Dr. Scott Couch and to all the members of the Centre for Research in Energy and the Environment at the Robert Gordon University. It was a fun 18 months. At the University of Edinburgh, I would like to thank all the members, past and present, of the Institute of Energy Systems. The character and potential of its members is exceptional. Finally, I cannot find the words to express my gratitude to Dr. Lucy Cradden, my colleague and partner since I moved to Edinburgh. There isn't enough honk in the world to repay my debt to you. Thank you and big love.

The PhD studentship was funded with an EPSRC doctoral training account administered by the SuperGen Marine Consortium (Phase 1).

The numerical method was coded using FORTRAN90 and compiled with the Intel FORTRAN Compiler and Intel Math Kernel Library (non-commercial).

The diagrams were prepared using gnuplot, Inkscape, Paraview (Kitware Inc.) and Python Matplotlib (John Hunter).

This document was prepared with $\LaTeX\,2_{\varepsilon}$.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

Mathew Bernard Robert Topper

Contents

Abstract **iii**

Acknowledgements **iv**

Declaration **vi**

Figures and Tables **xi**

1 Introduction **1**

1.1 Context for the Research 1

1.1.1 Tidal Energy as a Renewable Energy Resource 1

1.1.2 Research Aim 3

1.2 Review of Tidal Turbine Modelling 4

1.2.1 Wind Turbine Numerical Modelling 4

1.2.2 Boundary Element Methods for Marine Problems 6

1.2.3 Tidal Turbine Numerical Modelling 7

1.3 Review of the Key Literature 8

1.3.1 Historical Divergence of Applications 8

1.3.2 High Order Boundary Element Formulations 10

1.3.3 Hypersingular Boundary Integral Equations 13

1.4 Extended Outline of Thesis 17

1.4.1 Chapter 2: Mathematical Formulations 17

1.4.2 Chapter 3: Eulerian Stage Implementation 19

1.4.3 Chapter 4: Lagrangian Stage Implementation 20

1.4.4 Chapter 5 and chapter 6: Results 22

2 Mathematical Formulations 24

2.1 Introduction . . . 24

2.2 The Assumptions of Potential Flow . . . 25

2.2.1 The Effects of Scaling . . . 30

2.3 Physical Model . . . 32

2.3.1 Potential Flow . . . 32

2.3.2 Computational Domain . . . 33

2.4 Full Potential Formulation of Governing Equations . . . 34

2.4.1 Boundary Conditions . . . 34

2.4.2 Boundary Integral Equations . . . 37

2.4.3 Uniqueness of Solution and the Wake Strength . . . 43

2.4.4 The Numerical and Spatial Kutta Condition . . . 45

2.4.5 Initial Conditions for Mixed Eulerian-Lagrangian Time-Stepping . . 47

2.5 Perturbation Potential Formulation of the Governing Equations . . . 49

2.6 The Hypersingular Boundary Integral Equation . . . 52

2.7 The Method of Weighted Residuals . . . 55

2.7.1 The Collocation Method . . . 56

2.7.2 The Galerkin Method . . . 58

3 Eulerian Stage Implementation 59

3.1 Introduction . . . 59

3.2 Biquadratic Elements and Influence Functions . . . 61

3.3 Adaptive Numerical Integrations . . . 65

3.3.1 Evaluation of Far and Near Field Influences . . . 66

3.3.2 Evaluation of Self-Influences . . . 70

3.4 Linear Triangular Elements . . . 78

3.4.1 Adaptive Numerical Integration . . . 79

3.5 Linear System of Equations . . . 82

3.5.1 Wake-Free System . . . 83

3.5.2 System of Equations for Lifting Surfaces and the Kutta Condition . 85

3.6 Special Treatment of Boundaries . . . 87

3.6.1 Reflective Boundaries . . . 88

3.6.2 Double Nodes . . . 93

4 Lagrangian Stage Implementation 97
4.1 Introduction . . . 97
4.2 Numerical Differentiation . . . 98
4.3 Galerkin Gradient Method . . . 100
4.3.1 Coincident Integration . . . 106
4.3.2 Edge Adjacent Integration . . . 116
4.3.3 Application to a Vortex Sheet . . . 122
4.3.4 Validity at Sheet Edges . . . 125
4.4 Time-Stepping . . . 133

5 Results: Non-Lifting 136
5.1 Introduction . . . 136
5.2 Convergence Study of Steady Flow Past a Unit Sphere . . . 137
5.2.1 Generation of a Hemisphere . . . 137
5.2.2 Numerical Results . . . 141
5.3 Convergence Study of Steady Flow Within a Cube . . . 145
5.4 Evolution of Single and Counterpropagating Solitary Waves . . . 148
5.4.1 Accuracy Checks . . . 150
5.4.2 Evolution of a Single Soliton . . . 151
5.4.3 Counterpropagating Solitons . . . 156

6 Results: Lifting 165
6.1 Introduction . . . 165
6.2 Aerofoil Discretisation . . . 167
6.3 Wake Generation Algorithm . . . 171
6.3.1 Spatial Kutta Condition . . . 172
6.3.2 Off Wake Averaging . . . 173
6.3.3 Galerkin Gradient Method . . . 176
6.4 The CFL Condition for the Wake . . . 177
6.5 Wake Generation Without the Galerkin Gradient Method . . . 179
6.6 Wake Generation With the Galerkin Gradient Method . . . 189

7 Conclusions and Further Work 197

Appendices

A Review of the Kutta Condition 205
A.1 Kutta-Joukowski Hypothesis 205
A.2 The Kutta-Joukowski Theorem 206
A.3 The Kutta Condition 206
A.4 The Wake Shape and the Kutta Condition 209

B Derivatives of Shape Functions 212

C Derivation of Equation (3.38) 214

D Wall Boundary Correction 216

E Transformations of the Coincident Galerkin Integral 220
E.1 First Polar Transformation 220
E.2 Variable Substitution and Second Polar Transformation 223

F Reduction of Coincident Integrals to One Dimension 226
F.1 Integrals for Equation (4.21) 226
F.2 Integrals for Equation (4.20) 237

G Transformations for the Edge Adjacent Galerkin Integral 240
G.1 Application of the First Polar Transform 240
G.2 Application of the Second Polar Transform 242

H Collocation Method at the Wake Edges 244

Bibliography 248

Figures and Tables

Figures

2.1 The fluid domain under consideration. 33

2.2 The effect of circulation on the flow about an aerofoil. 36

2.3 The fluid domain under consideration (2D-slice). 37

2.4 Alternative orientations of the wake vortex sheet as it separates from the trailing edge of the aerofoil. 47

2.5 Limits on Wake 54

3.1 Nine node biquadratic panel and its image in parametric space. Adapted from Xü (1992, figure 4-1). 62

3.2 Normal vectors for identical panels with oppositely ordered perimeter node numberings. 63

3.3 Side lengths and minimum distance to field point for nine node panel. Recreated from Xü (1992, figure 4-2). 67

3.4 Orientation for greatest perspective angle for field points closest, (a) to a panel edge node and, (b), closest to a corner node. Recreated from Xü (1992, figure 4-3). 68

3.5 Panel subdivision for near-singular point, P. The shaded panels contain p_c, the nearest integration point to P. Adapted from Xü (1992, figure 4-4). 70

3.6 The degenerate triangle mapping from Cartesian to parametric (ρ_1, ρ_2) space. Recreated from Xü (1992, figure 4-5). 71

3.7 Triangularisations for nine-node elements for singular integration. 72

3.8 Subdivision of panels into triangles. 76

3.9 Included angle for a panel with large aspect ratio. 77

3.10 Linear triangular element. 79

3.11 Barycentric coordinates for an equilateral triangle. 80

3.12 Images of the original channel as a result of reflective boundaries for the bottom, left and right channel walls. 88

3.13 Domain and integration points for images generated by reflection in the bottom and right channel wall boundaries. 90

3.14 Distances to the integration point and its images for the channel in the (y, z) plane. 91

3.15 Double and triple nodes at the meeting of surfaces with differing normals. . . . 94

4.1 Galerkin integral types. 103

4.2 Polar coordinate transformation centred at (s,t). Adapted from Gray *et al.* (2004a), figure 1. 108

4.3 Polar coordinate transformation centred at $\tau = s, t = 0$. Adapted from Gray *et al.* (2004a), figure 2. 110

4.4 Integration domain for second polar transformation split between $s < 0$ and $s > 0$ and over the three triangles. Adapted from Gray *et al.* (2006). 113

4.5 Edge-adjacent integral and polar transformation. Adapted from Sutradhar *et al.* (2005). 116

4.6 Edge-adjacent integral, second polar transformation. Adapted from Sutradhar *et al.* (2005). 118

4.7 Wake limiting scheme with shared external point. 122

4.8 Opposite wake surfaces and points mapped onto a sphere. 126

4.9 Contributing panels for a point on one side of the wake. Σ_i are the red panels and Σ_{NZ} are both the red and striped panels. 127

4.10 Supporting panels approach the wake edge. 128

4.11 Simplistic illustrative geometry at the wake edge. 128

4.12 The impact of $[\phi]$ on the edge-adjacent integral for points close to edges. . . . 130

5.1 Exploded hemisphere discretised by ‘rings’ of quadrilateral panels. 138

5.2 Splitting the hemisphere into quadrants (a) and panelling each quadrant using rotated ‘diamond’ panels (b). 139

5.3 Diamond panel discretisation on the unit sphere. 139

5.4 Patch representation of the hemisphere. 140

5.5 Error in velocity potential for flow past a unit sphere. 142

5.6 Patterns of relative error for the three different panelling schemes. 143

5.7 Verification test for flow through a cube with one Dirichlet surface. 146

5.8 Average and maximum relative errors for steady flow through a channel containing a Dirichlet surface. 147

5.9 Wave profiles for soliton with wave height, $H' = 0.1$. 153

5.10 Wave profiles for soliton with wave height, $H' = 0.15$. 154

5.11 Wave profiles for soliton with wave height, $H' = 0.2$. 155

5.12 Volume conservation errors for solitons at 0.4 CFL number. 157

5.13 Total energy errors for solitons at 0.4 CFL number. 158

5.14 Collision of two counterpropagating solitons with height $H' = 0.1$. 161

5.15 Counterpropagating solitons with height $H' = 0.4$. 162

5.16 Time evolution of the amplitude of soliton in counterpropagating collision with height $H' = 0.4$. 164

5.17 3D representation of the final time-step for counterpropagating solitons with height $H' = 0.4$. 164

6.1 Curvature based node spacing for an aerofoil section. 169

6.2 Wing section with cap produced with the diamond panel discretisation scheme. 169

6.3 Direction and length of the Kutta strip. 172

6.4 Advancing the wake geometry for the first time step from $t = 0$ to $t = \Delta t$. . . 174

6.5 Position of points for the off-surface velocity averaging scheme applied at the wake edges. 175

6.6 Advancing the wake geometry for the second time step from $t = \Delta t$ to $t = 2\Delta t$. 177

6.7 Wake roll-up behind a lifting line. 180

6.8 Initial ϕ_n, geometry and Kutta strip. $\alpha = 5°$, $t = 0$ & $\Delta t = 0.02$. 181

6.9 Velocity induced without wake influence. $\alpha = 5°$, $t = 0.06$ & $\Delta t = 0.02$. . . . 181

6.10 Wake profile at $t = 1.1$ for $\alpha = 5°$, $C_\epsilon = 0.5$ and 5 spanwise panels. 184

6.11 Wake profile at $t = 1.1$ for $\alpha = 5°$, $C_\epsilon = 0.5$ and 10 spanwise panels. 184

6.12 Wake profile at $t = 1.1$ for $\alpha = 5°$, $C_\epsilon = 0.5$ and 20 spanwise panels. 185

6.13 Comparison of wakes at varying discretisation for slices in the x and y directions. 185

6.14 Wake profile at $t = 1.1$ for $\alpha = 5°$, $C_\epsilon = 0.2$ and 20 spanwise panels. 186

6.15 Wake profile at $t = 1.1$ for $\alpha = 5°$, $C_\epsilon = 0.1$ and 20 spanwise panels. 186

6.16 Comparison of wakes at varying C_ϵ for slices in the x and y directions. 187

6.17 Wake profile at $t = 1.1$ for $\alpha = 10°$, $C_\epsilon = 0.5$ and 20 spanwise panels. 187

6.18 Wake profile at $t = 1.1$ for $\alpha = 20°$, $C_\epsilon = 0.5$ and 20 spanwise panels. 188

6.19 Comparison of wakes at varying α for slices in the x and y directions. 188

6.20 Wake profile at $t = 1.1$ for $\alpha = 5°$, $C_\epsilon = 0.5$, 10 spanwise panels and zero ϵ at the free edges. 189

6.21 Comparison of wake profiles at varying C_ϵ for $\alpha = 5°$, viewed from positive z. . 191

6.22 Comparison of wakes at varying C_ϵ at $\alpha = 5°$ for slices in the x and y directions at $t = 0.3$. 191

6.23 Failure of wake at $t = 0.4$ for $\alpha = 5°$, $C_\epsilon = 0.5$ and 10 spanwise panels. 192

6.24 Comparison of wakes for varying C_ϵ at $\alpha = 10°$ for slices in the x and y-directions at $t = 0.3$. 192

6.25 Wake profile of the double edge node algorithm at $t = 0.8$ for $\alpha = 5°$, $C_\epsilon = 0.5$ and 10 spanwise panels. 194

6.26 Comparison of wakes slices in the x and y directions with double edge node algorithm and the wake averaging algorithm at $t = 0.8$. 194

6.27 Wake profile of the double edge node algorithm at $t = 0.4$ for $\alpha = 10°$, $C_\epsilon = 0.5$ and 10 spanwise panels. 196

6.28 Wake profile of the double edge node algorithm at $t = 0.6$ for $\alpha = 10°$, $C_\epsilon = 0.5$ and 10 spanwise panels. The nodes adjacent to the line of wake nodes furthest from the trailing edge are damped. 196

C.1 Triangle. 214

D.1 Velocity vectors tangent to the free surface in the parametric directions. 217

D.2 Five node differentiation scheme for a free surface point coincident to the channel wall in the s and t parametric directions. 219

F.1 $f(\Psi) = \cos(\Psi)\left(1 + 3^{-1/2}\tan\Psi\right)^3 \quad \forall \quad 0 < \Psi < \pi/3$. 228

F.2 $f(\Psi) = \cos^2(\Psi)\left(1 + 3^{-1/2}\tan\Psi\right)^3 \quad \forall \quad 0 < \Psi < \pi/3$. 230

F.3 $f(\Psi) = \left(1 + 3^{-1/2}\tan(\Psi)\right)^3 \quad \forall \quad 0 < \Psi < \pi/3$. 232

F.4 $f(\Psi) = \left(3^{-1/2}\tan(\Psi) - 1\right)^3 \quad \forall \quad 2\pi/3 < \Psi < \pi$. 234

F.5 $f(\Psi) = \left(3^{-1/2}\tan(\Psi) - 1\right)^3 \quad \forall \quad 2\pi/3 < \Psi < \pi$. 236

F.6 $f(\Psi) = \left(3^{-1/2}\tan(\Psi) - 1\right)^3 \quad \forall \quad 2\pi/3 < \Psi < \pi$. 237

H.1 Domain for collocation method in Cartesian and parameterised space. 245

Tables

5.1 Order of convergence of the present method for flow past a unit sphere. 144

Chapter 1

Introduction

1.1 Context for the Research

1.1.1 Tidal Energy as a Renewable Energy Resource

In recent times the way in which the human race uses and generates its energy has become of extreme social, economical and environmental importance. Predictions of damaging increases in mean global temperatures (see Solomon *et al.*, 2007) have increased the need for carbon emitting energy technologies to be replaced by low carbon alternatives. Additional long term economic factors are also playing their part in shifting momentum to new technologies, in particular the concerns about oil and natural gas supplies. Demand for oil and gas is expected to outstrip supply within the current century, leading to inflated prices and energy security issues. Unfortunately, although the resource and environmental issues are occurring simultaneously, they are not necessarily mutually supportive. For instance, once the price for oil has reached a certain value it can be economically synthesised from coal, providing no environmental benefits. In fact, if the market were left to choose a method for replacing dwindling oil resource then this is one of the most likely substitutes; even in light of population growth, coal reserves are estimated to last for hundreds of years (see Jaccard, 2005) and thus the cost is low.

With markets failing to deliver the necessary changes, it has become the responsibility of governments to intervene under the premise that the predicted environment and economical consequences of inaction will outweigh the costs incurred by immediate action. This is a tough political task as, particularly in the United Kingdom (UK), energy markets have followed a trend of liberalised trading which makes strategic decision making extremely challenging. Private investment is unlikely to match policy, unless the policy is seen to be long term and economically advantageous; a virtual impossibility when scientific and economic opinions of the market requirements are so uncertain. In addition, the likelihood of direct governmental intervention in the form of capital investment is now looking more unlikely, due to the recent financial crisis. Capital spending is set to fall significantly in order to reduce deficits endured to rescue the banking sector.

Despite the potential funding problems, the UK Government's Climate Change Act 2008 (see DEFRA, 2008) introduced for the first time legally binding targets for greenhouse gas emissions from within the UK. The targets set an 80% reduction of greenhouses gases by 2050 and a 26% reduction of carbon-dioxide (CO_2) emissions by 2020 with respect to 1990 levels. This is set in the context of Britain's pre-existing target to reduce CO_2 emissions by 12% as a ratified signatory of the Kyoto Protocol and a European Union proposal to cut EU wide greenhouse gas emissions by 20% by 2020 each with respect to 1990 levels. Electricity generation in the United Kingdom accounts for 37% of all CO_2 emissions and as part of the government's greenhouse gas reduction strategy it will seek to reduce this to zero by 2030 (see CCC, 2008). Given the enormity of this task it is foreseen that a varied mix of low/zero carbon electricity generating technologies will be required.

As such, a melting pot of renewable and sustainable energy technologies have begun to compete to become part of a 'post-carbon' energy mix. Some of these technologies are well established, such a nuclear, biomass and wind, others less so, such as solar photovoltaic and marine energy. In addition, carbon emitting technologies such as coal remain an attractive option for governments and investors alike, as carbon capture and storage technology promises to store away the greenhouse gases emitted in combustion. However, carbon capture and storage is still an unproven technology and thus, the future costs of this and many other of the new energy technologies are hard to predict and are, at present, highly contestable; ultimately, over the long term, energy cost is the most likely deciding factor as to which technologies will feature most. Hence, each industry is working hard to reduce costs via research and innovation.

Academic research is particularly important in the marine energy industry as the costs and risks of full scale testing at sea are extremely high. For both wave and tidal technologies, developers and consultants work closely with academic institutions such as The University of Edinburgh for guidance as to the state of the art in both resource and technology. It is hoped that numerical and physical experiments away from the sea will allow device developers to avoid costly mistakes that may retard confidence in the viability of this fledgling industry. Tidal current technology is particularly vulnerable to unforeseen eventualities at sea, as the costs of deployment and recovery are so high. As such, the academic community has been tasked with identifying and quantifying the risks faced when generating electricity by tidal currents.

The premise for generating power from tidal currents is the extraction of kinetic energy from the flows of water generated as the tidal bulges circumnavigate the globe. These tidal bulges are primarily driven by the combined rotation of the earth-moon system about their combined centre of mass. The moon's gravity draws one bulge of water towards it, while inertial effects produce a second bulge on the far side of the earth. Subsequently, as the earth rotates about its own axis faster than the earth and moon together, the tidal bulges move relative to the Earth's surface, producing the two pairs of high and low tides seen daily[1]. Although, in general, these flows of water are very slow, they can be significantly faster in regions of favourable geography such as the Pentland Firth located between the north east coast of Scotland and the Orkney Islands. Here tidal currents can exceed $8m/s$ in places. The Sustainable Development Commission (see Craig *et al.*, 2007) estimated that the annual power output from the Pentland Firth would be 636GWh/year from a 200MW capacity array of tidal energy converters. To put this in context, the United Kingdom's total electricity supplied in 2008 was 400TWh (see DUKES, 2009).

1.1.2 Research Aim

Interest in fully submerged horizontal axis, kinetic tidal energy extraction devices has become significant during the last three decades. These devices are, in part, analogous to a wind turbine, however, the tidal turbine will operate in a medium which is 800 times more dense, has the ability to cavitate (the sudden formation and collapse of low-pressure bubbles in liquids by means of mechanical forces, such as those resulting from rotation of a marine propeller) and is bounded by both a free surface and a

[1] In addition, the gravitational pull of the sun creates extreme high and low tides, known as the 'spring/neap cycle', which repeat every two weeks.

bottom boundary. In fact, the free surface is the principal driver for tidal flows at local geographic scale, the 'head' (or elevation) providing the potential energy to drive the current. These factors leads to a number of new research questions that may already be well understood for wind turbines. These include quantifying the maximum extractable energy by a single device (known as the Betz limit in wind turbine problems) and the role of the free surface on both the device and the wake of the device. At even greater resolution, surface waves and marine turbulence could contribute to reducing the performance and efficiency of these machines. Furthermore, the interaction of multiple devices or multiple rotors will further complicate the physics. It is expected that the outcomes of research into these issues will have a significant impact on the design of tidal turbines and the positioning of individual devices and farms.

The overarching goal of this study was to develop a means by which the free surface interaction with a horizontal axis tidal turbine at device scale could be qualitatively and quantitatively understood. Experimental testing of tidal energy devices is highly demanding, particularly scaling for both Reynolds and Froude numbers. In addition, the taking of fluid measurements in the near wake of a turbine is extremely challenging. Thus, a numerical approach was sought which could both accurately define the evolution of a free surface in time and model lifting bodies internal to the fluid.

1.2 Review of Tidal Turbine Modelling

1.2.1 Wind Turbine Numerical Modelling

At the onset of this project no numerical models for the near wake of tidal turbines yet existed. Thus, given the obvious analogies, wind turbine numerical modelling was the first resource for inspiration. Analysis, application and development of existing wind turbine models, to and for the marine environment, presented a clear research approach. Wind turbine research is 40 years old and the models designed for wind energy conversion are well understood and offer many potential solutions to tidal turbine problems. There are three main types of wind turbine model. These are 'Actuator Disk Models', 'Vortex Wake Models' and 'Navier-Stokes Models'.

Actuator disk models simulate a wind turbine by representing it as a permeable surface, normal to the wind direction, in which the blade forces are distributed evenly across the area of the disc. This method is used in the most basic, and most common, type of wind turbine modelling known as blade element/momentum theory. This theory uses empirical performance data for a known blade design together with a theoretical turbine with an infinite number of blades which is equivalent to the actuator disk. A basic example of an actuator disk model can be found in Manwell *et al.* (2002). Actuator disk models are often embedded within models that simulate the Navier-Stokes equations. The majority of these models have quasi-two-dimensional (2D) geometry combining 2D aerofoil data. A three dimensional (3D) actuator disk model has been developed by Sørensen and Shen (2002), which has the advantage of allowing the forces to be radially distributed along the blade lengths within a full 3D viscous Navier-Stokes solver. Nevertheless, aerofoil data is still required to accurately simulate the loading on the blades.

Vortex wake models use lifting lines or surfaces to represent the rotor blades and the shed vorticity in the wake of the turbine. At the blade's trailing edge the vortex strength is determined from the circulation generated about the blade by the incoming flow. Variations in this circulation generates vortex filaments along the spanwise direction of the blade which in turn are shed into the wake as the blade rotates over time. From the modelled blade and resultant vortex sheets, the flow field can be generated, combining the undisturbed flow and the induced velocity field. In three dimensions, 'boundary element[2]' or 'panel method' approaches allow the geometry of a turbine blade to be taken into account. Many examples can be found of vortex methods used in wind energy research such as those presented by Afjeh and Keith (1986), Simoes and Graham (1992) and Whale *et al.* (2000a). Vortex wake models have also been used to correct blade element momentum models (see Whale *et al.*, 2000b). Even the less well known Darrieus vertical axis turbine has been modelled using a vortex method by Ponta and Jacovkis (2001). Generally these method are inviscid and irrotational apart from the vortex wake, but additional viscous effects can be included by coupling a boundary layer solver in the region near to the blades. Overall, the primary advantage of vortex methods for lifting flows is their ability to track the evolution of the vortex wake in a Lagrangian manner. Eulerian grid based methods must find some alternative means

[2] The terms boundary element and boundary integral are often interchanged. Strictly, the boundary integral is the governing equation whilst the boundary element is the discretised version which is solved numerically.

of tracking the evolution of the wake, to ensure that the grid resolution is sufficient at areas of high vorticity.

The final model type is known as Navier-Stokes. The Navier-Stokes equations (see Katz and Plotkin, 2001, ch. 1) represent the most complete formulation for the physics of a fluid flow. Research into the aerodynamics of helicopters and fixed wing aircraft has led to the development of 'Computational Fluid Dynamics' (CFD) algorithms for wind turbines. The primary advances have been in the areas of grid generation and turbulence. Wind turbines do differ in operation from many typical aeronautical problems, as wind turbines operate at a much lower speed and often in stall conditions where the flow is separated from the blades. However, these methods have been shown to model wind turbines with great accuracy but they have the disadvantage of being extremely computationally expensive. Examples of such methods can be found in Duque *et al.* (2003), Sørensen and Shen (2002) and Xu and Sankar (2000).

1.2.2 Boundary Element Methods for Marine Problems

A close, but sometimes overlooked, relative to the marine turbine are marine propellers and hydrofoils. There has been a great deal of research into free surface and cavitation effects on propellers and hydrofoils using panel methods or boundary element methods. Professor Kinnas and colleagues have a long history of publications investigating marine propellers, numerical wave tanks and cavitating hydrofoils in the presence of a free surface, all based on boundary element methods (see Bal and Kinnas, 2002, 2003; Choi and Kinnas, 2003; Kinnas and Fine, 1993; Kinnas and Hsin, 1992; Lee and Kinnas, 2004, 2005a,b; Pyo and Kinnas, 1997; Young and Kinnas, 2003, 2004). In addition, Politis (2004) produced a free wake panel method for a marine propeller using a boundary element formulation. Zhu *et al.* (2006) have also developed a combined hydrofoil and free surface simulation using a boundary element method.

Boundary element methods also have a long and successful history in non-linear water wave modelling. Since the pioneering work of Longuet-Higgins and Cokelet (1976), numerous non-linear water wave simulations have been carried out in two and three dimensions (see Grilli *et al.* (1989, 2000) and Xue *et al.* (2001) for examples). Furthermore, numerical wave tanks encompassing wave generation by moving boundaries have been developed using boundary element methods, as was investigated by Forehand (1998).

The available literature clearly indicates there is an obvious, and previously exploited, union between the boundary element methods used to model aerofoils, wind turbines, propellers and hydrofoils and those used to model non-linear free surfaces and waves. As such, a boundary element provides a convenient solution to coupling the time dependant motion of a free surface and the wake of a submerged lifting body.

1.2.3 Tidal Turbine Numerical Modelling

Given the infancy of kinetic tidal energy extraction technology, it is no surprise that there are, as yet, no definitive numerical models that can successfully model the complete flow about a tidal turbine at device scale, including the interaction with a dynamic free surface. Over recent years, however, a small body of research has begun to formulate the building blocks for such models. Li and Çalişa (2010) have developed a hybrid viscid-inviscid discrete vortex method to predict the wake structure and acoustic intensities of a marine turbine. This model was successfully verified, even though no free surface was taken into account. Moreover, Baltazar and Falcão de Campos (2008) have applied a low order panel method to the problem, although this model once again contains no free surface. Furthermore, Navier-Stokes formulations have also been used to model tidal turbines. One such example is the vorticity transport equation formulation used by McCombes *et al.* (2009). Another is the commercial Computational Fluid Dynamics (CFD) approach taken by Gretton *et al.* (2009). These models can potentially resolve more fluid physics than the previous two (such as viscous effects), however, neither of these approaches is yet capable of resolving the dynamic wake and the free surface in conjunction. As such, the need remains for the development of a model capable of resolving both turbine wake and free surface.

As an alternative approach to the tidal turbine modelling problem, which naturally incorporates a robust free surface model, a quasi-potential lifting boundary element code was adopted within this thesis. The remaining sections of this chapter detail the reasoning behind this decision and introduce the novel theoretical additions to this mature modelling technique.

1.3 Review of the Key Literature

1.3.1 Historical Divergence of Applications

What follows is an in-depth discussion of the literature that was key to the development of the present thesis. This begins with a historical review of boundary element methods and their development in three particular fields. The development of these three distinct divisions is critical to the work within this thesis. Each branch has been built upon the fundamental basis of potential theory laid out by Kellogg (1929), nonetheless it would take another thirty years before numerical methods based on potential theory would be developed for each application. This delay can be attributed to the advancement of computer science; it wasn't until 1936 that the Church-Turing thesis laid out the basis for modern computing algorithms. By 1960 transistor based computing machines were commonplace for large industry and post-1960 would see transition to solid state transistor chips, which are still in use today. The availability of this computing power would lead scientists and engineers to develop numerical methods using potential theory for fracture mechanics, lifting and non-lifting flows, and water waves amongst others. Surprisingly, each formulation, although related through Kellogg, was developed nearly simultaneously and independently of the others.

The first potential flow based numerical methods relevant to the present project is attributed to the Douglas Aircraft Company (later to become the McDonnell Douglas Corporation and eventually to merge with Boeing). The first two dimensional numerical method for non-lifting flows about arbitrary bodies was developed by Smith and Pierce (1958). Hess and Smith (1962) would develop the three dimensional extension shortly afterwards. In 1972 the Douglas Aircraft Company model would be extended to three dimensional lifting flows by Hess (1972), by which time the important concepts of the vortex wake and numerical Kutta condition had become well understood in academia (see Djojodihardjo and Widnall, 1969). The vortex wake would be the key to allowing the viscous phenomenon of vortex shedding from the trailing edge of an aerofoil to be included in potential flow models; the Kutta condition (a consequence of the famous Kutta-Joukowski theorem) would govern the strength of the wake (see section 2.4 for more detail). The accumulation of this knowledge would become known as the *panel method*. For a more comprehensive historical background to the panel method see Hess

(1990).

Due to advances in aircraft technology and increased computing power, the use of panel methods in the aeronautical industry has been superseded by computational fluid dynamics. However, panel methods are still commonly used for other low-speed aeronautical or hydrodynamical problems such as wind turbines and propellers (see section 1.2.1). The decline of the use of panel methods in the aeronautical industry may well be an indicator as to why the theory of the method has not advanced greatly in recent years; a fact that is somewhat exploited in the present thesis.

Around the same time that Smith and Pierce began to formulate the use of boundary elements for potentials flows, Jaswon (1963) and Symm (1963) laid down the foundations for the use of potential theory based numerical methods in fracture mechanics. Over time, boundary element methods in fracture mechanics have become a highly distinct and advanced subset of the applications of potential theory. In the author's opinion it is currently the most advanced theoretically (see eg. Bonnet, 1999) and has made underlying advances in boundary element methods where, say, panel methods have not. In addition, the advances in fracture mechanics are not often transferred into the realm of other boundary element based methods. This is, in part, due to the different notation and formulations of fracture mechanics problems. Moreover, some - perhaps historical - unawareness of the other applications is also apparent. In particular, exciting advances in the solution to the hypersingular boundary integral equation for high order discretisations have been made by authors such as Guiggiani (1998) and Gray *et al.* (2004b). The solution to the hypersingular boundary integral equation is critical to determining the velocity on a vortex wake, which can then be used to advance the wake in time. Thus, such advances in their solution are ideally suited to application to lifting flows. Indeed, the opportunity to apply the work of Gray *et al.* (2004b) to lifting flows is taken for the first time herein. For a more detailed history of boundary integrals/elements in fracture mechanics see Cheng and Cheng (2005).

The final application of numerical methods based on potential flow impacting on the current work, is non-linear free surface water waves. The seminal paper of Longuet-Higgins and Cokelet (1976) showed that boundary element methods could be successfully applied to such problems. Although Longuet-Higgins and Cokelet make no reference to previous authors (not even Kellogg!), the possibility of application of panel methods to free surface problems was discussed in Hess (1972) and Giesing and Smith

(1967), although that paper used a linearised free surface boundary condition. The method of Longuet-Higgins and Cokelet forms the basis of many advanced boundary element methods for free surface problems such as those developed by Grilli *et al.* (2001) and Xue *et al.* (2001). The emphasis for boundary elements in free surface problems has been the accurate time evolution of waves governed by the non-linear free surface boundary conditions. To achieve this, methods using high order discretisations have been developed in order to accurately calculate free surface velocities, in addition to explicit time stepping schemes which allow the use of dynamic time steps at low computational cost (see Grilli *et al.*, 2001).

The separate development of the aforementioned fields is reunified within the present method. The requirement of a non-linear free surface is provided by a method based on the work of Longuet-Higgins and Cokelet and in particular, the high order 3D scheme developed by Xü (1992). Including submerged lifting surfaces requires an application of the work developed by Hess and colleagues. Amalgamations of free surface and lifting body have already been developed, such as in the work of Zhu *et al.* (2006). Frequently, however, such methods rely on low order discretisation. To enhance boundary element methods for lifting bodies near free surfaces a high order discretisation should be used throughout. This is achieved through the addition of work from fracture mechanics, and, in particular, the work of Gray *et al.* (2004b). A similar application of the work of Guiggiani (1998) has been applied to high order potential lifting flows by Bernasconi and Richelsen (2008). The apparent need for the use of higher order discretisations is discussed in the next section.

1.3.2 High Order Boundary Element Formulations

Before discussing the choice of order for the present boundary element method, a short explanation of the term ‘order’ is merited. *Order* in terms of boundary element methods, refers to the geometrical approximation to the surfaces bounding the volume of interest and the approximation of the field values on those surfaces, i.e. the velocity potential and it’s normal derivative. It is then clear that a piecewise constant method (the most common order for panel methods) is not a satisfactory approximation; the surface will have gaps and the potential will have discontinuities between elements. A higher order boundary element method is one which has linear order or higher. Beyond linear methods the advantages of higher order are accuracy and efficiency.

Less computational nodes are required, as order increases, to achieve the same level of accuracy, although methods beyond quadratic order are extremely rare.

As alluded to in the previous section, the reasons for striving for higher order methods are well documented, particularly in the development of boundary element methods for free surface flows. Xü (1992) carried out a detailed study into the feasibility of different element types, and concluded that a bi-quadratic element was required for sufficient accuracy for the free surface solver developed. Grilli *et al.* (2001) found that even higher order approximations were necessary to ensure stable simulations for extremely non-linear three dimensional water waves. Thus, the case for high order methods in free surface flows is clear.

The advantages of higher order discretisations for lifting flows have also been recognised previously. Pyo and Kinnas (1997) stated that,

> "It is a known fact that the use of a high-order panel method is more accurate than the low-order panel method or the discrete vortex method in computing the velocity field, since it precludes the appearance of instabilities in the vortex sheet due to the spurious numerical effects introduced by a too-crude representation."

Indeed, even as early as 1975, Hess (1975) realised that the use of high order approximations could improve the results for lifting flows. However, in the use of high order discretisations for lifting flows, comes a new difficulty, arising from the need to calculate the velocity on the wake surface. For free surface problems, this is not a difficulty as the potential on the free surface can be numerically differentiated to give the velocity. A vortex wake is different because only the difference in potential across the wake is known which does not provide the velocity when differentiated. The only way to calculate velocity is to calculate the direct gradient of the governing boundary integral equations. This leads to a set of equations known as the hypersingular boundary integral equations. Hypersingular refers to the order of singularity in these equations, being $\mathbf{O}(r^{-3})$ as $r \to \infty$, with r representing distance. This is one order higher than standard boundary integral equations which are strongly singular, i.e. $\mathbf{O}(r^{-2})$. The difficulty with hypersingular boundary integral equations is that their solution at the edges of discrete elements is difficult to obtain accurately. This did not hinder the use of constant panel methods as the collocation (computational) nodes are located at the centre of panels; however, high order methods are collocated at the panel edges and corners. Thus, as in Pyo and Kinnas (1997), previous high order methods have calculated the hypersingular boundary integral equation internal to the element and then interpolated back to the collocation points in order to circumvent this problem.

In the mid 1990s, Guiggiani and colleagues developed a method for solving the hypersingular boundary integral equations at the edges of panels subject to certain conditions. The method described in Guiggiani (1998), is reliant on the ability to perform a Taylor expansion at the panel edge, and thus a continuity requirement must be enforced between panels. This continuity requirement excluded the use of many common discrete elements, requiring elements based on B-spline approximations (e.g see Kim and Shin (2003)) which, as Xü (1992) states, has a very high computational burden (although the use of B-spline based elements has become much more common recently). In Gray *et al.* (2004b) a method was presented for which the solution to the hypersingular boundary integral equation could be found at panel edges of elements that did not conform to the strict requirements of the Guiggiani method; the method is known as a Galerkin boundary element method. The procedure developed by Gray *et al.* can be used to modify existing high order boundary element methods to include direct calculation of the velocity at vortex wake panel edges. This amalgamation is one which has not been attempted before.

Thus, the theoretical development of an improved high order method for lifting flows and free surfaces applying the work of Gray *et al.* (2004b) to a vortex wake was chosen as an obtainable outcome for the present thesis. Furthermore, such a method can be verified and validated by pre-existing analytical solutions, numerical predictions and experimental data for, say, solitary waves, aerofoils and hydrofoils. This takes a significant and original step along the road to developing a full tidal turbine and free surface simulation, while avoiding a number of pitfalls relating to the full problem. These issues stem mainly from a lack of validation data, a consequence of physical testing for tidal turbines being extremely new and also beset with uncertainty about the validity of experimental scaling. There is also a lack of high resolution data for the near wake of a tidal turbine and thus to properly validate this case, physical experiments would have to be undertaken in addition to the development of the model, which would be extremely time consuming. Therefore, expansion of the theoretical basis of an applicable model, such as a boundary element method, was regarded as preferable to attempting both a complicated numerical and experimental modelling process.

In closing this section, the author would like to introduce a quick caveat regarding the use of boundary element methods. Even though the strive for a robust high order method has been justified, the quantitative value of this type of numerical method must be addressed. A distinct difficulty with the validation of a boundary element method (particularly for lifting flows) is that beyond certain parameters (such as the angle of attack which induces stall in an aerofoil) experimental and numerical data can fail to match. A particular difficulty is that the strong assumptions of potential flow (see section 2.2) are often exceeded in real world conditions. This does not, in the author's opinion, reduce the value of boundary element methods, particularly for a problem like tidal turbines where little is previously understood. As Hess (1990) puts it,

> "Even when [the results] fail to give the proper experimental values, they are frequently useful in predicting the increment of effect of a proposed design change or in ordering various designs in terms of effectiveness."

In addition, Sarpkaya (1989), although sceptical about the quantitative abilities of vortex methods, stated that,

> "These pessimistic remarks are tempered by the fact that the most significant results are often qualitative judgements which provide insights into the real physics of the phenomenon."

The high order method proposed here can improve the accuracy of the potential flow based models. Nevertheless, it can not overcome the flaws inherent in the modelling assumptions themselves. Yet, for 'insight' into the effect of the free surface on the turbine and its wake, the model is expected to be extremely valuable. A true test of the model's validity to quantitative results for the full turbine problem may still be some time in coming. This will remain the case for all formulations of numerical model until high quality validation data becomes available.

1.3.3 Hypersingular Boundary Integral Equations

The hypersingular boundary integral equations (HBIEs) are the equations required to directly solve for the velocity field in a boundary element method. They are known as hypersingular as they contain an $\mathbf{O}(r^{-3})$ type singularity as $r \to 0$. The earliest technique for dealing with hypersingular boundary integral equations appeared in early commercial work with panel methods for the aeronautics industry. Famously, Hess

(1972) derived a velocity equation for a vortex sheet represented by dipoles by using Stokes' theorem to relate the surface integrals of dipoles to the Biot Savart Law (this regularisation is discussed as a special case in Gharakhani (1999)). Note that this form of regularisation is singular at panel edges. The most commonly used technique is to 'desingularise' the equations at panel edges by moving the collocation point slightly away from the edge. This technique is described by Epton (1992) who details a similar Stokes based regularisation. Only very recently have new methods been used to improve the accuracy and order of techniques in potential flow lifting solutions.

The Stokes regularisation technique has continued to be refined and streamlined in particular by the work of Sladek and Sladek (1992a,b), Balăs *et al.* (1989) and Gharakhani and Ghoniem (1997). These studies demonstrate a neat technique in exchanging an order of the singularity for a calculation of the local derivatives of the shape functions in the element of interest. An interesting point to note is that although Gharakhani and Ghoniem (1997) use the method for open surfaces, the regularisation technique described in Balăs *et al.* (1989) requires a closed surface and this necessity is repeated by Bonnet (1999). Gharakhani and Ghoniem (1997) also draw upon and expand the original work of Medina and Liggett (1988), which is a very readable application of a boundary element method to fracture mechanics and has closed form solutions to the singular integrals, negating the need for numerical quadrature. Gharakhani and Ghoniem (1997) evolve the technique for the higher order singularities. Unfortunately, although these techniques have proved extremely accurate for solving the HBIE on the surface within the panel it seems the technique does not extend to panel corners or edges. Bonnet (1999) showed that $C^{1,\alpha}$ continuity[3] for the potential is necessary to apply the above techniques and this condition is violated at the edges and corners of 'standard' elements. Bonnet (1999, chap. 5) gives an excellent review of identity based regularisation techniques.

Given the accuracy and maturity of these techniques for solving HBIEs away from element corners and edges, a wake evolution problem could be attempted by calculating the velocity at a point nearby to the collocation point and then relocating the collocation point there. This would provide an accurate solution for the velocity and thus the future position of the point, however, it would be necessary to interpolate the potential values

[3] $C^{k,\alpha}$ refers to the Hölder space for the k-th derivative of an arbitrary function. Hölder continuity is a stronger condition than regular continuity, yet does not imply differentiability.

to the evolved point, generating an error dependant on the order of the discretisation. This poses a dilemma in balancing how close the nearby point should be in order to get an accurate velocity without compromising the value of the potential. Other problems are also foreseeable, such as which panel to move the collocation point into when there are a number of panels meeting at a node and so on. Thus, without an extensive analysis, this technique is far from robust and in order to reduce errors an alternative approach must be sought.

An alternative method for regularising the HBIE was developed throughout the 90s by Guiggiani *et al.* (1992); Guiggiani (1995, 1998). Given the assumption of $C^{0,\alpha}$ continuity for potential and $C^{1,\alpha}$ continuity for it's normal derivative, Guiggiani was able to expand the field variables about the collocation point allowing a neat regularisation of the HBIEs. One important omission in the early work was a second free term that appears in the regularised HBIE formulae when curved elements are used. It was not until the publication of Guiggiani (1995) that this was reported, some five years after the initial publications. This issue was also highlighted by Mantič and París (1995). Guiggiani's work has been advanced by Karami and Derakhshan (1999) for higher order singularities. In conclusion, although this is a neat and accurate solution scheme, it does not overcome the problems of the continuity requirements at panel intersections.

A body of research has been undertaken to meet the continuity conditions with advanced elements, a particularly difficult challenge in three dimensions. The use of continuous elements has been successfully applied by Young (1996) and Wilde and Aliabadi (1998, 1999). Higher order B-spline elements have been used in free surface and lifting body simulations by Kim and Shin (2003) and Bernasconi and Richelsen (2008).

The regularisation of HBIEs without the need for complicated elements is obviously a highly desirable goal due to the ease of application to established 'standard' element routines, especially those of higher order. A 'smoothing' scheme was developed by Mantič *et al.* (1999); Graciani *et al.* (2000). This method negates the $C^{1,\alpha}$ continuity condition using linear elements. Sadly, the method has yet to be extended to three dimensions. The introduction to Mantič *et al.* (1999) provides an excellent review of HBIE solution techniques.

The publication of Gray *et al.* (2004a,b) finally provided a method which was theoretically capable of solving the HBIEs on three dimensional 'standard' boundary elements at corners and edges. In Gray (1995) the symmetric Galerkin approach to solving the HBIEs was applied to fracture analysis problems in 2D. Gray advanced the method to general derivatives for 2D linear elements in Gray *et al.* (1998). Gray and Kaplan (2001) presented the first results for using the Galerkin method in 3D, although this was lacking generality until the release of Gray *et al.* (2004a,b), when both normal and tangential derivatives for 3D hypersingular Galerkin methods with linear panels were presented. Gray *et al.* (2006) added to the analysis of Gray *et al.* (2004a) by deriving techniques for further simplifying the integrations involved. Gray's techniques have been applied by Sutradhar *et al.* (2005) for solving the problem of heat conduction in graded materials. Garzon *et al.* (2005) used Gray's method with a coupled level set/boundary integral wave simulation in order to time-step the free surface. Gray *et al.* (2004b) additionally showed the applicability of the Galerkin method to solving the problem of pressurised cracks. Given the similarity of cracks in fracture analysis to vortex sheets in fluid mechanics, it is hypothesised that the solution of the velocity on vortex wakes is a natural application of the Galerkin method. The available literature would also appear to indicate that this method has yet to be applied to quasi-potential lifting flows.

Early studies of the motion of vortex sheets using numerical methods are reported in work such as Baker *et al.* (1982) and Baker (1983) where a system of dipoles was used to represent a free surface as a vortex sheet. The study of vortex sheets has continued into the present century although most research concerns closed surfaces and are based on regularisations of HBIEs that necessitate the $C^{1,\alpha}$ continuity conditions. Pozrikidis (2000) used a Stokes regularisation that requires a closed surface to solve the potential gradients for three dimensional vortex sheets in the time domain. Grilli and Hu (1998) used a Taylor series expansion technique to regularise the vortex sheet. Brady *et al.* (1998) used Bézier interpolants to maintain $C^{1,\alpha}$ continuity. None of the above papers consider an open vortex sheet similar to that generated at the trailing edge of a lifting body. Bernasconi and Richelsen (2008) produced a grid free B-spline method to maintain the continuity required in order to use Guiggiani's method for finding the velocity at any point on an open vortex wake sheet.

In the current work, the vortex sheet generated at the trailing edge of the lifting body will be discretised using linear panels. Although this is one order of accuracy lower than the body and free surface panels it still represents an improvement over the traditional constant strength wake panels. The use of the Galerkin method to solve the velocity directly at collocation points may also prove advantageous when compared to higher order schemes which interpolate solutions to collocation points from panel centres. Using the gradient methods described for 'cracks' in Gray *et al.* (2004b) this study shall investigate the validity of reapplying the technique to calculate the velocity at collocation points for piecewise linear panels on a vortex wake sheet.

1.4 Extended Outline of Thesis

This section will summarise and outline the present thesis and its structure. Throughout this summary, pertinent literature relating to each part will be introduced along with consideration of alternative approaches. The thesis, in total, relies heavily on the work of Xü (1992) and Gray *et al.* (2004b). Xü's high order boundary element method was, in turn, based upon the seminal paper by Longuet-Higgins and Cokelet (1976) which dissects the problem of a time stepping, boundary element based, free surface method into an Eulerian stage and a Lagrangian stage. This algorithm is known as a mixed Eulerian Lagrangian method (MEL). The Eulerian stage implements the boundary element method in order to calculate the instantaneous solution at each time step. The Lagrangian stage involves time stepping of the geometry and field variable(s), subject to boundary conditions, which implicitly requires calculation of surface velocities. Due to the added complication in this process caused by the utilisation of the Galerkin method to calculate gradients, chapter 3 and chapter 4 are split into the Eulerian and the Lagrangian stage. Chapter 2 introduces the fundamental mathematics for which the numerical method will solve whilst chapter 5 and chapter 6 detail results for the non-lifting and lifting cases respectively.

1.4.1 Chapter 2: Mathematical Formulations

Chapter 2 begins with a discussion about the formulation of potential flow by reducing the full viscous fluid equations using scaling arguments as described in Katz and Plotkin (2001, chapter 1). The impacts of the chosen assumptions of inviscid and irrotational flow on the modelling is discussed alongside the alterations required to include lifting flows within a potential based model. This section concludes with a short discussion of the challenges of scale modelling of lifting bodies in the presence of a free surface.

Subsequently, it is shown that, under the assumptions of potential flow, the fluid is governed by the Laplace equation. The non-linear free surface boundary conditions (see Stoker, 1957) are defined in terms of the velocity potential, as are the boundary conditions on the channel walls, inlets and surfaces of interior bodies. The surfaces and boundary conditions that govern the wake of lifting bodies are also discussed. Finally, the fluid domain boundaries are divided into two subsets; on *Dirichlet* boundaries the normal derivative of the potential is unknown, on *Neumann* boundaries the potential itself is unknown.

From here, boundary element equations can be defined for the region of interest (a lifting body in a channel with a free surface) by following the derivation described by Katz and Plotkin (2001, chap. 1). However, here preference is given to coining the equations in terms of the potential and its derivative by taking the limit from the interior only, rather than in terms of sources and dipoles, where limits from both interior and exterior are taken. Uniqueness of the solution to the boundary integral equations is discussed and, in turn, the requirement for a condition at the trailing edge of a lifting body, known as the *Kutta* condition is defined and its evolution in time derived. For simplicity, the most basic Kutta condition is applied here (see Morino and Kuo, 1974), although more complicated alternatives have been presented in Morino and Bernardini (2001). Initial conditions are then given for the time dependant simulation.

The, now complete, governing equations are then presented in both full and perturbation potential form. The perturbation potential allows the far field velocity to be subtracted from the surface potential resulting in more accurate derivatives being calculated. These derivatives can either be calculated using numerical differentiation (see Xü, 1992) or, in the case of the wake, they must be calculated using the direct derivative of the boundary integral equation which produces the aforementioned hypersingular boundary integral equation.

Finally, the method of weighted residuals is utilised to describe how the boundaries of the domain may be discretised. This leads to many alternative discretisation methods, two of which are used in the present method. The first, the *collocation method*, is used for the standard boundary element method as in Xü (1992). The second, a *Galerkin method* is required for the solution to the hypersingular boundary integral equations derived by Gray *et al.* (2004b).

1.4.2 Chapter 3: Eulerian Stage Implementation

Chapter 3 introduces the numerical method instigated to solve the Eulerian, or instantaneous, stage of the mixed Eulerian Lagrangian method. This method provides the numerical solution to the boundary integral equations derived in chapter 2. The choice of order for the discretisation has already been discussed, however, there are also a number of alternative methods for solving the boundary integral equations. The key challenge to efficiency in a boundary element method is the formation and inversion of a full n by n matrix made up of the influences of each panel on each node. The computational burden of forming the matrix can be reduced by applying a technique known as the fast multipole method, developed by Rokhlin (1983). By grouping far field influences together the method can reduce computational burden from being proportional to n^2 to being proportional to n, a significant saving. Xü (1992) considered implementing such a method within his model, however concerns over accuracy led him to form the influence matrices by the traditional manner. In fact the first three dimensional wave solver that applies the multipole method was not developed until the work of Fochesato and Dias (2006). They applied the method to the free surface solver of Grilli *et al.* (2001) in order to improve its performance. The potential to solve boundary elements even faster by fast Fourier transforms (see Fata, 2008) means one of the key burdens of potential methods may soon be overcome. For the present method, however, the method of Xü (1992) is preferred.

Chapter 3 continues by defining the isoparametric, biquadratic influence functions required for the high order method. The far field influences are defined and discussed along with the adaptive algorithm proposed by Xü (1992) for which some minor adjustments are made. Subsequently, the singular integrals are reduced to a form that can be numerically integrated and the adaptive control for these integrals is also investigated. Once the influence functions for the nine node elements are defined, the same process is carried out for linear three node elements. These elements are required as the Galerkin boundary element method of Gray *et al.* (2004b) is derived in terms of these elements. Testing the hypothesis that the Galerkin method can calculate the wake velocity at collocation points does not require that the wake panels be biquadratic, just higher than constant order. The method can potentially be extended to biquadratic elements should it prove successful for the linear elements.

Once the influence functions are defined, the linear systems are derived for both non-lifting and lifting flows. Finally, some modifications of the linear systems, in order to model surface intersections and represent boundaries by reflection, are discussed. Note, once this stage is reached, the influence matrix is prepared for inversion. No particular solver is specified for the linear system as the ability of general algorithms (especially when operating in parallel) was seen to be sufficient. Indeed, it was found that more computational burden was imposed in forming the influence matrix than from inverting it.

1.4.3 Chapter 4: Lagrangian Stage Implementation

Once all the boundary data is known for a particular time-step, the mixed Eulerian Lagrangian algorithm moves onto the Lagrangian stage. The goal of this stage is to calculate the velocities on the dynamic surfaces and then time-step the geometry and field values dependant on the boundary conditions. This is a well established process for high order free surface solvers, utilising numerical differentiation. Often large sliding elements are used to calculate the surface derivatives, allowing central differencing schemes to be used in the majority of cases. Obviously, a skewed scheme must still be implemented at or near channel walls. Grilli *et al.* (2001) use a fourth order 25 node sliding panel to implement their numerical differentiation and a similar level of accuracy is used for the free surface derivatives within the present method and is described at the start of chapter 4. The accuracy of the evolution of the free surface is of particular interest in the results of chapter 5.

Calculating the velocity on the vortex wake emanating from the trailing edge of a lifting body is more complicated as was discussed earlier in the introduction. Even before calculating the velocity, some thought must be given to the orientation of the vortex wake as it leaves the lifting body. This process is another element of the Kutta condition that is used to provide the strength of the vortex wake, but also sets the requirement for the direction of the wake. This ‘spatial Kutta condition’ was best described by Mangler and Smith (1970). They discovered that the vortex wake can take different positions dependant on the spanwise vorticity along the trailing edge. The wake can leave tangentially to either side of the lifting body at the trailing edge or at trailing edge bisecting angle. In fact, only under very specific conditions is the third case true. Although the formulation of Mangler and Smith (1970) was for two dimensional steady flows, the work of Morino and Kaprielian (1985) extended the results to three dimensional unsteady flows with similar results. In the present method, for simplicity, the wake leaves the trailing edge at the bisecting angle of the lifting body. It was felt that the more complicated approach of Mangler and Smith (1970) was not necessary for the purposes of this thesis and could be applied as additional work.

Chapter 4 continues by describing the solution to the hypersingular boundary integral equation in order to find the velocity at computational nodes on the vortex wake. The applicability of the Galerkin method described in Gray *et al.* (2004b) to a vortex wake is demonstrated, with particular attention being given to the analogy of the crack problem in fracture mechanics (see Bonnet, 1999). Various formulae that were not formally published in Gray *et al.* (2004b), yet are necessary for the solution of the method, are derived and a discussion of the range of applicability of the method is conducted. The Galerkin method can not be applied at the wake edges and thus an alternative method for calculating the velocity must be used there and then fed to the Galerkin method. In fact, although the related collocation version of the method can be applied at the edges, the real solution at the wake edges is infinite (in the same sense as the velocity at the sharp trailing edge is infinite) and so an alternative solution must be derived for the edges. Due to time constraints, a basic average of off-surface velocities was applied as a 'first attempt' in order to feed the necessary information to the Galerkin method. An alternative approach may be to replace the edge of the vortex sheet with a vortex core, such as detailed in Lee and Kinnas (2005a), or vortex blobs as detailed in Hou *et al.* (2006). These extensions would add significant complications and thus are reserved for further work. Nevertheless, a more accurate solution to the velocity at the wake edges would be desirable in an otherwise robust modelling approach.

Once the algorithm has reached this point, the velocities on the dynamic surfaces are known. The final step, in order to complete the Lagrangian stage, is to move the geometry and field values forward in time. In boundary element methods there are two main alternatives for applying time-stepping; the first is an implicit scheme and the second an explicit scheme. An example of an explicit scheme was designed by Dold and Peregrine (see e.g. Dold (1992)), where they made use of the fact that partial time derivatives of the velocity potential also satisfy Laplace's equation. This allows higher-order material time derivatives of the free-surface position and velocity potential, ϕ, to be calculated, which in turn allows the use of higher-order truncated Taylor series for the time-stepping. This technique benefits from the fact that the solutions of Laplace's equation for $\phi_t, \phi_{tt}, \ldots$ are extremely low cost because the influence functions have already been calculated for the solution of $\nabla\phi = 0$. This process, although conceptually simple and efficient, is difficult to implement in practise for the reasons given in Forehand (1998, pg. 25), which include problems of deriving the boundary conditions for Laplace's equation applied to higher-order partial time derivatives of ϕ. The key advantages of Dold's time-stepping method is the reduced computational burden and the ease of implementation of a dynamic time-step.

Implicit time-stepping schemes, as used in the well known method of Longuet-Higgins and Cokelet (1976), are analytically simpler to implement but are more computationally demanding. This is because numerical integrators such as the Runge-Kutta fourth order scheme (RK4) and the Adams-Bashforth fourth order scheme (AB4) require intermediate time-steps. In the case of RK4, three additional Eulerian stage implementations are required (without the ability to reuse the influence matrix), and for AB4 one additional Eulerian stage is required. Dynamic time-stepping is also more computationally expensive as, when the time-step is changed, the AB4 integrator cannot be used as it requires information from three previous time-steps. Thus, when the time-step changes the RK4 method must be run at least 4 more times. Even with this deficit, however, the ease of implementation made it the reasonable choice for the present method and it is described at the end of chapter 4 along with a discussion of the required metrics for implementing dynamic time-stepping.

After time-stepping, the MEL method is complete and any post-processing (for internal velocities, pressures, energies, errors etc.) can be undertaken. The process is then repeated until a set time has been reached or some error metric is beyond tolerance.

1.4.4 Chapter 5 and Chapter 6: Results

With the mathematical and numerical foundations prepared for the numerical model, its implementation must be tested, a process known as verification. This was undertaken in two stages; the first stage examines whether the high order boundary element method, based on that of Xü (1992), would produce the expected order of convergence for steady problems and numerical stability for unsteady problems, with neither case including lifting bodies. To examine the quality of the present method a hierarchy of tests were performed which would examine various aspects of the code. These include the flow over a sphere in an infinite fluid domain and the steady flow through a rectangular channel with one Dirichlet boundary. These tests, in particular, allow the order of convergence of the method to be examined and compared to that of Xü.

With the order of convergence established, the time-stepping nature of the code can then be tested. An ideal test case for this problem is the evolution of solitary waves. This will evaluate the accuracy of the numerical differentiation routines and the numerical integration of the free surface boundary conditions. Solitary waves have an analytical solution within potential based methods and their celerity and form remains unchanged over time. This provides an excellent assessment of the dynamic time-stepping metrics and whether common 'saw-tooth' instabilities are a problem for the present method. In addition, the more challenging problem of counterpropagating solitary waves provides further evidence for the successful implementation of the code. Phenomena that can only be observed when using non-linear free surface boundary conditions are the correct quantity of wave run-up on impact of counterpropagating solitary waves and the formation of a wave residual after the two solitary waves detach again. These quantities can be compared to the results of other numerical studies such as Craig *et al.* (2006).

Once the implementation of the quadratic high order boundary element method has been examined, the application of the Galerkin gradient method to the wake can be attempted. Due to the challenges associated with the wake edges, a carefully designed algorithm must be followed in order to avoid the method exhibiting singular behaviour. Additionally, care must be taken during the time-stepping of the method to ensure that the spatial Kutta condition is enforced. Preliminary results for the development of the wake behind a lifting body in an infinite domain are examined. Unfortunately, there is little comparison data available to test the quality of the output. Comparisons can be made to the previous work of Pyo and Kinnas (1997), and simulations utilising the more complicated Galerkin method can be compared to simulations where the basic 'averaging' method, which is used to calculate the velocity at the wake edges, is applied over the entire wake sheet. Finally, the quality and stability of the evolution of the sheet using the Galerkin method is discussed and possible directions of work, in order to improve the results, are outlined.

Chapter 2

Mathematical Formulations

2.1 Introduction

The challenge of numerically modelling a lifting body in the presence of a free surface is extremely great. The complete physics that such a simulation should capture includes free-stream turbulence, cavitation, bottom friction and much else besides. Although much of the theory exists to tackle these problems, a simulation that included them all would be unfeasibly computationally expensive and extremely challenging to validate. Given these challenges, it is appropriate to seek an approximation to the real physics, where the computational expense is tractable and validation is practicable.

Thus the purpose of this chapter is to introduce the chosen assumptions and the models based upon them. This includes equations and boundary conditions for the simulation of a non-linear gravity driven free surface, and also those required for lifting bodies. The initial-boundary value problem is formulated for free surface flows over a finite domain, with and without submerged lifting and non-lifting bodies. Formulations are also derived for lifting and non-lifting bodies in an infinite domain.

Once the initial-boundary value problem has been derived, the solution procedure, known as a mixed Eulerian-Lagrangian method, is introduced. Within the Eulerian stage a boundary integral equation is formed in order to provide the unknown boundary data at each time step. Two formulations of the boundary integral equations are presented (full and perturbation) and compared, and the uniqueness of solutions is proved for both. For the Lagrangian stage, velocities must be calculated on the domain surfaces, which leads to a hypersingular boundary integral equation and the solution to this equation is discussed.

Finally, two alternative discretisation schemes for the developed equations are introduced. The first, a collocation method, is used for solving the boundary integral equation at the Eulerian stage. The second, the Galerkin method, is critical to the solution of the hypersingular boundary integral equations. Both are related to, and discussed in the context of, the weighted residual method.

2.2 The Assumptions of Potential Flow

A potential flow based model is an ideal choice for approaching the problem of a lifting body in the presence of a free surface. To support this statement, it is important to address the assumptions that are made in order to reduce the equations of motion to a state where a velocity potential can be defined and Laplace's equation applied. The two main assumptions are that the flow should be free from viscosity, referred to as *inviscid*, and that small fluid elements within the flow have zero spin, or *vorticity*, hence the fluid is described as *irrotational.* The next few paragraphs attempt to demonstrate the applicability of using these strong assumptions for this complicated physical problem.

The process begins by choosing an orthogonal reference frame, fixed on the earth with its origin at the undisturbed water level. x and y represent the two horizontal directions and z represents the vertical direction, positive pointing up. The vector $\mathbf{x}$ with coordinates (x, y, z) is defined to be the spatial position vector of a point in this frame. Within this frame of reference, consider the continuity equation and Navier-Stokes equation for a Newtonian fluid with a constant viscosity coefficient, μ, and constant density, ρ, given by

$$\nabla \cdot \mathbf{q} = 0, \tag{2.1}$$

$$\rho \left(\frac{\partial \mathbf{q}}{\partial t} + \mathbf{q} \cdot \nabla \mathbf{q} \right) = \rho \mathbf{f} - \nabla p + \mu \nabla^2 \mathbf{q} \tag{2.2}$$

where $\mathbf{q} = (u, v, w)$ is the velocity vector of the flow, t is time, $\mathbf{f}$ is the body force vector and p is the pressure.

In order to better understand the relative magnitudes of the individual terms of the governing equations a dimensional analysis is performed following Katz and Plotkin

(2001). To begin, some characteristic quantities are defined as

L	–	reference length,
V	–	reference speed,
T	–	characteristic time,
p_0	–	reference pressure,
f_0	–	body force.

and then, using the above, some non-dimensional variables can be defined, thus:

$$\begin{aligned} \mathbf{x}^* &= \frac{\mathbf{x}}{L}, \\ \mathbf{q}^* &= \frac{\mathbf{q}}{V}, \\ t^* &= \frac{t}{T}, \\ p^* &= \frac{p}{p_0}, \\ \mathbf{f}^* &= \frac{\mathbf{f}}{f_0}. \end{aligned} \tag{2.3}$$

The governing equations, given by equations (2.1) and (2.2), are now rewritten in terms of the non-dimensional quantities. Thus the continuity equation remains unchanged,

$$\nabla^* \cdot \mathbf{q}^* = 0, \tag{2.4}$$

and the momentum equation becomes

$$\left(\frac{L}{TV}\right)\frac{\partial \mathbf{q}^*}{\partial t^*} + \mathbf{q}^* \cdot \nabla^* \mathbf{q}^* = \left(\frac{Lf_0}{V^2}\right)\mathbf{f}^* - \left(\frac{p_0}{\rho V^2}\right)\nabla^* \mathbf{p}^* + \left(\frac{\mu}{\rho V L}\right)\left(\nabla^*\right)^2 \mathbf{q}^*. \tag{2.5}$$

All terms with asterisks in equations (2.4) and (2.5) are now of order one and thus their importance is determined by the values in the parentheses. For convenience, from now on, drop the asterisks from the non-dimensional variables. As can be seen, the non-dimensional momentum equation, given by equation (2.5), has four coefficients which govern the relative importance of certain terms in the equation. The most notable of these is on the far right of equation (2.5). The value in the parentheses represents the ratio of inertial to viscous forces and its inverse is known as the *Reynolds number*,

$$Re = \frac{VL}{\upsilon}, \tag{2.6}$$

where υ is the kinematic viscosity defined as

$$\upsilon = \frac{\mu}{\rho}. \tag{2.7}$$

The final term in the momentum equation being multiplied by Re^{-1} implies that if Re is high then this viscous term of the momentum equation becomes negligible. It should be noted that high Reynolds number does not guarantee viscosity can be neglected throughout the flow field. In fact this is only the case where the $\nabla^2\mathbf{q}$ term of equation (2.5) is approximately order 1. This is true of the outer flow field; however, close to solid boundaries, where $\nabla^2\mathbf{q} \gg 1$ the viscous term can no longer be discounted. Indeed, in these regions a boundary layer solution should be sought such that in the x direction,

$$\rho\left(u\frac{\partial x}{\partial x} + w\frac{\partial u}{\partial z}\right) = -\frac{\partial p}{\partial x} + \mu\frac{\partial^2 u}{\partial z^2} \tag{2.8}$$

and in the normal z direction,

$$\frac{\partial p}{\partial z} = 0. \tag{2.9}$$

The implication is that there are two distinct regions in the flow. The first is within the outer flow field where viscosity can be neglected and the Euler equations can be used to find the solution. In the second region the effects of viscosity can not be discounted and this occurs near boundaries within a certain boundary layer, of length δ. For large Reynolds number the proportion of boundary layer thickness, δ, to characteristic length can be derived from Schlichting (1960, pg. 130) as

$$\begin{aligned} &\delta \approx 5\sqrt{\frac{\upsilon L}{U}} \\ \Rightarrow\ &\frac{\delta}{L} \approx \frac{5}{L}\sqrt{\frac{\upsilon L}{U}} = 5\sqrt{\frac{1}{L^2}\frac{\upsilon L}{U}} = 5\mathrm{Re}^{-1/2}. \end{aligned} \tag{2.10}$$

Using this calculation, the boundary layer thickness for a typical hydrofoil will be in the order of 2mm, thus the extent of this region is very small compared to that of the outer flow region. However, equation (2.10) can only be used as an estimate of the size of the *viscous boundary layer* on the hydrofoil. For the tidal channel itself the *turbulent boundary layer* can occupy large proportions of the depth of the channel. The extent to which this boundary layer encroaches on the free surface (see Myers and Bahaj, 2008; Lueck and Lu, 1997) is still disputed, however this boundary layer is critical to the creation of the familiar log velocity profile seen in tidal channels. Possibly the biggest disadvantage of a potential based method is not capturing this log profile or the turbulence contained within the turbulent boundary layer. Even so, because the main interest of the present simulation is the effect of the free surface on the hydrofoil and its wake, it is considered an unfortunate but reasonable admission.

It is now prudent to discuss the second assumption, that of irrotational motion. In order for the flow to be considered irrotational, it must have no vorticity. Under what circumstances can this be the case? In particular, how can this be true given that it is useful to model some vorticity in order to better describe the physical problem?

Within boundary layers on solid surfaces the no-slip condition will naturally generate vorticity. The solid boundaries are not the only region that will generate vorticity however, especially when considering a free surface. Waves on a free surface have been shown to generate vorticity on both the solid bottom surface and at the free surface itself. Wave motion has two significant effects. The first and most commonly known is the generation of a boundary layer on the bottom boundary as described in Svendsen and Jonsson (1980). This boundary layer dissipates and reforms regularly for short waves and is confined to a very small distance from the bottom.

Longuet-Higgins (1992) has shown that at any free surface at which the tangential stress, τ_{ns}, vanishes there must be a surface vorticity $\omega = -2\kappa q$ where κ is the curvature and q is the tangential velocity. Therefore, there exists an additional boundary layer of length δ beneath the free surface where vorticity generation occurs. Longuet-Higgins observed that, at the second order of the wave steepness parameter, ak, this vorticity can escape through the Stokes boundary layer. Longuet-Higgins uses this phenomenon as a mechanism to describe the formation of capillary waves and bores on gravity waves, even in the absence of wind.

Dealing with this vorticity raises some important issues. *Kelvin's circulation theorem* (see Acheson, 1990, pg. 157) implies that if vorticity is present in an inviscid flow it can not be created or destroyed. Thus it is important to consider how to deal with the vorticity generated near the boundaries. Formally, the dominance of viscous forces near the boundaries will deal with much of this vorticity, dissipating it rapidly as it moves away from the boundary. This is also the case for the viscosity generated at the free surface when the waves are not too steep. We can therefore assume the vorticity is maintained in the small boundary layers near to the surfaces. However, as mentioned previously, it can prove very useful to model the time evolution of the vorticity generated on a lifting body surface when modelling the lifting flow of an aerofoil or hydrofoil.

It is a commonly known fact that fluid *circulation* around a body is the main cause of lift for wings (and even swerve on footballs!). Therefore, without a circulation around the aero/hydrofoil it would not lift. On the other hand, generating circulation here would be contradictory to Kelvin's circulation theorem and the flow could not be irrotational. The solution is to explicitly model the vorticity generated at the foil shedding into the trailing wake. This *vortex wake* has an equal and opposite circulation to that around the foil, allowing it to generate lift while allowing the flow to remain irrotational. The wake itself is obviously not irrotational and must be isolated from the irrotational domain. It is considered to be an infinitely thin vortex sheet, where there is a discontinuity of the tangential velocity on either side.

Vortex sheet modelling is one of the 'black arts' of potential flow based methods. Consider the assumption that the vortex sheet will emanate from the trailing edge of the foil. This is indeed the case for 'normal operation' of the foil where the angle of attack is not too steep. However, a potential model will not accurately model the characteristics of the foil with high angle of attack, or in 'stall', as the vorticity may not leave the foil at the trailing edge. In addition, the edges that bound the wake sheet pose a particular problem, in that the velocity here will be divergent (see Johnson (1994, pg. 546) regarding this behaviour for vortex sheets and Williams (1952) for the equivalent solution in fracture elastostatics) and this must be dealt with along with the impact this has on the rest of the sheet.

Choosing to discount generated vorticity at the other surfaces and within the turbulent boundary layer may cause some important physical effects to be missed. In this case, there is a need for more detailed viscous modelling at these boundaries. One method may be to predict the generation of additional vortex sheets, which can then be modelled using potential flow.

To summarise from the above discussion, it is clear that the inviscid, irrotational assumption is valid in the outer regions of the flow away from the boundaries and foil wake. On the boundaries themselves, vorticity is important and the flow is rotational. On the surface of the hydrofoil (provided it is not in stall) this vorticity is maintained within the viscous boundary layer described above. At the trailing edge of the aerofoil this vorticity is shed into the flow as a vortex sheet, a discontinuity in the speed of the fluid on either side of the boundary. This is known as the wake of the aerofoil and should be modelled to provide a unique solution to the problem. On the bottom and free surface, vorticity is generated and can separate. Provided the vorticity generated is not very strong then it can be assumed that this will be dissipated quickly by the viscous forces in the flow and therefore can be neglected. If the vorticity generated is large then there is a possibility that neglecting this physical effect may cause some errors in the results.

2.2.1 The Effects of Scaling

Before closing the discussion on potential flow, it is interesting to consider some typical characteristic values for the problem and the impact of recreating full scale physics at smaller scale. Obviously, for the Euler equations to hold, a large Reynolds number (see equation (2.6)) is required within the flow, so the first step is to check this. Consider a channel of 30m depth having a velocity of 5ms^{-1}. Within the channel is a hydrofoil of approximately 1m chord length. The Reynolds number for the hydrofoil (taking the kinematic viscosity of water to be $1 \times 10^{-6}\text{m}^2\text{s}^{-1}$) is $\text{Re}_B = 5 \times 10^6$ and for the channel it is $\text{Re}_C = 1.5 \times 10^9$. Thus, the potential flow approximation is valid in this case.

Another quantity of interest is the Froude number. The Froude number is a non-dimensional quantity that defines the relative importance of inertial and gravitational forces and it is given by

$$\text{Fr} = \frac{V}{\sqrt{Lg}}. \tag{2.11}$$

The inverse square of this term appears in the first term on the right hand side of equation (2.5) where f_0 is chosen to be gravitational acceleration, g.

Given that the premise of the model developed herein is to analyse free surface effects, it is important not to have a low Froude number for the channel - if it were very low then the dominance of gravity would suggest the free surface could be modelled as a rigid lid. For the channel described above, the value is $\text{Fr}_C = 0.29$ which is acceptable.

With these base figures established, consider the problem of scaling the channel whilst maintaining similar physical effects. In experimental modelling including a free surface, it is common to fix the value of the Froude number. With this restriction, the hydrofoil's Reynolds number can now be related to depth of the channel. To show this, consider a fixed Froude number for a channel of given depth, say L_C. Then the velocity needed to achieve the Froude number is given by

$$V = \sqrt{L_C g}\mathrm{Fr}_C. \tag{2.12}$$

By substituting equation (2.12) into the expression for the Reynolds number of the hydrofoil, observe that

$$\mathrm{Re}_B = \frac{\mathrm{Fr}_C\sqrt{L_C g}L_B}{\upsilon} \tag{2.13}$$

where L_B is the chord length of the hydrofoil. By estimating that the chord length will be one twentieth of the channel depth (i.e. $L_B = 0.05L_C$) the relationship of the Reynolds number of the hydrofoil to the channel depth is given by

$$\begin{aligned} \mathrm{Re}_B &= \frac{0.05\mathrm{Fr}_C\sqrt{L_C g}L_C}{\upsilon} \\ &= \left(\frac{0.05\mathrm{Fr}_C\sqrt{g}}{\upsilon}\right) L_C^{3/2} \\ \Rightarrow \qquad \mathrm{Re}_B &\propto L_C^{3/2}. \end{aligned} \tag{2.14}$$

Thus modelling the channel at tenth scale, whilst keeping the Froude number constant, would produce a Reynolds number of just three percent of the original value. Even then, this is still a Reynolds number in the turbulent regime. Despite this, it has been shown (see Bourgoyne *et al.*, 2003) that small variations in Reynolds number can impact on the characteristics of hydrofoils.

One potential solution to the scaling problem is apparent by observing that the Froude number is independent of the viscosity of the fluid being modelled. This offers the opportunity to produce exact full scale physics by using alternative fluids at smaller scales. For instance, for the characteristic quantities described above, petroleum could be used at half scale. At quarter scale, mercury would provide a near correct Reynolds number for the hydrofoil. Although this is an interesting observation, it is highly unlikely that scale experiments will be carried out in this manner as the aforementioned fluids are highly dangerous to work with.

2.3 Physical Model

Now that the assumptions of a potential flow solution have been described, the physical model using these assumptions, known as potential flow, can be introduced. The purpose of this section is to describe potential flow and introduce the domain to which this model will be applied.

2.3.1 Potential Flow

As detailed in the previous section, the assumption of high Reynolds number, within a well mixed fluid, allows the flow about a hydrofoil to be modelled as incompressible, inviscid and irrotational. For the flow to remain irrotational, irrespective of time, it must be governed by the principals of Kelvin's theorem (see Acheson, 1990, chapter. 5). Thus, all forces must be conservative, the total system energy should be unchanging, and the total circulation in the system must also be unchanging with time. With these constraints, for time dependent motion, the flow can be described using a scalar function $\phi(\mathbf{x}, t)$ in a three-dimensional Cartesian system, such that the fluid velocity $\mathbf{q}$ is given by

$$\mathbf{q} = \nabla\phi. \tag{2.15}$$

ϕ is known as the *velocity potential.* Substituting equation (2.15) into the continuity equation (given by equation (2.1)), yields

$$\nabla \cdot \mathbf{q} = \nabla \cdot \nabla\phi = \nabla^2\phi = 0. \tag{2.16}$$

This equation is known as *Laplace's equation* and is the governing equation of the flow.

Given the three-dimensional nature of the problem, the fluid domain, $D \subset \mathbb{R}^3$, is assumed to be simply-connected, but not necessarily bounded. In addition, for equation (2.16) to hold, ϕ must have a continuous second derivative within D, i.e. $\phi \in C^2(D)$. For the purposes of the model herein, $S \subset \mathbb{R}^2$, the boundary of D, is assumed to be an R–L surface as defined by Lynn and Timlake (1968). This type of surface provides a sufficiently smooth Lyapunov surface within discrete elements, whilst assuming that the surface as a whole is Kellogg regular. This facilitates the use of piecewise continuous elements whilst allowing for sharp corners between them, a requirement borne out by the discretisation of the equations discussed in section 2.7. Subsequently, it is well

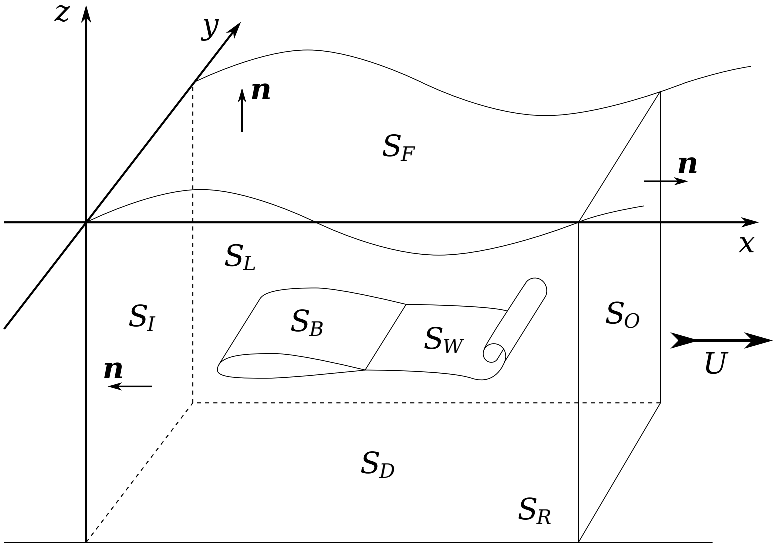


Figure 2.1: The fluid domain under consideration.

known that ϕ can be found anywhere interior to the domain D once suitable boundary values are known on S.

2.3.2 Computational Domain

The requirement is to solve the governing equation (equation (2.16)) for the domain described by figure 2.1. The fluid volume D has within it a lifting body, bounded by S_B, with a connected wake surface, S_W. Completing the boundary of D are the free surface, S_F, the inlet and outlet boundaries, S_I and S_O, and the bottom boundary, S_D. There are also the left and right vertical channel side wall boundaries, S_L and S_R. In the case where one or all of these surrounding boundaries are not present then a boundary at infinity, S_∞ must be included. Specific boundary conditions must be applied to the different boundaries, depending on whether the boundary is solid or free.

In traditional free surface potential problems, the boundary is subdivided into two subsets dependant on the known values on these subsets,

$$S = S_D \cup S_N, \tag{2.17}$$

where S_D describes a boundary for which ϕ is known, also known as a *Dirichlet* boundary, and S_N describes a boundary on which ϕ_n, the normal derivative of ϕ, is known, referred to as a *Neumann* boundary. Hereafter, the subscript n shall denote the normal derivative in the direction out of the fluid domain, unless otherwise stated.

The free surface, S_F, is a member of the Dirichlet subset and the other surfaces, apart from the wake surface, S_W, are Neumann surfaces. It turns out that the wake surface can be classified as neither, as it shall be shown that the quantity on the wake which describes its motion is known for all positions and times apart from at the intersection with the trailing edge of the lifting surface. Thus, when lifting bodies are included the total surface must have three subsets denoted

$$S = S_D \cup S_N \cup S_W. \tag{2.18}$$

2.4 Full Potential Formulation of Governing Equations

Now that the potential flow model and the domain to which it is applied have been formulated, the mathematical equations which describe the boundary conditions are required. In addition, a method for isolating the unknowns on the boundaries, known as the boundary integral equation method will be introduced in this section. With this method and these equations resolved, the theoretical framework is in place to produce a time domain simulation of the potential flow within the chosen domain.

2.4.1 Boundary Conditions

The required boundary conditions to simulate the physics within the domain illustrated in figure 2.1 are developed in this subsection. The far field velocity $\mathbf{U}(x, y, z)$, is introduced into the domain by boundary condition set on the inlet and outlet boundaries, S_I and S_O respectively. To conform with the irrotational assumption, $\mathbf{U}$ is chosen to be steady and uniform, having magnitude U in the direction of x . As such, the boundary conditions for S_I and S_O are

$$\phi_n(P) = -U \qquad P \in S_I \tag{2.19}$$

$$\phi_n(P) = U \qquad P \in S_O \tag{2.20}$$

where the unit normals on each surface, $\mathbf{n}$, are pointing out of the domain (for example, in figure 2.1, the normal would be $(-1, 0, 0)$ for S_I and $(1, 0, 0)$ for S_O). The remaining

Neumann boundaries, such as the submersed body, S_B, and bottom, have no crossing boundary conditions given by

$$\phi_n(P) = 0 \qquad P \in S_N - (S_I \cup S_O). \tag{2.21}$$

Note, that if the submerged body were translating, such as a turbine blade might, then the above equation would no longer hold. In the case where a Neumann boundary is the surface of a moving submersed body, the above condition generalises to

$$\phi_n(P) = \mathbf{u}(P) \cdot \mathbf{n} \tag{2.22}$$

where $\mathbf{u}(P)$ is the velocity of the body at point P and $\mathbf{n}$ is the unit normal, also at P.

The free surface boundary, S_F, (the only Dirichlet boundary in the problem) has some important physical considerations that must be adhered to. The first of these conditions is that a particle on the free surface must remain on the free surface. This is known as the *kinematic boundary condition* and is written,

$$\frac{D\mathbf{x}}{Dt} = \nabla\phi \qquad\qquad \mathbf{x} \in S_F. \tag{2.23}$$

The second condition arises from the Bernoulli equation for unsteady irrotational flow. Neglecting the effects of surface tension, the pressure felt at the free surface will be equal to that exerted by the external atmospheric pressure, p_F. Given this, the Bernoulli equation states that

$$\frac{\partial\phi}{\partial t} = -\frac{1}{2}\left(\left(\frac{\partial\phi}{\partial x}\right)^2 + \left(\frac{\partial\phi}{\partial y}\right)^2\right) - gz - \frac{p_F}{\rho} + c(t) \qquad \mathbf{x} \in S_F \tag{2.24}$$

where $c(t)$ is some function of time and g is the acceleration due to gravity. Adding a function of time alone to ϕ makes no difference to the flow velocity, given by equation (2.15), and so, if p_F is a constant, the last two terms in equation (2.24) can be absorbed into ϕ_t to give

$$\frac{\partial\phi}{\partial t} = -\frac{1}{2}\left(\left(\frac{\partial\phi}{\partial x}\right)^2 + \left(\frac{\partial\phi}{\partial y}\right)^2\right) - gz \qquad \mathbf{x} \in S_F. \tag{2.25}$$

Additionally, by noting that the material derivative of ϕ is

$$\frac{D\phi}{Dt} = \frac{\partial \phi}{\partial t} + \nabla\phi \cdot \nabla\phi \tag{2.26}$$

the Lagrangian form of equation (2.25) is written

$$\frac{D\phi}{Dt} = \frac{1}{2}\left|\nabla\phi\right|^2 - gz \qquad \mathbf{x} \in S_F. \tag{2.27}$$

Equations (2.25) and (2.27) are known as the *dynamic boundary condition* on the free surface.

As mentioned previously, there is a fundamental quantity on the wake which is known everywhere apart from at the trailing edge. To provide a value for this quantity at the trailing edge, a further boundary condition must be applied there. This boundary condition is a consequence of the *Kutta-Joukowski hypothesis* which postulates that the correct amount of circulation is achieved when the flow leaves the trailing edge of an aerofoil smoothly and with finite velocity. The subsequent and well known *Kutta-Joukowski theorem* (see Acheson, 1990, pg. 21) goes on to establish the relationship between circulation and lift. The concept is neatly illustrated in figure 2.2. Part (a) shows the irrotational flow with no circulation. Observe that the stagnation point is on the top side of the aerofoil and that the fluid velocity must be infinite at the trailing edge (as it must turn the sharp corner). In part (b) enough circulation is applied so that the velocity at the trailing edge is finite. This finite velocity condition on the flow at the trailing edge of the aerofoil is the required extra boundary condition.

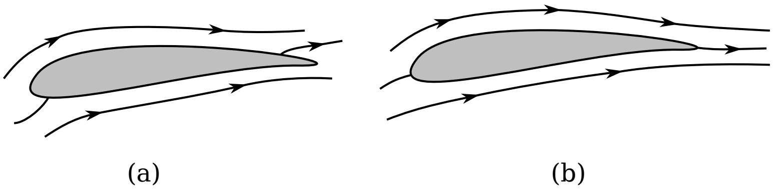


Figure 2.2: The effect of circulation on the flow about an aerofoil. Recreated from Acheson (1990).

The numerical implementation of the finite velocity trailing edge condition is known as the *Kutta condition*. The implementation of this condition has a significant impact on the design of the present method. As such, this is readdressed in section 2.4.3 and section 2.4.4, along with a brief introduction to the Kutta-Joukowski hypothesis and

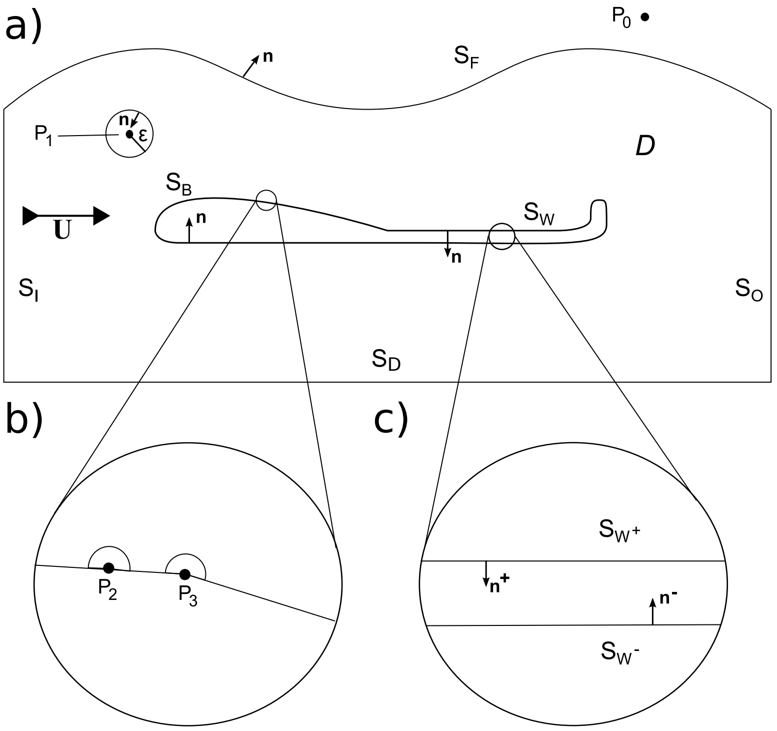


Figure 2.3: The fluid domain under consideration (2D-slice).

theorem.

2.4.2 Boundary Integral Equations

In order to find the unknown values on the Neumann and Dirichlet boundaries, a set of integral equations are derived which can then be solved simultaneously. These equations are known as boundary integral equations. They shall be derived from first principles by employing an application of Green's second identity (see Kellogg, 1929, pg. 215) in a similar manner to the derivation demonstrated in Katz and Plotkin (2001).

A two-dimensional slice of the three-dimensional computational domain is illustrated in figure 2.3. The requirement is to solve equation (2.16) for the 3D domain. Note that the left and right side vertical channel boundaries, S_L and S_R cannot be marked in figure 2.3. Also, recall that in the case where one or all of the surfaces bounding D are not present then a boundary at infinity, S_∞, must be included.

Following Katz and Plotkin, the divergence theorem is applied to a set of two scalar functions $\phi_1(x, y, z; t)$ and $\phi_2(x, y, z; t)$, resulting in

$$\int_S (\phi_1 \nabla \phi_2 - \phi_2 \nabla \phi_1) \cdot \mathbf{n} \, dS = \int_D \left(\phi_1 \nabla^2 \phi_2 - \phi_2 \nabla^2 \phi_1 \right) dD. \tag{2.28}$$

Equation (2.28) is known as Green's second identity. Here, the total surface, S, is the union of all the surfaces delineating D, i.e.

$$S = S_B \cup S_W \cup S_F \cup S_I \cup S_O \cup S_D \cup S_L \cup S_R.$$

In the next step, ϕ_1 and ϕ_2 are chosen such that

$$\phi_1 = \frac{1}{r} \qquad \text{and} \qquad \phi_2 = \phi,$$

where

$$\mathbf{R} = Q - P \qquad \text{and} \qquad r = \|\mathbf{R}\|.$$

$P = (x_p, y_p, z_p)$ is an arbitrary point within $\mathbb{R}^3$ (known as the field or collocation point) and $\mathbf{R}$ is the vector from P to $Q = (x_q, y_q, z_q)$, the integration point (or source point). Subsequently, equation (2.28) becomes

$$\int_S \left(\frac{1}{r} \nabla \phi - \phi \nabla \frac{1}{r} \right) \cdot \mathbf{n} \, dS = \int_D \left(\frac{1}{r} \nabla^2 \phi - \phi \nabla^2 \frac{1}{r} \right) dD. \tag{2.29}$$

Several limiting behaviours for equation (2.29) can now be demonstrated. First, consider a point, $P = P_0$, that lies external to the fluid domain. In this case r can not be zero for any point $Q \in D$ and thus both $\nabla^2(1/r)$ and $\nabla^2 \phi$ are always zero. Hence, equation (2.29) assumes the form

$$\int_S \left(\frac{1}{r} \nabla \phi - \phi \nabla \frac{1}{r} \right) \cdot \mathbf{n} \, dS = 0. \tag{2.30}$$

In the case of an internal point, such as $P = P_1$, r approaches zero as Q approaches P_1. To evaluate equation (2.29) for this case the singular point, P_1, must be isolated from the domain by surrounding it with a sphere of radius ϵ. As r is not zero–valued outside of the sphere surrounding P_1, equation (2.29) can now be written as

$$\int\limits_{S + \text{sphere } \epsilon} \left(\frac{1}{r} \nabla \phi - \phi \nabla \frac{1}{r} \right) \cdot \mathbf{n} \, dS = 0. \tag{2.31}$$

To integrate over the sphere surrounding P_1 a polar coordinate transformation is applied. The coordinate system for the sphere is transformed into a (r, θ, ψ) basis. Recasting equation (2.31) gives:

$$\int_{\text{sphere } \epsilon} \left(\frac{1}{r}\nabla\phi - \phi\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS + \int_S \left(\frac{1}{r}\nabla\phi - \phi\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS = 0. \tag{2.32}$$

As a result of the spherical coordinate system, on the sphere $\mathbf{n} = -\mathbf{e}_r$ and thus $\mathbf{n} \cdot \nabla\phi = -\partial\phi/\partial r$. Furthermore,

$$\nabla\frac{1}{r} = -\left(\frac{1}{r^2}\right)\mathbf{e}_r. \tag{2.33}$$

The integral over the sphere is now

$$\begin{aligned} &-\int_0^{2\pi}\int_0^{\pi}\left(\frac{1}{r}\frac{\partial\phi}{\partial r} + \frac{\phi}{r^2}\right) r^2 \sin(\theta)\, d\theta\, d\psi\bigg|_{r=\epsilon} \\ = &\left(-r\int_0^{2\pi}\int_0^{\pi}\frac{\partial\phi}{\partial r}\sin(\theta)\, d\theta\, d\psi - \int_0^{2\pi}\int_0^{\pi}\phi\sin(\theta)\, d\theta\, d\psi\right)\bigg|_{r=\epsilon}. \end{aligned} \tag{2.34}$$

Assuming ϕ and $\nabla\phi$ are well behaved at $P = P_1$, in the limit as $\epsilon \longrightarrow 0$ the first integral in the above expression vanishes and the second gives

$$\lim_{\epsilon \longrightarrow 0} \int_{\text{sphere } \epsilon} dS = -4\pi\phi(P_1). \tag{2.35}$$

Substituting equation (2.35) into equation (2.32) reveals that for the internal point

$$\phi(P_1) = \frac{1}{4\pi}\int_S \left(\frac{1}{r}\nabla\phi - \phi\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS. \tag{2.36}$$

A point is now chosen that lies on a smooth part of the boundary S of the volume D. Accordingly, the singular point can be isolated by a hemisphere. Thus, with similar working to that above, the result for point P_2, as seen in figure 2.3 part (b), is

$$\phi(P_2) = \frac{1}{2\pi}\int_S \left(\frac{1}{r}\nabla\phi - \phi\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS. \tag{2.37}$$

Should the point now lie on a sharp corner of S, such as the point $P = P_3$ (see figure 2.3, part (b)), the integral equation for that point is determined by the angle subtended by the portion of the sphere centred at P_3 which is internal to the fluid domain. This angle is known as the *solid angle*, α. Consequently, the integral equation for P_3 is given as

$$\alpha(P_3)\phi(P_3) = \int_S \left(\frac{1}{r}\nabla\phi - \phi\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS. \tag{2.38}$$

In order to determine α, the technique described by Xü (1992) is employed. Consider equation (2.38) applied to a constant potential $\phi = c$. Given ϕ will be unchanging in D the normal derivative of ϕ, $\nabla\phi \cdot \mathbf{n}$, will be zero. Hence, equation (2.38) becomes

$$\begin{aligned} & \alpha(P)\, c = c\int_S \left(-\nabla\frac{1}{r} \cdot \mathbf{n}\right)\, dS \\ \Rightarrow \qquad & \alpha(P) = -\int_S \nabla\frac{1}{r} \cdot \mathbf{n}\, dS. \end{aligned} \tag{2.39}$$

The r^{-1} kernel in equations (2.36), (2.37) and (2.38) is known as a Green's function, $G(P,Q)$. Using this notation, equation (2.38) can be rewritten for a generic point P as

$$\alpha(P)\phi(P) + \int_S \phi(Q)G_n(P,Q) - \phi_n(Q)G(P,Q)\, dS(Q) = 0, \tag{2.40}$$

where the normal derivative of G is

$$G_n = \nabla\left(\frac{1}{r}\right) \cdot \mathbf{n} = -\frac{\mathbf{n} \cdot \mathbf{R}}{r^3}\,. \tag{2.41}$$

When considering lifting surfaces in potential flow the circulation around the surface, inherent with the generation of lift, must be accounted for. For the flow to remain irrotational this circulation must be met with an equal and opposite circulation within the fluid domain so that the total circulation is unchanging. To achieve this a vortex sheet (surface S_W in figure 2.3) must be modelled downstream of the lifting surface. A potential flow solution including a vortex sheet is sometimes referred to as a *quasi-potential boundary element method* (Morino and Bernardini, 2001) as the irrotational vortex sheet must be excluded from the domain by assuming it is infinitesimally thin.

In a similar approach to that used by Katz and Plotkin (2001, chapter 3), consider the wake surface, S_W, as two separate surfaces, S_{W^+} and S_{W^-} (see figure 2.3 part (c)). For this geometry, equation (2.38) is rewritten

$$\begin{aligned} \alpha(P)\phi(P) = & \int_{S-S_W} \left(\frac{1}{r}\nabla\phi - \phi\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS + \\ & \int_{S_{W^+}} \left(\frac{1}{r}\nabla\phi - \phi\nabla\frac{1}{r}\right) \cdot \mathbf{n}^+\, dS + \int_{S_{W^-}} \left(\frac{1}{r}\nabla\phi - \phi\nabla\frac{1}{r}\right) \cdot \mathbf{n}^-\, dS. \end{aligned} \tag{2.42}$$

As the top and bottom wake surfaces must not move apart from each other, the normal derivative of ϕ must be continuous across them. As the normals on S_{W+} and S_{W-} are equal and opposite, i.e. $\mathbf{n}^+ = -\mathbf{n}^-$, when a unified normal is chosen, say $\mathbf{n} = \mathbf{n}^-$, the gradients of ϕ on either surface are also equal and opposite, i.e.

$$\nabla\phi \cdot \mathbf{n}^+ = -\nabla\phi \cdot \mathbf{n}^-. \tag{2.43}$$

Subsequently, equation (2.42) reduces to

$$\begin{aligned}\alpha(P)\phi(P) = \int_{S-S_W} \left(\frac{1}{r}\nabla\phi - \phi\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS \\ + \int_{S_{W+}} \left(\phi\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS - \int_{S_{W-}} \left(\phi\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS.\end{aligned} \tag{2.44}$$

By now considering the wake as one unified surface and by defining the doublet strength as $\mu = \phi^+ - \phi^-$ (where $^+$ and $^-$ indicate values on S_{W+} and S_{W-} respectively), the final governing equation for a lifting surface in potential flow is

$$\alpha(P)\phi(P) = \int_{S-S_W} \left(\frac{1}{r}\nabla\phi - \phi\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS + \int_{S_W} \mu \left(\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS. \tag{2.45}$$

In addition to developing the lifting formulation of the governing equations, attention must be given to the correct calculation of the solid angle. Applying the same technique used for equation (2.39), consider a constant potential c applied throughout the domain of equation (2.44). This gives,

$$\alpha(P)c = c\int_{S-S_W} -\left(\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS - c\int_{S_{W+}} \left(\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS + c\int_{S_{W-}} \left(\nabla\frac{1}{r}\right) \cdot \mathbf{n}\, dS. \tag{2.46}$$

Bringing the two sides of the wake together results in cancellation and equation (2.46) becomes

$$\alpha(P) = -\int_{S-S_W} \nabla\frac{1}{r} \cdot \mathbf{n}\, dS. \tag{2.47}$$

Although equation (2.47) is similar to equation (2.39) note that the surface of integration of the above equation is restricted to those panels *not on the wake*. Thus panels on the wake do not contribute to the solid angle.

Now that the governing boundary integral equations for lifting and non-lifting flows have been formed, it is useful to consider them in terms of the unknown quantities on the Neumann and Dirichlet boundaries in a similar style to Xü (1992, section 2.2). That is, the equations are rearranged so that the known quantities appear on the right hand side and the unknown quantities appear of the left hand side. Beginning by considering geometries without the inclusion of a wake, equation (2.40) for a point $P \in S_D$ can be written as

$$\begin{aligned}\iint_{S_N} \phi(Q) G_n(P,Q)\, dS(Q) - \iint_{S_D} \phi_n(Q) G(P,Q)\, dS(Q) = \\ -\alpha(P)\phi(P) - \iint_{S_D} \phi(Q) G_n(P,Q)\, dS(Q) + \iint_{S_N} \phi_n(Q) G(P,Q)\, dS(Q),\end{aligned} \tag{2.48}$$

and for a point $P \in S_N$,

$$\begin{aligned}\alpha(P)\phi(P) + \iint_{S_N} \phi(Q) G_n(P,Q)\, dS(Q) - \iint_{S_D} \phi_n(Q) G(P,Q)\, dS(Q) = \\ - \iint_{S_D} \phi(Q) G_n(P,Q)\, dS(Q) + \iint_{S_N} \phi_n(Q) G(P,Q)\, dS(Q)\,.\end{aligned} \tag{2.49}$$

For geometries including a wake surface, similar expressions can be written using equation (2.45). Thus, for $P \in S_D$ the equation becomes

$$\begin{aligned}\iint_{S_N} \phi(Q) G_n(P,Q)\, dS(Q) - \iint_{S_D} \phi_n(Q) G(P,Q)\, dS(Q) = \\ -\alpha(P)\phi(P) - \iint_{S_D} \phi(Q) G_n(P,Q)\, dS(Q) + \iint_{S_N} \phi_n(Q) G(P,Q)\, dS(Q) \\ + \iint_{S_W} \mu(Q) G_n(P,Q)\, dS(Q),\end{aligned} \tag{2.50}$$

and for $P \in S_N$,

$$\begin{aligned}\alpha(P)\phi(P) + \iint_{S_N} \phi(Q) G_n(P,Q)\, dS(Q) - \iint_{S_D} \phi_n(Q) G(P,Q)\, dS(Q) = \\ - \iint_{S_D} \phi(Q) G_n(P,Q)\, dS(Q) + \iint_{S_N} \phi_n(Q) G(P,Q)\, dS(Q) \\ + \iint_{S_W} \mu(Q) G_n(P,Q)\, dS(Q)\,.\end{aligned} \tag{2.51}$$

Equations (2.48) and (2.50) are Fredholm integral equations of the first kind and equations (2.49) and (2.51) are Fredholm integral equations of the second kind. The order of the kernels are weakly singular for G, i.e. $\mathbf{O}(r^{-1})$, and strongly singular for G_n, i.e. $\mathbf{O}(r^{-2})$ (see Guiggiani, 1998). Xü (1992, appendix. A) proves that, for a smooth C^{∞}-continuous surface, the strong singularity can be reduced to that of a weak singularity via means of a Taylor expansion, although this is not applicable at all points within a piecewise continuous surface. This difficulty is of particular interest to the present analysis.

2.4.3 Uniqueness of Solution and the Wake Strength

The uniqueness of solutions to lifting problems in potential flow is, in part, dependent on the connectivity of the domain to be evaluated. In two-dimensional lifting cases, domains of interest are always multiply connected and in such cases the uniqueness of the solution is dependent on a physical representation of the circulation generated about a wing (see Katz and Plotkin, 2001, page 32). In the case of simple bodies in a three-dimensional domain, the domain will be simply connected (a doughnut shaped body in three dimensions would result in a multiply connected flow domain) and thus it is easily shown (see Katz and Plotkin, 2001, page 30) that the solution is unique to within a constant and as the gradient of ϕ is the only quantity of interest, this is sufficient. However, as Katz and Plotkin point out, greater care must be taken when a wake is present in a three-dimensional potential flow field.

Morino and Bernardini (2001) point out the obvious lack of uniqueness in a lifting solution that would arise from not determining the doublet strength, μ, on the wake surface. The boundary integral formulation described by equations (2.50) and (2.51), is formulated in such a way as the wake strength, μ is almost an arbitrary value. Therefore, some physical considerations are required to set the wake strength.

The aim of Morino and Bernardini's 2001 paper was to determine the necessary conditions for a unique solution when formulating the doublet strength on the trailing edge. The three properties that they investigated were the Kutta-Joukowski theorem, the numerical Kutta condition, and the spatial Kutta condition (all introduced with the boundary conditions in section 2.4.1).

The Kutta-Joukowski theorem is interpreted in numerous ways by various authors. Morino and Bernardini (2001) choose to interpret it as two separate conditions whilst maintaining that they remain related to each other. Morino describes the Joukowski conjecture as the necessity of smooth, non-infinite, flow at the sharp trailing edge of an aerofoil, while the Kutta-condition is a numerical 'no pressure difference' condition, or

$$[p] = 0, \tag{2.52}$$

where '[ ]' denotes the difference between the top and bottom of the wake and p is the pressure.

Combining equation (2.52) with the Bernoulli equation provides an extremely useful result. In a similar manner to the result proved by Morino and Bernardini (2001) and Politis (2004), consider the unsteady Bernoulli equation at the trailing edge, given by

$$\frac{\partial \phi^+}{\partial t} + \frac{1}{2}\|\mathbf{v}^+\|^2 + \frac{p^+}{\rho} = \frac{\partial \phi^-}{\partial t} + \frac{1}{2}\|\mathbf{v}^-\|^2 + \frac{p^-}{\rho}\ . \tag{2.53}$$

Rearranging, it can be seen that

$$\frac{\partial \phi^+}{\partial t} - \frac{\partial \phi^-}{\partial t} + \frac{1}{2}\left(\|\mathbf{v}^+\|^2 - \|\mathbf{v}^-\|^2\right) + \frac{p^+}{\rho} - \frac{p^-}{\rho} = 0\ . \tag{2.54}$$

Utilising equation (2.52) and by observing that

$$\left(\|\mathbf{v}^+\|^2 - \|\mathbf{v}^-\|^2\right) = \left(\mathbf{v}^+ + \mathbf{v}^-\right) . \left(\mathbf{v}^+ - \mathbf{v}^-\right),$$

equation (2.54) becomes

$$\frac{\partial \phi^+}{\partial t} - \frac{\partial \phi^-}{\partial t} + \frac{1}{2}\left(\mathbf{v}^+ + \mathbf{v}^-\right) . \left(\mathbf{v}^+ - \mathbf{v}^-\right) = 0\ . \tag{2.55}$$

Now define the wake material derivative, D_W/Dt to be a material derivative with the velocity chosen to be the average of the velocity of the top and bottom of the wake, i.e.,

$$\frac{D_W}{Dt} = \frac{\partial}{\partial t} + \mathbf{v}_W \cdot \nabla \qquad \text{where} \tag{2.56}$$

$$\mathbf{v}_W = \frac{1}{2}\left(\mathbf{v}^+ + \mathbf{v}^-\right). \tag{2.57}$$

Rewriting equation (2.55) in terms of D_W/Dt then reveals

$$\frac{D_W}{Dt}\Big([\phi]\Big) = \frac{D_W}{Dt}\Big(\mu\Big) = 0. \tag{2.58}$$

The significance of equation (2.58) is that μ is a material property of the wake. That is, once a value of μ is known at a point on the wake surface, that value is transported along the wake with the wake velocity (i.e. the average fluid velocity on the top and bottom of the wake surface).

All that remains in order to guarantee uniqueness is the value of μ at the trailing edge, where the shed vorticity is generated, and the orientation of the vortex sheet as it separates from the aerofoil. In a time variant problem, this short section of the wake is known as the *Kutta strip*.

2.4.4 The Numerical and Spatial Kutta Condition

The Kutta condition describes the numerical implementation of the Kutta-Joukowski theorem. As Katz and Plotkin (2001) argued, unless the circulation about the wing is physically determined, the solution will not be unique. In order to maintain the irrotational assumption in the potential formulation, the circulation about the wing is balanced by an equal and opposite circulation in the wake. The circulation in the wake is manifest through the doublet strength and this quantity is transported with the wake as a material property. Thus, the doublet strength need only be calculated at the trailing edge of the aerofoil which, in turn, discerns the circulation about the aerofoil. To achieve this the Kutta-Joukowski theorem is applied.

Although equation (2.52) is relatively simple, its numerical application in potential flow is certainly not. A number of formulations and equations have been developed since Morino and Kuo (1974) first presented their well known formulation. They postulated that the no pressure difference condition could be implemented by relating the wake strength at the trailing edge to the difference in ϕ between the top and bottom surfaces of the aerofoil at the trailing edge, i.e

$$[\phi]_{\mathrm{TE}} = \phi_{\mathrm{upper}} - \phi_{\mathrm{lower}} \tag{2.59}$$

where $[\phi]_{\mathrm{TE}}$ is the value of $[\phi]$ on the wake at the trailing edge and ϕ_{upper} and ϕ_{lower} are the values of ϕ on the upper and lowers surfaces of the aerofoil at the trailing edge. Obviously, $[\phi]$ in equation (2.59) is identical to the doublet strength on the wake, μ. Thus, equation (2.59) provides both uniqueness and a means for calculating the strength of the wake sheet at the trailing edge. This *Morino type* Kutta condition is not the only available formulation and, indeed, may not provide an entirely accurate implementation of equation (2.52). This fact is illustrated by Kinnas and Hsin (1992), who chose to implement a more accurate scheme. However, from a practical point of view, the Morino type condition is much easier to implement than an iterative scheme or a scheme which explicitly limits velocity. Thus the Morino type Kutta condition is chosen for the present simulation. For the interested reader, a historical review of the Kutta condition and some of the alternative methods are presented in appendix A.

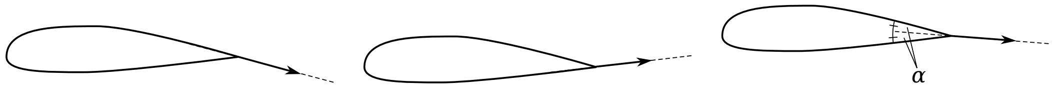
Prudence is still advisable when implementing equation (2.59). Unlike the constant panel methods used by Morino and Kuo when they formulated their Kutta condition, modern high order panel methods will collocate at the panel edges and, as such, allow the possibility for two coincident nodes to be placed at the trailing edge. It could be theorised that these nodes would then (without explicit continuity of ϕ) provide the best numerical representation of equation (2.59).

To disprove this theory, consider the well known fact that there exists a singularity in the solution at sharp corners (beyond a reflex dihedral angle of $\alpha > \pi$) and, in fact, from a mathematical perspective, it is this singularity that the Kutta condition seeks to cancel (this type of singularity is sometimes attributed to Kondrat'ev and Oleinik (1983)). The issue with attempting to resolve the strength of this singularity coincident to the trailing edge is that, as Bassanini *et al.* (1996) point out, the singularity exists *only in the limit* to the trailing edge. Therefore, an attempt to extract the solution from coincident nodes at the trailing edge is fruitless.

Ardito Marretta *et al.* (1999) have solved this problem for higher order schemes by using a similar approach to constant panel formulations. They applied the Kutta condition to the computational nodes immediately adjacent to the trailing edge rather than those on it. This approach would indicate that there is little to gain from using coincident trailing edge nodes to implement the Morino type Kutta condition.

In the theoretical implementation of the Morino Kutta condition, equation (2.50) and equation (2.51) would remain unchanged and the additional equation (2.59) solved latterly. In practical terms, however, due to the spatial discretisation, not all of S_W can remain of the left hand side of the equations and thus a strip of the wake, known as the Kutta strip, S_K, must be determined as part of the linear system using equation (2.59) directly. This approach is covered in greater detail within chapter 3.

Supplying the wake strength at the trailing edge is still not sufficient to render the



(a) Tangent to aerofoil top. **(b)** Tangent to aerofoil bottom. **(c)** Tangent to bisecting angle.

Figure 2.4: Alternative orientations of the wake vortex sheet as it separates from the trailing edge of the aerofoil.

Kutta condition completely satisfied. Equally important is the direction at which the wake sheet leaves the aerofoil (in particular, this removes the Kondratev and Oleinik singularity). Often, when thick lifting bodies are simulated with boundary element methods, the wake sheet is chosen to leave at the bisecting angle of the trailing edge. However, Mangler and Smith (1970), Bassanini *et al.* (1999) and Morino and Bernardini (2001) all suggest that this approach does not fully represent the physics. They point out that when fully removing the singularity from the trailing edge, the wake sheet is rarely positioned at the trailing edge angle bisector. Indeed, it is more likely to be tangent to one of the aerofoil surfaces. The alternative orientations for the wake sheet leaving the aerofoil trailing edge are shown in figure 2.4.

Mangler and Smith (1970) deduced that the wake will be tangent to either the top or bottom lifting surface dependent on the planform of the wing and the velocity tangential to trailing edge in the spanwise direction. Although Mangler and Smith's formulation was intended for steady flows, Bassanini *et al.* (1999) showed that similar conditions hold for lifting surfaces in unsteady flows. Thus, the wake sheet shedding at the bisecting angle, beyond very special cases, turns out to be non-physical. Still, the ease of implementing separation at the bisecting angle is highly attractive and thus it was decided that the added complication of implementing the Mangler and Smith's approach would not be beneficial to this study. The spatial Kutta condition is considered in further detail in appendix A.

2.4.5 Initial Conditions for Mixed Eulerian-Lagrangian Time-Stepping

The initial conditions for the formulation described in the previous section, excluding the wake surface are described in Xü (1992) by

$$\phi(\mathbf{x},0) = f_0(\mathbf{x}), \qquad \mathbf{x} \in S_D, \qquad f_0 \in C^0(S_D), \tag{2.60}$$

$$\phi_n(\mathbf{x},0) = g_0(\mathbf{x}), \qquad \mathbf{x} \in S_N, \qquad g_0 \in C^0(S_N), \tag{2.61}$$

where f_0 and g_0 are given. Additionally, the free surface geometry is chosen, nominally at rest. Also, any submerged bodies must be positioned within the fluid domain.

For lifting simulations at $t = 0$, a lifting surface with an attached *starting wake* is included where the starting wake leaves the trailing edge aligned with the bisecting angle. Equation (2.61) also holds for the lifting body and the initial condition on its wake is given by

$$\mu(\mathbf{x}, 0) = 0, \qquad \mathbf{x} \in S_W. \tag{2.62}$$

This condition ensures that, for higher order, the edges of the wake do not have a non-physical jump in potential. It also provides additional advantages for the calculation of the self-induced velocity on the wake as shall be discussed in chapter 4. Note that this condition does not apply to the Kutta nodes (those nodes on the wake at the trailing edge) as the values there shall be calculated as part of the solution.

With the boundary and initial conditions given, a mixed Eulerian-Lagrangian approach (Xü, 1992; Longuet-Higgins and Cokelet, 1976) is used to update the free surface and wake positions. As Xü (1992) described, once all of the velocity components on the free surface are calculated, equation (2.23) can be integrated to find the free surface position at time $t+\nabla t$. To find the the new potential value for $\phi(\mathbf{x}, t+\nabla t)$ on $S_F(t+\nabla t)$ equation (2.27) is integrated. Similarly the wake geometry is updated by integrating equation (2.57) to give the new geometry at time $t + \nabla t$. Due to equation (2.58) the value of $\mu(\mathbf{x}, t)$ on $S_W(t)$ is transported to the new position on $S_W(t + \nabla t)$ unaltered. The entire process is repeated for each time step.

Xü enforced that the key issues for implementing this process are the accurate calculation of ϕ and μ and, subsequently, the accurate calculation of the surface velocities. Of particular interest to this study is the calculation of the self induced wake velocity when using a high order piecewise discretisation.

2.5 Perturbation Potential Formulation of the Governing Equations

An alternative approach to formulating the governing integral equations is known as the *perturbation potential* formulation. This formulation is particularly useful for bodies submerged in a fluid of infinite domain as this approach assumes there is a constant free stream velocity which is applied at an imaginary surface at infinity, S_∞. This removes the need to discretise the boundaries in order to generate fluid flow.

The premise is to separate the whole potential into the separate contributions from the flow at infinity and the perturbation to that flow caused by the modelled surfaces within the computational domain. To demonstrate this, redefine the full potential as ϕ^* and recall that this full potential satisfies the boundary conditions given by equations (2.23) and (2.27) on the free surface, equations (2.21) and (2.22) for submersed bodies and equations (2.57) and (2.58) on the wake. In addition, the perturbation potential, ϕ, is related to the full potential by

$$\phi^* = \phi + \phi_\infty \tag{2.63}$$

where ϕ_∞ is the potential at infinity.

Assuming a constant flow at infinity having magnitude U in the x direction, such that $\mathbf{U} = (U, 0, 0)$, the potential at infinity is given by

$$\phi_\infty = Ux. \tag{2.64}$$

Substituting equation (2.64) into equation (2.63) gives

$$\phi^* = \phi + Ux. \tag{2.65}$$

As the flow at infinity is now superimposed throughout the domain, the Neumann no-crossing boundary condition on a stationary body is altered to

$$\phi_n = -\mathbf{U} \cdot \mathbf{n}. \tag{2.66}$$

In addition, for the channel illustrated in figure 2.1, surfaces S_I and S_O no longer form part of the solution. Indeed, perturbation potential simulations can be performed with just the S_B and S_∞ boundaries with the addition of S_W for a lifting problem. Such a geometry would describe a traditional *Neumann problem* and is useful for verification tests as well as testing the wake model in the absence of the free surface (see chapter 6). For simulating a channel using perturbation potential, the surfaces S_F, S_D, S_L and S_R must also be included.

The free surface boundary conditions are also altered in perturbation potential as a consequence of equation (2.65). The kinematic boundary condition, given in full potential by equation (2.23), becomes

$$\frac{D\mathbf{x}}{Dt} = \nabla\left(\phi + Ux\right) = \nabla\phi + \mathbf{U} \qquad \mathbf{x} \in S_F. \tag{2.67}$$

Now consider the effect of substituting equation (2.65) into the dynamic boundary condition given by equation (2.27). Maintaining the frame of reference, the material derivative is given by

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \nabla\phi^* \cdot \nabla. \tag{2.68}$$

Expanding equation (2.27) in terms of perturbation potential, the left hand side becomes

$$\begin{aligned}\frac{D\phi^*}{Dt} &= \frac{\partial\phi^*}{\partial t} + \nabla\phi^* \cdot \nabla\phi^* \\ &= \frac{\partial\phi + Ux}{\partial t} + \nabla\phi^* \cdot (\phi + Ux) \\ &= \frac{\partial\phi}{\partial t} + \nabla\phi^* \cdot \phi + U\frac{\partial\phi}{\partial x} + U^2 \\ &= \frac{D\phi}{Dt} + U\frac{\partial\phi}{\partial x} + U^2.\end{aligned} \tag{2.69}$$

Subsequently, the right hand side is given by

$$\begin{aligned}\frac{1}{2}\left|\nabla\phi^*\right|^2 - gz &= \frac{1}{2}\left(\left(\frac{\partial(\phi + Ux)}{\partial x}\right)^2 + \left(\frac{\partial\phi}{\partial y}\right)^2 + \left(\frac{\partial\phi}{\partial z}\right)^2\right) - gz \\ &= \frac{1}{2}\left(\left(\frac{\partial\phi}{\partial x}\right)^2 + \left(\frac{\partial\phi}{\partial y}\right)^2 + \left(\frac{\partial\phi}{\partial z}\right)^2 + 2U\frac{\partial\phi}{\partial x} + U^2\right) - gz \\ &= \frac{1}{2}\left|\nabla\phi\right|^2 + U\frac{\partial\phi}{\partial x} + \frac{1}{2}U^2 - gz.\end{aligned} \tag{2.70}$$

Bringing the left and right hand sides together and noting that because U^2 is invariant in time it will not influence $\nabla\phi^*$, the perturbation potential dynamic boundary condition is given by

$$\frac{D\phi}{Dt} = \frac{1}{2} |\nabla\phi|^2 - gz \qquad \mathbf{x} \in S_F. \tag{2.71}$$

For the wake boundary conditions, equation (2.57) is altered slightly to include the free stream velocity. The perturbation potential form of the boundary condition is

$$\mathbf{v}_W = \frac{1}{2}\left(\mathbf{v}^+ + \mathbf{v}^-\right) + \mathbf{U}. \tag{2.72}$$

Equation (2.58) is unaltered; the same quantity added to ϕ on the top and bottom of the wake will not alter $[\phi] = \mu$.

The use of perturbation potential provides an additional advantage over the full potential formulation because the altered free surface boundary conditions provide additional numerical accuracy. This is a consequence of the full potential incorporating the, potentially large, potential at infinity. This may then be susceptible to truncation error, eliminating small variations of ϕ^* on the free surface. In contrast, the perturbation potential will only give the variation of the free surface potential from the potential at infinity and thus small variations will not be disguised. Even with this caveat, it should be noted that the full potential solution provides a useful check on the results of the perturbation potential formulation.

Equations (2.50) and (2.51) or (2.48) and (2.49) will continue to provide the governing equations for the perturbation potential with the full potential exchanged with the perturbation potential. Similarly, the initial conditions take a similar form to that of the full potential formulation.

An additional consideration for the perturbation potential is the value of the solid angle with the use of the boundary at infinity. For simulations using S_∞, the solid angle contribution from this surface is simply 4π, thus the total solid angle is given by

$$\alpha(P) = -\int_{S_B} \nabla\frac{1}{r} \cdot \mathbf{n}\, dS + 4\pi, \tag{2.73}$$

recalling that the wake (should it be present) continues not to contribute.

For perturbation potential within a channel S_I and S_O do not contribute to the solution. However, they must still be included within the integration domain of equation (2.39). For a simple geometry, such as a rectangular inlet/outlet, this can be achieved analytically, provided the free surface potential is zero. Otherwise an analytical solution to a strong singularity is required, which is far from trivial. In addition, for a variable geometry, say, if the free surface were to be allowed to deform there, the analytical result would be again more complicated. Thus, in general the inlet and outlet boundaries are discretised even in the perturbation potential case.

2.6 The Hypersingular Boundary Integral Equation

In order to advance time dependent boundary conditions and to examine quantities of interest within the flow field, it is important to be able to determine the velocity both within the volume and on the surfaces bounding the domain. Within the volume, away from surfaces, equation (2.36) can be analytically differentiated to provide a formula for the result. The gradient of equation (2.36) in the principal directions (x, y, z) is sought. Following the notation of Gray *et al.* (2004a), these derivatives are represented by $d/d\mathbf{E}_k$, where $\mathbf{E}_k$, for $k = 1, 2$ and 3, is the unit vector in the x, y and z directions, respectively. Differentiating equation (2.36) produces

$$\begin{aligned} \frac{\partial \phi(P)}{\partial \mathbf{E}_k} &= \frac{1}{4\pi} \int_S \nabla \left[\left(\frac{1}{r} \right) \phi_n - \phi \left(\frac{\mathbf{n} \cdot \mathbf{R}}{r^3} \right) \right] \cdot \mathbf{E}_k \, dS \\ &= \frac{1}{4\pi} \int_S \frac{-\mathbf{E}_k \cdot \mathbf{R}}{r^3} \phi_n - \phi \left(\frac{\mathbf{n} \cdot \mathbf{E}_k}{r^3} - 3 \frac{(\mathbf{n} \cdot \mathbf{R})\,(\mathbf{E}_k \cdot \mathbf{R})}{r^5} \right) \, dS. \end{aligned} \tag{2.74}$$

Equation (2.74) is valid for all points $P \in D \backslash S$ and is non singular. Hence, no analytical manipulation of the equation is required and the accuracy of points nearby to S can be maintained via adaptive numerical schemes.

Three alternative approaches are available for calculating the gradient of ϕ at points $P \in S$. The first is numerical differentiation; numerical differentiation schemes use backwards, forwards or central differences to calculate the gradients on surfaces. On surfaces such as the free surface, numerical differentiation is perfectly adequate and is discussed in chapter 4. Difficulties with implementing numerical differentiation arise when there are either too few points available or when a gradient is required at the very edge of a surface (as skewed differencing schemes are less accurate). Unfortunately, when wakes are considered and just the doublet strength, μ, is known, numerically differentiation will not suffice. This results in the gradient of the difference in ϕ between the top and bottom wake surface being found, rather than the *average* of the gradients of ϕ on the top and bottom surfaces as required by equation (2.58).

An alternative approach for determining velocities on a surface is to directly differentiate equation (2.38) applied to a general point, $P \in S$. This results in

$$\alpha(P)\frac{\partial \phi(P)}{\partial \mathbf{E}_k} = \int_S \frac{-\mathbf{E}_k \cdot \mathbf{R}}{r^3}\phi_n - \phi\left(\frac{\mathbf{n}\cdot\mathbf{E}_k}{r^3} - 3\frac{(\mathbf{n}\cdot\mathbf{R})\,(\mathbf{E}_k\cdot\mathbf{R})}{r^5}\right)\,dS. \qquad (2.75)$$

The difficulties with evaluating equation (2.75) stem from the $\mathbf{O}(r^{-3})$ (denoting order) terms multiplying ϕ. In this form, a direct solution to the equation requires that the surface be smooth everywhere (see Guiggiani, 1998).

The third method, and that chosen herein, is an alternative limiting procedure favoured by Gray *et al.* (2004a,b). Their approach was to use the 'off-surface' integral equation, given by equation (2.36), and take the limiting value as the point approaches the surface. Returning to G-notation, this modifies equation (2.38) so that

$$\phi(P) = \lim_{P_I \to P}\frac{1}{4\pi}\int_S G\phi_n - \phi G_n\,dS, \qquad (2.76)$$

where P_I is a point interior to the domain and P is a point on a S. Subsequently, taking the gradient of equation (2.76) produces

$$\frac{\partial \phi(P)}{\partial \mathbf{E}_k} = \lim_{P_I \to P}\frac{1}{4\pi}\int_S \frac{\partial G}{\partial \mathbf{E}_k}\phi_n - \phi\frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}\,dS. \qquad (2.77)$$

Notice the lack of solid angle in equations (2.76) and (2.77) as the 4π persists in the limit as P_I approaches P.

The r^{-3} type singularity in equations (2.75) and (2.77) is known as *hypersingular* and, as such, the equations are known as *hypersingular boundary integral equations*. It is now possible to demonstrate how this limiting approach can be applied to the wake vortex sheet.

To begin, assume that the velocity on all surfaces bounding D excluding S_W is known. The velocity on the wake surface, S_W, is sought, given by equation (2.57). Essentially, the average of the velocity in the limit to the top and bottom of the wake surface is required. The equation to provide this velocity shall be developed in a similar manner to equation (2.45).

Consider the wake as two separate surfaces as seen in figure 2.5. In this situation there are a number of different approaches to taking the limits to the wake and subsequently unifying surfaces S_W^+ and S_W^-. One approach is to examine the limiting

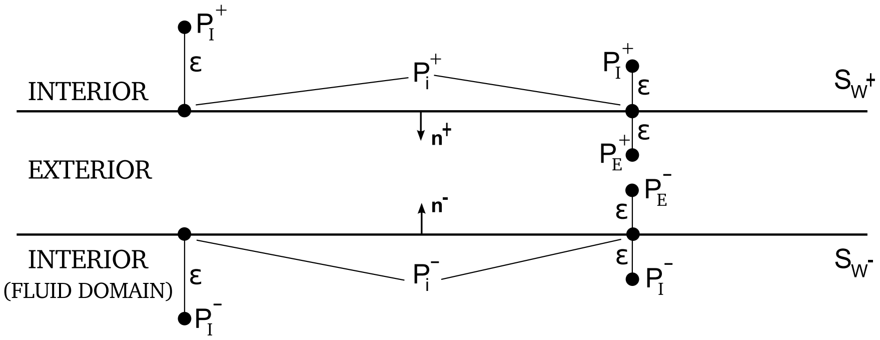


Figure 2.5: Limits on Wake

behaviour of each side independently and then bring the two sides together. This process is demonstrated in greater detail hereafter.

For the limit to the upper surface S_W^+, define an internal and an external point, P_I^+ and P_E^+, respectively. Given that P_I^+ is internal to the flow the equation for the velocity as $P_I^+ \to P^+$ is identical to equation (2.77), and therefore

$$\frac{\partial \phi(P^+)}{\partial \mathbf{E}_k} + \lim_{P_I^+ \to P^+} \frac{1}{4\pi} \int_S \phi(Q) \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}(P_I^+, Q) - \frac{\partial G}{\partial \mathbf{E}_k}(P_I^+, Q)\phi_n(Q)\, dQ = 0. \quad (2.78)$$

The limit to P^+ from the exterior point, P_E^+, is found by differentiating equation (2.30). Applying the limit gives

$$\lim_{P_E^+ \to P^+} \int_S \phi(Q) \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}(P_E^+, Q) - \frac{\partial G}{\partial \mathbf{E}_k}(P_E^+, Q)\phi_n(Q)\, dQ = 0. \quad (2.79)$$

With the interior and exterior limits established, Gray *et al.* (2004b) employ an innovative technique to reduce the integration to a few panels local to the point of interest. The first step of the technique is to subtract the external limit, equation (2.79), from the internal limit, equation (2.78), to give

$$\frac{\partial \phi(P)}{\partial \mathbf{E}_k} = \left\{ \lim_{P_I \to P} - \lim_{P_E \to P} \right\} \frac{1}{4\pi} \int_S \frac{\partial G}{\partial \mathbf{E}_k} \phi_n(Q) - \phi(Q) \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}\, dQ \quad (2.80)$$

where the '+' superscript has been dropped for convenience. This process reduces the integration to include only singular parts, thus reducing the computational effort.

In an identical manner, a second equation can be formed for the other side of the wake. Using these two equations the average velocity velocity can be found, provided a suitable limiting technique is used to unify S_W^+ and S_W^-. This is an integral process to the regularisation of these equations.

Regularisation is required as equation (2.80) does not reduce the order of the hypersingular integral enough for it to be solved numerically. For this, a Galerkin boundary element formulation must be used. The full derivation of these equations forms part of chapter 4. In order to apply this technique, the formulation of both collocation and Galerkin discretisation of the boundary integral equations is reviewed.

2.7 The Method of Weighted Residuals

With the fundamental mathematics defined, the question of discretisation of the equations must be addressed. A number of common methods for discretising boundary integral equations exist, and, as Brebbia (1978) showed, they can be related and categorised by considering the *method of weighted residuals*. To describe this method, consider an integral equation given by

$$\mathcal{L}\Upsilon(\mathbf{x}) - f(\mathbf{x}) = 0 \tag{2.81}$$

where $\mathcal{L}$ is an integral operator and $\Upsilon(\mathbf{x})$ is the exact solution to the function $f(\mathbf{x})$. This solution may be approximated by a combining a complete series chosen from a set of linearly independent functions (called basis, shape or interpolation functions) given by $\psi_j(\mathbf{x})$. This approximation is written as

$$\Upsilon \approx \upsilon(\mathbf{x}) = \sum_{j=1}^{n} \beta_j \psi_j. \tag{2.82}$$

where β are unknown coefficients which must be determined. Subsequently, substitution of equation (2.82) into equation (2.81) will result in an error or *residual*, $\mathcal{R}(\mathbf{x})$, of the form

$$\mathcal{R}(\mathbf{x}) = \mathcal{L}\left(\sum_{j=1}^{n} \beta_j \psi_j\right) - f(\mathbf{x}) \neq 0. \tag{2.83}$$

The principle behind the weighted residual method is to force $\mathcal{R}$ to zero by means of a weighted average over the domain. This can be written as

$$\iint_S \mathcal{R} w_j \, dS = \iint_S \left[\mathcal{L}\upsilon - f\right] w_j \, dS = 0 \tag{2.84}$$

where w_j are a set of weighting functions. Equation (2.84) provides n algebraic equations for the unknown constants, β_j, and thus provides Υ.

Brebbia showed that the method of weighted residuals could be applied directly to the Laplace equation in order to form the boundary integral equations, utilising the Green's function as the weighting. However, the primary interest here is how the weighted residual method may be used to describe the discretisation of the boundary integral equations.

The method of weighted residuals has five common sub-methods, dependent on which weightings are chosen. These are known as collocation, sub-domain, least-squares, Galerkin and the method of moments. The methods applicable to the present study are the collocation method (for the ϕ boundary integral) and the Galerkin method (for the hypersingular velocity equation). These are now discussed in turn.

2.7.1 The Collocation Method

For the *collocation method* the weighting function, w_j, is chosen to be the Dirac delta function given by

$$\delta(\mathbf{x} - \mathbf{x}_j) = \begin{cases} 1 & x = x_j, \\ 0 & \text{otherwise}. \end{cases} \tag{2.85}$$

This weighting provides the most common boundary element discretisation and is equivalent to solving equations (2.50) and (2.51) using interpolation or shape functions. If $\psi_j(\mathbf{x})$ are globally defined functions over S then the method is known as a *global method*; if $\psi_j(\mathbf{x})$ are defined over piecewise smooth subregions of S then the method is known as a *local method*. In a similar approach to Xü (1992), the standard boundary integral equations are solved using local piecewise smooth approximating functions, allowing greater flexibility in the geometry and the boundary conditions.

Following Xü (1992, section 2.3), divide S_D, S_N and S_W into piecewise smooth elements defined by

$$S_D = \bigcup_{m=1}^{M_D} \Sigma_m, \tag{2.86}$$

$$S_N = \bigcup_{m=1}^{M_N} \Sigma_m, \tag{2.87}$$

$$S_W = \bigcup_{m=1}^{M_W} \Sigma_m, \tag{2.88}$$

where the total number of elements is $M = M_D + M_N + M_W$ and Σ_m is a generic element. The discretised versions of equations (2.48) and (2.49) are

$$\sum_{m=1}^{M_N} I_d - \sum_{m=1}^{M_D} I_s = -\alpha(P)\phi(P) - \sum_{m=1}^{M_D} I_d + \sum_{m=1}^{M_N} I_s \qquad P \in S_D \tag{2.89}$$

and

$$\alpha(P)\phi(P) + \sum_{m=1}^{M_N} I_d - \sum_{m=1}^{M_D} I_s = -\sum_{m=1}^{M_D} I_d + \sum_{m=1}^{M_N} I_s \qquad P \in S_N. \tag{2.90}$$

In addition, equations (2.50) and (2.51) can now be written as

$$\sum_{m=1}^{M_N} I_d - \sum_{m=1}^{M_D} I_s = -\alpha(P)\phi(P) - \sum_{m=1}^{M_D} I_d + \sum_{m=1}^{M_N} I_s + \sum_{m=1}^{M_W} I_w \qquad P \in S_D \tag{2.91}$$

and

$$\alpha(P)\phi(P) + \sum_{m=1}^{M_N} I_d - \sum_{m=1}^{M_D} I_s = -\sum_{m=1}^{M_D} I_d + \sum_{m=1}^{M_N} I_s + \sum_{m=1}^{M_W} I_w \qquad P \in S_N. \tag{2.92}$$

The terms appearing in equations (2.89)–(2.92) are defined by

$$I_s = \iint_{\Sigma_m} \phi_n(Q) G(P,Q)\, dS(Q), \tag{2.93}$$

$$I_d = \iint_{\Sigma_m} \phi(Q) G_n(P,Q)\, dS(Q), \tag{2.94}$$

$$I_w = \iint_{\Sigma_m} \mu(Q) G_n(P,Q)\, dS(Q), \tag{2.95}$$

and Xü refers to these as the *elemental integrals*. The finite part of the singular integral is taken if $P \in \Sigma_m$.

As Xü (1992) expressed, at this point an approximation to the piecewise generic elements has yet to be described. Given this, all of the S_W elements remain on the right hand side of equations (2.91) and (2.92) while equation (2.59) does not, as yet, form part of the solution procedure.

2.7.2 The Galerkin Method

The *Galerkin method* is applied by Gray *et al.* (2004b) to desingularise the hypersingular boundary integral equation and thus facilitate direct calculation of velocity on the wake. The method uses the local element's shape functions as its weighting functions, i.e. $w_m = \psi_m$.

In a similar way to the collocation method, the Galerkin method facilitates the use of piecewise smooth elements to approximate the surface, S. However, an additional surface integration is required which re-poses the hypersingular boundary integral equation as

$$\int_S \hat{\psi}_j(P) \frac{\partial \phi(P)}{\partial \mathbf{E}_k}\, dP = \frac{1}{4\pi} \int_S \hat{\psi}_j(P) \int_S \frac{\partial G}{\partial \mathbf{E}_k} \phi_n(Q) - \phi(Q) \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}\, dQ\, dP. \tag{2.96}$$

This formulation requires that the left hand side of the equation be integrated over the panels which 'support' the point P (the non-zero shape functions local to P are denoted $\hat{\psi}$). Additionally, there is a further surface integration to be carried out on the right hand side. This allows the difficult singularities to be analytically reduced to a greater extent than is feasible for a collocation method. This advantage is exploited by Gray *et al.* and the application of this technique to the wake is shown in chapter 4.

Chapter 3

Eulerian Stage Implementation

3.1 Introduction

The difficulties of developing an accurate, unified, potential flow method for solving the governing equations developed in chapter 2 are extensive. A dynamic free surface in close proximity to submerged lifting bodies belies the use of simplistic constant and linear approximations. It is well known that accurate free surface simulations using boundary integral formulations require higher order elements. Indeed, it is generally accepted that higher order elements are both desirable and more efficient in boundary element methods than those of zeroth and first order.

Although higher order solutions are prominent in free surface wave simulations using boundary integrals, their presence in lifting solutions (particularly dynamic simulations) is much more sparse. However, the advantages of a higher order approach have been recognised in the existing literature (see Pyo and Kinnas, 1997).

One inherent difficulty, which may explain the lack of high order methods applied to this problem, is the accurate calculation of the hypersingular boundary integral equation, necessary to determine the self induced velocity on surfaces such as vortex wakes. For the free surface this problem has been circumvented using numerical differentiation, rather than solving the equations directly. Repeating this approach on a vortex wake is not possible as the conserved quantity on the wake, μ, does not provide the average velocity when differentiated.

Solving the hypersingular boundary integral equation at the edges of higher order elements is the goal of this thesis and is discussed in detail in chapter 4. The present chapter is concerned with developing a high-order boundary element method to which the new solution technique can be applied. The application of a high accuracy differentiating scheme requires a reliable, transparent and well respected high order boundary element method to produce accurate solutions to the general boundary problem described in chapter 2. The method chosen is the non-linear free surface solver of Xü (1992).

The method of Xü may not be the most modern (with regards to its speed of computation), however it is accurate and relatively simple to reproduce. In Xü (1992, chap. 3) there are a number of numerical experiments which compare various super- and iso-parametric elements. It was shown that the constant panel method did not have a desirable level of accuracy for a mixed Neumann-Dirichlet problem, and also that linear iso-parametric elements did not yield accurate enough velocities on the free surface. Although B-spline approximations provided the most accurate solutions, they were found to be too computationally costly. Thus, a biquadratic iso-parametric element was chosen by Xü and the method is faithfully recreated herein with some alterations to the published adaptive integration scheme (see section 3.3).

In addition to reproducing the scheme for the biquadratic elements, it was necessary to produce a similar scheme for the linear elements that would be used on the wake. The solution to the hypersingular boundary integral equation on the wake requires linear triangular elements (other elements are possible but not developed here), and, thus, similar elements would be used to calculate the influences from the wake. Thus, an adaptive integration scheme for these elements is developed and discussed in section 3.4.

With the algorithm required to calculate the elemental integrals developed, the linear system to be solved must be described for domains with and without wakes. When wakes are present, the Kutta condition must be applied and the impact of this on this linear system is illustrated. Additional equations appended to the linear system provide the wake strength at the trailing edge for each time step. The linear systems with and without wakes are described in section 3.5. Additional adjustments must be made to account for the presence of the channel boundaries. In particular, should a reflective boundary be chosen for the bottom or side walls, the Green's function must be altered to account for this. Also, where a reflective boundary is not used, a double or triple node representation is used to ensure continuity of ϕ at the intersection between boundaries. For the free surface/channel wall boundary, additional equations can be written to correct the free surface potential ensuring that the free surface does not disconnect from, or pass through, the channel wall. The equations to undertake this correction are detailed in appendix D, whilst a more basic approach was deemed satisfactory for the simulations presented within the thesis.

3.2 Biquadratic Elements and Influence Functions

This section and section 3.3 strongly follow the work of Xü (1992, sections 4.3 and 4.5) in order to present a complete methodology for the numerical model developed herein. As the model is an amalgamation of multiple theories, the opportunity is taken to unify the symbols and terminology of the standard boundary element method with those of the Galerkin boundary element method presented in chapter 4. Some alterations and additional theory are also provided, which are distinct from Xü (1992), and these will be highlighted as and when required.

In developing the panels or elements for their boundary element method, Xü wished to design a formal *admissible grid* test into the code, based upon the Taylor expansion of ϕ within a parametric element. For the polynomial expansion to be useful, it was necessary to enforce *geometric isotropy*, which would leave the polynomial approximation invariant after the transformation of coordinates. To facilitate the quadratic approximation and satisfy the above restrictions, a nine node Lagrangian element was chosen.

The generic element, Σ_m, defined in chapter 2, is now chosen to have nine nodes and shall be described using curvilinear coordinates. The orthogonal parametric coordinates (s, t) are used to map the domain of Σ_m to a square in (s, t) space with limits $s \in [-1, 1]$, $t \in [-1, 1]$ as depicted in figure 3.1. All variables of interest, (x, y, z, ϕ, ϕ_n), are defined on the nine nodes of the panel (note, μ is not included here as nine node panels are not used on the wake). The values within the panel can be determined from the nine nodal values and the shape or interpolation functions. For a general field function, Φ, the interpolated value in (s, t) is given by

$$\Phi(s, t) = \sum_{j=1}^{9} \psi_j(s, t)\Phi_j \tag{3.1}$$

where Φ_j is the nodal value at the j-th node. Xü gave the nine shape functions as

$$\begin{cases} \psi_j(s,t) = \dfrac{1}{4}s(s+s_j)t(t+t_j) & j = 1,3,5,7, \\ \psi_j(s,t) = \dfrac{1}{2}(1-t_j^2s^2-s_j^2t^2)\left[t_jt(1+t_jt)+s_js(1+s_js)\right] & j = 2,4,6,8, \\ \psi_9(s,t) = (1-s^2)(1-t^2). \end{cases} \tag{3.2}$$

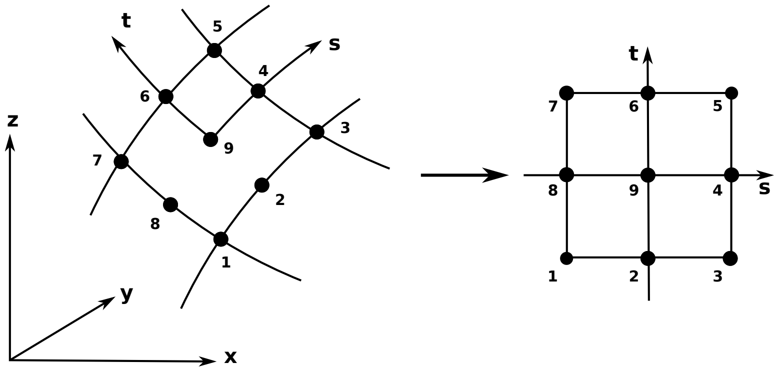


Figure 3.1: Nine node biquadratic panel and its image in parametric space. Adapted from Xü (1992, figure 4-1).

The vector from a general point, P, to the integration point Q, on S, is given as

$$\begin{aligned} \mathbf{R} &= Q - P \\ &= (x_q - x_p,\ y_q - y_p,\ z_q - z_p). \end{aligned} \tag{3.3}$$

The position of Q on the panel Σ_m is estimated by being parameterised into (s,t) space using equation (3.1). Therefore,

$$(x_q, y_q, z_q) = Q(s,t) = \left(\sum_{j=1}^{9}\psi_j x_j,\ \sum_{j=1}^{9}\psi_j y_j,\ \sum_{j=1}^{9}\psi_j z_j\right) \tag{3.4}$$

where x_j, y_j and z_j correspond to the Cartesian coordinates of the nodes on Σ_m.

Subsequently,

$$\mathbf{R} = \left(\sum_{j=1}^{9}\psi_j x_j - x_p,\ \sum_{j=1}^{9}\psi_j y_j - y_p,\ \sum_{j=1}^{9}\psi_j z_j - z_p\right), \tag{3.5}$$

and the length of $\mathbf{R}$ is given by $r = |\mathbf{R}|$.

The derivatives of the shape functions are also necessary for the calculation of the elementary integrals and influence functions. These are not provided within Xü (1992), but are listed here in appendix B. The derivatives are denoted by Q_s and Q_t, where the subscripts indicate the s and t partial derivatives respectively.

For a parameterised surface, the normal can be estimated using cross product of Q_s and Q_t. Thus, the unit normal is given by

$$\mathbf{n} = \frac{Q_s \times Q_t}{|Q_s \times Q_t|}. \tag{3.6}$$

The components of $Q_s \times Q_t = (j_1, j_2, j_3)$ are

$$\begin{cases} j_1 = \dfrac{\partial y_q}{\partial s}\dfrac{\partial z_q}{\partial t} - \dfrac{\partial y_q}{\partial t}\dfrac{\partial z_q}{\partial s}, \\ j_2 = \dfrac{\partial z_q}{\partial s}\dfrac{\partial x_q}{\partial t} - \dfrac{\partial z_q}{\partial t}\dfrac{\partial x_q}{\partial s}, \\ j_3 = \dfrac{\partial x_q}{\partial s}\dfrac{\partial y_q}{\partial t} - \dfrac{\partial x_q}{\partial t}\dfrac{\partial y_q}{\partial s}. \end{cases} \tag{3.7}$$

Note that the direction of the normal is governed by the ordering of the nine nodes in a generic panel. As can be seen in figure 3.2, the sense of the normal is given by the right hand rule relative to the node numbering. This is, if the fingers in your clenched right hand point in the direction of increasing perimeter node number then the sense of the normal is given by the direction in which your thumb is pointing.

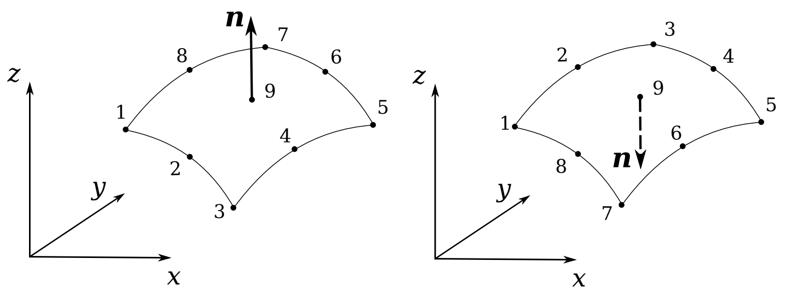


Figure 3.2: Normal vectors for identical panels with oppositely ordered perimeter node numberings.

The differential surface area for the parameterised element, Σ_m, is given by

$$dA = J(s,t)ds\;dt \tag{3.8}$$

where

$$J(s,t) = |Q_s \times Q_t| \tag{3.9}$$

is known as the Jacobian. This represents the local change in area between the parameter and physical space.

By comparing equations (3.7) and (3.9), observe that the Jacobian can be calculated using

$$J(s,t) = \sqrt{j_1^2 + j_2^2 + j_3^2}. \tag{3.10}$$

Further to Xü (1992), the normal and the Jacobian can be related by

$$\mathbf{n} = \frac{1}{J(s,t)} \left(j_1, j_2, j_3\right). \tag{3.11}$$

which will prove to be a useful result for simplifying later expressions.

The approximation to the elemental integrals defined by equations (2.93) and (2.94) is given by

$$I_{s,i} = \sum_{j=1}^{9} \phi_{n,j} S_j^{(m)}(P_i), \tag{3.12}$$

$$I_{d,i} = \sum_{j=1}^{9} \phi_j D_j^{(m)}(P_i). \tag{3.13}$$

where i is the global index for the collocation point. $\phi_{n,j}$ and ϕ_j are the field function values for a particular local node, j, on an element, Σ_m, and Xü defines $S_j^{(m)}$ and $D_j^{(m)}$ as the influence functions associated with those nodes. In particular, the influence functions are given in Xü (1992) by

$$S_j^{(m)} = \iint_{\Sigma_m(s,t)} \frac{\psi_j(s,t) J(s,t)}{r(s,t)} \, ds \, dt, \tag{3.14}$$

$$D_j^{(m)} = \iint_{\Sigma_m(s,t)} -\frac{\mathbf{n} \cdot \mathbf{R}}{r^3(s,t)} \psi_j(s,t) J(s,t) \, ds \, dt. \tag{3.15}$$

The domain of integration for $\Sigma_m(s,t)$ is $s \in [-1,1]$, $t \in [-1,1]$.

Note that the form of equation (3.15), appearing in Xü (1992), can be simplified with the use of equation (3.11). Observe that

$$\begin{aligned} \mathbf{n} \cdot \mathbf{R} &= \frac{1}{J(s,t)} \left((j_1, j_2, j_3) \cdot (r_1, r_2, r_3) \right) \\ &= \frac{1}{J(s,t)} (j_1 r_1 + j_2 r_2 + j_3 r_3) \end{aligned} \tag{3.16}$$

and thus equation (3.15) becomes

$$D_j^{(m)} = \iint_{\Sigma_m(s,t)} -\frac{(j_1 r_1 + j_2 r_2 + j_3 r_3)}{r^3(s,t)} \psi_j(s,t)\, ds\, dt. \tag{3.17}$$

The discretised form of the solid angle, as described by equation (2.47), is required. Presented here is an alternative to that given by Xü (1992), formulated as

$$\alpha(P_i) = -\sum_{m=1}^{M_S} \sum_{j=1}^{9} D_j^{(m)}. \tag{3.18}$$

M_S is the total number of panels in the surface subset, Σ_S, where Σ_S corresponds to all of the nine node panels, i.e. those panels that are not on the wake.

Note that as

$$\sum_{j=1}^{9} \psi_j = 1, \tag{3.19}$$

equation (3.18) can be written as

$$\alpha(P_i) = \sum_{m=1}^{M_s} \iint_{\Sigma_m(s,t)} \frac{(j_1 r_1 + j_2 r_2 + j_3 r_3)}{r^3(s,t)}\, ds\, dt. \tag{3.20}$$

3.3 Adaptive Numerical Integrations

The purpose of this section is to develop an algorithm for evaluating the influence functions given by equations (3.14) and (3.15) in the singular, near-singular and far field cases. The adaptive method of Xü (1992) is recreated here and any alterations or modifications to the method are clearly highlighted throughout the section.

The essence of the adaptive quadrature scheme is to deliver a fixed level of accuracy for each element by subdividing elements and using increasing order of Gauss-Legendre quadrature. The adaptive scheme is controlled via a number of non-dimensional metrics which examine the distance between point and panel, perspective angle from point to panel and panel aspect ratio. For the near field, the target values of these metrics are not clearly described within Xü (1992). With this in mind, and a target level of accuracy sought (a local relative error of 10^{-4}), a bespoke scheme was developed using similar non-dimensional metrics.

To regularise the weakly singular integrals, Xü employs a degenerate triangle mapping. This technique maps a triangular domain to a square one, 'stretching out' the singular point into a line. This is an alternative method to the polar transformation technique used extensively later in this document to regularise the hypersingular boundary integral equation. The triangle mapping is used by Xü to employ element subdivision in the singular cases, which benefits the adaptive control of accuracy.

3.3.1 Evaluation of Far and Near Field Influences

When the field point, P, lies away from the surface, S, the boundary integral equations, given by equations (3.14) and (3.15), are continuous and non-singular. Xü (1992) used Gauss-Legendre quadrature to numerically integrate the influence functions. Accuracy in the simulation was maintained by a two stage process. As the field point approaches the surface, the order of quadrature is gradually increased to a certain level. Beyond that, the panels are subdivided, dependant on the location of the collocation point relative to the local integration panel, and then quadrature applied over the subdivided panels.

For the far and near field influences the panels are subdivided in the s and t parametric directions. The number of subdivisions in the s and t directions are given by m_1 and m_2 respectively. Xü approximated the influence functions using the formula

$$\iint_{\Sigma_m(s,t)} \mathcal{F}(s,t)\, ds\, dt = \frac{1}{m_1 m_2} \sum_{k=1}^{m_1} \sum_{l=1}^{m_2} \sum_{p=1}^{\nu(k,l)} \sum_{q=1}^{\nu(k,l)} w_p^{(\nu)} w_q^{(\nu)} \mathcal{F}(s_p^{(\nu)}, t_q^{(\nu)}) \tag{3.21}$$

where w_p and w_q are the Gaussian weights for s and t, ν is the given order of quadrature and $\mathcal{F}$ is the function to be integrated. An equal number of quadrature points are used in the s and t directions, however the number can vary across subelements.

Within Xü (1992) there are three parameters for determining the order of Gauss-Legendre quadrature and number of subdivisions for a particular node-panel pairing. They are:

1. $\varrho_1 = r_{\text{min}}^2$: r_{min} is the minimum distance from the field point to one of the nodes of the integration element as seen in figure 3.3.

2. ϱ_2: the cosine of the greatest perspective angle viewed from the field point towards

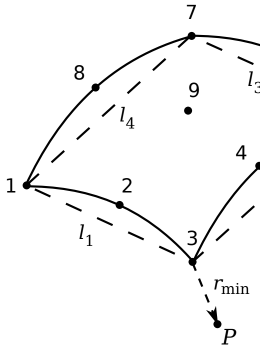


Figure 3.3: Side lengths and minimum distance to field point for nine node panel. Recreated from Xü (1992, figure 4-2).

the element. This is given by

$$\varrho_2 = \cos(\theta) = \frac{a^2 + b^2 - c^2}{2ab} \tag{3.22}$$

where the perspective angle, θ and a, b and c are illustrated in figure 3.4.

3. ϱ_3: an estimated "aspect ratio" of the element, defined by

$$\varrho_3 = \sqrt{\frac{l_1^2 + l_3^2}{l_2^2 + l_4^2}} \tag{3.23}$$

where l_i, $i = 1, 2, 3, 4$, are the distances between the four vertices as illustrated in figure 3.3.

Ideally, ϱ_1 would be a non-dimensional, normalised quantity (otherwise there is potential for the same level of quadrature to be applied to panels with equal $r_{\min}$ but with much different surface area). Thus, the original definition of ϱ_1 given by Xü (1992) is inadequate in its given form. An alternative definition for ϱ_1 developed as part of the present method is:

ϱ_1 should be normalised by either the average of the lengths, l_i, i.e.

$$\varrho_1 = \left(\frac{r_{\min}}{\frac{1}{4}\left(l_1 + l_2 + l_3 + l_4\right)} \right)^2 = 16 \left(\frac{r_{\min}}{l_1 + l_2 + l_3 + l_4} \right)^2 \tag{3.24}$$

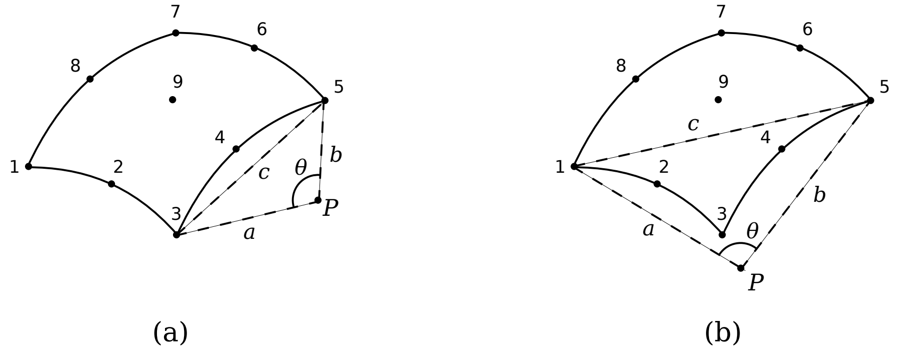


Figure 3.4: Orientation for greatest perspective angle for field points closest, (a) to a panel edge node and, (b), closest to a corner node. Recreated from Xü (1992, figure 4-3).

or should be normalised by the root mean square of the lengths given by

$$\varrho_1 = \left(\frac{r_{\min}}{\sqrt{\dfrac{l_1^2 + l_2^2 + l_3^2 + l_4^2}{4}}} \right)^2 = \frac{4r_{\min}^2}{l_1^2 + l_2^2 + l_3^2 + l_4^2}. \tag{3.25}$$

Additionally, the definition of ϱ_2 given by Xü (1992) exhibits some ambiguity. The problem lies with the determination of the length c. The original technique is described,

> "Since $\theta \in [0, \pi], \varrho_2 \in [-1, 1]$, we first determine the local node number which is closest to the field point... then the maximum dimension c of the element viewed at P_i is defined as the maximum distance between any two nodes observed at P_i.".

The confusion is illustrated well by figure 3.4. Parts (a) and (b) show the greatest perspective angle for alternative closest nodes, however, using the definition given above, the length of c in (b) would be used for both the field points in (a) and (b) and this seems incorrect. Thus, a better approach was developed herein which explicitly locates the greatest angle, θ, rather than the greatest length, c.

To correctly evaluate ϱ_2, take c to be each of the four sides of the panel (those with lengths, l_i) in turn as well as the two diagonals of the panel. Subsequently, calculate the resulting six cosines (or values of ϱ_2). Of the six cosines, the lowest will provide the greatest perspective angle as $\varrho_2 \in [-1, 1]$.

During the Eulerian stage of the mixed Eulerian-Lagrangian approach the grid is fixed while the boundary integral equations are evaluated. Thus the quantities l_1, l_2, l_3, l_4 and ϱ_3 do not vary and may be calculated just once per time step.

The key parameter for controlling the order of quadrature is ϱ_1. Although levels of quadrature for particular bounds of ϱ_1 are given in Xü (1992), little information is provided as to how those bounds were calculated. Thus, a number of numerical tests were carried out as part of the development of the present method which evaluated the numerical error of a field point as it approached a discretised panel. As a result of these tests, it was found that the order of quadrature for the far field integrals can be determined by

$$\nu = \begin{cases} 2, & \text{if } \varrho_1 > 196, \\ 3, & \text{if } 49 < \varrho_1 \leq 196, \\ 4, & \text{if } 16 < \varrho_1 \leq 49, \\ 5, & \text{if } 9 < \varrho_1 \leq 16, \end{cases} \tag{3.26}$$

in order to achieve a local target accuracy of 1×10^{-4}. For these values of ϱ_1 no subdivision is required.

For the near-singular integrals, when $\varrho_1 \leq 9$, ϱ_2 is used to determine whether further subdivision is necessary. It was found that further subdivision was required when $\varrho_2 < 2^{-1/2}$. If ϱ_2 is within this range then the panels are subdivided in the manner of Xü (1992) with the parametric directions being determined by the aspect ratio, ϱ_3. The resulting formulae are

$$\begin{cases} m_1 = \text{int}\left[2\varrho_3\right], & m_2 = 2, & \text{if } \varrho_3 > 1 \\ m_1 = 2, & m_2 = \text{int}\left[\dfrac{2}{\varrho_3}\right], & \text{if } \varrho_3 < 1 \end{cases} \tag{3.27}$$

where 'int[ ]' represents the integer part. The order of quadrature within the subelement is variable depending on its location relative to the collocation point. Following the notation of Xü (1992), subelements are denoted by $\Sigma_m^{(k,l)}$ while Σ_m^c represents the set of subelements containing p_c, the closest node to the collocation point. Xü considered

$$\nu = \begin{cases} 5, & \text{if } \Sigma_m^{(k,l)} \in \Sigma_m^c, \\ 4, & \text{otherwise,} \end{cases} \tag{3.28}$$

to be an effective level of quadrature for the subdivided elements. An example of such a subdivision is illustrated in figure 3.5. Here the closest node is $p_c = 2$ and the shaded panels represent Σ_m^c.

If $\varrho_2 > 2^{-1/2}$ then no subdivision is required. Rather than using the values proposed

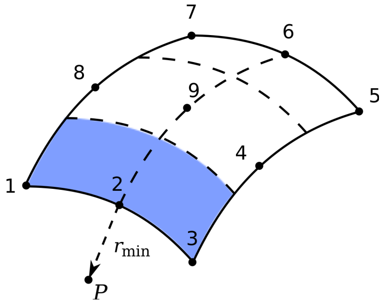


Figure 3.5: Panel subdivision for near-singular point, P. The shaded panels contain p_c, the nearest integration point to P. Adapted from Xü (1992, figure 4-4).

in Xü (1992), the bounds on the quadrature are once again developed through numerical experiment. An appropriate set of limits is

$$\nu = \begin{cases} 6, & \text{if } 4 < \varrho_1 \leq 9, \\ 7, & \text{if } 2 < \varrho_1 \leq 4, \\ 8, & \text{if } 0 < \varrho_1 \leq 2. \end{cases} \tag{3.29}$$

The remaining case is when $\varrho_1 = 0$. In this case the collocation point is coincident with one of the points on the panel being integrated and thus the boundary integral equation is singular.

3.3.2 Evaluation of Self-Influences

Over the course of the integration of equations (2.89)–(2.92) the collocation point, P, will at some point require the influence from a panel within which it is contained. This is described by Xü as a *self-influence* calculation.

The boundary integral equation contains two types of singularity that must be regularised. The first, part of the source influence function, is known as a *weakly singular* integral and is $\mathbf{O}(r^{-1})$. The second, part of the doublet influence function, is known as a *strongly singular* integral and is $\mathbf{O}(r^{-2})$.

Evaluating the singular kernels increases in difficulty when panels with a higher order discretisation than constant are used. For constant panels, the collocation point is in the centre of the panel and a number of analytical regularisation schemes can be applied. It is often reported that constant panel methods lacked the accuracy required and thus came a move to linear panels. Here, the collocation points are at the edges of flat panels and thus, a discontinuity in ϕ_n is present there. Methods for dealing with the difficulties of regularising linear panels were developed by Swedlow and Cruse (1971) and others.

In order to further increase accuracy, quadratic and higher order panels were used. Subsequently, three alternative approaches were developed to regularise these higher order panels. The first, and least common, was a special quadrature with singular weightings. The second, and probably most common approach, is a polar coordinate transformation of the singular panel as described in Liu and Lu (1988). The final approach, and that chosen by Xü, employs the degenerate triangle mapping presented by Li *et al.* (1985). As Xü points out, the final quadrature for the polar transformation and the triangle mapping differ in only one parametric direction. Xü elects to implement the triangle mapping method due to the ease of panel subdivision it offers and, thus, the method is readdressed herein.

The degenerate triangle mapping is developed following Xü (1992, section 4.5.2). Xü began by defining a plane triangular domain, T, in three dimensions, that had three vertices located at $V_1(\mathbf{x}_1)$, $V_2(\mathbf{x}_2)$ and $V_3(\mathbf{x}_3)$. Then, for an integrand, $\mathcal{F}$, with a weak singularity at V_1, the degenerate triangle mapping was defined as

$$\mathbf{x} = (1-\rho_1)\mathbf{x}_1 + \rho_1(1-\rho_2)\mathbf{x}_2 + \rho_1\rho_2\mathbf{x}_3 \tag{3.30}$$

where ρ_1 and ρ_2 are triangle polar coordinates. Xü denoted the image of the triangle domain in (ρ_1, ρ_2) space as T', having bounds $\rho_1 \in [0,1]$, $\rho_2 \in [0,1]$. The images of the vertices of T, V_i, are denoted by V_i' respectively. The key to the analysis is that the singular point, V_1, now corresponds to the line $\overline{V_1''V_1'}$, as can be seen in figure 3.6.

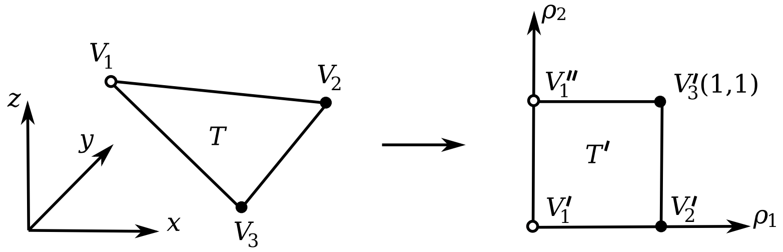


Figure 3.6: The degenerate triangle mapping from Cartesian to parametric (ρ_1, ρ_2) space. Recreated from Xü (1992, figure 4-5).

In order to apply the above mapping to the biquadratic iso-parametric element, Σ_Q, the element must be divided into triangles such that every triangle has *one and only one vertex* which coincides with the singular point. Such a triangularisation is illustrated in figure 3.7. Following the triangularisation, each triangle in (s, t) parameter space is mapped to the square in the second parameter space (ρ_1, ρ_2). The weak singularity is then analytically removed and the strong singularity is shown to cancel with the solid angle contribution of the panel.

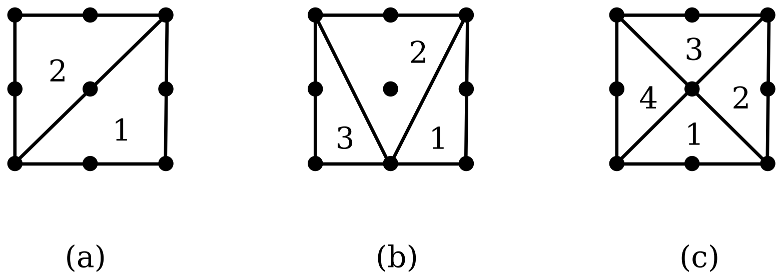


Figure 3.7: Triangularisations for nine-node element when the singular point is a) at a corner, b) at an edge and c) at the centre of the element. Recreated from Xü (1992, figure 4-6).

When singular, the collocation point, P_i, will be coincident with the integration panel at a node $k = 1, 2, \ldots, 9$. Xü (1992) was able to show that the shape functions, given by equations (3.2), were related to the (ρ_1, ρ_2) parameter space by

$$\psi_j(s,t) = \begin{cases} 1 + \rho_1 \varphi_j(\rho_1, \rho_2) & \text{if } j = k \\ \rho_1 \varphi_j(\rho_1, \rho_2) & \text{if } j \neq k \end{cases} \tag{3.31}$$

where $j = 1, 2, \ldots, 9$ and functions φ_j are non-zero at $\rho_1 = 0$. Although the form of φ_j is quite complicated and dependant on how the biquadratic element is triangularised, the functions φ_j do not need to be explicitly calculated. To avoid the direct calculation of the functions developed in Xü (1992), simply rearrange equation (3.31), such that the functions φ_j are given by

$$\varphi_j = \begin{cases} \dfrac{\psi_j - 1}{\rho_1} & \text{if } j = k \\ \dfrac{\psi_j}{\rho_1} & \text{if } j \neq k. \end{cases} \tag{3.32}$$

Using the result given in equation (3.31), the vector, $\mathbf{R}$ given by equation (3.3), now becomes

$$\begin{aligned}\mathbf{R} = Q - P &= (x_q - x_k, y_q - y_k, z_q - z_k) \\ &= \left(\sum_{j=1}^{9} \psi_j x_j - x_k, \sum_{j=1}^{9} \psi_j y_j - y_k, \sum_{j=1}^{9} \psi_j z_j - z_k\right) \\ &= \left(\rho_1 \sum_{j=1}^{9} \varphi_j x_j,\ \rho_1 \sum_{j=1}^{9} \varphi_j y_j,\ \rho_1 \sum_{j=1}^{9} \varphi_j z_j\right) \\ &= \rho_1 \left(X(\rho_1, \rho_2), Y(\rho_1, \rho_2), Z(\rho_1, \rho_2)\right).\end{aligned} \tag{3.33}$$

Thus, the length of $\mathbf{R}$ is now given by

$$r = \rho_1 R(\rho_1, \rho_2) \tag{3.34}$$

where

$$R(\rho_1, \rho_2) = \sqrt{X^2(\rho_1, \rho_2) + Y^2(\rho_1, \rho_2) + Z^2(\rho_1, \rho_2)}. \tag{3.35}$$

The Jacobian of the transformed triangle must also be evaluated. The components defined by equation (3.7) are transformed such that

$$j_i = j_i(s(\rho_1, \rho_2), t(\rho_1, \rho_2)), \qquad i = 1, 2, 3. \tag{3.36}$$

The alteration to the differential area resulting from the transformation is given by

$$\begin{aligned}dS &= J(s, t)\ ds\ dt \\ &= J(s(\rho_1, \rho_2), t(\rho_1, \rho_2))\ ds(\rho_1, \rho_2)\ dt(\rho_1, \rho_2) \\ &= J(\rho_1, \rho_2) J_\mu(\rho_1, \rho_2)\ d\rho_1\ d\rho_2\end{aligned} \tag{3.37}$$

where

$$J_\mu = 2\rho_1 A_\mu. \tag{3.38}$$

J is given by equation (3.10) and A_μ is the area of the μ-th triangle in (s, t) space. This result is reported in both Xü (1992) and Liu and Lu (1988), however not explicitly proved. The proof of equation (3.38) is given in appendix C.

Equations (3.34) to (3.38) can now be substituted into the influence functions. For the source influence, observe that

$$S_j^{(m)} = 2\sum_{\mu=1}^{\kappa}\int_0^1\int_0^1 \frac{\psi_j(\rho_1,\rho_2)J(\rho_1,\rho_2)A_\mu}{R(\rho_1,\rho_2)}\,d\rho_1\,d\rho_2 \tag{3.39}$$

where $j = 1, 2, \ldots, 9$ and the number of triangles $\kappa \in [2, 3, 4]$ correspond to the singular point being at a corner node, a side node, or the centre node, respectively.

For the doublet influence there are two cases. For the case were $j \neq k$, equation (3.15) is rewritten as

$$D_j^{(m)} = -2\sum_{\mu=1}^{\kappa}\int_0^1\int_0^1 \frac{\mathcal{K}(\rho_1,\rho_2)}{R^3(\rho_1,\rho_2)}\varphi_j(\rho_1,\rho_2)A_\mu\,d\rho_1\,d\rho_2 \tag{3.40}$$

noting that

$$\begin{aligned}\mathbf{n}\cdot\mathbf{R} &= \frac{\rho_1\left(j_1X(\rho_1,\rho_2)+j_2Y(\rho_1,\rho_2)+j_3Z(\rho_1,\rho_2)\right)}{J(\rho_1,\rho_2)}\\ &= \frac{\rho_1\mathcal{K}(\rho_1,\rho_2)}{J(\rho_1,\rho_2)}.\end{aligned} \tag{3.41}$$

When $j = k$ the doublet influence is

$$D_j^{(m)} = -2\sum_{\mu=1}^{\kappa}\int_0^1\int_0^1 \frac{\mathcal{K}(\rho_1,\rho_2)}{\rho_1R^3(\rho_1,\rho_2)}\left[1+\rho_1\varphi_j(\rho_1,\rho_2)\right]A_\mu\,d\rho_1\,d\rho_2. \tag{3.42}$$

The first term of the integrand of equation (3.42) is singular as $\rho_1 \to 0$. Fortunately, the very same term can be shown to cancel with the solid angle contribution of the panel.

To demonstrate this, in a more verbose manner than that given in Xü (1992), consider calculating the solid angle contribution at a point P_i from a panel, Σ_Q, where P_i is coincident to one of the nodes on Σ_Q. The solid angle for P_i is given by equation (3.20) and for the panel Σ_Q the contribution is given by

$$\alpha(P_i) = -\iint_{\Sigma_Q} -\frac{(j_1r_1+j_2r_2+j_3r_3)}{r^3(s,t)}\,ds\,dt. \tag{3.43}$$

Note that equation (3.43) is singular at P_i. To deal with this singularity the degenerate triangle mapping is applied to the right hand side of equation (3.43), the result of which is

$$2\sum_{\mu=1}^{\kappa}\int_0^1\int_0^1 \frac{\mathcal{K}(\rho_1,\rho_2)}{\rho_1R^3(\rho_1,\rho_2)}A_\mu\,d\rho_1\,d\rho_2. \tag{3.44}$$

Comparing equation (3.44) with equation (3.42) it can be seen that the solid angle contribution from the singular panel will completely cancel the first term of the singular doublet influence when the integration point and the collocation point are the same node, i.e. $j = k$.

The implication of this cancellation is that only equation (3.40) is used to calculate the singular doublet influences of those panels containing the collocation point. The solid angle contribution for the entire singular panel is also cancelled and thus does not contribute to the solid angle, $\alpha(P_i)$. Equation (3.20) becomes, in effect,

$$\alpha(P_i) = \sum_{\substack{m=1 \\ m \notin Q_c}}^{M_s} \iint_{\Sigma_m(s,t)} \frac{(j_1 r_1 + j_2 r_2 + j_3 r_3)}{r^3(s,t)}\, ds\, dt \tag{3.45}$$

where Q_c is the set indices of panels containing P_i.

In order to apply Gauss-Legendre quadrature to the singular panels the quadrature points must be transformed from $\rho_i \in [0,1]$, to $\tau_i \in [-1,1]$ $(i = 1,2)$. The mapping

$$\rho_i = (\tau_i + 1)/2, \qquad i = 1,2, \tag{3.46}$$

is used. Once this mapping is applied, equations (3.39) and (3.40) become

$$S_j^{(m)} = \frac{1}{2} \sum_{\mu=1}^{\kappa} \int_{-1}^{1} \int_{-1}^{1} \frac{\psi_j(\tau_1,\tau_2) J(\tau_1,\tau_2) A_\mu}{R(\tau_1,\tau_2)}\, d\tau_1\, d\tau_2, \tag{3.47}$$

$$D_j^{(m)} = -\frac{1}{2} \sum_{\mu=1}^{\kappa} \int_{-1}^{1} \int_{-1}^{1} \frac{\mathcal{K}(\tau_1,\tau_2)}{R^3(\tau_1,\tau_2)} \varphi_j(\tau_1,\tau_2) A_\mu\, d\tau_1\, d\tau_2. \tag{3.48}$$

As stated in Xü (1992), the singular $D_j^{(m)}$ contains a more severe singularity than $S_j^{(m)}$. Thus to maintain the same level of accuracy, a higher order of quadrature is required for $D_j^{(m)}$ over $S_j^{(m)}$. In order to remove the necessity for very high orders of quadrature, Xü chose to subdivide the triangles prior to performing the degenerate triangle mapping on the resulting sub-triangles. The subdivision is illustrated in figure 3.8 parts (a) and (b) by the dotted lines. Let the *included angle* at the singular point of the triangular domain in physical (Cartesian) space be θ. This angle is estimated in a similar manner to the angle evaluated for ϱ_2, using equation (3.22) with a, b and c defined as in figure 3.8, and is used to set a new parameter ϱ_4.

If $\varrho_4 < 0.15$ then subdivision is required. Two alternative parameters for control-

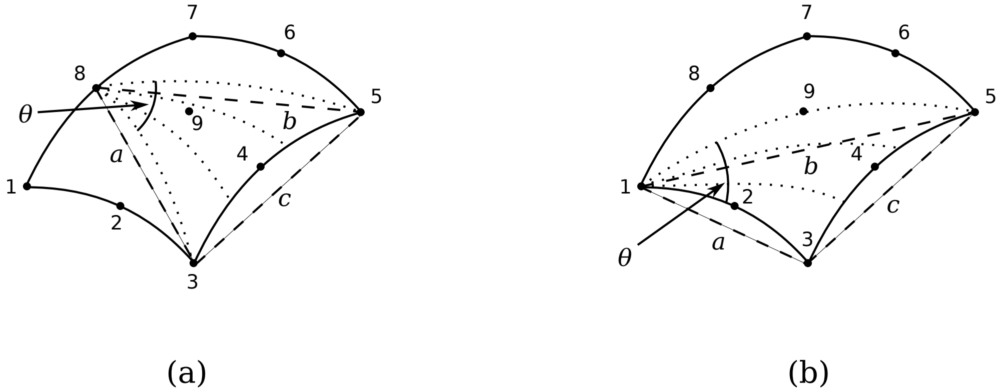


Figure 3.8: Subdivision of panels into triangles where the singular node lies a) on an edge or b) on a corner. The lengths a, b, c and the angle θ required for ϱ_4 is illustrated. Adapted from Xü (1992, figure 4-7).

ling the subdivision are investigated. First, consider using the parameter defined by equation (3.23), ϱ_3, in the formula

$$m_3 = \text{int}[2\varrho_3] \tag{3.49}$$

where m_3 are the required number of subdivisions. This approach to calculating m_3 was used in Xü (1992). Although the approach may seem plausible, consider the case where the panel is thin in the s parametric direction and fat in the t parametric direction as can be seen in figure 3.9. In this situation ϱ_3 will be very small and thus equation (3.49) may not produce any subdivision at all. Thus, it was decided that, because of the issues highlighted above, an alternative methodology for evaluating m_3 should be developed. A more feasible control based on ϱ_3 may be

$$\begin{cases} m_3 = \text{int}\left[2\varrho_3\right], & \text{if } \varrho_3 > 1 \\ m_3 = \text{int}\left[\dfrac{2}{\varrho_3}\right], & \text{if } \varrho_3 < 1. \end{cases} \tag{3.50}$$

An alternative control strategy can be derived using just the parameter, ϱ_4. An initial starting point may be to set

$$m_3 = \text{int}\left[\frac{2}{\varrho_4}\right]. \tag{3.51}$$

An issue occurs with equation (3.51), however, when $\theta = \pi/2$ as ϱ_4 will be zero. To counteract this problem a constant can be added to the denominator of equation (3.51)

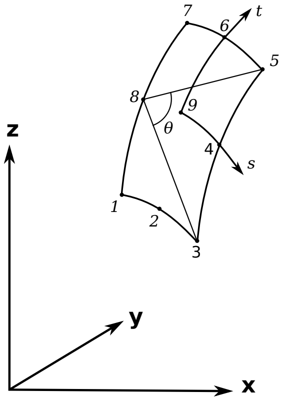


Figure 3.9: Included angle for a panel with large aspect ratio.

resulting in

$$m_3 = \mathrm{int}\left[\frac{2}{\varrho_4 + 1}\right]. \tag{3.52}$$

Equation (3.52) removes the singularity at $\theta = \pi/2$, however for the range $0 \leq \varrho_4 \leq 0.15$, this equation would result in m_3 being equal to one. To improve equation (3.52) so that m_3 is always greater than two consider choosing

$$m_3 = \mathrm{int}\left[\frac{2.3}{\varrho_4 + 1}\right]. \tag{3.53}$$

Equation (3.53) provides $m_3 > 2, \forall \varrho_4 \in [0, 0.15]$. However, m_3 gets very large as $\theta \to \pi$ and $\varrho_4 \to -1$. Thus the finalised formula is given by

$$m_3 = \mathrm{int}\left[\frac{2.3 + 2x}{\varrho_4 + 1 + x}\right], \tag{3.54}$$

where x is some quantity decided upon experimentally. A suitable choice is $x = 0.15$. Within the subdivided panels, fourth order Gauss-Legendre quadrature is found to provide the desired level of accuracy.

When $\varrho_4 > 0.15$ the formula

$$\nu = \mathrm{int}\left[-4.5\varrho_4 + 9.6\right] \tag{3.55}$$

is used to simplify the more complicated expression used in Xü (1992, equation (4.77)).

3.4 Linear Triangular Elements

Ideally, the biquadratic elements developed in the last section could also be extended to the wake, however there are some practical difficulties and theoretical restrictions that currently make this undesirable. In particular, the wake can only spawn one line of nodes per time-step. This precludes the use of a biquadratic panel in the early time-steps although as the wake grows, panels could be converted from three to nine node. The additional incentive for using three node panels on the wake is that the solution scheme for the hypersingular integral equation (see chapter 2) is currently only developed for use with linear elements.

There was no requirement for a linear element in Xü (1992) and, thus, the standard boundary element for a linear triangular element must be developed here from scratch. In principle, however, the approach is extremely similar to the formulation for biquadratic elements. In order to proceed, the linear element and appropriate weightings must be defined. The element, described by Gray *et al.* (2004a), is a three nodal linear triangular element with shape functions defined by

$$\begin{aligned}\tilde{\psi}_1(s,t) &= \frac{\sqrt{3}\,(1-s)-t}{2\sqrt{3}},\\ \tilde{\psi}_2(s,t) &= \frac{\sqrt{3}\,(1+s)-t}{2\sqrt{3}},\\ \tilde{\psi}_3(s,t) &= \frac{t}{\sqrt{3}},\end{aligned} \tag{3.56}$$

within the equilateral parameter space, $-1 \leq s \leq 1$, $0 \leq t \leq \sqrt{3}(1-|s|)$. $\tilde{\psi}$ is used to differentiate from the nine node shape functions denoted by ψ. The element is shown in figure 3.10. The points P and Q, the vector $\mathbf{R}$ and length r are calculated in similar manner to the nine node panels. Equation (3.3) is unchanged, although, Q is now given by

$$(x_q, y_q, z_q) = Q(s,t) = \left(\sum_{j=1}^{3} \tilde{\psi}_j x_j,\ \sum_{j=1}^{3} \tilde{\psi}_j y_j,\ \sum_{j=1}^{3} \tilde{\psi}_j z_j \right). \tag{3.57}$$

The s and t derivatives of the linear shape functions are available in appendix B. The definitions for the normal and Jacobian are given by equations (3.6) to (3.11) and note that these quantities are constant across each linear triangular panel.

Given that the three node linear panels are only applied to the wake, only the wake

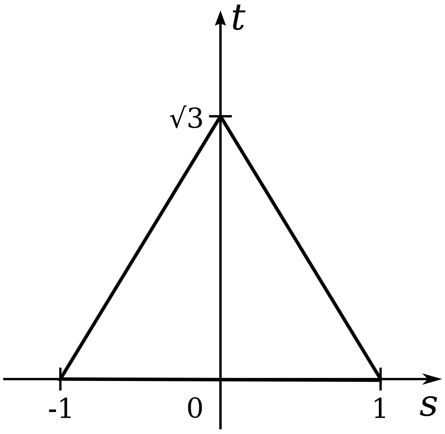


Figure 3.10: Linear triangular element.

elemental integral is required. This is approximated by

$$I_w = \sum_{j=1}^{3} \mu_j W_j^{(m)}(P_i) \tag{3.58}$$

where

$$W_j^{(m)} = \iint_{\Sigma_m(s,t)} -\frac{\mathbf{n}\cdot\mathbf{R}}{r^3(s,t)}\tilde{\psi}_j(s,t)J(s,t)\,ds\,dt. \tag{3.59}$$

and Σ_m is a generic three node linear panel on the wake. As a consequence of equation (3.16), equation (3.59) can be re-coined as

$$W_j^{(m)} = \iint_{\Sigma_m(s,t)} -\frac{(j_1r_1 + j_2r_2 + j_3r_3)}{r^3(s,t)}\tilde{\psi}_j(s,t)\,ds\,dt. \tag{3.60}$$

3.4.1 Adaptive Numerical Integration

Unlike the case for adaptive numerical integration for the nine node panels on S_D and S_N, the field point, P, will never be on the wake surface, S_W, as all (with the exception of the Kutta strip) values of μ are known at this stage of the calculation. Given the requirement to include a reasonably large far field velocity, it is expected that the closest that a field point will get to the wake points, without singular integration, is $\varrho_1/2$ where ϱ_1 is defined by either equation (3.24) or (3.25). Some limits on the size of the panels nearest the trailing may be appropriate in this case. In addition, the method of quadrature must change for the panels on S_W as the integration domain is now over a triangle rather than a square and this is addressed in the section below.

Near and Far Field Influences

A triangle quadrature scheme must be applied in order to calculate the influence $W_j^{(m)}$. These schemes are normally defined in terms of barycentric (also know as area, or areal) coordinates on the triangle (see Coxeter, 1989). Defined on the *natural* or equilateral triangle, the barycentric coordinates are given as (α, β, γ), as illustrated in figure 3.11. The origin is at the centre of mass of the triangle with the three coordinates defined in the directions of the vertices. At each vertex, one coordinate is equal to one and the others are zero.

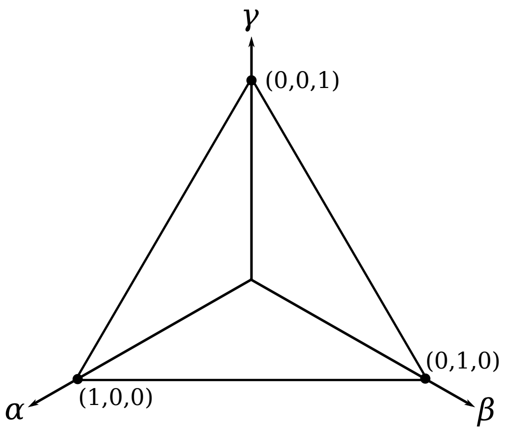


Figure 3.11: Barycentric coordinates for an equilateral triangle.

Most quadrature tables for triangular domains are presented using barycentric coordinates and in order to make use of them, without rewriting the shape functions, the quadrature points must be converted into the (s, t) parametric space. If a generic triangle in (s, t) space has vertices (p_1, p_2, p_3) then the barycentric expansion for that triangle is given by

$$\begin{aligned} s &= \alpha\, p_1(s) + \beta\, p_2(s) + \gamma\, p_3(s), \\ t &= \alpha\, p_1(t) + \beta\, p_2(t) + \gamma\, p_3(t). \end{aligned} \tag{3.61}$$

Thus, using the parametric domain illustrated in figure 3.10, the conversion is given by

$$\begin{aligned} s &= -\alpha + \beta, \\ t &= \sqrt{3}\gamma. \end{aligned} \tag{3.62}$$

The quadrature formula over the triangular domain for near and far field influences is given by

$$\iint_{\Sigma_m(s,t)} \mathcal{F}(s,t)\; ds\; dt = A \sum_{p=1}^{\nu} w_p^{(\nu)} \mathcal{F}(s(\alpha_p^{(\nu)}, \beta_p^{(\nu)}, \gamma_p^{(\nu)}), t(\alpha_p^{(\nu)}, \beta_p^{(\nu)}, \gamma_p^{(\nu)})) \tag{3.63}$$

where ν is the order of quadrature, w_p are the weightings, and the area of the generic triangular panel is $A = \sqrt{3}$. Tables containing points and weightings for integration over triangular domains can be found in Dunavant (1985).

Numerical experiments were undertaken for isolated panels, as part of the present work, in order to develop a basic adaptive scheme which could control the order of quadrature used to evaluate the integrals on the wake panels. The results of these experiments suggested a scheme such as

$$\nu = \begin{cases} 1, & \text{if } \varrho_1 > 3000, \\ 3, & \text{if } 16 < \varrho_1 \leq 3000, \\ 12, & \text{if } \varrho_1 \leq 16. \end{cases} \tag{3.64}$$

Here ϱ_1 is evaluated in a similar manner to equation (3.24) or (3.25) using just three side lengths rather than four. One caveat to this is that for $\varrho_1 \leq 0.5$ the order of quadrature does not produce the desired level of accuracy. In fact, it was found that higher orders of quadrature were still inadequate. Thus, at this stage it is most likely that some subdivision would be needed. However, for the reasons stated above, the need for this is perceived to be unlikely and so no subdivisions are used in the present scheme, noting that this could be conceivable as further work.

Singular Integrations

The singular integrations for the linear triangular panels are, in essence, dealt with in a very similar fashion to the nine node panels. One considerable advantage of the 'flat' geometry of a linear triangular panel is that $\mathbf{n} \cdot \mathbf{R} = 0$ for singular nodes. Given that only the strong integral is required on the wake, when the field point is singular at the trailing edge, the influence from the wake panel is conveniently zero.

It should be noted that using triangular elements may be useful on other surfaces. For instance, the problem of capping a solid body, like a wing, is much more straightforward using triangles. Indeed, the linear triangular elements have been used in this study to demonstrate the accuracy and speed advantages of using quadratic panels (see chapter 5). In this case, the singular source integral will be required for a triangular panel. Fortunately, this is identical to the integral of a single triangle for the nine node panels given by equation (3.39) without the summation over the other triangles (i.e $\kappa = 1$).

Beyond these comparison tests, the linear triangular panels are not used for the non-wake surfaces in the later simulations. The reasoning behind this decision is the inherent accuracy problems and an undesirable development cost in developing the adaptive scheme to deal with them.

3.5 Linear System of Equations

With the numerical methods in place to calculate the fundamental integrals defined by equations (2.93)–(2.95), the linear system established to solve the discretised integral equations is now described. To achieve this, the governing equations can be written using a 'per node' representation. Two systems are presented: first is the system of equations without the inclusion of a wake. This will just contain the Dirichlet and Neumann boundaries. The second system will include the wake surface and address the required adjustment to implement the Kutta condition at the trailing edge of the lifting surface.

Begin by defining the total number of nodes on S as N and the total number of elements as M. There are three types of unknowns on the nodes, (ϕ, ϕ_n or μ). The complete surface, S, is partitioned into separate surfaces, $S = S_D \cup S_N \cup S_K \cup S_W$ where S_D, S_N and S_W are the Dirichlet, Neumann and wake surfaces respectively. As linear panels are to be used on the wake, the Kutta strip, S_K, reduces from a surface to a line (the line along the trailing edge of the lifting surface). On S_D, ϕ is known and ϕ_n is sought. On S_N, ϕ_n is known, and ϕ is sought. On S_K, no information is known and μ is sought while on S_W, μ is known and no other information is required. Thus, no equations are written for nodes on S_W and the values there only contribute to the right hand side of the linear system. Also, μ on S_K is found via an auxiliary equation known as the Kutta condition and is written separately to the boundary integral equations. Thus the total number of collocation points in the boundary integral shall be known as N_c and N_c equations are written to determine the unknowns on S_D and S_W with N_k

additional equations for the number of nodes on S_K.

3.5.1 Wake-Free System

The formation of the linear system for the domain containing no wake is demonstrated. The governing equations are given by equations (2.89) and (2.90). Using equations (3.12) and (3.13), the full discrete equations can be derived.

For $P_i \in S_D$ the integral equation is written as

$$\sum_{m=1}^{M_N}\sum_{j=1}^{9}\phi_j D_j^{(m)} - \sum_{m=1}^{M_D}\sum_{j=1}^{9}\phi_{n,j} S_j^{(m)} = \\ -\alpha_i\phi_i - \sum_{m=1}^{M_D}\sum_{j=1}^{9}\phi_j D_j^{(m)} + \sum_{m=1}^{M_N}\sum_{j=1}^{9}\phi_{n,j} S_j^{(m)}, \quad (3.65)$$

and subsequently for $P_i \in S_N$ the equation is

$$\alpha_i\phi_i + \sum_{m=1}^{M_N}\sum_{j=1}^{9}\phi_j D_j^{(m)} - \sum_{m=1}^{M_D}\sum_{j=1}^{9}\phi_{n,j} S_j^{(m)} = \\ -\sum_{m=1}^{M_D}\sum_{j=1}^{9}\phi_j D_j^{(m)} + \sum_{m=1}^{M_N}\sum_{j=1}^{9}\phi_{n,j} S_j^{(m)}. \quad (3.66)$$

The difficulty with interpreting the above equations is that should a node lie within more than one panel (such as the corner of four adjoining panels) then it will appear in the above equations more than once. To reduce the level of complication, let

$$\sum_m$$

denote the summation over the number of panels containing the global integration point, defined by k, then define two global quantities

$$D_{i,k} = \sum_m D_k^{(m)}, \quad (3.67)$$

$$S_{i,k} = \sum_m S_k^{(m)}. \quad (3.68)$$

Using, equations (3.67) and (3.68) (following Otta *et al.*, 1992), the discretised equations for each field node in the system, i, form an $N_c \times N_c$ system together with the global integration nodes, k. Hence, the governing integral equation (given by equation (2.40)) can be written as

$$\alpha_i\phi_i + D_{i,k}\phi_k - S_{i,k}\phi_{n,k} = 0 \qquad i = k = 1, 2, \ldots, N_c. \tag{3.69}$$

The next task is to demonstrate how equation (3.69) can be used to form a linear system of the form

$$\mathcal{A}\mathcal{F} = \mathcal{B}. \tag{3.70}$$

Define the number of nodes on the Neumann surfaces to be N_N and the number of nodes on the Dirichlet surfaces to be N_D. The global integration point, k, can then be split into two groups dependant on which boundary it lies. Subsequently, in a similar manner to Otta *et al.* (1992), the coefficient matrix is described by

$$\mathcal{A}_{ik} = \begin{cases} (D_{i,k} + \delta_{i,k}\alpha_i) & k = 1, 2, \ldots, N_N, \\ S_{i,k} & k = N_N + 1, N_N + 2, \ldots, N_N + N_D, \end{cases} \tag{3.71}$$

where δ is the Kronecker delta. The right hand side of the linear system is given by

$$\mathcal{B}_i = \sum_{k=1}^{N_N} S_{i,k}\phi_{n,k} + \sum_{k=N_N+1}^{N_N+N_D} (D_{i,k} + \delta_{i,k}\alpha_i)\phi_k. \tag{3.72}$$

Thus, the linear system will have the form

$$\mathcal{A}_{ik} \left(\begin{array}{c} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_{N_N} \\ \hline \phi_{n,N_N+1} \\ \phi_{n,N_N+2} \\ \vdots \\ \phi_{n,N_N+N_D} \end{array} \right) = \mathcal{B}_i \tag{3.73}$$

by choosing $\mathcal{F}_i$ to be ordered such that

$$\mathcal{F}_i = \begin{cases} \phi_i & i = 1, 2, \dots, N_N, \\ \phi_{n,i} & i = N_N + 1, N_N + 2, \dots, N_N + N_D. \end{cases} \tag{3.74}$$

The linear system given by equation (3.73) can be inverted to give $\mathcal{F}$ and, thus, the unknown values of ϕ and ϕ_n.

3.5.2 System of Equations for Lifting Surfaces and the Kutta Condition

Implementation of the Kutta condition, described in section 2.4.4, is critical to the modelling of lifting flows using boundary element methods. This section describes the numerical implementation of the Morino type Kutta condition described by equation (2.59). Additionally, the linear system used to solve the boundary element problem given by equation (3.73) must be adjusted as a consequence of the Kutta condition whilst also including the influence of the wake surface.

Equations (2.91) and (2.92), which are the governing equations including the wake, can now be expressed making use of the elemental integrals given by equations (3.12), (3.13) and (3.58). Observe that for $P_i \in S_D$ the integral equation is written as

$$\sum_{m=1}^{M_N}\sum_{j=1}^{9} \phi_j D_j^{(m)} - \sum_{m=1}^{M_D}\sum_{j=1}^{9} \phi_{n,j} S_j^{(m)} = \\ - \alpha_i \phi_i - \sum_{m=1}^{M_D}\sum_{j=1}^{9} \phi_j D_j^{(m)} + \sum_{m=1}^{M_N}\sum_{j=1}^{9} \phi_{n,j} S_j^{(m)} + \sum_{m=1}^{M_W}\sum_{j=1}^{3} \mu W_j^{(m)}, \tag{3.75}$$

and subsequently for $P_i \in S_N$ the equation is

$$\alpha_i \phi_i + \sum_{m=1}^{M_N}\sum_{j=1}^{9} \phi_j D_j^{(m)} - \sum_{m=1}^{M_D}\sum_{j=1}^{9} \phi_{n,j} S_j^{(m)} = \\ - \sum_{m=1}^{M_D}\sum_{j=1}^{9} \phi_j D_j^{(m)} + \sum_{m=1}^{M_N}\sum_{j=1}^{9} \phi_{n,j} S_j^{(m)} + \sum_{m=1}^{M_W}\sum_{j=1}^{3} \mu W_j^{(m)}. \tag{3.76}$$

In light of the numerical Kutta condition, equations (3.75) and (3.76) do not provide the complete linear system for lifting surfaces. These equations assume that μ is known at every node on the wake, however, this is not the case as the Kutta nodes coincident to the trailing edge are considered unknowns at this stage of the calculation and must be determined by values from the lifting surface. In order to express these unknown values recall that the number of Kutta nodes is N_K and the number of nodes on the wake surface (including the Kutta nodes) is N_W. Expressing the governing equations in terms of the global integration index, k, the general equation is given by

$$\alpha_i\phi_i + D_{i,k}\phi_k - S_{i,k}\phi_{n,k} - W_{i,k}\mu_k = 0 \qquad \begin{cases} i = 1, 2, \ldots, N_N + N_D, \\ k = 1, 2, \ldots, N_N + N_D + N_W. \end{cases} \tag{3.77}$$

In light of equation (3.77), the various components of the linear system must be redefined with the Kutta condition included. Thus the coefficient matrix, $\mathcal{A}$, is now determined by

$$\mathcal{A}_{ik} = \begin{cases} (D_{i,k} + \delta_{i,k}\alpha_i) & k = 1, 2, \ldots, N_N, \\ S_{i,k} & k = N_N + 1, N_N + 2, \ldots, N_N + N_D, \\ W_{i,k} & \begin{aligned} k = {} & N_N + N_D + 1, N_N + N_D + 2, \ldots, \\ & N_N + N_D + N_K, \end{aligned} \end{cases} \tag{3.78}$$

and for the right hand side,

$$\mathcal{B}_i = \sum_{k=1}^{N_N} S_{i,k}\phi_{n,k} + \sum_{k=N_N+1}^{N_N+N_D} (D_{i,k} + \delta_{i,k}\alpha_i)\phi_k + \sum_{k=N_N+N_D+N_K+1}^{N_N+N_D+N_W} W_{i,k}\mu_k, \tag{3.79}$$

for $i = 1, \ldots, (N_N + N_D)$. This will produce a linear system of the form

$$\mathcal{A}_{ik} \begin{pmatrix} \phi_1 \\ \vdots \\ \phi_{N_N} \\ \hline \phi_{n,N_N+1} \\ \vdots \\ \phi_{n,N_N+N_D} \\ \hline \mu_{N_N+N_D+1} \\ \vdots \\ \mu_{N_N+N_D+N_K} \end{pmatrix} = \mathcal{B}_i \tag{3.80}$$

where $\mathcal{F}_i$ has been ordered such that

$$\mathcal{F}_i = \begin{cases} \phi_i & i = 1, 2, \ldots, N_N, \\ \phi_{n,i} & i = N_N + 1, N_N + 2, \ldots, N_N + N_D \\ \mu_i & \begin{aligned} i = {} & N_N + N_D + 1, N_N + N_D + 2, \ldots, \\ & N_N + N_D + N_K. \end{aligned} \end{cases} \tag{3.81}$$

The apparent difficulty with this linear system is that there are $N_N + N_D$ equations for $N_N + N_D + N_K$ unknowns. Thus an auxiliary equation is required for each N_K Kutta nodes. This is conveniently provided by the numerical Kutta condition. By labelling the node on the upper surface of the wing adjacent to the trailing edge to be KT and the similar node for the lower side to be KB, the value of μ_K for a node on the Kutta strip is given by

$$\mu_K = \phi_{KT} - \phi_{KB}. \tag{3.82}$$

Thus the additional equations in the linear system are defined by

$$\left.\begin{aligned} \mathcal{A}_{i,KT(i)} &= -1 \\ \mathcal{A}_{i,KB(i)} &= 1 \\ \mathcal{A}_{i,i} &= 1 \\ \mathcal{B}_i &= 0 \end{aligned}\right. \qquad i = N_N + N_D + 1, \ldots, N_N + N_D + N_K. \tag{3.83}$$

The linear system now has $N_N + N_D + N_K$ equations for $N_N + N_D + N_K$ unknowns.

3.6 Special Treatment of Boundaries

For the domain illustrated in figure 2.1, the formulation of the boundary element method would require that the input, output, left, right, and bottom boundaries should be discretised in a manner similar to solid bodies within the domain, using panels having a Neumann boundary condition. For the input and output boundaries the value of ϕ_n is set equal to the free stream velocity, $\mathbf{U}_x$, and for the side and bottom boundaries a no crossing condition is set requiring, $\phi_n = 0$.

The disadvantage of such an approach to discretising the channel boundaries is that more panels must be used, and thus computational time must be expended,

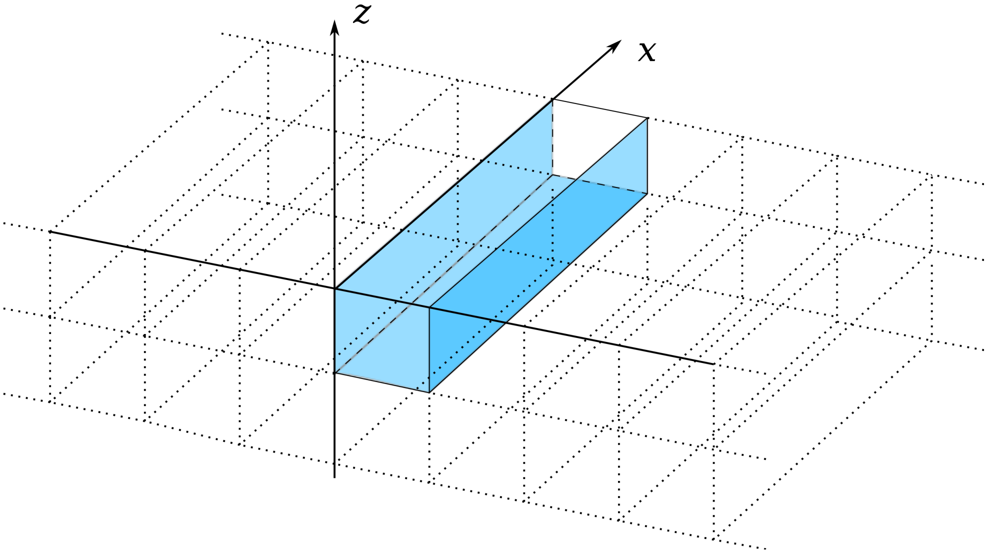


Figure 3.12: Images of the original channel as a result of reflective boundaries for the bottom, left and right channel walls.

recalling that for N nodes the influence matrix will be full and of order N^2. Thus it is advantageous to reduce the number of extraneous panels to a minimum.

The input and output boundaries can be discounted by using the perturbation potential formulation rather than the full potential formulation. The perturbation potential formulation is discussed in section 2.5, noting that care must be taken to correctly calculate the solid angle when the inlet and outlet boundaries do not form part of the solution. For the left, right and bottom boundaries the ideal approach is to use reflective boundaries for each. This approach is certainly feasible for the bottom boundary, and for one side boundary, however, when attempting to apply both side boundaries, the level of difficulty increases substantially, as discussed in the next section.

3.6.1 Reflective Boundaries

Using reflective boundaries on both the left and right sides of the channel results in infinite images of the channel extending in both the positive and negative y-directions, as can be seen in figure 3.12. Within the figure, the no-crossing boundaries of the original domain are shaded.

The problem of infinite repetition for wave problems is not novel. In a similar manner, when periodic boundaries are required on both the left and right hand sides of the domain, an infinite number of ‘copies’ of the domain are produced. To apply the periodic boundaries to the original domain, the influences of all of these copies must be included into the Greens functions of the boundary integral equations.

A great deal of research was undertaken in the 1990s in order to solve these problems. Works by Breit (1991), Newman (1992) and Xü (1992) devised Greens functions that satisfied periodic boundary conditions in either one (singly) or two (doubly) periodic directions. The work of Newman concentrated on increasing the rate of convergence of the resulting infinite series of images.

A periodic representation may be ideal for wave simulations using boundary elements, however, as the primary interest in the current simulation is flow in a channel with solid planar bottom and side boundaries, reflective rather than periodic conditions must be imposed. The resulting series is not dissimilar to that of the periodic problem, nonetheless, the focus of this project did not allow time for the solution to be derived. Thus, a method was sought were some discretisation could be reduced while avoiding the need to solve an infinite series of Greens functions. The result is the computational domain illustrated in figure 3.13. The influences from the point Q and the image points, Q', Q'', and Q''' can all be integrated into the Greens function without the need to solve any infinite series. Note that one side wall must still be discretised (the non-shaded, $y = 0$, side wall).

To derive the Greens function for the domain illustrated in figure 3.13 enforce no-crossing boundaries at $y = y_r$ and $z = -d$ (note from figure 3.13 that $y_r < 0$). Three image domains are created which are positioned as in figure 3.13. Now consider the influence from a particular point, Q, in the original domain, in addition to the influence from the images, Q', Q'', and Q'''. The distances from each of these points are given by $\mathbf{R}$, $\mathbf{R}'$, $\mathbf{R}''$ and $\mathbf{R}'''$ and the associated Greens function are G, G', G'', and G'''. The values of ϕ and ϕ_n are identical at each point. Thus, writing the contribution to the integral equation from these points gives

$$\begin{aligned} &\phi_n G - \phi G_n + \phi_n G' - \phi G'_n + \phi_n G'' - \phi G''_n + \phi_n G''' - \phi G'''_n \\ &= \phi_n \left(G + G' + G'' + G''' \right) - \phi \left(G_n + G'_n + G''_n + G'''_n \right). \end{aligned} \tag{3.84}$$

Provided the field point, P, does not lie on the reflective boundary, equation (3.84) demonstrates that the reflection can be represented by incorporating the influence of

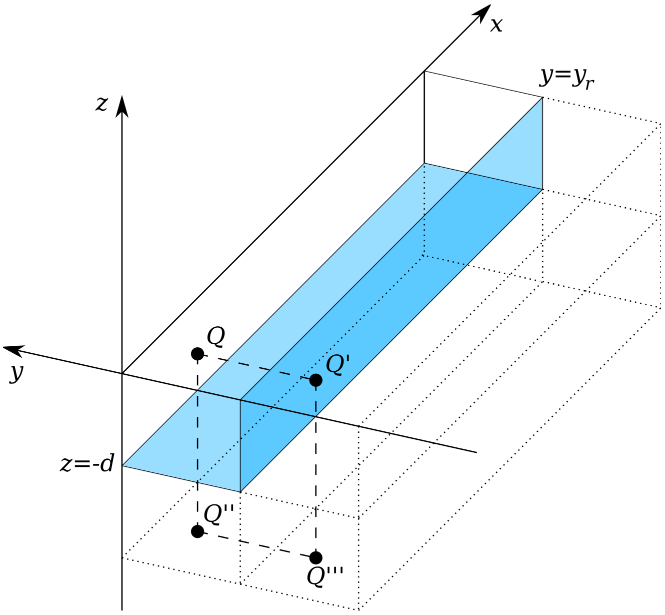


Figure 3.13: Domain and integration points for images generated by reflection in the bottom and right channel wall boundaries.

the images into the Greens function (the cases where P lies on one or both reflective boundaries is dealt with later). Thus, a combined Greens function is sought, such that,

$$\begin{aligned} G^C &= G + G' + G'' + G''' \\ &= \frac{1}{\mathbf{R}} + \frac{1}{\mathbf{R}'} + \frac{1}{\mathbf{R}''} + \frac{1}{\mathbf{R}'''} \\ &= \frac{1}{P-Q} + \frac{1}{P-Q'} + \frac{1}{P-Q''} + \frac{1}{P-Q'''}. \end{aligned} \tag{3.85}$$

Figure 3.14 shows the image points in the (z, y) plane and the associated distances to the image points. The positions of the image points are given by $Q = (x_q, y_q, z_q)$, $Q' = (x_q, y'_q, z_q)$, $Q'' = (x_q, y_q, z'_q)$ and $Q''' = (x_q, y'_q, z'_q)$. Thus y'_q and z'_q must be determined subject to the position of the right channel wall, y_r and the depth of the channel, d.

The resulting formula are

$$y'_q = 2y_r - y_q \tag{3.86}$$

$$z'_q = -2d - z_q \tag{3.87}$$

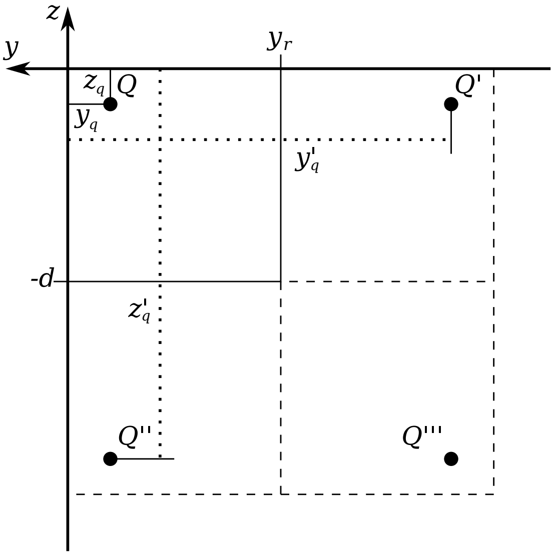


Figure 3.14: Distances to the integration point and its images for the channel in the (y, z) plane.

assuming that the depth, $d > 0$. To illustrate the effect in (s, t) space consider

$$\begin{aligned} y'_q(s,t) &= \sum_{j=1}^{9} \psi_j \left(2y_r - y_j\right) \\ &= 2y_r \sum_{j=1}^{9} \psi_j - \sum_{j=1}^{9} \psi_j y_j \\ &= 2y_r - \sum_{j=1}^{9} \psi_j y_j \end{aligned} \tag{3.88}$$

and thus $\mathbf{R}'$ is given by

$$\left(\sum_{j=1}^{9} \psi_j x_j - x_p,\ 2y_r - \sum_{j=1}^{9} \psi_j y_j - y_p,\ \sum_{j=1}^{9} \psi_j z_j - z_p \right) \tag{3.89}$$

with corresponding length, r'.

$\mathbf{R}''$ and $\mathbf{R}'''$ are defined similarly and the combined source influence is given by

$$S_j^{(m)} = \iint_{\Sigma_m(s,t)} \psi_j(s,t) J(s,t) \left(\frac{1}{r(s,t)} + \frac{1}{r'(s,t)} + \frac{1}{r''(s,t)} + \frac{1}{r'''(s,t)} \right) ds\, dt. \tag{3.90}$$

Similarly, the doublet influence is given by

$$\begin{aligned} D_j^{(m)} = -\iint_{\Sigma_m(s,t)} & \psi_j(s,t)J(s,t)\times \\ & \left(\frac{\mathbf{n}\cdot\mathbf{R}}{r^3(s,t)} + \frac{\mathbf{n}\cdot\mathbf{R}'}{(r')^3(s,t)} + \frac{\mathbf{n}\cdot\mathbf{R}''}{(r'')^3(s,t)} + \frac{\mathbf{n}\cdot\mathbf{R}'''}{(r''')^3(s,t)}\right) ds\, dt. \end{aligned} \tag{3.91}$$

The self influence case is straightforward to handle. The self influences are calculated using equations (3.47) or (3.48) and then the non-singular influences from the image points added to give the total influence.

Slightly more complicated is the case where the integration point, Q, lies on one or both of the reflective boundaries. In this case, certain images no longer contribute. For example, should Q lie on the right wall then the influence from Q' and Q''' is removed. If it lies on the bottom then the influence from Q'' and Q''' is ignored and, finally, if Q lies on the intersection of the the two boundaries then only the non-reflected influence is required. This holds true for the non-singular influences and the singular source influence. However, the case is different for the singular doublet influence due to the solid angle calculation.

In a similar fashion to the source and doublet influence, the calculation for the solid angle using reflective boundaries must also include the influences from the images. The solid angle for the domain given in figure 3.14 is

$$\begin{aligned} \alpha(P_i) = \sum_{m=1}^{M_s} \iint_{\Sigma_m(s,t)} & \frac{(j_1 r_1 + j_2 r_2 + j_3 r_3)}{r^3(s,t)} + \frac{(j_1 r_1' + j_2 r_2' + j_3 r_3')}{(r')^3(s,t)} + \\ & \frac{(j_1 r_1'' + j_2 r_2'' + j_3 r_3'')}{(r'')^3(s,t)} + \frac{(j_1 r_1''' + j_2 r_2''' + j_3 r_3''')}{(r''')^3(s,t)}\, ds\, dt \end{aligned} \tag{3.92}$$

when P_i is not on the reflective boundaries. If P_i is on the right side wall then the r' and r''' terms are dropped and if it lies on the bottom then the r'' and r''' terms are dropped. If P_i lies on the intersection between the two reflective surfaces then just the original r term remains.

In the case of the singular doublet influence, the first term in the solid angle, equation (3.92), will cancel with the matching term from equation (3.42) with only the contributions from the reflective images remaining. Thus, the singular doublet influence for the reflective kernel is given by

$$
\begin{aligned}
D_j^{(m)} = &-2\sum_{\mu=1}^{\kappa}\int_0^1\int_0^1 \frac{\varphi_j(\rho_1,\rho_2)\mathcal{K}(\rho_1,\rho_2)A_\mu}{R^3(\rho_1,\rho_2)}\,d\rho_1\,d\rho_2 \\
&- \iint_{\Sigma_m(s,t)} \psi_j(s,t)J(s,t)\left(\frac{\mathbf{n}\cdot\mathbf{R}'}{(r')^3(s,t)}+\frac{\mathbf{n}\cdot\mathbf{R}''}{(r'')^3(s,t)}+\frac{\mathbf{n}\cdot\mathbf{R}'''}{(r''')^3(s,t)}\right)ds\,dt \qquad (3.93)
\end{aligned}
$$

with the image kernels being removed depending on the position of P_i in a similar fashion as previously.

3.6.2 Double Nodes

The “L-shaped” boundary solution described in the previous section is a useful method for reducing the amount of discretisation required for modelling the channel. However, for the remaining channel wall (and when reflective boundaries are not used) there are difficulties when modelling the intersection of surfaces with differing or similar boundary conditions. This problem is particularly prevalent where the free surface boundary, having a Dirichlet boundary condition, meets a side wall or inlet boundary, which have Neumann boundary conditions. In this case enforcing the condition that the free surface must remain attached to the wall also adds increased difficulty.

An excellent review of these *corner problems* is available in Forehand (1998, section 2.3.2). The focus of his review is predominantly two-dimensional methods, however, three-dimensional techniques for dealing with these issues are also covered. In particular the papers of Grilli *et al.* (1989) and Grilli and Svendsen (1990) are outlined in Forehand (1998) and similar techniques are applied to the present method.

At a moving boundary, the discontinuity between ϕ_n on the free surface and ϕ_n on the boundary can lead to a singularity in the solution. Although there are no moving boundaries other than the free surface in the current method, it is clear that similar methods used to correct the singularity identified at a moving boundary can be used to correct the problems at a boundary side wall. Due to implementation problems, formal results from the channel side wall correction do not form part of this thesis. The theoretical background is available in appendix D, nonetheless. For the problems investigated in this thesis, which do not exhibit a significant cross-flow component, the free-surface nodes nearest a channel wall are simply translated back to that wall at each time-step. Clearly, this approximation has significant room for improvement in future work.

Also described by Grilli and Svendsen (1990) is the concept of *double nodes.* Double nodes imply that at a meeting of two discretised surfaces, instead of the elements on either surface sharing nodes at the intersection, both surfaces have their own coincident overlapping nodes. For a three dimensional domain (with all the channel boundaries discretised) there exists the possibility of having triple nodes. Double and triple nodes are illustrated in figure 3.15.

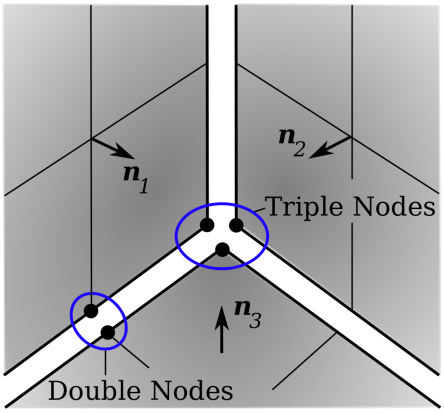


Figure 3.15: Double and triple nodes at the meeting of surfaces with differing normals.

The double (or triple) node representation encompasses the different normal vectors within the elements on the intersecting surfaces (i.e. $\mathbf{n}_i, i = 1, 2, 3$ in figure 3.15). Each double or triple node has the same spatial position as its coincident nodes and each node has its own equation. The form of these equations depend on which conditions the intersecting boundaries have. True of all double nodes is that their ϕ_n values will differ and that each pair or triplet of nodes must have equal or continuous ϕ explicitly imposed. The two cases which occur for double nodes within the present simulation are now described.

Dirichlet-Neumann Intersections

This case occurs when the free surface boundary intersects a channel side wall or the discretised inlet/outlet boundaries. On the free surface the potential at the node $I1$, ϕ_{I1}, is known along with the normal derivative, $(\phi_{I2})_n$, at the corresponding double node on the Neumann boundary, which is denoted $I2$. By enforcing continuity at the double nodes, the unknown ϕ_{I2} is set equal to ϕ_{I1}. Although this implies that the equation for the point $I2$ is unnecessary, it is advisable to maintain a row in the linear system by rewriting the equation for $I2$ to explicitly enforce the continuity of ϕ, whilst setting the coefficients of $I2$ to zero in the other equations.

As $I1$ and $I2$ are coincident (albeit belonging to different surfaces) it follows that

$$S^{(m)}_{I1,k} = S^{(m)}_{I2,k} \qquad D^{(m)}_{I1,k} = D^{(m)}_{I2,k} \qquad \alpha_{I1} = \alpha_{I2}. \tag{3.94}$$

Thus, the modifications to equation (3.69) for a Dirichlet-Neumann intersection using double nodes $I1$ on the Dirichlet boundary and $I2$ on the Neumann boundary is enforced by the requirement for the continuity of ϕ. The equation which replaces the equation for the $I2$ node is

$$\mathcal{M}\phi_{I2} = \mathcal{M}\phi_{I1}, \tag{3.95}$$

where

$$\mathcal{M} = \max(\mathcal{K}_{ii}), \tag{3.96}$$

and $\mathcal{K}_{ii}$ is the value of the leading diagonal of the matrix for point i.

Neumann-Neumann Intersections

This case occurs when Neumann boundaries (such as the side walls, bottom, inlet or outlet) intersect. Using $I1$ and $I2$ to represent the double nodes on either side of the boundary, the boundary integral equation for these points will be identical as the unknown $\phi_{I1} = \phi_{I2}$ is sought for both points. In this case, the linear system will be singular, similar to the Dirichlet-Neumann case, and so the solution at just one of the nodes, say $I1$, is calculated using the boundary integral equation and continuity of ϕ is used to replace the equation for the other node. As ϕ appears on the left hand side of the linear system for a Neumann boundary, the equation for $I2$ is rewritten as

$$\mathcal{M}\phi_{I2} - \mathcal{M}\phi_{I1} = 0. \tag{3.97}$$

Triple Nodes

There are two possible boundary condition combinations of triple nodes where three surfaces with distinct normals meet for this simulation. These are a Neumann-Neumann-Dirichlet intersection or a Neumann-Neumann-Neumann intersection. The methods for dealing with these nodes are identical to the double node cases except that continuity of ϕ must additionally be enforced for the extra node, $I3$. Thus the boundary integral equation for $I3$ should be replaced with

$$\mathcal{M}\phi_{I3} = \mathcal{M}\phi_{I1} \tag{3.98}$$

for the Neumann-Neumann-Dirichlet case (assuming $I1$ is the Dirichlet node). For the Neumann-Neumann-Neumann case the equation should be

$$\mathcal{M}\phi_{I3} - \mathcal{M}\phi_{I1} = 0, \tag{3.99}$$

assuming $I1$ is the node for which the boundary integral equation is implemented. Note that continuity between $I3$ and $I2$ could also be enforced and the author believes this to be equivalent to implementing equations (3.98) and (3.99).

Chapter 4

Lagrangian Stage Implementation

4.1 Introduction

This chapter describes the Lagrangian stage of the mixed Eulerian-Lagrangian approach used for the time-stepping boundary element method. By this stage, the Eulerian system has been solved and all field values are known at all points. With this information the velocities can then be calculated on the dynamic surfaces which can then be time-stepped using the chosen boundary conditions and an appropriate integration scheme.

This process is well understood for free surface simulations which employ boundary element methods. Therefore, the bulk of this chapter is concerned with the problem of advancing the position of the wake in time. This is the most challenging part of lifting solutions using boundary element methods and a problem not normally tackled with higher order elements. If they are used then the interpolation of the calculated velocities from the element centres to the edge nodes (Pyo and Kinnas, 1997) is the typical compromise. This approximation is enforced due to the need to regularise the hypersingular boundary element equation for velocity (equation (2.74)) at panel edges.

Since the publication of Guiggiani *et al.* (1992) a method for directly evaluating the hypersingular boundary integral equation at panel edges has been available. The method developed by Guiggiani *et al.* relies upon using a Taylor series expansion to reduce the order of the singularity at the point of interest. Such an expansion is only applicable subject to having C^2 continuity at the point. Thus, the traditional piecewise elements of many boundary element formulations are not applicable to this method at edges and corners. C^2 continuity can be guaranteed by utilising B-spline approximations and the work of Guiggiani *et al.* has been applied to lifting surfaces by

Bernasconi and Richelsen (2008). However, B-spline approximations to the boundary integral equations provide a greater challenge to producing a consistent level of accuracy over more simple interpolants due to the difficult design of adaptive integration schemes (see Xü, 1992).

The recent work of Gray *et al.* (2004b) was the first to provide a method for the direct evaluation of the hypersingular boundary integral equation at the edges and corners of high order continuous three-dimensions elements which lack C^2 continuity at the intersections. His regularisation using the *Galerkin boundary element method* has been successfully applied to problems of fracture mechanics (Paulino and Sutradhar, 2006) and has a general formulation that is transferable to any model utilising a boundary integral formulation. In particular, the method has been shown to be applicable to cracks. A crack is modelled as a discontinuity in displacement, ϕ, identical to the discontinuity of velocity potential, ϕ, used to model a vortex sheet. To the author's knowledge, this is the first application of the method of Gray *et al.* to potential flow and in particular to the direct evaluation of the velocity on vortex wakes. The Galerkin method is derived in section 4.3 and its applicability to vortex sheets is also validated there.

Prior to the description of the Galerkin method, the calculation of velocity on the free surface using numerical differentiation is described. Once all of the velocities on the surfaces have been calculated then these surfaces can be advanced in time subject to their boundary conditions. The final section of this chapter considers the integration schemes commonly used to carry out the time-stepping and introduces a number of metrics which can be used to dynamically control the time-step.

4.2 Numerical Differentiation

The most common method of calculating a gradient (such as velocity) on a surface within the domain of a boundary element method is numerical differentiation. In particular, once the values of ϕ_n on the free surface have been calculated, numerical differentiation can be applied to the values of ϕ to calculate the complete velocity field upon the surface. This information is then available for time-stepping or for inspection. The general process for extracting the velocity components from the parametric derivatives of ϕ is described by Xü (1992). The method for calculating the gradient in the parametric directions is described by Forehand (1998, appendix D).

Following Xü (1992, sec. 4.8), a general smooth surface $F : F(x, y, z) \in \mathbb{R}^3$ is described by an array of collocation points at

$$\mathbf{x}_{uv}, \text{ for } u = 0, 1, 2, \ldots, U \text{ and } v = 0, 1, 2, \ldots, V,$$

where U and V are integers. The curvilinear coordinate system, parameterised by u and v, allows any point, $\mathbf{x} = \mathbf{x}(u, v)$, on F to remain uniquely defined should the surface become multivalued as occurs during wave breaking. On this surface, the normal vector is defined in a similar manner to equation (3.6), with $\mathbf{x}_u$ and $\mathbf{x}_v$ replacing Q_s and Q_t. The components of these derivatives and the normal can be written

$$\begin{aligned} \mathbf{x}_u &= (x_u, y_u, z_u)\,, \\ \mathbf{x}_v &= (x_v, y_v, z_v)\,, \\ \mathbf{n} &= (n_x, n_y, n_z) \end{aligned} \tag{4.1}$$

where the subscripts x, y and z denote components and the subscripts u and v denote partial derivatives.

As $\mathbf{x}(u, v)$ is a global parametric representation of F, the field variables on F can be represented similarly and thus the chain rule provides the following relations for the derivatives of ϕ:

$$\begin{aligned} \phi_u &= \phi_x x_u + \phi_y x_u + \phi_z z_u, \\ \phi_v &= \phi_x x_v + \phi_y y_v + \phi_z z_v, \\ \phi_n &= \phi_x n_x + \phi_y n_y + \phi_z n_z. \end{aligned} \tag{4.2}$$

The linear system described by the equations given in (4.2) can be written as

$$\begin{pmatrix} \phi_x \\ \phi_y \\ \phi_z \end{pmatrix} \begin{bmatrix} x_u & y_u & z_u \\ x_v & y_v & z_v \\ n_x & n_y & n_z \end{bmatrix} = \begin{pmatrix} \phi_u \\ \phi_v \\ \phi_n \end{pmatrix}. \tag{4.3}$$

In order to solve equation (4.3), $\mathbf{x}_u$ and $\mathbf{x}_v$ must be calculated for all the collocation points. Xü discusses in detail the method for calculating these parametric gradients and found that a 7 point finite difference formula was comparable in accuracy to a bi-cubic spline fit with much reduced computational effort. There is also reference to the inherent drop in accuracy at edges and corners where skewed rather than central differencing schemes must be used. To compensate, more points can be used to increase the accuracy although this may not always be practical or beneficial dependant on the topology of F.

In general a 5 point differentiation scheme should suffice with the possibility of increasing to 7 points at edges and corners as described in Forehand (1998, appendix D). The coefficients used for this differencing scheme are also used for the coefficients in the boundary correction scheme discussed in appendix D, and equation (D.15).

An alternative method for evaluating the free surface velocities is to implement the Galerkin boundary integral approximation to the hypersingular velocity equation. The implementation of this equation should improve the accuracy over numerical differentiation, particularly in the corners, but due to time restraints the more basic numerical integration is used on the free surface. Numerical differentiation is inapplicable to vortex sheets (due to the reasons discussed in section 2.6) so the Galerkin method provides a necessary tool for evaluating the velocity on the wake.

4.3 Galerkin Gradient Method

A *Galerkin boundary element method*, as developed by Gray (1995); Gray *et al.* (1998, 2004a,b), can be applied to solve for the velocity on the wake vortex sheet by regularising equation (2.77). The Galerkin equation differs from an ordinary boundary element equation (in the form of equation (2.40)) by integrating over the surface a *second time* using the interpolation (or shape) functions as the weightings for the discretisation, as discussed in section 2.7. Thus, as in Gray *et al.* (2004b), the Galerkin form of equation (2.80) is

$$\int_S \hat{\psi}_i(P)\frac{\partial \phi(P)}{\partial \mathbf{E}_k}\,dP = \left\{\lim_{P_I\to P} - \lim_{P_E\to P}\right\}\int_S \hat{\psi}_i(P)\,\frac{1}{4\pi}\int_S \frac{\partial G}{\partial \mathbf{E}_k}\phi_n(Q) - \phi(Q)\frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}\,dQ\,dP, \quad (4.4)$$

where $\hat{\psi}_i$ are the non-zero shape functions at a particular node, P_i.

The additional surface integration on the right hand side of equation (4.4) is key to the success of this technique. As there are now four, rather than the normal two, integrations to be carried out, additional analytical manipulations can be made to the equations. Through these manipulations the divergent terms in the hypersingular kernel can be explicitly identified. Then, by taking the difference in the inner and outer limits the divergent terms are cancelled analytically. The remaining non-singular integrals can be solved numerically. Unfortunately, the additional surface integration on the left hand side of the equation forms a linear system which must be solved in order to recover the gradient at the collocation points.

The above technique, which was developed for three dimensional domains in Gray *et al.* (2004a,b), is currently only applicable to linear triangular elements. Given that the present simulation uses biquadratic panels on the lifting bodies, it would be preferable if these could also be used for their wakes. Professor Gray has indicated that the technique is applicable to higher order elements, yet, the time required to develop a Galerkin method for such elements was beyond the scope of this project. In addition, as this is the first time that this method has been applied to vortex wake sheets, evaluation of the method using linear elements will provide an indication as to the value of developing the method for higher order elements. Thus, in the manner of Gray *et al.* (2004a,b), the three nodal linear triangular element with shape functions defined by equation (3.56) is used to discretise equation (4.4). The element is shown in figure 3.10. For ease of reading the tildes have been dropped from the 'ψ's representing the shape functions for the element.

The potential $\phi(Q)$ is interpolated by

$$\phi(s^*, t^*) = \sum_{j=1}^{3} \phi(Q_j)\psi_j(s^*, t^*), \tag{4.5}$$

where s^* and t^* represents the parameter space for the Q integration. 's' and 't' now describe the parameter space for the P integration.

It is clear that, in a similar manner to equation (2.45), the ϕ_n integrals in equation (4.4) will cancel when the two sides of the wake are brought together. Thus, the task here is to establish the effect of the Galerkin limiting process on the hypersingular part. The problem, using a particular pair of elements, Σ_P and Σ_Q, is to regularise

$$\left\{\lim_{P_I \to P} - \lim_{P_E \to P}\right\} \phi(Q) \int_{\Sigma_P} \psi_l(P) \int_{\Sigma_Q} \psi(Q) \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}\, dQ\, dP \tag{4.6}$$

where

$$\frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}(P,Q) = \frac{1}{4\pi}\left(\frac{\mathbf{n}\cdot\mathbf{E}_k}{r^3} - 3\frac{(\mathbf{n}\cdot\mathbf{R})\,(\mathbf{E}_k\cdot\mathbf{R})}{r^5}\right) \tag{4.7}$$

and ψ_l is the local weighting for P within the element Σ_P.

To simplify the notation, the difference in the internal and external limits of equation (4.6) shall be assumed until explicitly calculated within the analysis. Facilitating the opposing limits for equation (4.6) is the definition of the distance vector, $\mathbf{R}$. To distinguish between the internal and external limits let

$$\mathbf{R}^{\pm} = Q - (P \pm \epsilon\mathbf{N}), \tag{4.8}$$

where $\mathbf{N}$ is the normal to element Σ_P and is equivalent to $\mathbf{n}^{+}$ as defined in figure 2.5. The '±' in equation (4.8) allows P to approach the boundary from the external and internal domains, respectively, as $\epsilon \to 0$. A few important relations are resultant from these definitions. Notably that

$$\mathbf{N}\cdot\mathbf{R}^{\pm} = \mp\epsilon; \tag{4.9}$$

$$\mathbf{E}_k\cdot\mathbf{R} = R_k, \tag{4.10}$$

being careful to remember that the subscript k corresponds to the particular directional component sought.

What follows is a detailed account demonstrating the analysis by which equation (4.6) is reduced to the calculation of two singular quantities. The first is known as the *coincident integral* and describes the integration of a panel that contains P being coincident to the integration panel. The second is known as the *edge adjacent integral* and refers to the integration of a panel containing P that shares an edge with the integration panel. The *vertex adjacent integrals* and the non-singular integrals are completely cancelled. A vertex adjacent integral is when a panel containing P shares a common vertex with the integration panel. These concepts are depicted in figure 4.1. These cancellations are another major advantage of this approach; the vast majority of the integrals (i.e. the non-singular and vertex adjacent integrals) have zero contribution and, therefore, only the small number of local edge-adjacent and coincident integrals contribute to the right hand side of equation (4.4).

The coincident and edge adjacent integrals are developed in full in the following two

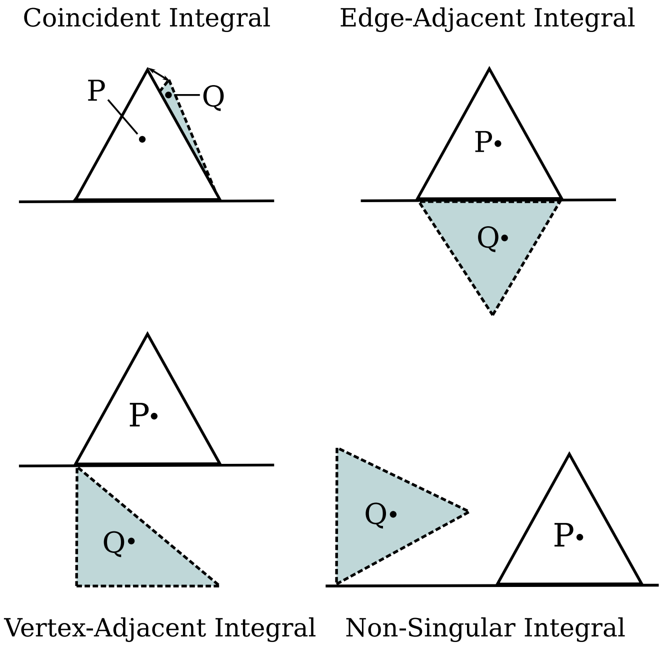


Figure 4.1: Galerkin integral types.

sections. The most singular part of the coincident integral was derived in Gray *et al.* (2004b), with reference to the similar derivation presented in Gray *et al.* (2004a). For completeness, the derivation is presented in section 4.3.1 in full, following the method of Gray *et al.* (2004b). All of the integrals omitted from Gray *et al.* (2004b) are included utilising techniques from Gray *et al.* (2004a) to regularise them. Additional detail relating to the derivation of all of the integrations are available in the appendices. Also, the method to further reduce the dimension of the remaining numerical integrations, that was presented in Gray *et al.* (2006), is applied here to the coincident integrals for the gradient for the first time. Section 4.3.2 demonstrates the regularisation of the edge adjacent integral. This is not explicitly demonstrated in Gray *et al.* (2004b) and is presented here using the techniques presented to solve the related integral in Gray *et al.* (2004a). Throughout both sections, symbolic computing was utilised to aid in the production of the expressions. As both derivations are quite involved, a summary of the techniques used in the regularisation of each type of integral is presented below.

Following the development of the expressions for calculating the gradient of ϕ, the novel approach to apply these results to a vortex sheet is presented in section 4.3.3. The key to demonstrating the applicability is establishing a suitable limiting procedure for the two ‘sides’ of the open wake as it is closed together. By linking the Galerkin limit to the top and bottom of the wake with the limit closing the top and bottom of the wake, it can be shown that ϕ can be replaced by the difference in ϕ between the two surfaces, or $\mu = [\phi]$, the known quantity on the wake. Thus, just the application of a constant coefficient to the expressions developed in section 4.3.1 and section 4.3.2 is required to calculate the gradient on the wake. Section 4.3.4 goes on to establish the validity of the Galerkin method at the wake edges where the 'closing' of the wake is less well defined. By examining the form of the integrals as the open wake edge is approached, the importance of the wake edges having zero $[\phi]$ is highlighted. In addition, despite the fact that zero $[\phi]$ allows the Galerkin method to be applied directly to the wake edges, the reasons why this is *not* in practice possible are discussed at the end of the section.

Coincident Integration

A coincident integration occurs when the field and integration panels are the same. The strategy for identifying and cancelling the divergent terms in this situation requires that two polar transformations are made. The first polar transform is centred on the parameter space of the integration panel, (s, t). Immediately after the polar transformation, the external limit is subtracted from the internal limit, although at this stage the limit to the surface, $\epsilon \to 0$, can not be taken. The integral's two innermost integrations are now in terms of the polar coordinates ρ and θ and there remains two integrals to be solved, one of order ρ^{-3} and the other of order ρ^{-2}. The impact of the first polar transformation is that the domain must be split into three subtriangles, and each subtriangle solved separately. Fortunately, the result from one subtriangle is applicable to the other two, by rotating the element.

The next stage is to evaluate the ρ integral. The process is different for the integrals of different orders of ρ. For the order ρ^{-2} integral, ϵ may be set to zero immediately after the initial integration has been carried out. The remaining θ integration is directly integrable, noting that singularities are present at $s = \pm 1$; as these are at the extreme bounds of the integration, they are of no concern. In contrast, the result, when evaluated directly, is unwieldy and, so, a substitution and second polar transformation is carried out to reduce the integral to two dimensions. This, once again, splits the integral into a further three parts. Although numerical quadrature can be applied to the two dimensional integral resulting from the regularisation, it was chosen instead to apply the technique developed in Gray *et al.* (2006) (which demonstrates how the coincident integrals in Gray *et al.* (2004a) can be further reduced) to reduce the integrals to one dimension by analytical means. This is achieved by reordering the integral so that the more tractable s-integral is innermost, then evaluating the integral over each triangular domain in turn. The resulting one dimensional integrals are detailed in section F.1.

In a similar fashion to the order ρ^{-2} integral, a substitution and second polar transform must also be carried out for the order ρ^{-3} integral. This step is much more important in this case, as ϵ may not be set to zero prior to the transformations. In fact, as was shown in Gray *et al.* (2004b), the additional step of subtracting a zero valued integral of θ, containing ϵ^{-1}, from the integral must be taken before the second transformation. Once the substitution and polar transform have been carried out, further manipulation is required before ϵ may be set to zero. An application of the binomial theorem to the resulting expression, and by discarding the resulting ϵ^4 term, finally produces an expression for which the limit to the surface may be taken. Once again, the results of reducing the remaining two dimensional integrals to a single dimension are detailed in section F.2.

Edge Adjacent Integration

An edge adjacent integral occurs when the field and integration panels share an edge. The regularisation of the edge adjacent integral follows a similar process to that of the coincident integral, in that two polar transformations are required. The two elements share the $t = t^* = 0$ edge and, thus, the first polar transform is centred at $s = s^*, t = 0$. This results in the integral being split into two domains. In contrast to the coincident integral, the difference between the internal and external limits are *not* taken at this point. This is due to the forms of the distance vector $\mathbf{R}$ which differs for each limit; at this stage, each limit is advanced separately.

Prior to any of the integrals being evaluated, the second polar transform is applied to the integrals, after the t-integral is swapped with the θ-integral to become the second innermost. This is possible, in part, because the limits of the θ-integral are independent of t. The next polar transform is then undertaken, which splits the domain into two more parts. Once this stage is reached, it is critical to take the difference in the interior and exterior limits. If this is not done, then divergent $\log(\epsilon)$ terms will persist. Now the innermost integral may be evaluated and following that the limit as $\epsilon \to 0$ may be taken. This deals with the divergent terms in the integral. The remaining integral may be further simplified by observing that the s-integral is independent of the other two integrals and, as such, may be evaluated first. In addition, the separate domains of the two polar transformations may be reunified, leaving a single two dimensional integral to be solved numerically over a simple domain.

4.3.1 Coincident Integration

The first case of singular integration to be tackled is when both Σ_P and Σ_Q are the same and this is known as the coincident integral. The derivation of the most singular integral in this section can be found in Gray *et al.* (2004b). The development of the integral is presented here in detail along with the less singular integral, making reference to the techniques in Gray *et al.* (2004a) where necessary. Symbolic computing was used to ensure the accuracy of the resulting expressions.

To begin, following Gray *et al.* (2004b), transform equation (4.6) to the parameter space, which gives

$$\sum_{j=1}^{3} \phi(Q_j) \int_{-1}^{1} \int_{0}^{\sqrt{3}(1-|s|)} \psi_l(s,t) \times \\ \int_{-1}^{1} \int_{0}^{\sqrt{3}(1-|s^*|)} \psi_j(s^*,t^*) J_P^2 \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}} \, dt^* \, ds^* \, dt \, ds. \quad (4.11)$$

The Jacobian of the transformations, J_P, is the difference in area between the physical and parameter space triangles and is constant. The J_P^2 term arises as the Jacobians for Σ_P and Σ_Q are the same for the coincident integral, i.e. $J_Q = J_P$.

The regularising of equation (4.11) involves two polar transformations, the first of which transforms the (s^*, t^*) parameter space into (s, t). In particular (s^*, t^*) is replaced by a polar coordinate system centred at (s, t) defined by

$$s^* - s = \rho \cos(\theta), \quad (4.12)$$

$$t^* - t = \rho \sin(\theta). \quad (4.13)$$

The resulting polar coordinates are illustrated in figure 4.2. The effect of this transformation on $\mathbf{R}$ is demonstrated in section E.1, the result of which is

$$\mathbf{R}^{\pm} = \begin{pmatrix} a_1\rho \mp \epsilon N_1 \\ a_2\rho \mp \epsilon N_2 \\ a_3\rho \mp \epsilon N_3 \end{pmatrix}. \tag{4.14}$$

The coefficients, a_1, a_2 and a_3, are also defined in section E.1. As $\mathbf{N} \cdot \mathbf{R} = 0$, $r^2 = (a^2\rho^2 + \epsilon^2)$ where $a^2 = a_1^2 + a_2^2 + a_3^2$. Also note that the coefficients of ρ in $\mathbf{R}$, a_1, a_2 and a_3, are now simply functions of θ and the nodal coordinates of element Σ_P.

It is desirable to recast the governing integral equation, equation (4.11), in terms of the new coordinate system. First examine the effect on the hypersingular kernel, given by

$$J_P^2 \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}} = \frac{J_P^2}{4\pi}\left(\frac{\mathbf{N}_k}{r^3} - 3\frac{(\mp\epsilon)\left(R_k^{\pm}\right)}{r^5}\right). \tag{4.15}$$

Subtracting the external from the internal limit, equation (4.15) becomes

$$\begin{aligned} \frac{J_P^2}{4\pi}\left[\left(\frac{\mathbf{N}_k}{r^3} - 3\frac{(+\epsilon)\left(R_k^{-}\right)}{r^5}\right) - \left(\frac{\mathbf{N}_k}{r^3} - 3\frac{(-\epsilon)\left(R_k^{+}\right)}{r^5}\right)\right] &= -3\frac{J_P^2\epsilon}{4\pi r^5}\left(R_k^- + R_k^+\right) \\ &= -6\frac{J_P^2\epsilon a_k\rho}{4\pi r^5}. \end{aligned} \tag{4.16}$$

The form of the shape function $\psi_j(s^*, t^*)$ under the transformation defined by equations (4.12) and (4.13) is

$$\psi_j(s, t, \rho) = c_{j,0}(s, t) + c_{j,1}(\theta)\rho \tag{4.17}$$

where the coefficients are defined in equation (E.14).

Subsequently, the transformation of the internal most integral of equation (4.11) reveals

$$\frac{J_P^2}{4\pi}\int_{\theta_1}^{\theta_2}\int_0^{\rho_L}\left[c_{j,0}(s,t) + c_{j,1}(\theta)\rho\right]\left(-6\frac{a_k\rho\epsilon}{(a^2\rho^2+\epsilon^2)^{5/2}}\right)\rho\,d\rho\,d\theta. \tag{4.18}$$

As a result of the transformation, the integral must be divided into three separate regions. The lower subtriangle, as seen in figure 4.2, has limits given by

$$\begin{aligned} \rho_L &= -\frac{t}{\sin(\theta)}, \\ \theta_1 &= -\frac{\pi}{2} - \tan^{-1}\left(\frac{1+s}{t}\right), \\ \theta_2 &= -\frac{\pi}{2} + \tan^{-1}\left(\frac{1-s}{t}\right). \end{aligned} \tag{4.19}$$

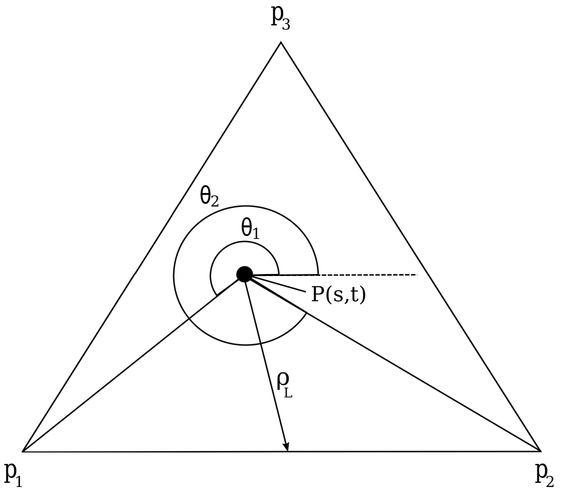


Figure 4.2: Polar coordinate transformation centred at (s,t). Adapted from Gray *et al.* (2004a), figure 1.

Eventually, all three subtriangles must be evaluated, nonetheless, only the lower subtriangle need be evaluated as the other two can use the same analysis by simply rotating the element. By inspecting equation (4.18), it can be seen that there are two integrals which must be calculated,

$$\frac{J_P^2 c_{j,0}}{4\pi} \int_{\theta_1}^{\theta_2} \int_0^{\rho_L} \left(-6 \frac{a_k \rho^2 \epsilon}{(a^2\rho^2 + \epsilon^2)^{5/2}} \right) d\rho \, d\theta \tag{4.20}$$

and

$$\frac{J_P^2}{4\pi} \int_{\theta_1}^{\theta_2} c_{j,1}(\theta) \int_0^{\rho_L} \left(-6 \frac{a_k \rho^3 \epsilon}{(a^2\rho^2 + \epsilon^2)^{5/2}} \right) d\rho \, d\theta. \tag{4.21}$$

The final goal is to remove the singularities from the above integrals and to reduce the number of integrations required to as few as possible, including the s and t integrals which encompass the above expressions. Equation (4.21) shall be evaluated first, noting that this integral was omitted from the derivation in Gray *et al.* (2004b). However, the process of evaluating this integral is relatively simple, utilising the techniques presented in Gray *et al.* (2004a,b).

To begin, the innermost integral of equation (4.21) is evaluated first:

$$\begin{aligned}\frac{J_P^2}{4\pi}\int_{\theta_1}^{\theta_2} c_{j,1}(\theta)\int_0^{\rho_L}\left(-6\frac{\rho^3\epsilon}{(a^2\rho^2+\epsilon^2)^{5/2}}\right)d\rho\, d\theta& \\ &= \frac{J_P^2}{4\pi}\int_{\theta_1}^{\theta_2} c_{j,1}(\theta)a_k(\theta)\left[\frac{2\epsilon\left(2\epsilon^2+3a^2\rho^2\right)}{a^4\left(a^2\rho^2+\epsilon^2\right)^{3/2}}\right]_0^{\rho_L} d\theta \\ &= \frac{J_P^2}{4\pi}\int_{\theta_1}^{\theta_2} c_{j,1}(\theta)a_k(\theta)\left(\frac{2\epsilon\left(2\epsilon^2+3a^2\rho_L^2\right)}{a^4\left(a^2\rho_L^2+\epsilon^2\right)^{3/2}}-\frac{4}{a^4}\right)d\theta.\end{aligned} \tag{4.22}$$

It is now possible to safely set ϵ to zero in equation (4.22). What remains is the θ integral, given by

$$\frac{J_P^2}{4\pi}\int_{\theta_1}^{\theta_2} -4\frac{c_{j,1}a_k}{a^4}\, d\theta. \tag{4.23}$$

At this stage, the divergent quantity in this integral has been removed and equation (4.23) may be integrated directly to reduce the integral to two dimensions. There are potential problems at $s = \pm 1$, but as these point lie at the extremes of the domain they are of no concern. Of greater concern is the unwieldy nature of the expression which results from directly evaluating equation (4.23), and, therefore, to make the result more manageable, the substitution and second polar transformation demonstrated in Gray *et al.* (2004a,b) is applied. The substitution is given by

$$\theta = -\frac{\pi}{2} + \tan^{-1}\left(\frac{\tau - s}{t}\right), \tag{4.24}$$

where $-1 \leq \tau \leq 1$, and the second polar transformation is defined by

$$\begin{aligned}\tau &= \Lambda\cos(\Psi) + s,\\ t &= \Lambda\sin(\Psi).\end{aligned} \tag{4.25}$$

The domain of integration for the second polar coordinate transformation is illustrated in figure 4.3. The integral must be, once again, split into three separate domains. The impact of these transformations on the variables, shape functions and the limits of integration is detailed in section E.2. The form of the ψ_l shape functions after the transformations are

$$\psi_l(s,\Lambda) = k_{l,0} + k_{l,1}\Lambda \tag{4.26}$$

where the coefficients are given in equation (E.19). Section E.2 also shows that

$$dt\, d\theta = \sin(\Psi)\; d\Psi\, d\Lambda. \tag{4.27}$$

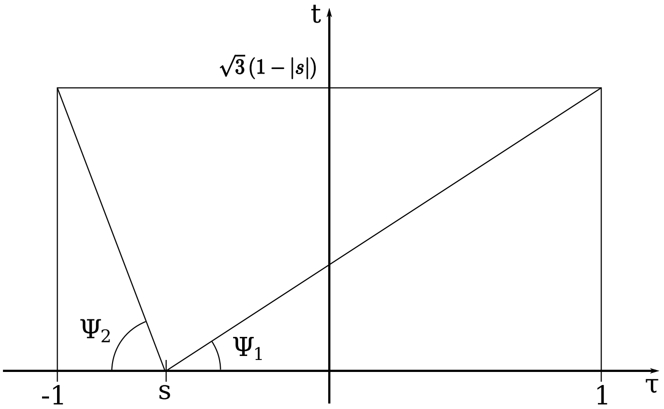


Figure 4.3: Polar coordinate transformation centred at $\tau = s, t = 0$. Adapted from Gray *et al.* (2004a), figure 2.

Transforming equation (4.23) and re-introducing the outer variables of integration produces an integral of the form

$$-\frac{J_P^2}{\pi}\int_{-1}^{1} ds \int^{\Psi} \frac{c_{j,1}a_k}{a^4}\sin(\Psi)\, d\Psi \int_0^{\Lambda_L} k_{l,0} + k_{l,1}\Lambda\, d\Lambda. \tag{4.28}$$

Observing figure 4.3, the limits of integration for Ψ over the three domains are defined by $0 \le \Psi \le \Psi_1(s)$, $\Psi_1(s) \le \Psi \le \pi - \Psi_2(s)$ and $\pi - \Psi_2(s) \le \Psi \le \pi$ where

$$\begin{aligned}\Psi_1(s) &= \tan^{-1}\left(\frac{\sqrt{3}\,(1-|s|)}{1-s}\right),\\ \Psi_2(s) &= \tan^{-1}\left(\frac{\sqrt{3}\,(1-|s|)}{1+s}\right).\end{aligned} \tag{4.29}$$

The corresponding limits for Λ, given by $\Lambda_L(\Psi)$, for the three triangles are

$$\begin{aligned}\Lambda_1(\Psi) &= \frac{1-s}{\cos\Psi},\\ \Lambda_2(\Psi) &= \frac{\sqrt{3}\,(1-|s|)}{\sin\Psi},\\ \Lambda_3(\Psi) &= -\frac{1+s}{\cos\Psi}.\end{aligned} \tag{4.30}$$

Hence, the inner integral of equation (4.28) is evaluated revealing

$$\begin{aligned}\int_0^{\Lambda_L} k_{l,0} + k_{l,1}\Lambda \, d\Lambda. &= \left[k_{l,0}\Lambda + \frac{k_{l,1}\Lambda^2}{2}\right]_0^{\Lambda_L} \\ &= \Lambda_L\left(k_{l,0} + \frac{k_{l,1}\Lambda_L}{2}\right).\end{aligned} \tag{4.31}$$

The remaining integral to be evaluated is

$$-\frac{J_P^2}{\pi}\int_{-1}^{1} ds \int^{\Psi} \frac{c_{j,1}a_k}{a^4}\sin(\Psi)\Lambda_L\left(k_{l,0} + \frac{k_{l,1}\Lambda_L}{2}\right) d\Psi. \tag{4.32}$$

The integrand of the Ψ integral with all of the coefficients replaced is examined with $j = 1$ and $l = 1$ which results in

$$\frac{\left(-\frac{1}{2}\cos(\Psi) + \frac{1}{6}\sqrt{3}\sin(\Psi)\right)\left(A_1\cos(\Psi) - B_1\sin(\Psi)\right)}{(\chi)^2} \times \\ \frac{1}{2}\tan(\Psi)\left(1-s\right)^2\left(1 - \frac{1}{6}\sqrt{3}\tan(\Psi)\right) \tag{4.33}$$

where

$$\chi = \left(A_1\cos(\Psi) - B_1\sin(\Psi)\right)^2 + \left(A_2\cos(\Psi) - B_2\sin(\Psi)\right)^2 + \\ \left(A_3\cos(\Psi) - B_3\sin(\Psi)\right)^2. \tag{4.34}$$

Clearly, equation (4.33) is too complicated to evaluate analytically. However, the integral in equation (4.32) can still be reduced to a single numerical integration by utilising the method presented in Gray *et al.* (2006). In Gray *et al.* (2006) the method was used to reduce the coincident integrals produced in Gray *et al.* (2004a) to one dimension. To the best of the author's knowledge, this is the first time this technique has been used for the coincident integrals of the gradient calculation. The method works by exchanging the order of integration of the s and Ψ integrals, allowing the simpler s integrand to be evaluated first. This gives the coincident integral the form

$$-\frac{J_P^2}{\pi}\int \frac{c_{j,1}a_k}{a^4}\sin(\Psi)\,d\Psi \int \Lambda_L\left(k_{l,0} + \frac{k_{l,1}\Lambda_L}{2}\right) ds. \tag{4.35}$$

Although there are no obvious singularities for the range of s in expression (4.35) it is still convenient to deal with this third analytical integration in two separate regions.

These regions are $-1 < s < 0$ and $0 < s < 1$.

The various domains of the integrals can be seen in figure 4.4. Each triangle in figure 4.3 is marked by a different pattern and the domains for $s < 0$ and $s > 0$ are illustrated in a) and b) respectively. The function $\Psi(s)$, which is transformed to $\alpha(\Psi)$, is marked in both the a) and b) diagrams. This function is a legacy from splitting the domain into positive and negative s and this causes the second triangle to be split into two separate integrals in both the case of $s < 0$ and $s > 0$. The function α is derived by rearranging Ψ for s. To demonstrate, when $s < 0$,

$$\begin{aligned}
& & \Psi &= \tan^{-1}\left(\frac{\sqrt{3}(1+s)}{1-s}\right) \\
&\Rightarrow & \frac{-1-s}{-1+s} &= \frac{1}{3}\sqrt{3}\tan(\Psi) \\
&\Rightarrow & s &= \frac{-3+\sqrt{3}\tan(\Psi)}{3+\sqrt{3}\tan(\Psi)} \\
&\therefore & \alpha &= \frac{\beta-1}{\beta+1},
\end{aligned} \tag{4.36}$$

where $\beta = \frac{1}{\sqrt{3}}\tan(\Psi)$. When $s > 0$, α must be redefined as

$$\alpha = -\left(\frac{\beta+1}{\beta-1}\right). \tag{4.37}$$

Note that, in an effort to avoid redefining β, a slightly different α is used for $s > 0$ than is seen in Gray *et al.* (2006).

The resulting one-dimensional integrals, derived from equation (4.21), can be found in section F.1. Along with deriving the integrals, the integral kernels of the resulting numerical integration are checked to ensure they are non-singular over the domain of integration.

The second integral to be evaluated, equation (4.20), requires additional steps to successfully reduce it to a 1D numerical integration. This is the subject of Gray *et al.* (2004b, section 3) and is reproduced here for completeness, before taking the extra step of reducing the integral to one dimension. The process is begun by evaluated the inner

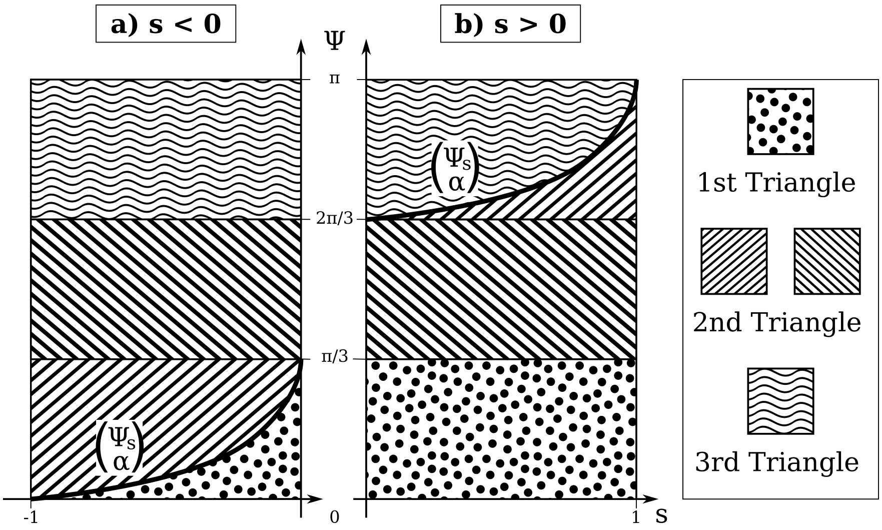


Figure 4.4: Integration domain for second polar transformation split between $s < 0$ and $s > 0$ and over the three triangles. Adapted from Gray *et al.* (2006).

integral equation,

$$\begin{aligned}\frac{J_P^2 c_{j,0}}{4\pi}\int_{\theta_1}^{\theta_2}\int_0^{\rho_L}\left(-6\frac{a_k\rho^2\epsilon}{(a^2\rho^2+\epsilon^2)^{5/2}}\right)d\rho\, d\theta.&\\ &= \frac{J_P^2 c_{j,0}}{2\pi}\int_{\theta_1}^{\theta_2}\left[-\frac{1}{\epsilon}\frac{a_k(\theta)\rho^3}{(\epsilon^2+a^2\rho^2)^{3/2}}\right]_0^{\rho_L} d\theta\\ &= \frac{J_P^2 c_{j,0}}{2\pi}\int_{\theta_1}^{\theta_2} -\frac{1}{\epsilon}\frac{a_k(\theta){\rho_L}^3}{(\epsilon^2+a^2{\rho_L}^2)^{3/2}}\, d\theta.\end{aligned} \tag{4.38}$$

Unlike equation (4.22) it is not possible to set ϵ to zero in equation (4.38) due to the presence of the ϵ^{-1} term. Gray *et al.* deals with this problem by considering the limiting form of equation (4.22), obtained by replacing $\left(\epsilon^2+a^2{\rho_L}^2\right)^{-3/2}$ with $a^{-3}\rho_L^{-3}$. In this situation the integral reduces to

$$\frac{J_P^2 c_{j,0}}{2\pi}\int_0^{2\pi} -\frac{1}{\epsilon}\frac{a_k(\theta)}{a^3}\, d\theta = -\frac{J_P^2 c_{j,0}}{2\pi\epsilon}\int_0^{2\pi}\frac{a_k(\theta)}{a^3}\, d\theta = 0. \tag{4.39}$$

The result of equation (4.39) relies upon being able to integrate over the range $0 \leq \theta \leq 2\pi$. There are, however, difficulties achieving this if the point $P(s,t)$ should be at a corner or on an edge. However, a well know fact dictates that the edges and corners have zero volume and thus do not contribute to the integral. This, combined with a successful outcome of the analytic integration scheme, indicate that these special cases are not of significance to this integral.

With the applicability of equation (4.39) assured, the integrand of the limiting case can be subtracted from the integrand of equation (4.22) for all the subtriangles. This results in the new kernel,

$$-\frac{1}{\epsilon}\frac{a_k {\rho_L}^3}{(\epsilon^2 + a^2 {\rho_L}^2)^{3/2}} - \left(-\frac{1}{\epsilon}\frac{a_k}{a^3}\right) = -\frac{1}{\epsilon}\frac{a_k\left(a^3\rho_L^3 - \left(\epsilon^2 + a^2\rho_L^2\right)^{3/2}\right)}{a^3\left(\epsilon^2 + a^2\rho_L^2\right)^{3/2}}. \tag{4.40}$$

The form of equation (4.40) still prevents the safe setting of ϵ to zero for the reasons detailed in Gray *et al.* (2004b, pg. 300). In order to finally deal with this singular term the transformations described in equations (4.24) and (4.25) must be applied to the integral.

It is useful at this point to restate the entire integral to be evaluated. The remaining integral is

$$\frac{J_P^2}{2\pi}\sum_{j=1}^{3}\phi(Q_j)\int_{-1}^{1} ds \int_{0}^{\sqrt{3}(1-|s|)} \psi_l(s,t) c_{j,0}(s,t)\, dt \times \int_{\theta_1}^{\theta_2} -\frac{1}{\epsilon}\frac{a_k\left(a^3\rho_L^3 - \left(\epsilon^2 + a^2\rho_L^2\right)^{3/2}\right)}{a^3\left(\epsilon^2 + a^2\rho_L^2\right)^{3/2}}\, d\theta. \tag{4.41}$$

The effect of transformations (4.24) and (4.25) on ψ_l has already been derived. The impact on $c_{j,0}$ is given by equation (E.23) and can be seen to have identical form to that of ψ_l (see equation (4.26)). The inner-most integrand of equation (4.41) must also be transformed resulting in

$$\frac{1}{\epsilon}\left(\frac{a_k}{a^3} - \frac{a_k\Lambda^3}{(\epsilon^2 + a^2\Lambda^2)^{3/2}}\right). \tag{4.42}$$

Thus, the full integral in (s, Ψ, Λ) space is

$$\frac{J_P^2}{2\pi}\int_{-1}^{1} ds \int^{\Psi} \sin(\Psi)\, d\Psi \times \int_{0}^{\Lambda_L} \frac{1}{\epsilon}\left(k_{l,0} + k_{l,1}\Lambda\right)^2 \left(\frac{a_k}{a^3} - \frac{a_k\Lambda^3}{(\epsilon^2 + a^2\Lambda^2)^{3/2}}\right) d\Lambda. \tag{4.43}$$

Now consider the inner-most integral in equation (4.43) and the contribution to it from the lowest order term in Λ in $(k_{l,0}+k_{l,1}\Lambda)^2$, i.e. the $k_{l,0}^2$ term. That is, consider

$$\begin{aligned}
\mathrm{I}_{\Lambda^0} &= \int_0^{\Lambda_L} \frac{1}{\epsilon} k_{l,0}^2 \left(\frac{a_k}{a^3} - \frac{a_k \Lambda^3}{(\epsilon^2 + a^2\Lambda^2)^{3/2}} \right) d\Lambda \\
&= \left[-\frac{a_k k_{l,0}^2 \left(a^2\Lambda^2 + 2\,\epsilon^2 - \Lambda\, a\sqrt{\epsilon^2 + a^2\Lambda^2} \right)}{a^4 \epsilon\, \sqrt{\epsilon^2 + a^2\Lambda^2}} \right]_0^{\Lambda_L} \\
&= -\frac{a_k k_{l,0}^2 \left(a^2{\Lambda_L}^2 + 2\,\epsilon^2 - \Lambda_L a\sqrt{\epsilon^2 + a^2{\Lambda_L}^2} - 2\,\epsilon\, \sqrt{\epsilon^2 + a^2{\Lambda_L}^2} \right)}{a^4 \epsilon\, \sqrt{\epsilon^2 + a^2{\Lambda_L}^2}}.
\end{aligned} \tag{4.44}$$

Unfortunately, the result of equation (4.44) will still not permit the setting of ϵ to zero. However an application of the binomial theorem to the $\Lambda_L a\sqrt{\epsilon^2 + a^2{\Lambda_L}^2}$ term reveals that

$$\begin{aligned}
\Lambda_L a\sqrt{\epsilon^2 + a^2{\Lambda_L}^2} &= \Lambda_L^2 a^2 \left(1 + \frac{\epsilon^2}{\Lambda_L^2 a^2} \right)^{1/2} \\
&= \Lambda_L^2 a^2 \left(1 + \frac{1}{2}\frac{\epsilon^2}{\Lambda_L^2 a^2} + \mathbf{O}(\epsilon^4) \right).
\end{aligned} \tag{4.45}$$

Given that the $\mathbf{O}(\epsilon^4)$ terms and above in equation (4.44) are sufficiently small, it is reasonable to ignore them and thus

$$\Lambda_L a\sqrt{\epsilon^2 + a^2{\Lambda_L}^2} \approx \Lambda_L^2 a^2 + \frac{1}{2}\epsilon^2. \tag{4.46}$$

With the substitution of equation (4.46) into the result of equation (4.44) the limit may be taken, revealing

$$\lim_{\epsilon \to 0} \left[-\frac{1}{2} \frac{a_k k_{l,0}^2 \left(3\,\epsilon - 4\,\sqrt{\epsilon^2 + a^2{\Lambda_L}^2} \right)}{a^4 \sqrt{\epsilon^2 + a^2{\Lambda_L}^2}} \right] = \frac{2 a_k k_{l,0}^2}{a^4}\,. \tag{4.47}$$

Through similar processes, the Λ and Λ^2 integrals (from the $2k_{l,0}k_{l,1}\Lambda$ and $k_{l,1}^2\Lambda^2$ terms in $(k_{l,0}+k_{l,1}\Lambda)^2$) are found to be zero in the limit as $\epsilon \to 0$. Thus, the remaining integral to be evaluated is

$$\frac{J_P^2}{\pi} \int_{-1}^{1} k_{l,0}^2\, ds \int^{\Psi} \frac{a_k \sin(\Psi)}{a^4}\, d\Psi. \tag{4.48}$$

This integral is now reduced to one numerical integration by exchanging the order of

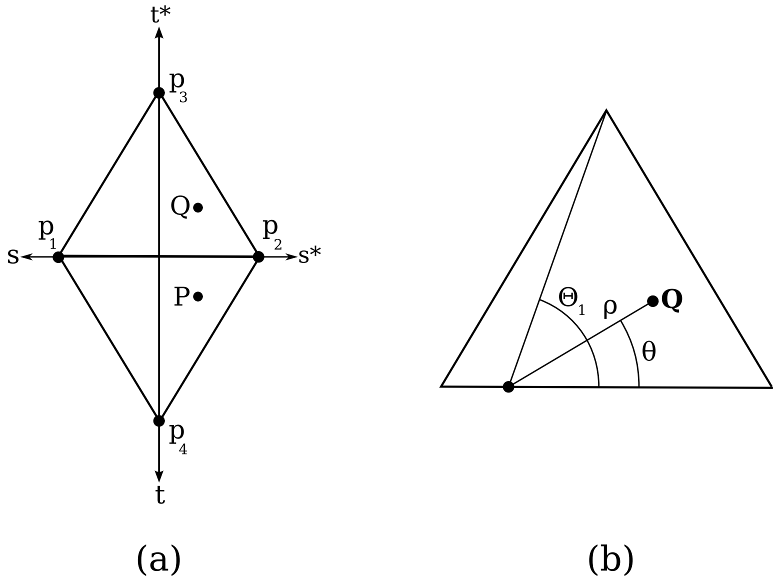


Figure 4.5: Edge-adjacent integral and polar transformation. Adapted from Sutradhar *et al.* (2005).

integration in a similar manner to the method used to reduce equation (4.32). The resulting integrals are listed in section F.2.

4.3.2 Edge Adjacent Integration

The next stage in the analysis is the evaluation of the edge adjacent integral. This type of integral was not explicitly derived for the gradient method in Gray *et al.* (2004b). The method used for the edge adjacent integrals in the accompanying Gray *et al.* (2004a) paper can be applied to the gradient, and this is demonstrated herein. The orientation of elements which require such an integration is illustrated in figure 4.5(a) with suitable choice of s, t, s^* and t^* axes. The elements containing the points P and Q share the vertices p_1 and p_2 and the edge at $t = t^* = 0$. Also note that along the shared edge $s = -s^*$.

The distance vector, $\mathbf{R}$, under these conditions is derived in section G.1. In addition, the Q vector must be transformed into the (s, t) parameter space. To achieve this Gray *et al.* (2004a) employed a polar transformation centred about $s^* = -s$ and $t^* = 0$ which is described by

$$\begin{aligned} s^* &= \rho\cos(\theta) - s, \\ t^* &= \rho\sin(\theta). \end{aligned} \tag{4.49}$$

The transformation splits the integration into two domains ($0 < \theta < \Theta_1$, $\Theta_1 < \theta < \pi$), as can be seen in figure 4.5(b). Consequently, $\mathbf{R}$ is now a function of (ρ, θ, t).

As is seen in appendix G, the form of r^2 is different dependent on whether the internal or external limit is taken. For the $\mathbf{R}^-$ vector (see equation (4.8)) the result is

$$r_-^2 = \epsilon^2 + 2\left(\mathbf{N}\cdot\mathbf{R}^0\right)\epsilon + b_{01}(\theta)\rho^2 + b_{11}(\theta)\rho t + b_{10}t^2 \tag{4.50}$$

where

$$\mathbf{N}\cdot\mathbf{R}^0 = b_{20}(\theta)\rho + b_{02}t \tag{4.51}$$

and $\mathbf{R}^0 = Q - P$ or, otherwise, $\mathbf{R}^\pm$ with ϵ set to zero. The coefficients are defined within equation (G.7). The result for the $\mathbf{R}^+$ vector is

$$r_+^2 = \epsilon^2 - 2\left(\mathbf{N}\cdot\mathbf{R}^0\right)\epsilon + b_{01}(\theta)\rho^2 + b_{11}(\theta)\rho t + b_{10}t^2. \tag{4.52}$$

The difficulty that arises from having two different r^2 terms is that the denominators of the internal and external integrands will not be equal. Hence, cancellation of singular quantities at an early stage is not forthcoming. It is chosen, instead, to advance both the internal and external limits to the point of the first analytical integration. The difference is then taken and, finally, integrated.

The form of the integral to be solved is

$$\sum_{j=1}^{3}\phi(Q_j)\int_{-1}^{1}ds\int_{0}^{\sqrt{3}(1-|s|)}\psi_l(s,t)dt \times \\ \left[\int_0^{\Theta_1(s)}d\theta\int_0^{L_1}F\rho\,d\rho + \int_{\Theta_1(s)}^{\pi}d\theta\int_0^{L_2}F\rho\,d\rho\right] \tag{4.53}$$

where

$$F = \psi_j(s,\theta,\rho)J_P J_Q\frac{\partial^2 G}{\partial\mathbf{E}_k\partial\mathbf{n}}. \tag{4.54}$$

From figure 4.5, the limits of the inner integrals are

$$\Theta_1(s) = \frac{\pi}{2} - \tan^{-1}\left(\frac{s}{\sqrt{3}}\right) \tag{4.55}$$

and

$$L_1 = \frac{\sqrt{3}\,(1+s)}{\sin(\theta) + \sqrt{3}\cos(\theta)}, \qquad L_2 = \frac{\sqrt{3}\,(1-s)}{\sin(\theta) - \sqrt{3}\cos(\theta)}. \tag{4.56}$$

Given that Θ_1 is solely a function of s it is possible to reorder the θ and t integrals. The result is that the complete integral now has the form

$$\int_{-1}^{1} ds \int_{0}^{\Theta_1(s)} d\theta \int_{0}^{\sqrt{3}(1-|s|)} \psi_l(s,t)dt \int_{0}^{L_1} F\rho\, d\rho + \int_{-1}^{1} ds \int_{\Theta_1(s)}^{\pi} d\theta \int_{0}^{\sqrt{3}(1-|s|)} \psi_l(s,t)dt \int_{0}^{L_2} F\rho\, d\rho. \tag{4.57}$$

Due to the singularity at $t = \rho = 0$, it is prudent to make one further polar transformation centred about this point. This transformation is given by

$$\begin{aligned} \rho &= \Lambda\cos(\Psi), \\ t &= \Lambda\sin(\Psi). \end{aligned} \tag{4.58}$$

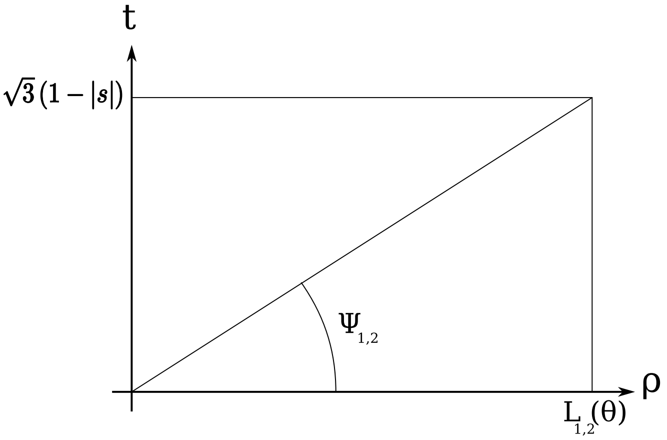


Figure 4.6: Edge-adjacent integral, second polar transformation. Adapted from Sutradhar *et al.* (2005).

As can be seen from figure 4.6, the domain of integration must, once again, be split in two. As was shown in Sutradhar *et al.* (2005), this results in the following four integrals:

$$\int_0^{\Psi_1} d\Psi \int_0^{L_{11}} F\cos(\Psi)\Lambda^2\, d\Lambda\; + \int_0^{\Psi_1} d\Psi \int_0^{L_{12}} F\cos(\Psi)\Lambda^2\, d\Lambda + \\ \int_0^{\Psi_2} d\Psi \int_0^{L_{11}} F\cos(\Psi)\Lambda^2\, d\Lambda\; + \int_0^{\Psi_2} d\Psi \int_0^{L_{12}} F\cos(\Psi)\Lambda^2\, d\Lambda\;. \tag{4.59}$$

From figure 4.6 the Λ limits are

$$L_{n1} = \frac{L_n(\theta)}{\cos(\Psi)}, \qquad\qquad L_{n2} = \frac{L_n(\theta)}{\sin(\Psi)}. \tag{4.60}$$

It is not necessary to list the values of the Ψ_1 and Ψ_2 limits, as later it will be shown that the Ψ integration domains can be unified following the Λ integration.

Subsequently, the forms of r_+^2 and r_-^2 under the transformation are:

$$\begin{aligned} r_-^2 &= b_2\Lambda^2 + b_1\epsilon\,\Lambda + \epsilon^2, \\ r_+^2 &= b_2\Lambda^2 - b_1\epsilon\,\Lambda + \epsilon^2, \end{aligned} \tag{4.61}$$

where

$$\begin{aligned} b_2 &= b_{11}\cos(\Psi)\sin(\Psi) + b_{10}\sin^2(\Psi) + b_{01}\cos^2(\Psi)\,, \\ b_1 &= b_{20}\sin(\Psi) + b_{02}\cos(\Psi)\,. \end{aligned} \tag{4.62}$$

The shape functions, ψ_j and ψ_l have the form

$$\begin{aligned} \psi_j(s,\theta,\psi,\Lambda) &= c_{j,0}(s) + c_{j,1}(\theta,\psi)\Lambda, \\ \psi_l(s,\psi,\Lambda) &= k_{l,0}(s) + k_{l,1}(\psi)\Lambda, \end{aligned}$$

where the coefficients are defined by equations (G.10) and (G.12). The distance vector, $\mathbf{R}$, under the second polar transformation is presented in section G.2 and the form of the $\mathbf{n}\cdot\mathbf{R}$ term becomes

$$J_Q\left(\mathbf{n}\cdot\mathbf{R}^{\pm}\right) = j_{1q}\Lambda \mp J_Q\left(\mathbf{n}\cdot\mathbf{N}\right)\epsilon, \tag{4.63}$$

where $j_{1q} = J_Q\left(d_1n_1 + d_2n_2 + d_3n_3\right)$ with the d_k terms defined in equation (G.13).

$\mathbf{E}_k \cdot \mathbf{R}$ remains denoted as R_k and the product with J_P has the form

$$J_P R_k^{\pm} = j_{1p}\Lambda \mp J_P \epsilon \tag{4.64}$$

where $j_{1p} = J_P d_k$.

The integral for the exterior limit, for one of the triangles, is given by

$$\int^{\Psi} \cos(\Psi)\, d\Psi \int^{L} \left(c_{j,0}(s) + c_{j,1}(\theta,\Psi)\Lambda\right)\left(k_{l,0}(s) + k_{l,1}(\Psi)\Lambda\right) \times \\ \left(\frac{J_P J_Q \mathbf{n}_k}{r_+^3} - 3\frac{\left(j_{1q}\Lambda - J_Q\left(\mathbf{n}\cdot\mathbf{N}\right)\epsilon\right)\left(j_{1p}\Lambda - J_P\epsilon\right)}{r_+^5}\right)\Lambda^2\, d\Lambda. \tag{4.65}$$

It is critical at this point to subtract the external limit from the internal limit. If they are integrated separately they will have divergent $\log(\epsilon)$ terms. Collecting together the shape functions, the powers of Λ are combined such that

$$\begin{aligned} &\left(c_{j,0}(s) + c_{j,1}(\theta,\psi)\Lambda\right)\left(k_{l,0}(s) + k_{l,1}(\psi)\Lambda\right) \\ &\qquad = c_{j,0}k_{l,0} + \left(c_{j,0}k_{l,1} + c_{j,1}k_{l,0}\right)\Lambda + c_{j,1}k_{l,1}\Lambda^2 \\ &\qquad = a_1(s) + a_2(s,\theta,\Psi)\Lambda + a_3(\Psi)\Lambda^2. \end{aligned} \tag{4.66}$$

For the hypersingular part, the result of the Λ integration, once the interior and exterior integrands have been subtracted and the limit as $\epsilon \to 0$ has been taken is $a_1 f$ where

$$f = \frac{1}{f_1}\left(f_2 \ln\left(-\frac{b_1 + 2\sqrt{b_2}}{b_1 - 2\sqrt{b_2}}\right) + f_3\right) \tag{4.67}$$

and

$$\begin{aligned} f_1 &= b_2^{5/2}\left(16b_2^2 - 8b_2b_1^2 + b_1^4\right), \\ f_2 &= 3j_pj_q\left(16b_2^2 - 8b_2b_1^2 + b_1^4\right), \\ f_3 &= 4\sqrt{b_2}\,\left[\left(-16\, j_qJ_P + 8\, J_Q(\mathbf{N}\cdot\mathbf{n})\, J_Pb_1 - 16\, j_pJ_Q(\mathbf{N}\cdot\mathbf{n})\right){b_2}^2 + \right. \\ &\qquad \left. 20\, j_qj_pb_1b_2 - 3\, j_qj_p{b_1}^3\right]. \end{aligned} \tag{4.68}$$

Likewise, the result for the strongly singular part is $a_1 g$ where

$$g = \frac{g_1}{g_2}\left(g_2 \ln\left(-\frac{b_1 + 2\sqrt{b_2}}{b_1 - 2\sqrt{b_2}}\right) + g_3\right) \tag{4.69}$$

and

$$\begin{aligned} g_1 &= \frac{J_P J_Q n_k}{b_2^{3/2}}, \\ g_2 &= \left(4b_2 - b_1^2\right), \\ g_3 &= 4\sqrt{b_2} b_1. \end{aligned} \tag{4.70}$$

Note that only the lowest power of Λ in equation (4.66) contributes to the result of the integration as the result of the higher powers is zero. Thus, the coefficient a_1 appears in the result of both the hypersingular and strongly singular parts of the integral. By observing that equations (4.67) and (4.69) are independent of L and that a_1 depends only on s, the θ and Ψ integral domains can now be re-unified, given the new ranges, $0 \leq \theta \leq \pi$ and $0 \leq \Psi \leq \pi/2$. In addition, as the s integration is independent from the others it is possible to reorder the integral such that equation (4.65) can now be written

$$\int_0^\pi d\theta \int_0^{\pi/2} \cos(\Psi)(f+g)\, d\Psi \int_{-1}^1 a_1\, ds. \tag{4.71}$$

The result of evaluating the inner most integral is

$$\int_{-1}^1 a_1 ds = \int_{-1}^1 \left(\frac{1}{2} + \frac{1}{2}s\right)^2 ds = \frac{1}{4}\left[s + s^2 + \frac{s^3}{3}\right]_{-1}^1 = \frac{2}{3}\,. \tag{4.72}$$

Hence, the remaining integral, to be solved numerically, is:

$$\frac{1}{6\pi}\int_0^\pi d\theta \int_0^{\pi/2} \cos(\Psi)(f+g)\, d\Psi. \tag{4.73}$$

The above is too complicated to be reduced any further analytically, however the numerical integration is over a simple domain, that of $0 \leq \theta \leq \pi$, $0 \leq \Psi \leq \pi/2$.

Equation (4.73) is the final expression for the edge adjacent integral and Galerkin gradient method. It can be shown that the vertex-adjacent and non-singular integrals cancel in the limiting process and thus make no contribution to the final result. Also note that there is no dependence on the edge-adjacent integral to 'cancel' any non-bounded expressions resulting from the coincident integral, as is the case in Gray *et al.* (2004a). The independence of the coincident and edge-adjacent integrals from each other is important in applying the Galerkin gradient method over a non-closed surface.

4.3.3 Application to a Vortex Sheet

It is proposed that the analysis in the last two sections can be applied to a vortex sheet (see Gray *et al.* (2004b, sec. 6) for the equivalent crack problem in fracture mechanics). To demonstrate this, an alternative limiting scheme is considered. Observing the right hand part of figure 2.5, the Galerkin derivative scheme can be applied to the wake top and bottom individually and then summed (see Gray *et al.*, 2004b, sec. 6). Although that analysis is plausible, it is perhaps not robust enough to be unequivocal.

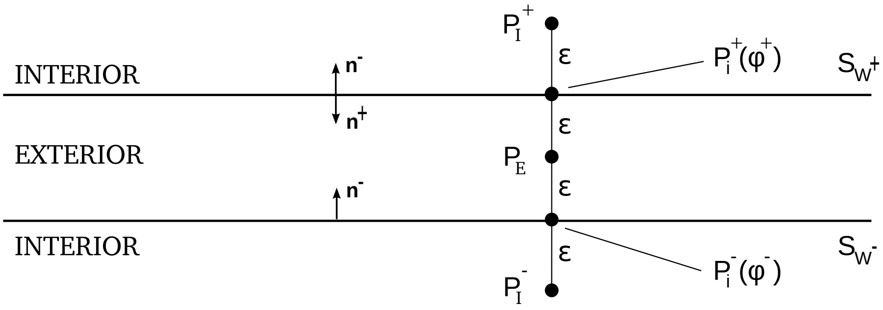


Figure 4.7: Wake limiting scheme with shared external point.

Figure 4.7 illustrates another possible scheme which is described herein. This scheme aims to unify both the Galerkin limit to the top and bottom surfaces and the limit closing the two surfaces. This is achieved through a shared exterior point, P_E. By writing out the equations for each side in full (with a unified normal), when the limit $\epsilon \to 0$ is taken, the full expression for the velocity on the wake should remain. The desired outcome is that the expressions will be functions of $[\phi] = \phi^+ - \phi^-$. The process is begun by restating the equation for the velocity of the wake, i.e.

$$\mathbf{v} = \frac{1}{2}\left(\frac{\partial \phi^+}{\partial \mathbf{E}_k} + \frac{\partial \phi^-}{\partial \mathbf{E}_k}\right). \tag{4.74}$$

The result of writing out the Galerkin form of the right hand side of equation (4.74) in full is

$$\frac{1}{2}\Bigg[\left\{\lim_{P_I^+ \to P^+} - \lim_{P_E \to P^+}\right\} \sum_{j=1}^{3} \phi^+(Q_j) \int_{\Sigma_P} \psi_l(P^+) \int_{\Sigma_Q} \psi_j \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^+}\, dQ\, dP\, + \\ \left\{\lim_{P_I^- \to P^-} - \lim_{P_E \to P^-}\right\} \sum_{j=1}^{3} \phi^-(Q_j) \int_{\Sigma_P} \psi_l(P^-) \int_{\Sigma_Q} \psi_j \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^-}\, dQ\, dP\Bigg]. \tag{4.75}$$

A shared normal is required and if $\mathbf{n}^-$ is chosen then the ϕ^- limit is identical to the analysis already carried out. The effect of exchanging the $\mathbf{n}^+$ normal for the ϕ^+ limit is now investigated. Most notably, the internal and external limits are reversed. The external limit is now

$$\mathbf{R}^+ = Q - (P - \epsilon\mathbf{N}), \tag{4.76}$$

while the internal limit is

$$\mathbf{R}^- = Q - (P + \epsilon\mathbf{N}). \tag{4.77}$$

Note, for the coincident integral, that $\mathbf{N} = \mathbf{n}^-$. Also note that equation (4.10) still holds, however, equation (4.9) has been altered to

$$\mathbf{N} \cdot \mathbf{R}^\pm = \pm\epsilon \; . \tag{4.78}$$

Consequently, comparing the result for the ϕ^- limit of the coincident integral (given by equation (4.16)) to the difference in limits for ϕ^+ it can be seen that

$$\begin{aligned}\frac{J_P^2}{4\pi}\left[\left(\frac{\mathbf{N}_k}{r^3} - 3\frac{(-\epsilon)\left(R_k^-\right)}{r^5}\right) - \left(\frac{\mathbf{N}_k}{r^3} - 3\frac{(+\epsilon)\left(R_k^+\right)}{r^5}\right)\right] &= 3\frac{J_P^2\epsilon}{4\pi r^5}\left(R_k^- + R_k^+\right) \\ &= 6\frac{J_P^2\epsilon a_k\rho}{4\pi r^5} \; .\end{aligned} \tag{4.79}$$

The coincident integral for the velocity equation becomes

$$\begin{aligned}\frac{1}{2}\Bigg[&\sum_{j=1}^{3}\phi^+(Q_j)\int_{\Sigma_P}\psi_l(P^+)\int_{\Sigma_Q}\psi_j(Q)\frac{6J_P^2\epsilon a_k\rho}{4\pi r^5}\,dQ\,dP \\ &- \left(\sum_{j=1}^{3}\phi^-(Q_j)\int_{\Sigma_P}\psi_l(P^-)\int_{\Sigma_Q}\psi_j(Q)\frac{6J_P^2\epsilon a_k\rho}{4\pi r^5}\,dQ\,dP\right)\Bigg] \; .\end{aligned} \tag{4.80}$$

Fortuitously, now that the limit as $\epsilon \to 0$ has been taken, P^+ and P^- are coincident. Hence, choosing $\phi^+ - \phi^- = [\phi]$, expression (4.80) becomes

$$\frac{1}{2}\left[\sum_{j=1}^{3}[\phi]\,(Q_j)\int_{\Sigma_P}\psi_l(P)\int_{\Sigma_Q}\psi_j(Q)\frac{6J_P^2\epsilon a_k\rho}{4\pi r^5}\,dQ\,dP\right] \; . \tag{4.81}$$

Thus, the results of the coincident integral for ϕ^- also hold for $[\phi]$, noting that a factor of '$-1/2$' must be applied.

Accordingly, the edge adjacent integral for the ϕ^+ limit must now be evaluated. Again, a unified normal, $\mathbf{n}^-$, is used in the analysis. Most of the initial analysis for the edge adjacent integral is unaltered. The main impact is on $\mathbf{R}$, being defined by equations (4.76) and (4.77). Therefore the form of r_+^2 and r_-^2, once the two polar transformations described in section 4.3.2 have been carried out, is altered from equation (4.61) to

$$\begin{aligned} r_-^2 &= b_2\Lambda^2 - b_1\epsilon\,\Lambda + \epsilon^2, \\ r_+^2 &= b_2\Lambda^2 + b_1\epsilon\,\Lambda + \epsilon^2, \end{aligned} \tag{4.82}$$

with coefficients defined by equation (4.62). Equation (4.63) is also altered to

$$J_Q\left(\mathbf{n}\cdot\mathbf{R}^\pm\right) = j_{1q}\Lambda \pm J_Q\left(\mathbf{n}\cdot\mathbf{N}\right)\epsilon \tag{4.83}$$

and equation (4.64) becomes

$$J_P R_k^\pm = j_{1p}\Lambda \pm J_P\epsilon. \tag{4.84}$$

Subsequently, the innermost integral, applying the difference between the internal and external limits, is

$$\begin{aligned} \int^L & \left(c_{j,0}(s) + c_{j,1}(\theta,\Psi)\Lambda\right)\left(k_{l,0}(s) + k_{l,1}(\Psi)\Lambda\right) \times \\ & \left[\left(\frac{J_P J_Q \mathbf{n}_k}{r_-^3} - 3\frac{\left(j_{1q}\Lambda - J_Q\left(\mathbf{n}\cdot\mathbf{N}\right)\epsilon\right)\left(j_{1p}\Lambda - J_P\epsilon\right)}{r_-^5}\right) - \right. \\ & \left.\left(\frac{J_P J_Q \mathbf{n}_k}{r_+^3} - 3\frac{\left(j_{1q}\Lambda + J_Q\left(\mathbf{n}\cdot\mathbf{N}\right)\epsilon\right)\left(j_{1p}\Lambda + J_P\epsilon\right)}{r_+^5}\right)\right]\Lambda^2\, d\Lambda. \end{aligned} \tag{4.85}$$

Thus, the result of the hypersingular, Λ integral is

$$a_1 f = \frac{a_1}{f_1}\left(f_2 \ln\left(-\frac{b_1 + 2\sqrt{b_2}}{b_1 - 2\sqrt{b_2}}\right) + f_3\right), \tag{4.86}$$

and the strongly singular part is

$$a_1 g = \frac{a_1 g_1}{g_2}\left(g_2 \ln\left(-\frac{b_1 + 2\sqrt{b_2}}{b_1 - 2\sqrt{b_2}}\right) + g_3\right). \tag{4.87}$$

The f_i and g_i coefficients are given in equations (4.68) and (4.70).

The limits of integration are now unified and the s-integral evaluated in the manner of equation (4.71). Finally, combining the results for the ϕ^+ and ϕ^- limits produces

$$\frac{1}{2}\left[\frac{1}{6\pi}\sum_{j=1}^{3}\phi^+\int_0^{\pi}d\theta\int_0^{\pi/2}\cos(\Psi)(f+g)\,d\Psi + \left(\frac{-1}{6\pi}\sum_{j=1}^{3}\phi^-\int_0^{\pi}d\theta\int_0^{\pi/2}\cos(\Psi)(f+g)\,d\Psi\right)\right] \tag{4.88}$$

and once again noting that in the limit as $\epsilon \to 0$, P^+ and P^- are unified and $[\phi] = \phi^+ - \phi^-$. Therefore, equation (4.88) becomes

$$\frac{1}{2}\left[\frac{1}{6\pi}\sum_{j=1}^{3}[\phi]\int_0^{\pi}d\theta\int_0^{\pi/2}\cos(\Psi)(f+g)\,d\Psi\right]. \tag{4.89}$$

Equations (4.81) and (4.89) demonstrate that the analysis for the ϕ^- side of the wake is identical to that for the collapsed wake sheet with ϕ^- replaced by $[\phi]$ and a factor of $-1/2$ applied.

4.3.4 Validity at Sheet Edges

Although the result of the previous section is highly convenient, the validity of the closing of the wake sheet is still yet to be determined. The above analysis requires two distinct sides, where equations for ϕ^+ and ϕ^- are defined. Crucially, when closing the two sides it is vital that the equations do not require information from the opposite side or it will not be possible to use the unified field value $[\phi]$.

To visualise the problem, consider the wake mapped to the surface of the sphere. The equator of the sphere represents the wake edge, formed once the wake sides are brought together. This geometry, showing two opposite points which will be unified, can be seen in figure 4.8.

Such a geometry is required to facilitate cancellation between the internal and external domain. The challenge, when bringing the two surfaces together, is to separate the two sides, S^+ and S^-, so that equations of either side, at a point P^+ or P^-, have no dependency on each other. Figure 4.9 illustrates the panels supporting the point of interest (i.e. those panels with non-zero shape functions at P_i) and the edge adjacent panels that must be included in the right hand side of the Galerkin integration.

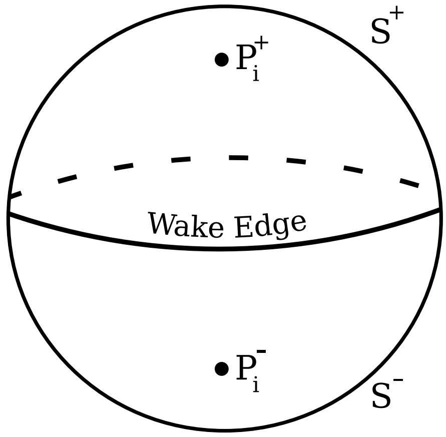


Figure 4.8: Opposite wake surfaces and points mapped onto a sphere.

The discretisation of the left and right hand side of the governing equations offers insight into the execution of the Galerkin method. The governing integral equation for the wake velocity is written

$$\int_{\Sigma_i} \hat{\psi}_i(P) \left(\frac{\partial \phi^+(P)}{\partial \mathbf{E}_k} + \frac{\partial \phi^-(P)}{\partial \mathbf{E}_k} \right) dP = \left\{ \lim_{P_I \to P} - \lim_{P_E \to P} \right\} \int_{\Sigma_i} \hat{\psi}_i(P) \int_{\Sigma_{\mathrm{NZ}}} [\phi(Q)] \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}} \, dQ \, dP. \quad (4.90)$$

Here, Σ_i represents the set of panels which have non-zero shape functions, $\hat{\psi}_i$, at point P_i on Σ_P. Σ_{NZ} are the set of panels containing Q which are coincident or edge adjacent to the panels contained in Σ_i.

When considering the role of the left hand side of equation (4.90), it is useful first to consider a single panel from the set Σ_i. This panel is referred to as $\Sigma_{i,m}$. Within $\Sigma_{i,m}$, the shape function at P_i, $\hat{\psi}_i$, is, once again, redefined as ψ_l for some $l \in [1, 2, 3]$. The sum of the gradients can then be written in terms of the nodal values of $\Sigma_{i,m}$ such that

$$\left(\frac{\partial \phi^+(P)}{\partial \mathbf{E}_k} + \frac{\partial \phi^-(P)}{\partial \mathbf{E}_k} \right)(s,t) = \sum_{j=1}^{3} 2 v_{\mathrm{av}}(p_j) \psi_j(s,t), \quad (4.91)$$

where $j = 1, 2, 3$ and v_{av} is the average velocity. Thus the left hand side of equation (4.90) becomes

$$2 \sum_{j=1}^{3} v_{\mathrm{av}}(p_j) \int_{-1}^{1} \int_{0}^{\sqrt{3}(1-|s|)} \psi_l(s,t) \psi_j(s,t) J_P \, dt \, ds. \quad (4.92)$$

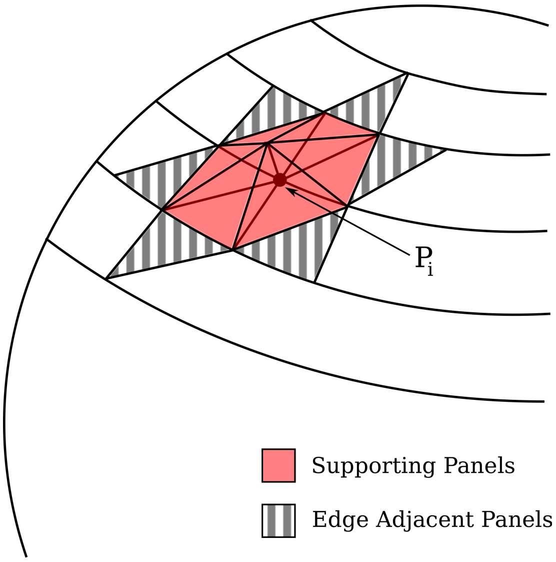


Figure 4.9: Contributing panels for a point on one side of the wake. Σ_i are the red panels and Σ_{NZ} are both the red and striped panels.

Equation (4.92) illustrates that the Galerkin method requires the average velocity at the six nodes surrounding P_i (for the geometry depicted in figure 4.9).

Now considering the right hand side of equation (4.90), $[\phi]$ is estimated for a panel Σ_Q by the values at the panel's nodes. Thus, for a particular panel, Σ_n that belongs to the set Σ_{NZ},

$$\int_{\Sigma_n} [\phi(Q)] \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}} \, dQ = \sum_{j=1}^{3} [\phi(q_j)] \int_{-1}^{1} \int_{0}^{\sqrt{3}(1-|s|)} \psi_j(s,t) J_Q \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}} \, dt \, ds. \quad (4.93)$$

Note that the domain of Σ_{NZ} will extend beyond the set of panels contained in Σ_i. To stop the left and right hand side of equation (4.90) from depending on points on the opposite side of the wake from P_i, the ability to truncate the equations at the edges is critical.

Figure 4.10 illustrates how the support required for the left hand side of equation (4.90) could extend beyond the wake edges. Starting at (i), the velocity at the circular point is desired. The left hand side of equation (4.90) requires the average velocities at the six points surrounding the circle, one of which is the square point. Moving onto (ii), the support for the square point is shown. Again the left hand side

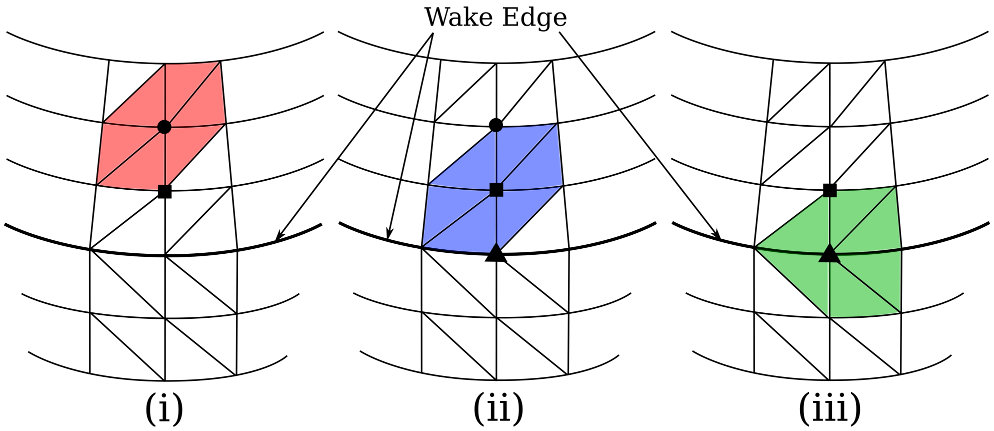


Figure 4.10: Supporting panels approach the wake edge.

demands one of these points is the triangular point which lies on the wake edge. If the Galerkin method is to be used to determine the velocity at the triangle, then the support shown in (iii) is required. The required support of this point would appear to invalidate the use of the Galerkin method as points on the opposite side of the wake are required. However, it can be demonstrated that in the special case of $[\phi]$ being zero at the wake edge, the equations do, in fact, truncate.

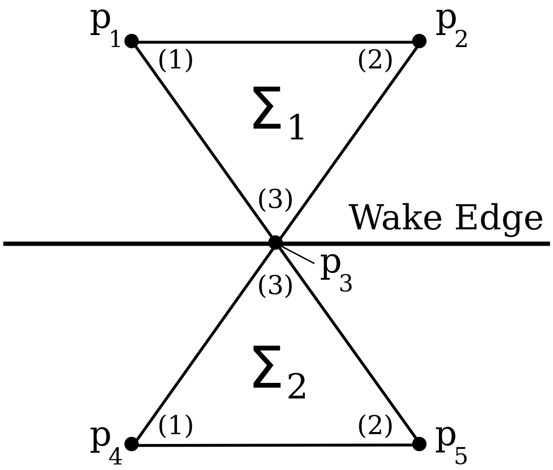


Figure 4.11: Simplistic illustrative geometry at the wake edge.

As an illustration, consider the special case where just two panels are in support of a point at the wake edge (such a situation may arise at the corner of the wake). Depicted in figure 4.11, the numbers in parentheses represent the local node numbering and the opposite ordering of local numbering ensures a combined normal once the panels are brought together.

Considering the point p_3, supported by panels Σ_1 and Σ_2, the left hand side of the discretised equation reads

$$\sum_{i=1}^{2} \int_{\Sigma_i} \hat{\psi}_i \frac{\partial \phi}{\partial \mathbf{E}_k} dP. \tag{4.94}$$

Now estimate the gradient so that

$$\frac{\partial \phi}{\partial \mathbf{E}_k} = \sum_{j=1}^{3} \psi_j \frac{\partial \phi_j}{\partial \mathbf{E}_k}. \tag{4.95}$$

Thus, equation (4.94) becomes

$$\begin{aligned} &\sum_{i=1}^{2} \sum_{j=1}^{3} \frac{\partial \phi_j}{\partial \mathbf{E}_k} \int_{\Sigma_i} \psi_j \hat{\psi}_i \, dP = \\ &\qquad \frac{\partial \phi_1}{\partial \mathbf{E}_k} \int \psi_1 \psi_3 \, dP + \frac{\partial \phi_2}{\partial \mathbf{E}_k} \int \psi_2 \psi_3 \, dP + \frac{\partial \phi_3}{\partial \mathbf{E}_k} \int \psi_3 \psi_3 \, dP + \\ &\qquad\qquad \frac{\partial \phi_4}{\partial \mathbf{E}_k} \int \psi_1 \psi_3 \, dP + \frac{\partial \phi_5}{\partial \mathbf{E}_k} \int \psi_2 \psi_3 \, dP + \frac{\partial \phi_3}{\partial \mathbf{E}_k} \int \psi_3 \psi_3 \, dP. \end{aligned} \tag{4.96}$$

Grouping the relevant terms observe that the right hand side of equation (4.96) becomes

$$\begin{aligned} &\left(\frac{\partial \phi_1}{\partial \mathbf{E}_k} + \frac{\partial \phi_4}{\partial \mathbf{E}_k} \right) \int \psi_1 \psi_3 \, dP + \\ &\qquad \left(\frac{\partial \phi_2}{\partial \mathbf{E}_k} + \frac{\partial \phi_5}{\partial \mathbf{E}_k} \right) \int \psi_2 \psi_3 \, dP + 2 \frac{\partial \phi_3}{\partial \mathbf{E}_k} \int \psi_3 \psi_3 \, dP. \end{aligned} \tag{4.97}$$

Closing the two sides of the wake, the points p_1 and p_2 are now coincident with points p_4 and p_5, respectively. Equation (4.97) may now be written

$$2\mathbf{v}_{\text{av1}} \int \psi_1 \psi_3 \, dP + 2\mathbf{v}_{\text{av2}} \int \psi_2 \psi_3 \, dP + \mathbf{v}_3 \int \psi_3 \psi_3 \, dP, \tag{4.98}$$

where $\mathbf{v}_{\text{av1}}$ and $\mathbf{v}_{\text{av2}}$ are the average velocities across the closed wake sheet at nodes 1 and 2, as required. The velocity at the wake edge, $\mathbf{v}_3$, is also present, assuming that the velocity at the top and bottom of the wake is identical here. For this to be true, it is critical that there is only one value of ϕ at p_3. Therefore it can be seen that the value of $[\phi]$ at p_3 is

$$[\phi] = \phi^+(p_3) - \phi^-(p_3) = \phi(p_3) - \phi(p_3) = 0. \tag{4.99}$$

Thus, $[\phi]$ must be zero at the wake edge.

The above result confirms that the Galerkin equations for the wake will truncate at the wake edge if and only if the value of $[\phi]$ is zero. If this is not the case (such as on the trailing edge) then the velocity at the edges must be supplied to the Galerkin method in order to truncate the equations.

The requirements of the domain of the right hand side of equation (4.90) remains to be determined. The difficulties concerning the right hand side of the Galerkin method are demonstrated in figure 4.12. On the left of the diagram the trailing edge has non-zero $[\phi]$ and the Galerkin equation for a point one line of panels from the edge will require an edge adjacent integral from a panel on the opposite side of the wake. This will invalidate the method for such points. This complicates the use of the Galerkin method at the trailing edge, as the velocity can only be sought *two points* away from the edge. Consequently, the velocity must also be known *a priori* one point away from the trailing edge and $[\phi]$ must be known on the trailing edge. This information can be provided if the velocity of a point at the trailing edge, determined by the Kutta condition, is carried for an extra time-step.

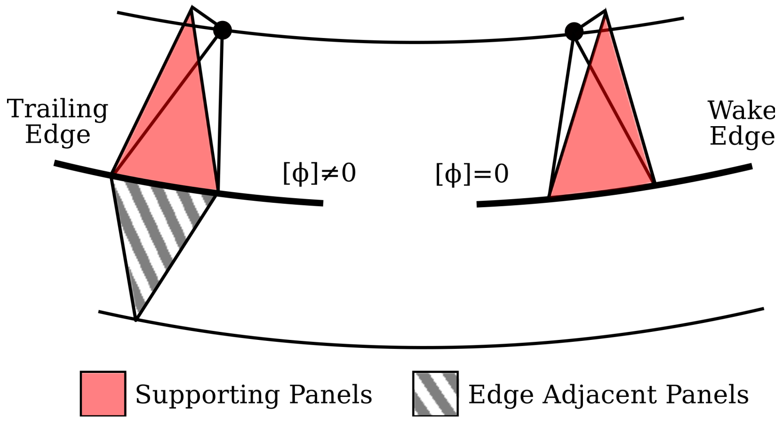


Figure 4.12: The impact of $[\phi]$ on the edge-adjacent integral for points close to edges.

The need for an edge-adjacent integral from the opposite side does not arise for points that lie one panel away from a wake edge (where $[\phi] = 0$) as shown on the right of figure 4.12. As this integral only depends on nodes lying on the edge, having zero value there will nullify this integral and thus the Galerkin method can be used for such points. All that remains is to show that the right hand side of equation (4.90) can be used on the wake edge itself when $[\phi]$ is zero.

Returning to the domain illustrated by figure 4.11, the right hand side of the Galerkin equation is given by

$$\left\{ \lim_{P_I \to P} - \lim_{P_E \to P} \right\} \int_{\Sigma_i} \hat{\psi}_i(P) \int_{\Sigma_{\mathrm{NZ}}} \phi(Q) \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}} \, dQ \, dP. \tag{4.100}$$

Assuming that ϕ can be estimated by the values at the nodes of the triangles by

$$\phi = \sum_{j=1}^{3} \psi_j \phi_j, \tag{4.101}$$

then equation (4.100) is discretised as

$$\sum_{i=1}^{2} \sum_{j=1}^{3} \phi_j \left\{ \lim_{P_I \to P} - \lim_{P_E \to P} \right\} \int_{\Sigma_i} \int_{\Sigma_{\mathrm{NZ}}} \hat{\psi}_i \psi_j \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}} \, dQ \, dP. \tag{4.102}$$

Assuming the difference in limits, and restricting the domain to panels Σ_1 and Σ_2, expanding equation (4.102) reveals

$$\begin{aligned} &\phi_1 \int_{\Sigma_1} \int_{\Sigma_1} \psi_3 \psi_1 \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^+} \, dQ \, dP + \phi_2 \int_{\Sigma_1} \int_{\Sigma_1} \psi_3 \psi_2 \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^+} \, dQ \, dP + \\ &\phi_3 \int_{\Sigma_1} \int_{\Sigma_1} \psi_3 \psi_3 \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^+} \, dQ \, dP + \phi_4 \int_{\Sigma_2} \int_{\Sigma_1} \psi_3 \psi_1 \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^-} \, dQ \, dP + \\ &\phi_5 \int_{\Sigma_2} \int_{\Sigma_1} \psi_3 \psi_2 \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^-} \, dQ \, dP + \phi_3 \int_{\Sigma_2} \int_{\Sigma_1} \psi_3 \psi_3 \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^-} \, dQ \, dP. \end{aligned} \tag{4.103}$$

By enforcing a shared normal observe that

$$\frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^-} = - \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^+}. \tag{4.104}$$

Thus bringing the wake sides together, equation (4.103) reduces to

$$\begin{aligned} (\phi_1 - \phi_4) \int_{\Sigma_1} \int_{\Sigma_1} \psi_3 \psi_1 \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^+} \, dQ \, dP \, + \\ (\phi_2 - \phi_5) \int_{\Sigma_1} \int_{\Sigma_1} \psi_3 \psi_2 \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^+} \, dQ \, dP \, + \\ (\phi_3 - \phi_3) \int_{\Sigma_1} \int_{\Sigma_1} \psi_3 \psi_3 \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^+} \, dQ \, dP. \end{aligned} \tag{4.105}$$

By setting $(\phi_1 - \phi_4) = [\phi]_1$ and $(\phi_2 - \phi_5) = [\phi]_2$ and recognising that at the wake edge point $[\phi] = 0 = \phi_3 - \phi_3$, the result is

$$[\phi]_1 \int_{\Sigma_1} \int_{\Sigma_1} \psi_3 \psi_1 \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^+} \, dQ \, dP + [\phi]_2 \int_{\Sigma_1} \int_{\Sigma_1} \psi_3 \psi_2 \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}^+} \, dQ \, dP \tag{4.106}$$

as required.

With the other panels in Σ_{NZ} following similarly, this proves that both the left and right hand sides of the Galerkin equation are applicable to a wake edge where $[\phi]$ is zero. In spite of this, it was found in numerical experiments that the Galerkin method was unstable on an open surface, even when the edges are set to zero. In discussions with the creator of the method, Professor Gray described that the solution to crack problems is unbounded at the edges even with zero $[\phi]$ (see Williams, 1952). In fact, this result is expected for wakes edges, also; it makes sense that the wake edge should have infinite velocity here, as the flow will be "turning the corner" between the top and bottom of the vortex sheet. Enforced by Johnson (1994, pg. 546), the velocity here will naturally be divergent. Since the solution is sought exactly at the wake edge, the singularity which is cancelled at the trailing edge by the Kutta condition has now simply been transformed to the wake edges. This was not an issue with previous methods that sought the velocity at the centre of discretised panels.

An alternative method must be used to calculate the velocity at the wake edges. As the Galerkin method is stable for a closed surface, truncating it by supplying values at the wake edge will 'trick' the method into operating as if the surface were closed and the velocity at the wake edges bounded. The kinematics of $[\phi]$, described by equation (2.58), make it convenient to use $[\phi] = 0$ at the wake edges. This allows the potential use of an alternative, *collocation* type method at the wake edges (see appendix H). The zero value of $[\phi]$ cancels the worst singularities resulting from the collocation method, however, it was surmised that the singular behaviour at the edges would continue to be reproduced and thus this method was abandoned. Finally, in the absence of time to develop a more advanced alternative, a basic off surface averaging technique was chosen to supply the wake edge values to the Galerkin gradient method. The implementation of this method is described in section 6.3.2.

4.4 Time-Stepping

With initial conditions supplied, the methods described above can be used to generate the velocity field on any surface within domain D. The kinematic and dynamic boundary condition on the free surface, given by equations (2.23) and (2.27), in conjunction with wake evolution equation (see equation (2.57)) and the Kutta condition, can be used to update the free surface and wake geometry for each subsequent time-step. This sections examines the schemes used to integrate the boundary conditions in time.

Both implicit and explicit schemes can be used, each having significant advantages and disadvantages. The development of the explicit scheme is accredited to Dold and Peregrine (1986); Dold (1992). They found that all the derivatives of the Laplace equation with respect to time also satisfy the Laplace equation, i.e,

$$\nabla^2\phi = \nabla^2\phi_t = \nabla^2\phi_{tt} = \ldots = \nabla^2\phi_{nt} = 0. \tag{4.107}$$

As a consequence of equation (4.107), a truncated Taylor series can be used to time-step forward the position and velocity potential of the dynamic surfaces. Given that the majority of computational effort in boundary elements comes from forming the influence matrices, these can be reused when calculating the later time derivatives of ϕ. This is highly advantageous over implicit schemes that require the results from varying geometries and thus differing influence matrices. Additionally, as an explicit scheme has no requirement on previous or intermediate solutions, the time-step can be varied without expense. For an implicit scheme, a change of time-step will involve additional expense as previous time-steps may no longer be used.

The challenge with explicit schemes is the necessity to calculate high order derivatives in order to satisfy the boundary conditions. The formulae describing the boundary conditions not only become more complicated, but when using numerical differentiation to calculate the derivatives, the accuracy of the higher derivatives reduces (thus requiring more points in the differencing formulae). Nevertheless, these difficulties were overcome for 2D simulations by Dold and Peregrine (1986) and for 3D simulation by Grilli *et al.* (2001).

Although an explicit scheme would be preferential to an implicit scheme for the reasons given above, in the present simulations an implicit scheme was chosen. Primarily, this decision was made to avoid additional difficulty where it was unnecessary, however there were additional considerations taken into account. Unlike the work of Grilli *et al.* and Dold and Peregrine, the current simulation contains a lifting surface and the lifting surface has a dynamic wake. As was established earlier, the velocity of the wake cannot be determined using numerical differentiation. Thus, a direct solution to the hypersingular boundary element equation is required to give the spatial derivatives on the wake. As such, the next higher order spatial derivative on the wake would require a supersingular integral to be solved at the collocation points. For a piecewise linear discretisation, this is yet to be achieved. However, the explicit time-stepping approach could still be feasible for B-spline discretisations.

The application of a mixed Eulerian-Lagrangian time-stepping scheme to free surfaces is attributed to Longuet-Higgins and Cokelet (1976). They used a fourth order Adams-Bashforth-Moutlon (ABM4) predictor-corrector scheme for their time integrations. As this method requires knowledge of three previous time-steps, a fourth order Runge-Kutta (RK4) multi-stage scheme is used to provide the initial steps. The ABM4 scheme requires one additional *predictor* step to be calculated before it can time-step and the RK4 requires three simulations of *mini-steps* prior to completing the time-step.

Dynamic choice of the time-step, Δt, is not desirable using an ABM4/RK4 scheme as the more expensive RK4 algorithm must be employed each time Δt changes. However, in order to maintain stability in the solution, it is necessary to change the time-step as the simulation progresses based on a number of metrics.

Dommermuth *et al.* (1988) carried out a stability analysis of the linearised free surface problem for the ABM4 and RK4 schemes. They found that ABM4 was weakly unstable and that RK4 was stable on condition that

$$\Delta t^2 \leq \frac{8\Delta x}{\pi g} \tag{4.108}$$

where Δx is the spatial reference length.

Xü (1992) extended this condition by additionally requiring compliance with the Courant-Friedrichs-Lewy condition based on the wave speed, c, given by

$$\Delta t = C_n \Delta x \; c \tag{4.109}$$

for $0 < C_n < 1$, where C_n is the Courant-Friedrichs-Lewy (CFL) number. If the simulation has no wave speed, such as for pumped flow, then c can be substituted with the free stream velocity, U.

Using the same process as Xü (1992), a fixed value of C_n is chosen for each simulation. The spatial reference, Δx, is found by calculating the minimum node spacing on the grid prior to each time step. Once Δx has been found, a value of Δt is then given by equation (4.109). Finally, the values of Δt and Δx are substituted into equation (4.108) to check for compliance and failing that Δt is further reduced until equation (4.108) is satisfied. The process is then repeated for each time step. To avoid unnecessary use of the RK4 scheme, the time step is only changed if the new Δt is smaller and, if so, the new time step is then reduced further such that $\Delta t_{\text{new}} = \alpha \Delta t$ where, for the present study, $\alpha = 0.9$. This avoids the time step converging with increasingly small changes. The entire procedure is known throughout the rest of the thesis as the *dynamic time stepping* procedure.

The impact of implementing these various stability conditions on the free surface calculations is illustrated in chapter 5. Should a lifting surface be included in the flow a wake surface will also need to be evolved over time. Although the author is unaware of any stability analysis for unsteady wakes in panel methods, the time-step may require further restriction in order to allow roll-up where curvature of the wake surface becomes great. If the velocities on the wake are also large then a reduced time-step may become necessary to avoid the wake crossing itself. In fact, the spatial Kutta condition used to generate the wake from the aerofoil trailing edge fixes the CFL number for the wake as is seen in section 6.4.

Chapter 5

Results: Non-Lifting

5.1 Introduction

This chapter is dedicated to establishing the accuracy and applicability of the boundary element solver, developed in the previous chapters, to free surface and non-lifting body problems. To gain confidence in the ability of the method, a number of verification stages are required. These stages include checks on the convergence of the method, and the accuracy of the integration, by comparison with analytical solutions such as that for potential flow about a sphere. Previously, a number of local, single panel tests were completed in order to develop the adaptive numerical scheme in chapter 3. The tests in this chapter will provide an estimate for the global accuracy of the method at a particular grid resolution.

Given that the flow about a sphere is a Neumann problem, a test that involves the solution of a Dirichlet surface is also desirable. For this, a channel with a free surface and constant uniform velocity applied was studied. This problem not only facilitates a check for a simple mixed Neumann/Dirichlet problem, it will also test the implementation of the channel boundaries. These can either be discretised fully or part discretised using reflective surfaces. Time-stepping is used for the first time here (allowing an examination of the stability of the solution); the intersection of the free surface and the channel wall is also of great interest. In addition, checks on mass and energy conservation will provide further evidence towards the quality of the simulations.

With a basic test of the mixed Neumann/Dirichlet problem complete, a comprehensive test of the free surface boundary conditions is required. A good option for such a test is to generate a solitary wave and track its progress over time. Theoretically, for an infinite domain, this wave, known as a *soliton*, should be invariant in form as it travels. For the channels developed in this method, the domains are not infinite, and, thus, the soliton will be impeded by the channel ends at some point. However, if the channel is long enough, a period of time is expected where the form of the soliton will be nearly invariant. Additional numerical procedures can also be tested at this stage such as smoothing of instabilities and adaptive time-stepping.

5.2 Convergence Study of Steady Flow Past a Unit Sphere

Within this section the numerical results for the steady potential flow past a sphere are analysed. The analytical solution is well known and provides a good test of the accuracy of the numerical method at a particular grid resolution. With this information an estimated order of convergence can be calculated and Richardson extrapolation carried out to provide a prediction of the numerical solution at infinite grid resolution.

One important barrier to carrying out the sphere test is forming the geometry using nine node bi-quadratic panels. Unlike triangular panels, surfaces mapped with rectangular panels are difficult to 'cap' or, in other words, finish a surface at a point. This process is not only important for meshing the sphere but also for capping the ends of lifting surfaces. Thus a method for providing surface closure without the use of degenerate triangles is presented.

5.2.1 Generation of a Hemisphere

To produce a grid for the simulation of the flow past a sphere, two hemispheres must be panelled using nine node bi-quadratic panels and then joined. Along the equator of each hemisphere the panels must be aligned regularly in order to easily join the surfaces. The sphere may be discretised in a variety of ways. One way is to create rings of panels, placed one above the other, until the top is reached. The advantages of this approach are that the sphere may be easily built from sections, or two-dimensional slices, reducing in radius as the pole is approached. In addition, any shape can be built this way, including caps of aerofoil sections. The disadvantages are twofold: first, maintaining the number of panels used on the ring at the equator as the pole is approached will produce very distorted panels on the later rings. Secondly, there is no obvious way to finish the last ring without using degenerate panels to form a point. These problems are evident in

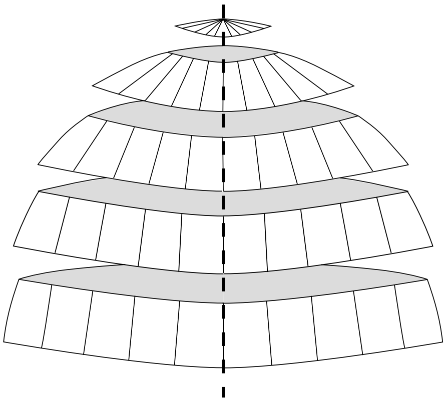


Figure 5.1: Exploded hemisphere discretised by 'rings' of quadrilateral panels.

figure 5.1.

An alternative solution is to discretise the hemisphere in quadrants, rather than as a whole surface. For instance consider the hemisphere made up of four quarters as seen in figure 5.2a. These patches still finish at a point, so in order to place a non-degenerate panel here, one of the nine node panels must be rotated by forty five degrees. This *diamond panel* provides a useful means to discretise the tip of the quadrant and also to reduce the number of panels in each ring as the pole is approached from the equator. Figure 5.2b illustrates how the process works. A column of diamond panels climbs above the equatorial row of panels to the tip of the quadrant. Approaching the tip, one panel is removed from the first ring and two from each subsequent ring. In general this surface can still be produced from nodes on circular sections. All the nodes lie on a section, apart from the edge (non-corner or centre) nodes of the diamond panels. The position of these nodes must be estimated. Because of this, the diamond panel approach can still be used for other surfaces, such as aerofoils. A sphere discretised using the diamond panel technique can be seen in figure 5.3.

A serious disadvantage of the diamond panel approach is a loss of accuracy of the numerical derivatives caused by nodes being far from collinear. Because the node numbering is irregular at these panels and the change of direction so severe, the numerical differentiation scheme is reduced in accuracy and is more complicated to implement.

In an effort to analyse the impact of using the diamond panel approach on a hemisphere, an alternative approach was required for comparison. The second approach moves away from building a hemisphere in strips by choosing to split the hemisphere

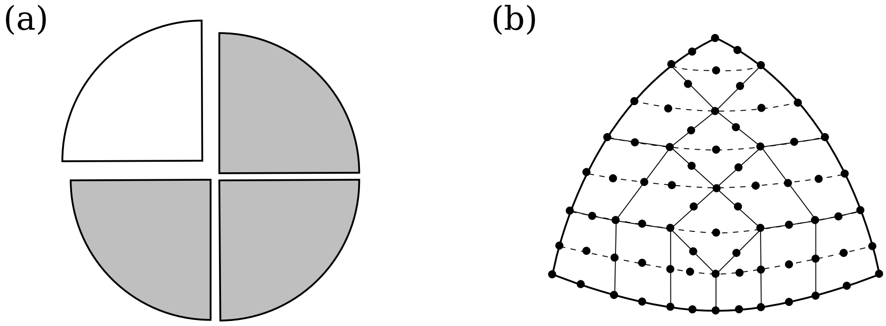
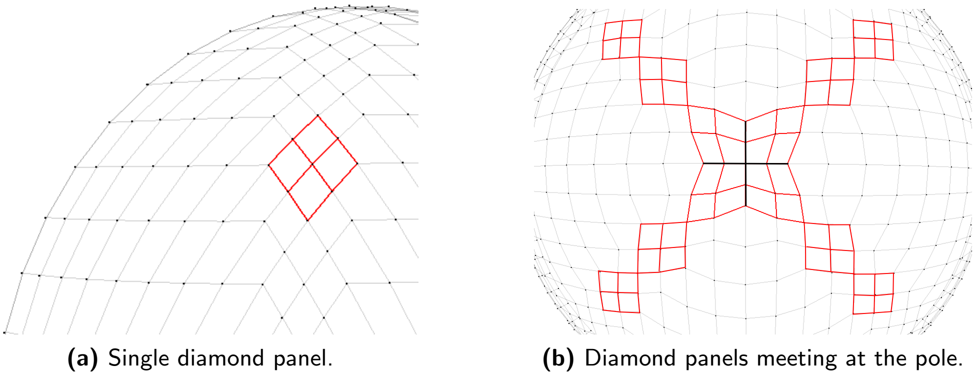


Figure 5.2: Splitting the hemisphere into quadrants (a) and panelling each quadrant using rotated 'diamond' panels (b).

into twelve equal *patches*. The reference patch is enclosed by four planes through the origin. These planes are:

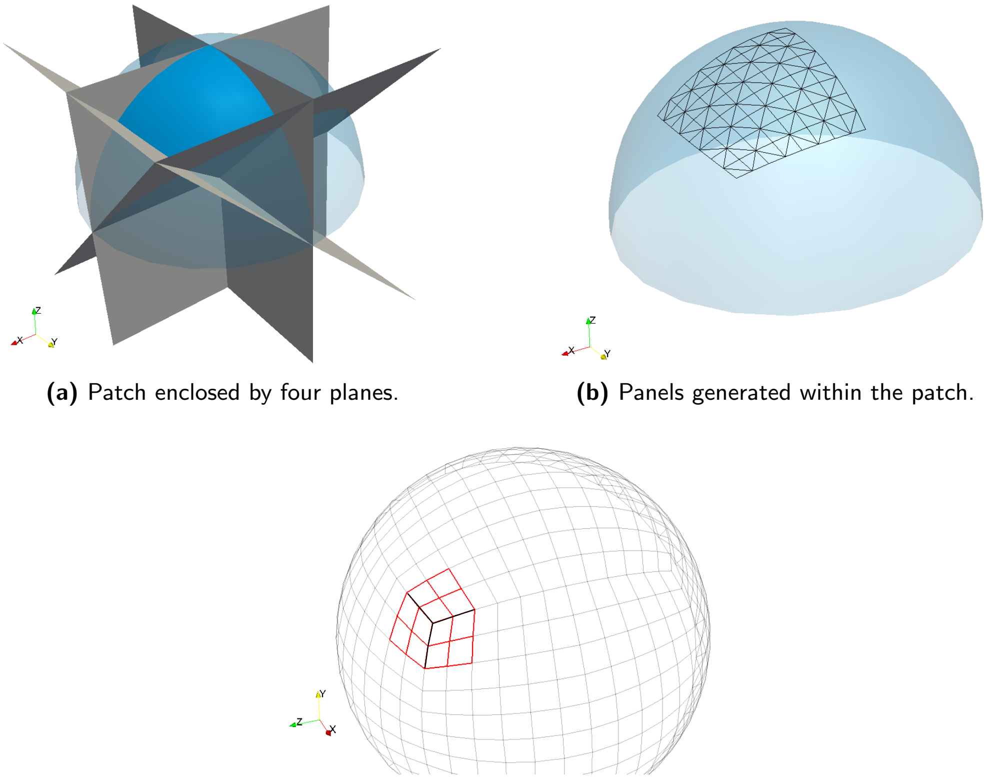
1. The x–z plane,
2. The y–z plane,
3. The plane with normal $(0, -1/\sqrt{2}, 1/\sqrt{2})$,
4. The plane with normal $(-1/\sqrt{2}, 0, 1/\sqrt{2})$.

These planes and the resulting patch are illustrated in figure 5.4a. The patch is a spherical polygon, bounded by the arcs of four great circles. These arcs can be divided into a number of sub-arcs along the edges of the patch with evenly spaced nodes. By passing planes through these nodes and the origin, and then calculating the intersection of these planes and a unit sphere, a set of nodes, and hence panels, for the patch can be formed. The result of this process is illustrated in figure 5.4b.


(a) Single diamond panel.

(b) Diamond panels meeting at the pole.

Figure 5.3: Diamond panel discretisation on the unit sphere.



(a) Patch enclosed by four planes.

(b) Panels generated within the patch.

(c) Join between patches.

Figure 5.4: Patch representation of the hemisphere.

Over an entire sphere or hemisphere, the patch meshing scheme generates a more regular grid than the diamond scheme. However, this meshing scheme suffers some similar difficulties to the diamond panel scheme. Figure 5.4c shows one of eight regions on a sphere where panels meet at a 120° angle to each other. At these points numerical differentiation is less accurate, as well as being more challenging to implement. Nonetheless, in comparison to the diamond panel scheme, these problematic nodes are significantly less in number. An additional drawback of the patch meshing scheme is that in order to discretise shapes other than a sphere, the sphere or hemisphere must be discretised first and then transformed to the chosen shape. For more challenging shapes (such as aerofoil caps) these transformations are not easy to calculate and it is also difficult to control the spacing between nodes.

5.2.2 Numerical Results

The purpose of this section is to verify the Eulerian stage of the boundary element method using a Neumann boundary problem. This is achieved by employing a grid convergence study similar to those developed for computational fluid dynamics methods (see Roache, 1998). By systematically refining the grid resolution and comparing the error at each stage, an estimate of the error at infinite grid refinement can be found. In addition, the order of grid convergence can be established. Given that the discretisation of the panels is quadratic, second order convergence is a desirable result although this may be slightly impeded by the adaptive numerical quadrature capping the local integrals to an error of 1×10^{-4}. This process will also provide a comparison of the two gridding methods developed in the previous section, the premise being that the patch type gridding scheme will be more accurate than the diamond type scheme, particularly for a maximum local error.

The analytical solution for the flow past a uniform sphere must be formulated in terms of the perturbation potential, as no discretised boundaries exist to enforce a fluid velocity external to the sphere. The analytical results for this case (as formulated in Katz and Plotkin (2001)) is given by

$$\phi = \frac{1}{2}Ux, \qquad \phi_n = -\mathbf{n} \cdot \mathbf{U} \tag{5.1}$$

where $\mathbf{U} = (U, 0, 0)$ is the far field velocity and $\mathbf{n}$ is the outward normal on the surface of the sphere. As part of the test, ϕ_n is specified on the sphere and the value of ϕ is sought. As relative errors are examined, the analytical and numerical values are normalised by the absolute maximum of the solution (in this case $\phi = 1/2$) to avoid the zero in the analytical result. This particular test was also carried out by Xü (1992) for his QBEM (Quadratic Boundary Element Method) scheme and a comparison at this point is prudent.

The results of the convergence study are graphed in figure 5.5 with the distribution of error on the sphere for each panelling scheme illustrated in figure 5.6. Note that three node linear triangles were also tested for comparison.

The approximate order of convergence for each panelling scheme is given in table 5.1. As expected the triangular panelling scheme has the lowest order of convergence, with the patch panelling being the highest. The apparent deficit of the nine-node diamond

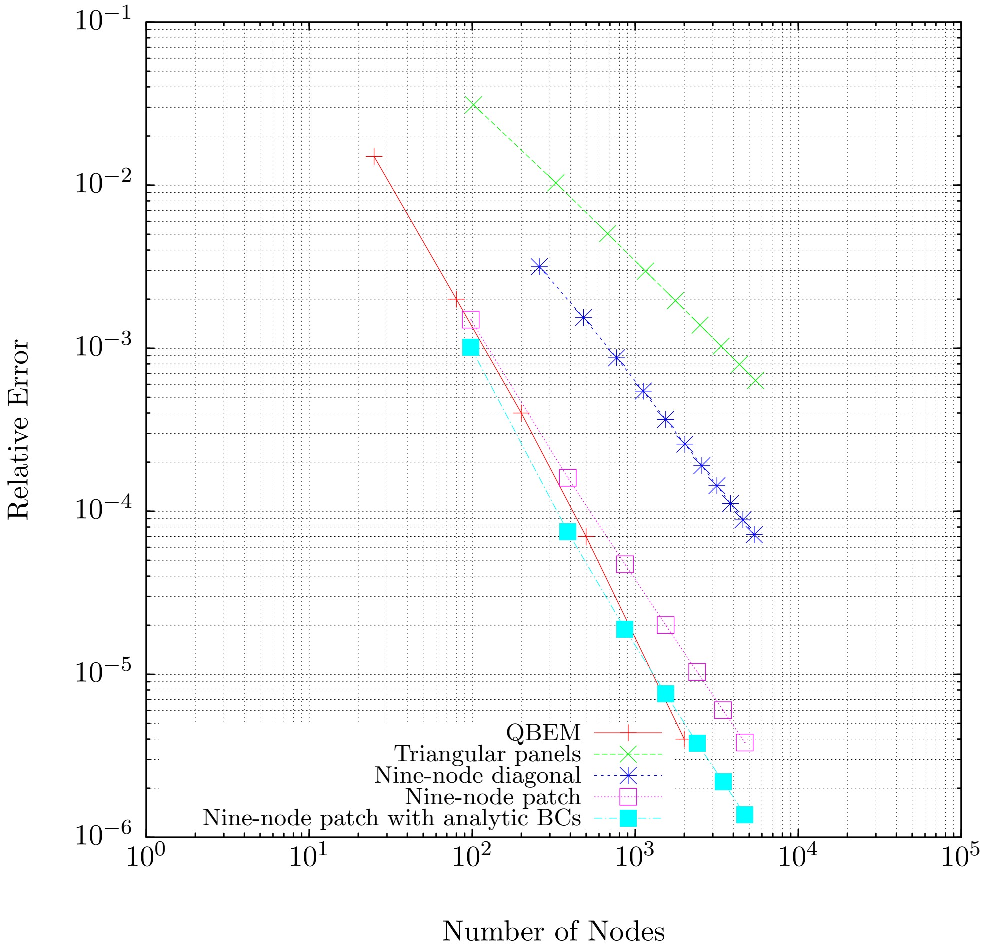
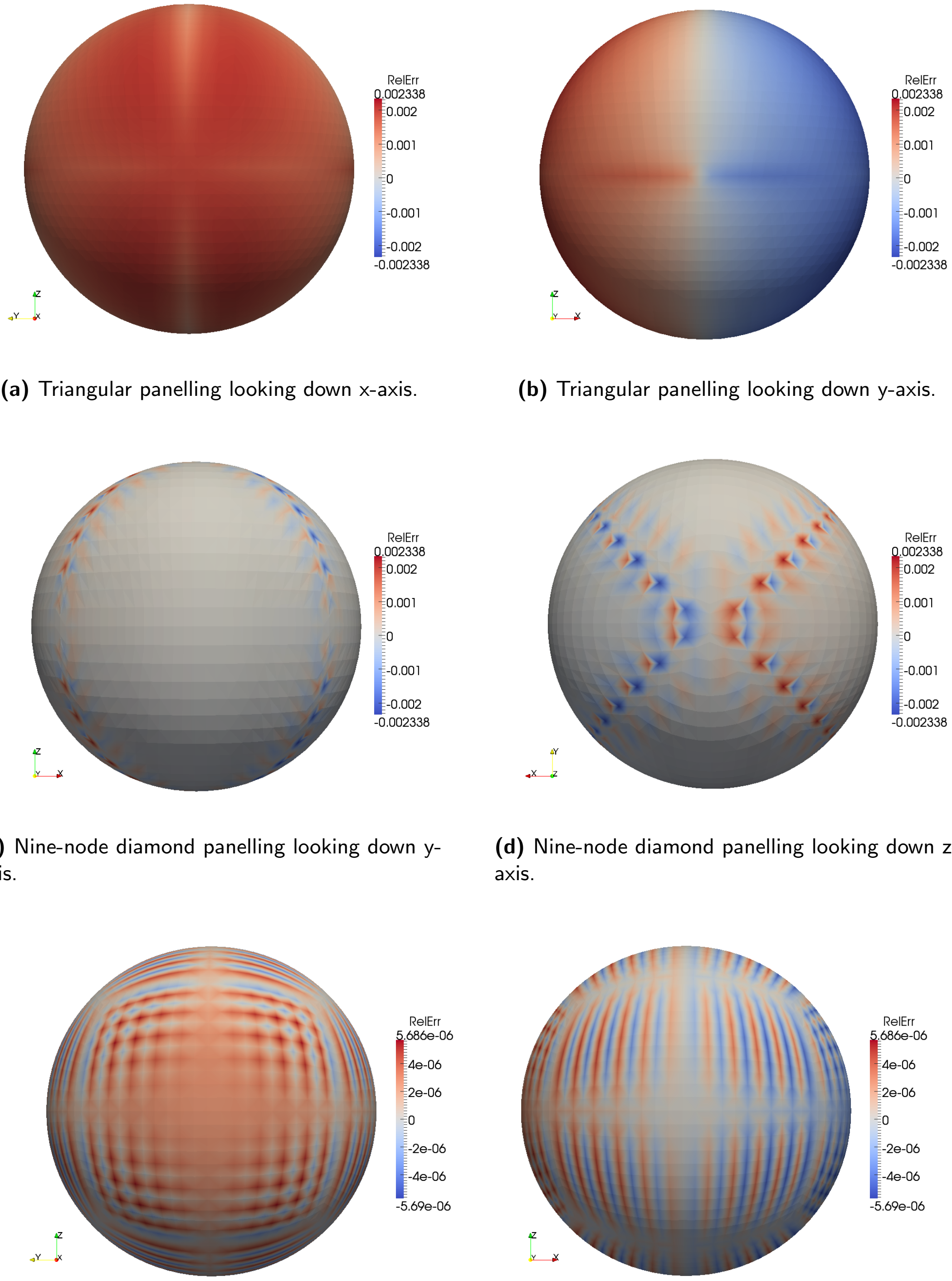


Figure 5.5: Error in velocity potential for flow past a unit sphere for the present method using alternative panelling schemes and QBEM developed in Xü (1992).



(a) Triangular panelling looking down x-axis.

(b) Triangular panelling looking down y-axis.

(c) Nine-node diamond panelling looking down y-axis.

(d) Nine-node diamond panelling looking down z-axis.

(e) Nine-node patch panelling looking down x-axis.

(f) Nine-node patch panelling looking down y-axis.

Figure 5.6: Patterns of relative error for the three different panelling schemes.

panelling scheme is illuminated by figure 5.6c and figure 5.6d. Here, large relative errors in comparison to the rest of the sphere can be seen in the vicinity of the diamond panels. It is surmised that it is these exaggerated errors that are responsible for the drop in the convergence rate. For the triangular panel case, figure 5.6a and figure 5.6b show a pattern of error that varies reasonably in line with the magnitude of the input ϕ_n. It is anticipated that these errors could be reduced with a more complicated adaptive scheme, particularly for the self-influences. However, as the occurrence of self-influences will be low (just at the trailing edge of the foil), the deficit in the global accuracy caused by using the panels to discretise the wake is manageable. In contrast to the diamond panelling scheme, the error for the patch panelling scheme is much more evenly distributed, as can be seen in figure 5.6e and figure 5.6f. Note that the error values are much less than the other two panelling schemes, indicated by the greatly reduced maximum and minimum on the scales of figure 5.6e and figure 5.6f.

Panelling Scheme	Order of Convergence
Triangular	1
Nine-node diamond	1.3
Nine-node patch	1.5
QBEM	1.9

Table 5.1: Order of convergence of the present method for flow past a unit sphere with various panelling schemes and also QBEM developed in Xü (1992).

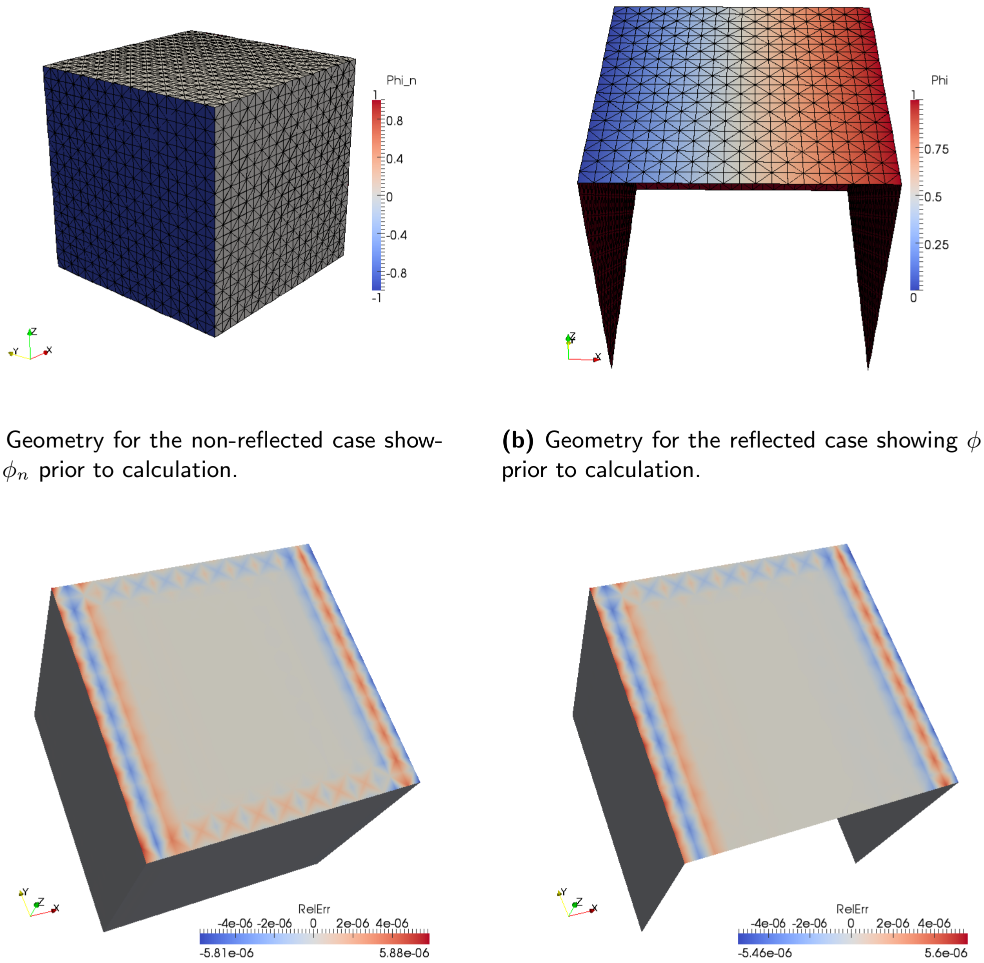
One surprising aspect of this convergence study was the deficit in performance in comparison to QBEM of even the best panelling scheme for the present method. The patch panelling scheme achieved an approximate order of convergence of 1.5 whereas QBEM was reported to have an order of convergence of 1.9. The reasons for this deficiency are not entirely clear. Given that the approximations to the panels are quadratic, an order of convergence close to 2 would seem a reasonable requirement. There are, however, a number of differences between the method developed in this thesis and QBEM. Firstly, the matrix inversion methods differ. Where QBEM used a preconditioned GMRES algorithm to invert the influence matrix, the current method relies on the Intel Math Kernel libraries. Also, whereas QBEM is solved in a vectorised manner it was found that loops proved to be more efficient for the current method, particularly when parallelised. Of most concern is the self-doublet influence (given by equations (3.40) and (3.42)) where no simple analytical test was found to evaluate the accuracy for an isolated panel or test the coding in isolation. This meant relying solely on the adaptive parameters of QBEM for the control of these integrals.

Even with these difficulties, it can be seen from figure 5.5 that the present method matches the accuracy of QBEM using the patch panelling scheme and exceeds it when analytical values are used to generate ϕ_n, i.e. the exact normals are used rather than the numerically computed normals. Due to the reduced order of convergence of the current scheme in comparison to QBEM, the accuracy of QBEM will outstrip the current method as more unknowns are used. However, it is unknown if QBEM could maintain this order of convergence once the target accuracy for the integral equations was surpassed, as results beyond a few thousand unknowns were not published in Xü (1992).

5.3 Convergence Study of Steady Flow Within a Cube

The previous study of flow past a unit sphere provided an excellent verification for a Neumann problem using the current method. Now a test involving Dirichlet boundaries is required, in which a test of the full potential, and the use of both reflective and non-reflective kernels, would be highly desirable. Such a test is available by examining the flow within a cube, where one or more boundaries are Dirichlet. To avoid the analytical solution for the potential being identically zero on the Dirichlet boundaries, the full potential method must be used, with the input and output Neumann boundaries enforcing the magnitude and direction of the flow throughout the cube. The other Neumann boundaries of the fluid domain are set with a no-crossing boundary condition and, if desired, two such boundaries can be removed with the use of a reflective Green's function.

The geometry for the reflected and non-reflected cases are illustrated in figure 5.7a and figure 5.7b respectively. For the non-reflected case the faces of the cube are divided such that there are five Neumann boundaries, two of which are an inlet and outlet where the boundary conditions $\phi_n = \pm U$, respectively, are set (the inlet is seen as blue in figure 5.7a). The other Neumann boundaries are no-crossing boundaries. The top face is a Dirichlet boundary, with boundary condition $\phi = Ux$. The reflected geometry is similar apart from two of the no-crossing boundaries are not discretised, being replaced by reflections. With the geometry and boundary conditions established,



(a) Geometry for the non-reflected case showing ϕ_n prior to calculation.

(b) Geometry for the reflected case showing ϕ prior to calculation.

(c) Resulting relative error in ϕ_n for non-reflected case.

(d) Resulting relative error in ϕ_n for reflected case.

Figure 5.7: Verification test for flow through a cube with one Dirichlet surface.

the Neumann boundaries are solved for ϕ and the Dirichlet boundary is solved for ϕ_n.

Post calculation, figure 5.8 shows the results for increasing numbers of unknowns. The order of convergence for both the reflective and non-reflective cases is approximately 0.4. This could be interpreted as a disappointing result, apart from the fact that the method was beyond the target accuracy, even using the coarsest grid resolution. The importance of the target accuracy can be seen in the maximum relative errors. These are maintained below 10^{-5} by the adaptive numerical method, but, as expected, do not fall with increasing resolution. As can be seen in figure 5.7c and figure 5.7d, these errors occur along the edges of the faces, where the proximity to other faces has the

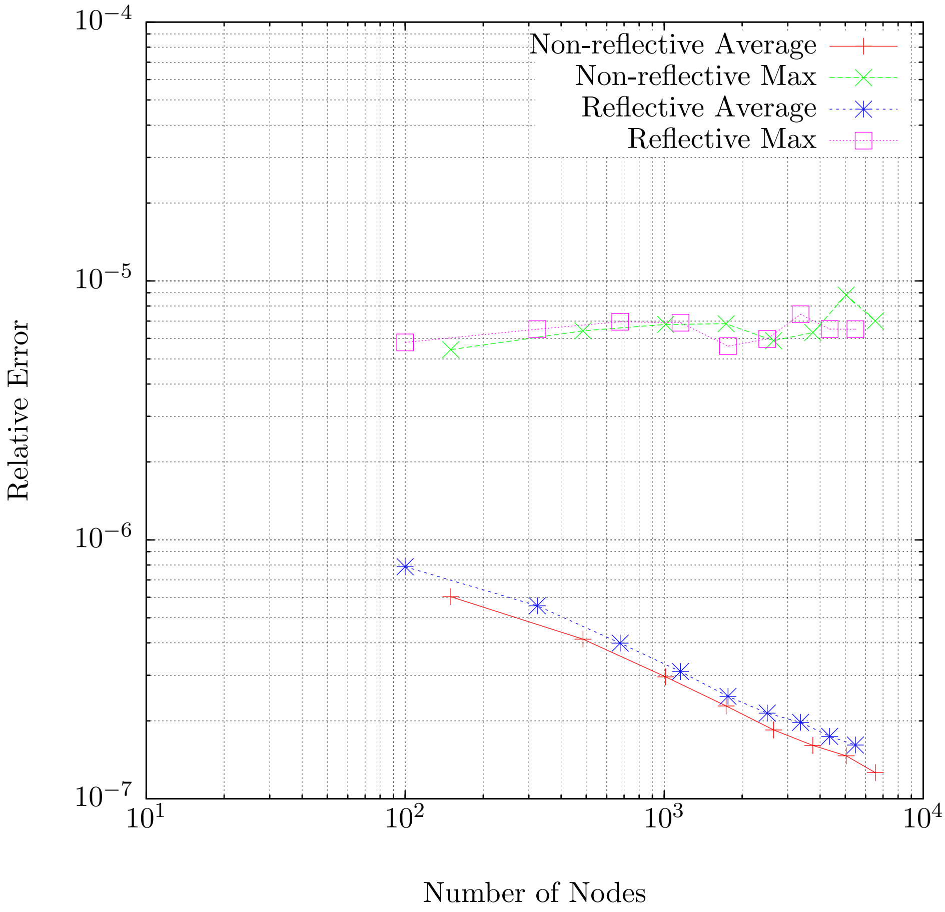


Figure 5.8: Average and maximum relative errors for steady flow through a channel containing a Dirichlet surface.

most impact. With increasing unknowns, there are additional panels in close proximity to other faces, thus increasing the number of nodes having the maximum allowed local error defined by the adaptive quadrature scheme (see section 3.3). Additionally, the flat surfaces do not make full advantage of the quadratic geometric approximation within the panels.

An interesting observation for this case is highlighted by the differences between figure 5.7c and figure 5.7d. The errors resulting from the proximity to nearby faces can be seen at the edges of the uppermost surface. By comparing the two figures, it can be seen that these errors interact (or superimpose themselves) to form errors of greater magnitude in the corners, indicated by the increased brightness of the colours there (also observe the destructive interference that occurs). Thus, there are less of these 'peak' errors in the reflected case as there is no surface at the reflective boundary and thus no induced errors there. This would imply, as expected, that the less boundaries used in the simulation, the more accurate it will be. This may seem to be contradictory to the results of figure 5.8 which shows the average relative errors of the reflected and non-reflected case for a particular number of nodes. Under scrutiny, this is not a fair comparison, as the number of nodes for the reflected case in this test was significantly less than for the non-reflected case. Thus, it is true to say that the reflected case will be more accurate than the non-reflected when using a *similar grid resolution*. In conclusion, for a channel simulation, the best results should be expected from the reflected channel geometry using the perturbation potential. Ideally, the other channel wall would be reflective too, but this results in a much more complicated Green's function and is left as further work.

5.4 Evolution of Single and Counterpropagating Solitary Waves

Testing of the time dependant free surface boundary conditions constitutes the next verification of the numerical model. This involves the quantitative examination of the implementation of the free surface boundary conditions. There are two parts to this implementation; the numerical differentiation applied to obtain velocities on the free surface and the numerical integration used to propagate the position of the free surface, and the value of ϕ upon it. The evolution in time of a solitary wave, or soliton, is an excellent test case for the free surface boundary conditions (and hence the related theory). The existence of solitary waves was first noticed by John Scott Russell in 1834 whilst observing canal boats on the Union canal at Hermiston, near Edinburgh. Naming the waves 'waves of translation', he famously recorded how the wave on the Union canal

> "continued its course along the channel apparently without change of form or diminution of speed."

Later to become known as solitary waves, they exhibit a finely balanced equilibrium between dispersive effects and the non-linearity of the governing equations. Their properties include the ability to travel in time without losing shape, amplitude or speed and the ability to interact with other solitary waves and remain (nearly) unaltered by the collisions. Indeed, in an inviscid fluid, a solitary wave should propagate, unchanging, for indefinite time.

The properties of solitary waves provide an excellent test for the present numerical method. Once a solitary wave has been generated, its form, celerity and energy can be monitored over time to give insight into the abilities of the model. By reversing, in space, the initial conditions of the solitary wave a second *counterpropagating* wave can be made to collide and then pass through the original solitary wave. As reported in Craig *et al.* (2006), the emerging waves should be nearly identical to their state before the collision; however, Craig *et al.* also show that a small reduction in celerity and amplitude should be expected for both waves, this energy loss being transferred to small residual waves.

Solitary waves have been used for verification of numerical codes developed by the likes of Grilli *et al.* (2001), Garzon *et al.* (2005), etc.; the generation, breaking and interactions of solitons remain a popular field of research. The only drawback of using solitons for verification of the present method is the lack of current flow in the exact solutions. As no far field current is present in the simulations, the full and perturbation boundary conditions are equal and thus both modes of the model are equivalent. To establish the accuracy and ability of the numerical model for free surface flows with waves and current in full and perturbation potential, a further test case would have to be devised.

5.4.1 Accuracy Checks

The performance of the numerical model can be evaluated by examining the total energy of the system. It is therefore necessary to calculate kinetic and potential energy within the numerical method. In addition, the ability to monitor the global mass/volume flux during simulations will provide a test for mass/volume conservation. In particular, the total volume flux across S should be zero:

$$\iint_S \phi_n(\mathbf{x}, t)\, dS = 0. \tag{5.2}$$

The volume error is the deviation of the left hand side of equation (5.2) from zero. In addition, a tolerance can be placed on this value to act as an emergency abort for the simulation, should the solution begin to diverge.

The kinetic energy is derived by integrating $\rho \mathbf{v}^2/2$ across the volume of the domain. This leads to

$$E_k = \frac{\rho}{2} \iiint_D (\nabla\phi)^2\, dV. \tag{5.3}$$

By noting that

$$(\nabla\phi)^2 = \nabla \cdot (\phi\nabla\phi), \tag{5.4}$$

equation (5.3) can be transformed into a surface integral by applying Gauss' theorem. Equation (5.3) therefore becomes

$$\begin{aligned} \frac{\rho}{2} \iiint_D (\nabla\phi)^2\, dV &= \frac{\rho}{2} \iiint_D \nabla \cdot (\phi\nabla\phi)\, dV \\ &= \frac{\rho}{2} \iint_S (\phi\nabla\phi) \cdot \mathbf{n}\, dS \\ \Rightarrow \qquad E_k &= \frac{\rho}{2} \iint_S \phi\phi_n\, dS. \end{aligned} \tag{5.5}$$

To derive the potential energy $\rho g z$ is integrated across the computational domain, resulting in

$$E_p = \rho g \iiint_D z\, dV. \tag{5.6}$$

To transform equation (5.6) into a form which is suitable for the application of Gauss' theorem, a vector field is sought such that

$$\nabla \cdot \mathbf{F} = \frac{\partial F_1}{\partial x} + \frac{\partial F_2}{\partial y} + \frac{\partial F_3}{\partial z} = z. \tag{5.7}$$

The above is true if $F_1 = F_2 = 0$ and $F_3 = z^2/2$. Thus, equation (5.6) can now be written as

$$\begin{aligned} \rho g \iiint_D z \, dV &= \rho g \iiint_D \nabla \cdot \mathbf{F} \, dV \\ &= \frac{\rho g}{2} \iint_S z^2 n_3 \, dS. \end{aligned} \tag{5.8}$$

As the contribution from the bottom surface is constant it can be treated as a datum. Thus, the potential energy can be defined solely in terms of the free surface such that

$$E_p = \frac{\rho g}{2} \iint_{S_F} \eta^2 n_3 \, dS \tag{5.9}$$

where η is the free surface elevation. Finally, the total energy is simply given by

$$E_t = E_k + E_p. \tag{5.10}$$

The numerically computed energy values are calculated and compared to the analytical solution at each time-step.

5.4.2 Evolution of a Single Soliton

This test case examines the time dependant propagation of a single solitary wave travelling in the positive x-direction. An exact solution for a solitary wave is provided by the method of Tanaka (1986) which is based on Cauchy's integral theorem using the complex velocity potential. Originally, the solitary waves generated by Tanaka's method were parametrised by the dimensionless crest velocity; however, the method was modified by Cooker (1990) to allow the non-dimensional wave height to be used instead. Grilli and Subramanya (1993) developed a code for computing this analytical solitary wave employing these techniques. The waves generated by Tanaka's method provide the initial geometry and potential distribution for the numerical method at a given resolution. The method also provides details of the wave energy and celerity, which will be used to compare with results from the simulated waves. Note that the output from Tanaka's method is two-dimensional and thus a 'per unit width' comparison must be made to the present three-dimension simulations.

The numerical simulations within this chapter are carried out in non-dimensional form. Choosing scales so that both the channel depth and the acceleration due to gravity are unity gives the following non-dimensional variables,

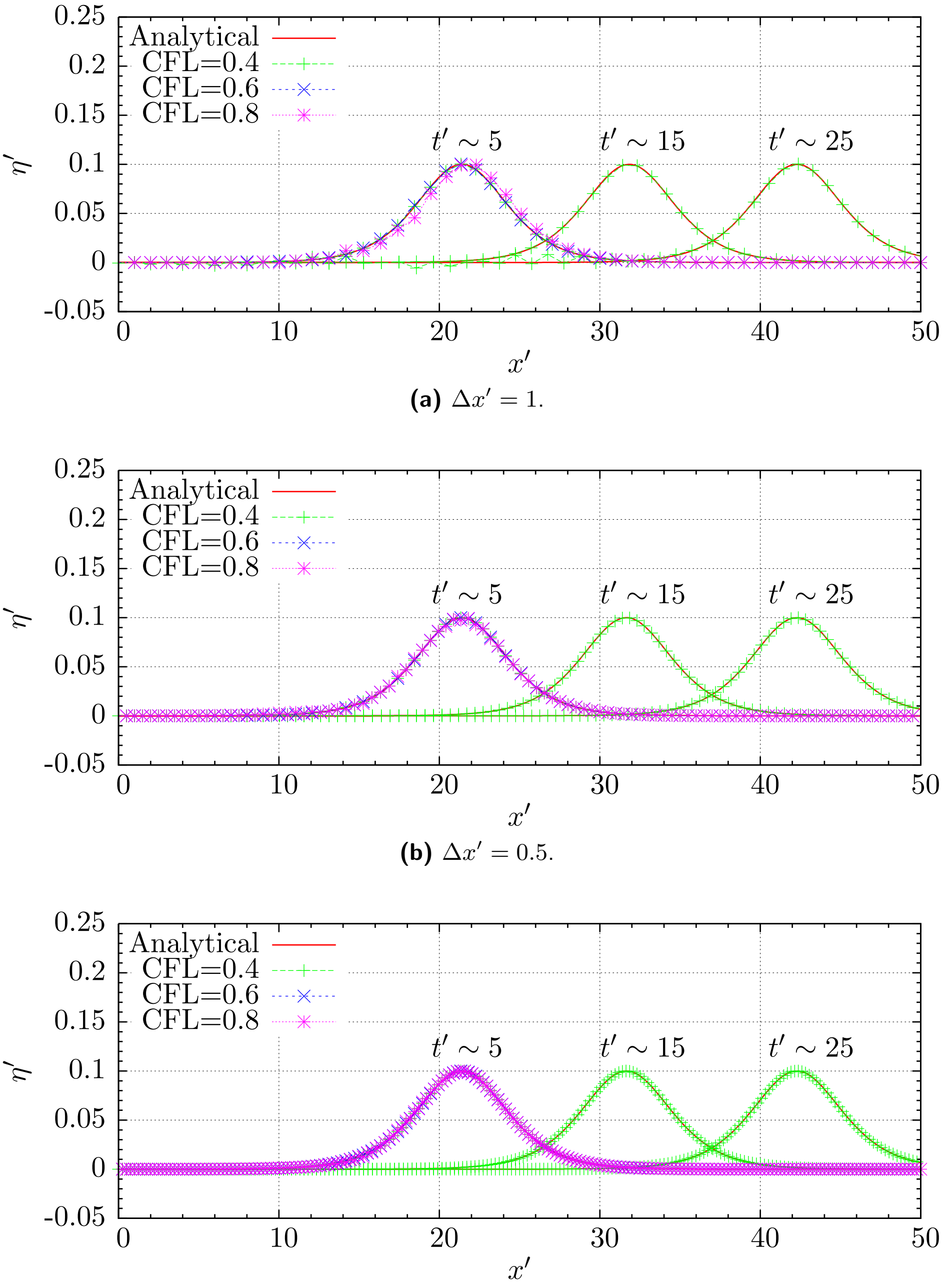
$$\mathbf{x}' = \frac{\mathbf{x}}{h_0}, \qquad t' = t\sqrt{\frac{g}{h_0}}, \qquad c' = \frac{c}{\sqrt{gh_0}} \tag{5.11}$$

where h_0 is the channel depth and c is the wave celerity. The waves generated by Tanaka's method are laterally truncated at 0.1% of the wave height and then used as initial conditions for the simulated waves. The length of the channel is set such that the wave will not encounter the end of the domain by the final time-step. The width of the channel is arbitrary, but is nominally set to unity although the node spacing is forced to be the same in the x and y direction, thus the width may vary slightly.

The single soliton test provides useful insight into the dynamic time-stepping (see section 4.4) within the present code and, in particular, the impact of the CFL number on the stability of the results. Three different non-dimensional initial wave heights were investigated, $H_0' = 0.1$, 0.15 and 0.2. For each wave height, three grid discretisations were tested, $\Delta x' = 1$, 0.5 and 0.25 and each discretisation was time-stepped dynamically with CFL numbers 0.8, 0.6 and 0.4. Each simulation was stepped to a maximum time of $t' = 30$. Figure 5.9, figure 5.10 and figure 5.11 show wave profiles for a slice in the x-direction at approximate times $t \sim 5$, 15 and 25. The times are approximate as the dynamic time-stepping will decide and alter the length of steps dependant on the given CFL number. An analytical profile is provided for each time that is shifted to coincide with the simulated profile using a CFL number of 0.4. η' is the non-dimensional free surface height.

Each of the figures show that there is a sensitivity to the CFL number regardless of wave height and discretisation. Only the simulations run with the CFL number of 0.4 consistently reached $t' = 30$. In addition, the stability of the higher CFL number simulations increased along with the wave height. This may seem counter intuitive given that the number of nodes per wavelength reduces as the waves gain in amplitude due to the fixed node spacing in x'. The effect appears to arise as an artifact of the dynamic time-stepping which reduces the time-step as the waves increase in amplitude. This is most likely due to the increased curvature at the wave crests causing the minimum node spacing to reduce in comparison to lower wave heights.

Figure 5.12 and figure 5.13 show the effect of increased discretisation at each wave height for simulations with 0.4 CFL number. Improvement in accuracy can be

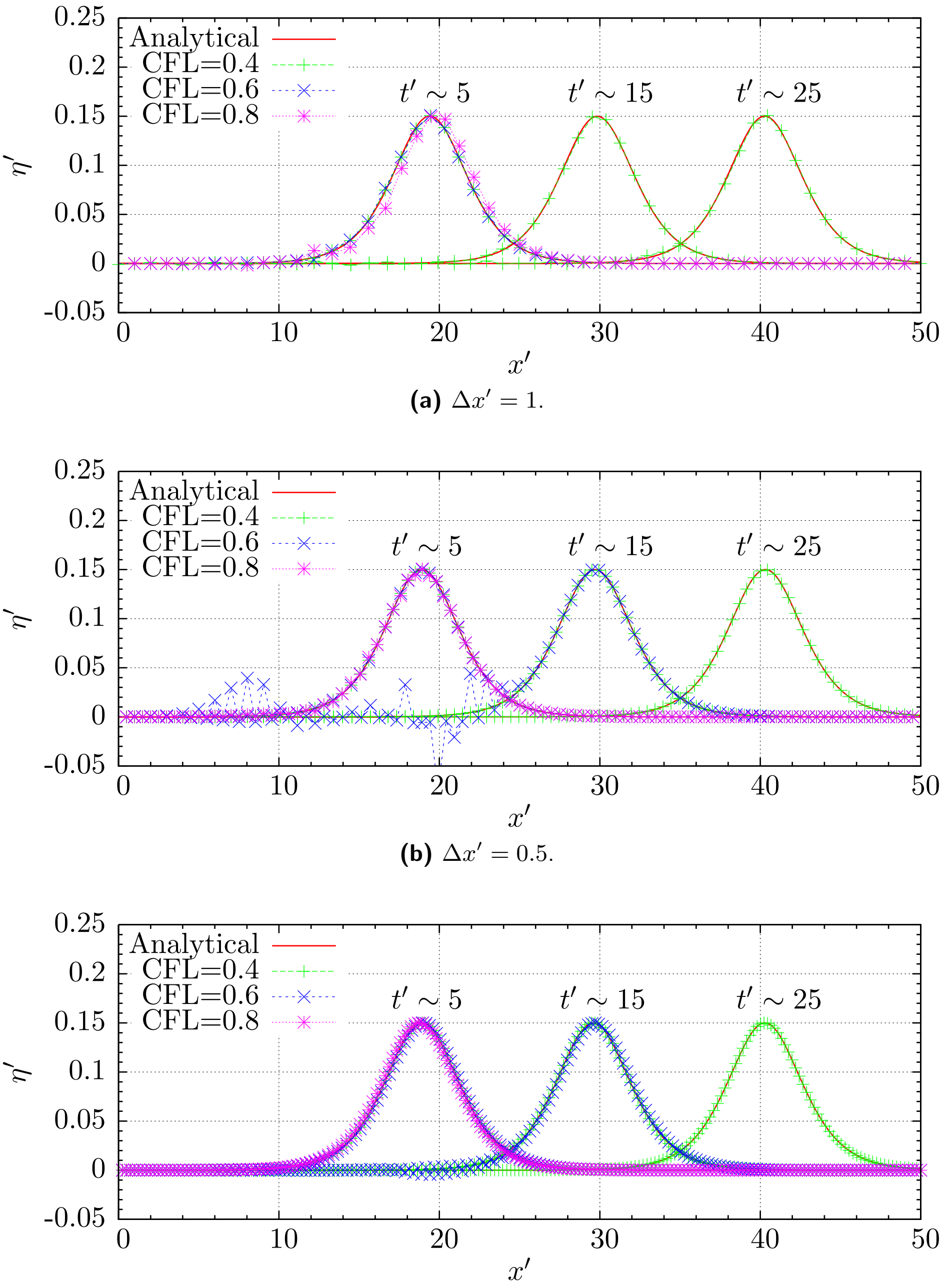


(a) $\Delta x' = 1$.

(b) $\Delta x' = 0.5$.

(c) $\Delta x' = 0.25$.

Figure 5.9: Wave profiles for soliton with wave height, $H' = 0.1$.

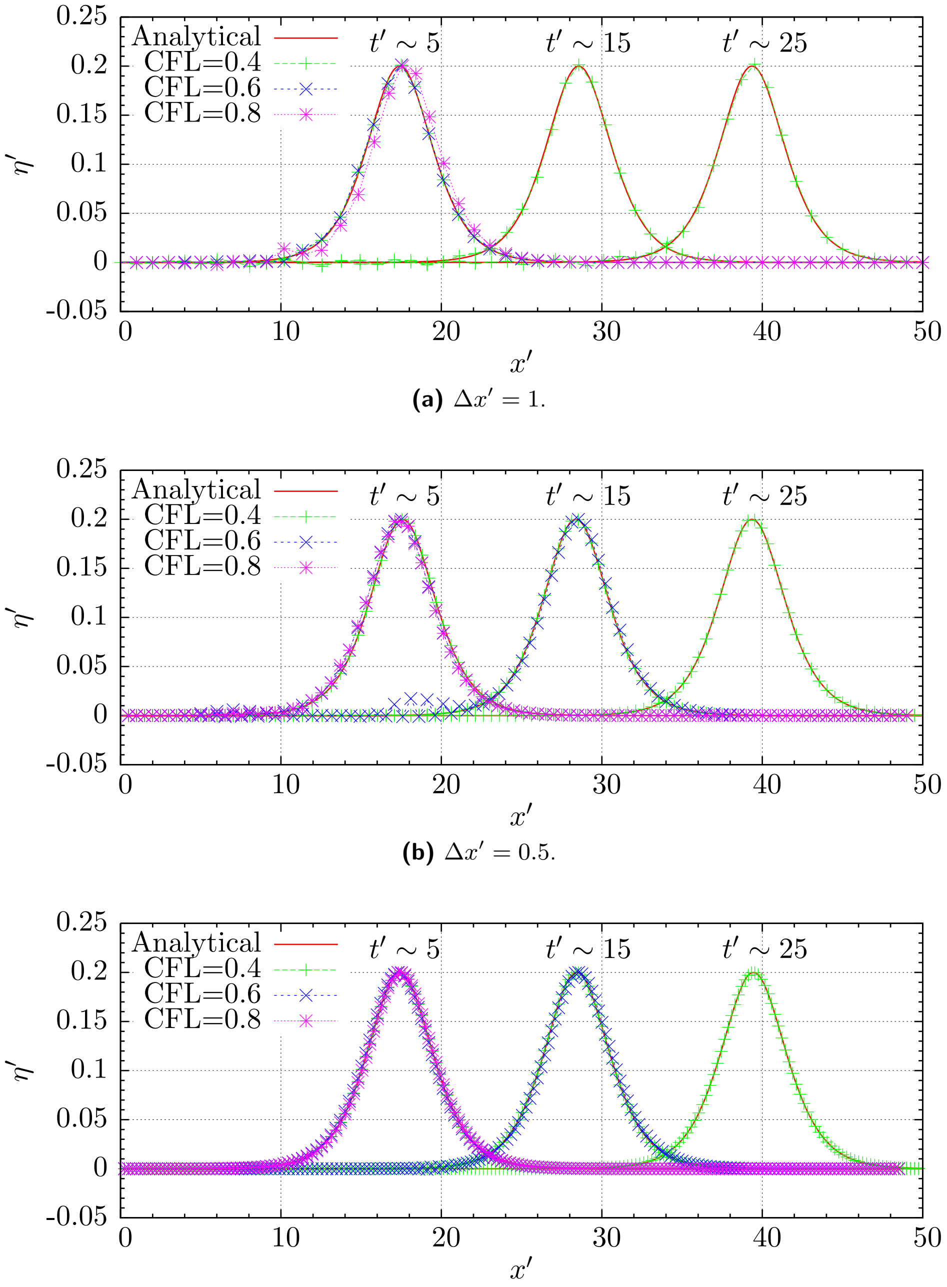


(a) $\Delta x' = 1$.

(b) $\Delta x' = 0.5$.

(c) $\Delta x' = 0.25$.

Figure 5.10: Wave profiles for soliton with wave height, $H' = 0.15$.



(a) $\Delta x' = 1$.

(b) $\Delta x' = 0.5$.

(c) $\Delta x' = 0.25$.

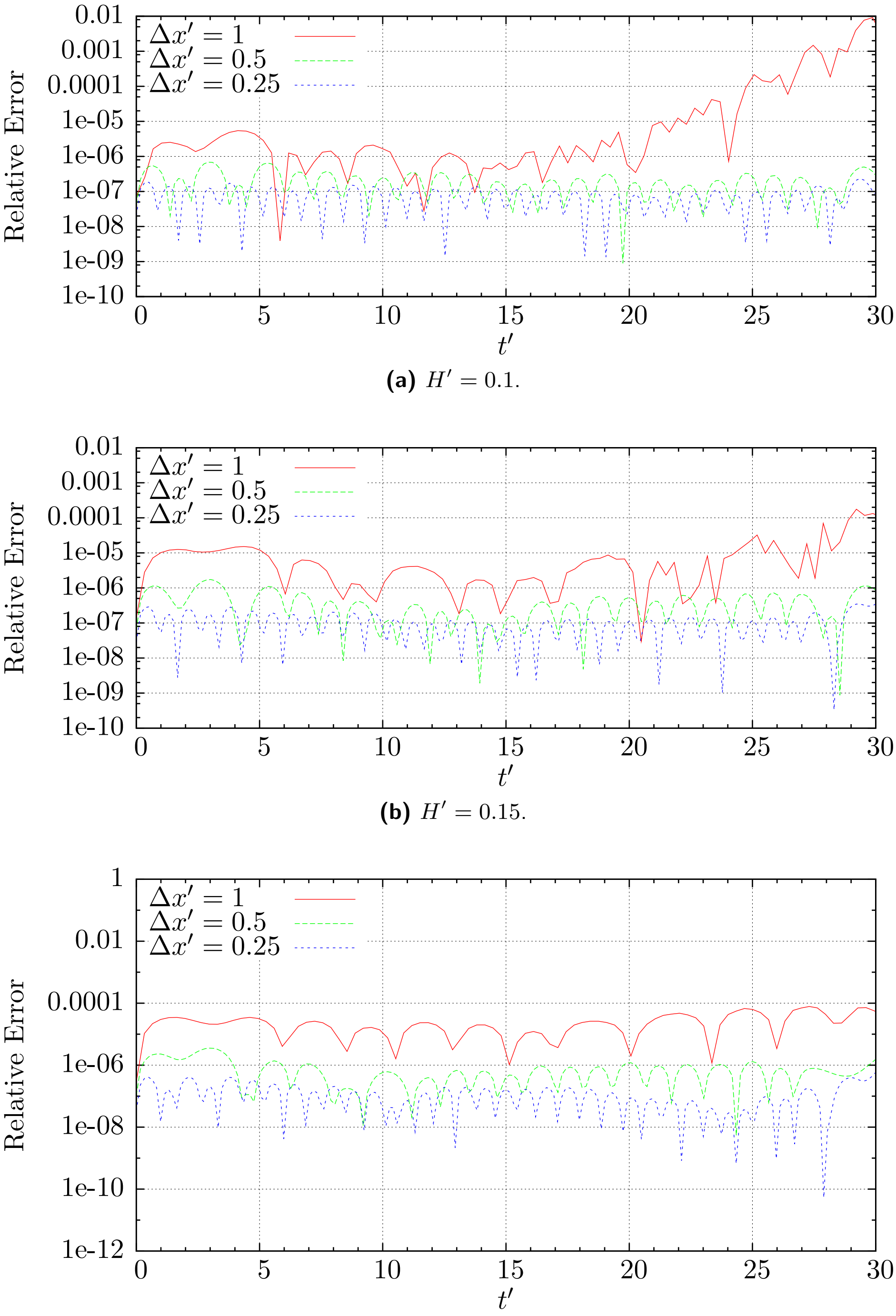
Figure 5.11: Wave profiles for soliton with wave height, $H' = 0.2$.

observed between each wave height when the discretisation is increased from $\Delta x' = 1$ to $\Delta x' = 0.5$. In contrast, the difference for $\Delta x' = 0.25$ is only significant at the highest wave amplitude. The results indicate that there is consistent stability over the whole simulation and potentially over greater time periods. A noteworthy observation is the non-existence of the apparently inherent saw-tooth instability (see Tsai and Yue, 1996). This corroborates the opinions of Broeze *et al.* (1993) and Grilli *et al.* (1989) who stated that the onset of these instabilities are a numerical accuracy problem rather than the onset of physical phenomenon.

To summarise, the progression of solitary waves of varying non-dimensional wave height were simulated with differing node discretisation and CFL number. It was found that the simulations were stable up to $t' = 30$ for CFL number < 0.5, and that the duration of stability for higher CFL numbers increased with wave amplitude. This phenomena may be due to the higher curvature (and therefore concentration of nodes) at the wave crests of the high amplitude waves. All simulations at 0.4 CFL number improved in accuracy as the discretisation was increased from $\Delta x' = 1$ to $\Delta x' = 0.5$. Only for wave height $H' = 0.2$ did the $\Delta x' = 0.25$ discretisation further improve the accuracy of the results. The saw-tooth Helmholtz instability was not encountered in any of the 0.4 CFL number simulations.

5.4.3 Counterpropagating Solitons

The counterpropagating soliton case consists of two solitons, of equal size or otherwise, travelling in opposing directions, which interact and then separate from each other. The extraordinary property of solitons that undergo such interactions is that their form, amplitude and celerity are *substantially* unchanged from their initial state prior to the collision. Contrary to the results derived from the Korteweg-de Vries (KdV) equation (see Acheson, 1990, page 108), the colliding solitons are not totally unaffected by the interaction. As detailed in Craig *et al.* (2006), interaction between solitons governed by the Euler equations (and, thus, potential flow) cause the post collision waves to exhibit a small loss in amplitude and celerity. The total energy of the two waves must be conserved, however, and this is transferred to small residual waves, trailing the resulting solitons and which in time will separate from them. Craig *et al.* successfully quantified the interactions of counterpropagating solitons in terms of their amplitudes after collision, phase lag, run-up and wall residence time. These last two

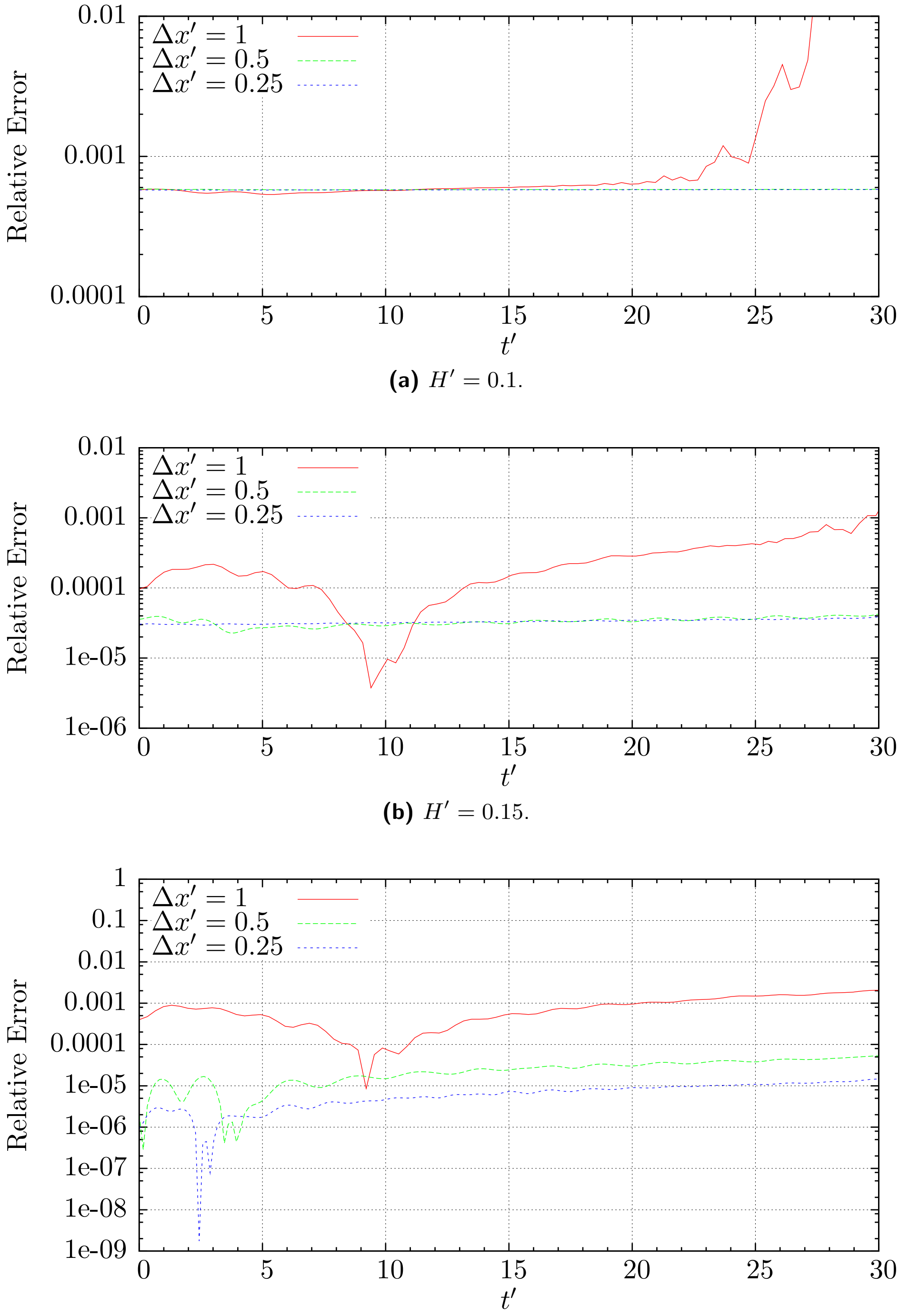


(a) $H' = 0.1$.

(b) $H' = 0.15$.

(c) $H' = 0.2$.

Figure 5.12: Volume conservation errors for solitons at 0.4 CFL number.



(a) $H' = 0.1$.

(b) $H' = 0.15$.

(c) $H' = 0.2$.

Figure 5.13: Total energy errors for solitons at 0.4 CFL number.

quantities will be defined later.

For the case of two identical counterpropagating solitary waves, the wave crests have displacements given by

$$x = c_1 t + a_1, \qquad\qquad x = c_2 t + a_2, \tag{5.12}$$

prior to the collision where the solution is asymptotic as $t \to -\infty$. Some time after the collision, the waves will once again exhibit an asymptotic solution as $t \to \infty$ where the progress of the crests is given by

$$x = c_1^+ t + a_1^+, \qquad\qquad x = c_2^+ t + a_2^+. \tag{5.13}$$

To define the point of intersection of the two waves, it is important to avoid the ambiguity arising from altered celerities before and after the collision. Craig *et al.* (2006) chose the midpoint of the wave intersection in time as the correct moment to measure the phase shift. If this time is given by τ then the phase shift is defined as

$$(a_j^+ - a_j) + \tau(c_j^+ - c_j) \qquad\qquad j = 1, 2. \tag{5.14}$$

The time τ is simple to define for colliding counterpropagating solitons with equal wave heights, yet is less obvious when the waves are of different heights. Equation (2.1) from Craig *et al.* (2006) can be applied to determine τ in such cases; for simplicity, only waves of equal height are attempted here.

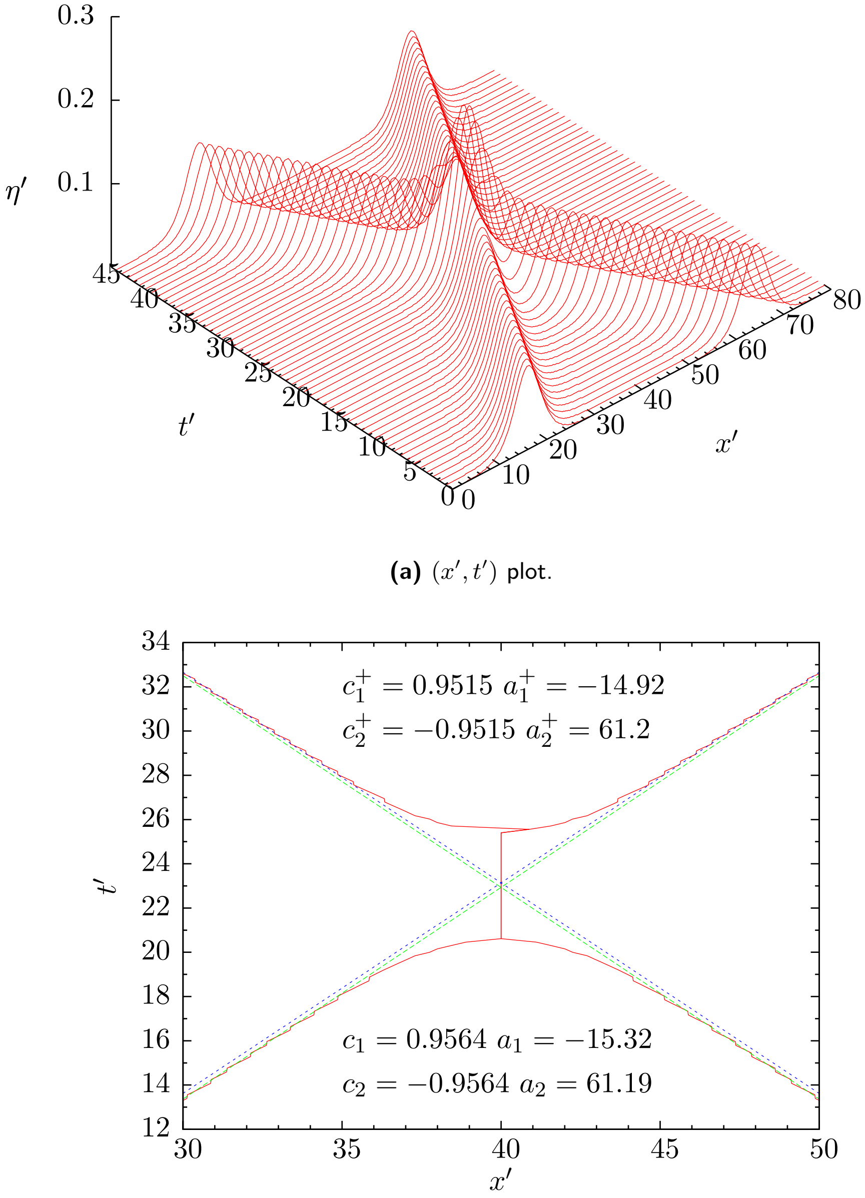
The *run-up* of colliding solitary waves is defined as the maximum amplitude of the superimposed wave. This is expected to extend beyond twice the amplitude of the separated waves. Also, the *wall residence time* is simply the period of time for which the two soliton crests are indistinct during the collision.

All of these quantities are compared, using the present numerical method, to two similar tests carried out in Craig *et al.* (2006) for solitons of height $H' = 0.1$ and $H' = 0.4$. Craig *et al.* went on to formally quantify the residual waves from these collisions but this effort is omitted here. One concern is that the resolution of the present simulation would not be sufficient to accurately recreate the residual waves, which are much smaller in amplitude and wavelength. This is also a concern in tracking the wave crests, as a computational node may not occupy the exact crest position for each time-step. Finally, the duration of the present experiments are significantly shorter than those in Craig *et al.* (2006) due to time constraints. Each wave is allowed to travel for a period and a half before the simulation is terminated. As a consequence of the single soliton test cases, the simulations were run in dynamic mode with a CFL number of 0.4 and the node spacing was set to $\Delta x' = 0.25$.

Results for the collision between counterpropagating solitons of wave height $H' = 0.1$ are given in figure 5.14. Figure 5.14a depicts the evolution in time of a 2D slice in the x-direction. The interaction and subsequent resultant waves can clearly be seen, although at this wave height, the residual waves were not apparent. Figure 5.14b shows the trajectories of the two wave crests before and after the intersection. Also drawn on this figure are the asymptotes that describe the long term evolution of the post and pre collision solitons. These were estimated by a least squares fit of a straight line to the extreme data points. From this graph the wall residence time can be seen to be $t' = 4.8$ which is comparable to figures 2(b) and 6 in Craig *et al.* (2006). The time at the midpoint of the intersection is $\tau' = 23$ and, using the values of c_j, c_j^+, a_j, a_j^+ determined by the least squares fits, the non-dimensional phase shift was found to be 0.2873 for the first wave and 0.1227 for the second wave. The result for the second wave is nearby to the result of 0.1370 reported in Craig *et al.* (2006) figure 2, whilst the result for the first wave is significantly different. This is probably due to the least squares approximation of the asymptotes and the difficulties in tracking the wave crests. The maximum run-up was 0.2055. In general, because of the small size of this wave, the numerical model struggled to capture some of the small scale features, in particular the residual waves. Increased node resolution would improve this but at obvious computational costs.

Figure 5.15 shows the time evolution plot and crest trajectories for the $H' = 0.4$ counterpropagating soliton collision. In figure 5.15a the residual waves can clearly be made out towards the end of the simulation ($t' > 20$). The crest trajectories in figure 5.15b include asymptotes estimated in a similar fashion to the $H' = 0.1$ waves. From this diagram the wall residence time was found to be $t' = 2.89$. The mid point of the intersection is $\tau' = 11.9$. The phase shift of the first wave was found to be 0.68035 and for the second it was 0.60165. This is considerably higher than the result in Craig *et al.* (2006) figure 3 that quotes a value of 0.3257. The maximum run-up was 9.25, which compares well with the results in Craig *et al.* (2006) figure 5.

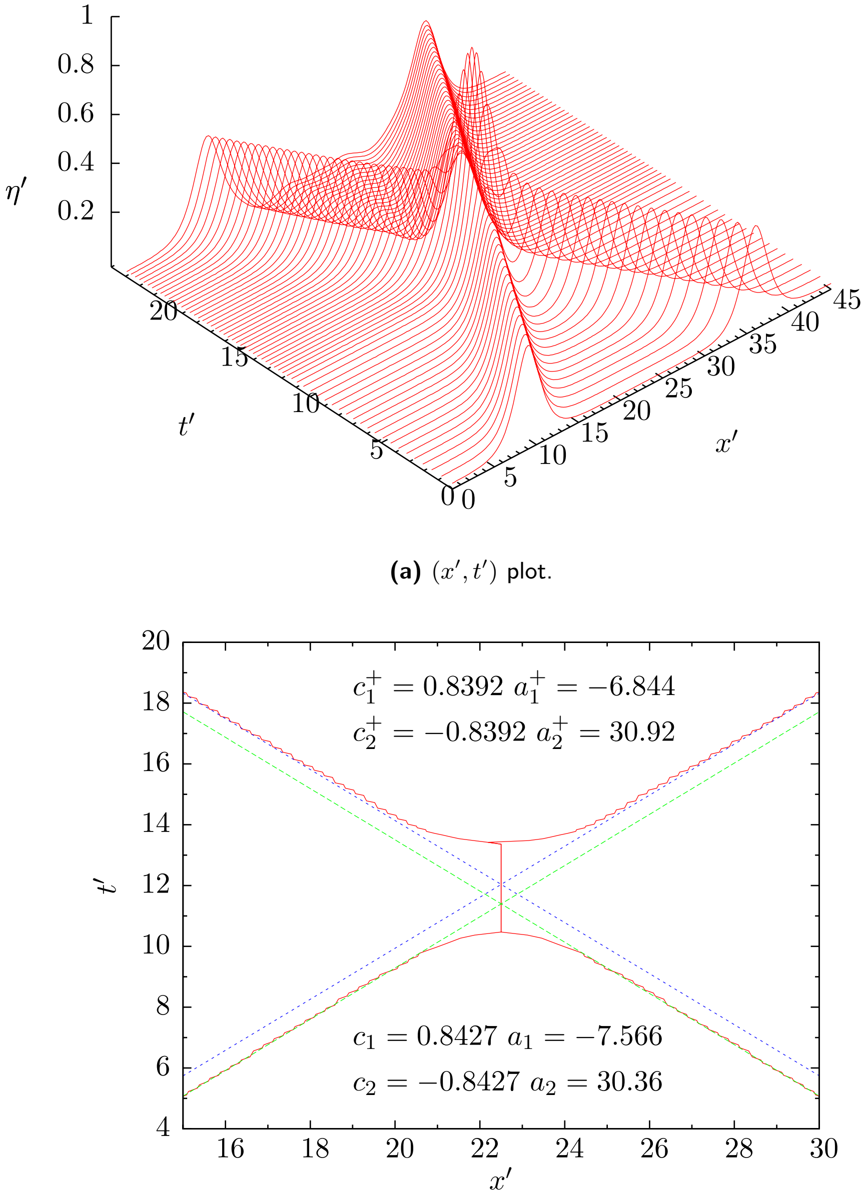
Regarding the error in the phase shift, the same reasons given for the $H' = 0.1$ case



(a) (x', t') plot.

(b) Crest trajectories and asymptotes.

Figure 5.14: Collision of two counterpropagating solitons with height $H' = 0.1$.



(a) (x', t') plot.

(b) Crest trajectories and asymptotes.

Figure 5.15: Counterpropagating solitons with height $H' = 0.4$.

are still valid for this test case, but additional factors must be taken into consideration. Figure 5.16 shows the amplitude of the first wave before, during and after the collision. The figure is comparable to figure 4 in Craig *et al.* (2006) and shows very similar attachment and detachment times for the wave crests. The important point to note is that in figure 4 of Craig *et al.* (2006) the resulting wave takes an additional 60 units of time to recover its final wave height. As can be seen from figure 5.16, the present simulation stops just ten units of time after the intersection of the waves. The wave height deficit at this time is very similar to that of figure 4 of Craig *et al.* (2006) and thus implies that the wave crest has yet to reach its asymptotic trajectory. This could be a key factor in the high error in the phase shifts found for this case. As can be seen in figure 5.17, the residual waves are still attached to the resulting solitary waves. It is anticipated that until these waves detach, the solitary wave will not recover its asymptotic form. Further work could investigate whether the present model exhibits the same behaviour as the waves modelled by Craig *et al.* for the extended period beyond the wave intersection.

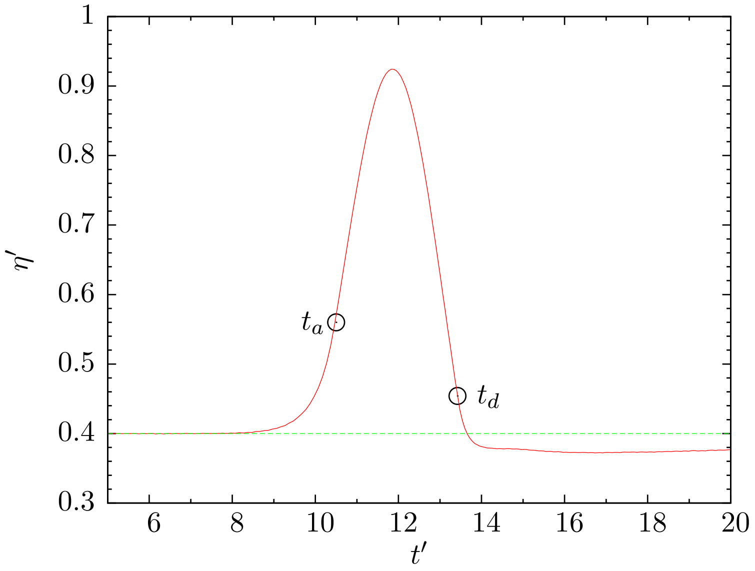


Figure 5.16: Time evolution of the amplitude of soliton in counterpropagating collision with height $H' = 0.4$. t_a and t_d are the attachment and detachment times respectively.

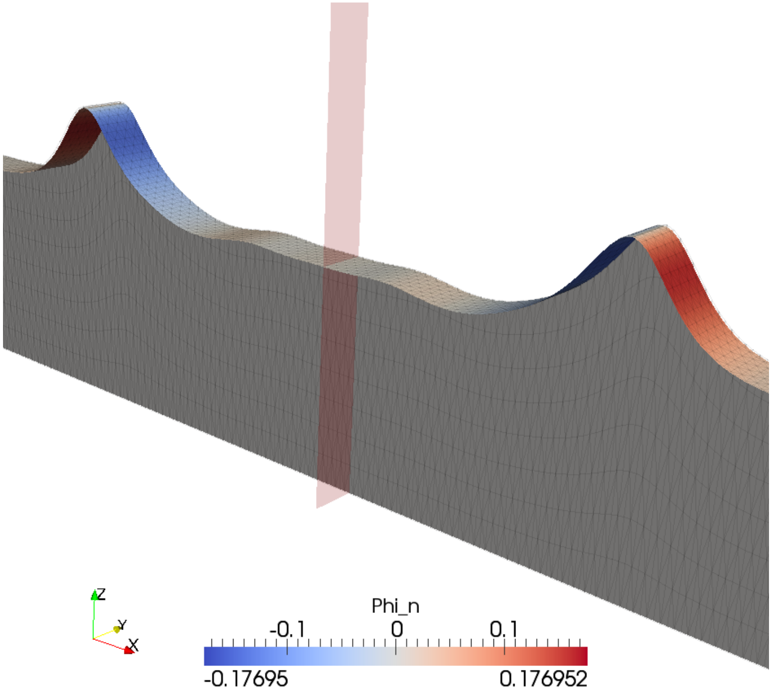


Figure 5.17: 3D representation of the final time-step for counterpropagating solitons with height $H' = 0.4$. The z-coordinate scale is multiplied by a factor of 10. The red plain marks the intersection point of the two waves.

Chapter 6

Results: Lifting

6.1 Introduction

The purpose of the following chapter is to illustrate the application of the Galerkin gradient method to a lifting body. As the focus of the chapter is evaluating the performance of the Galerkin gradient method, the simulation of a hydrofoil in an infinite fluid is isolated from the overarching goal of simulating a hydrofoil under a free surface; without establishing the effectiveness of the Galerkin gradient method, there is no value in attempting to simulate a hydrofoil under a free surface. Note that, a hydrofoil in an infinite fluid, when simulated using the present method is, for all intents and purposes, equivalent to simulating an aerofoil in an infinite fluid. Thus, given its classic nature, this chapter simulates an aerofoil, rather than a hydrofoil.

The aerofoil simulation requires a number of stages in order to create a computational environment which will be compatible with the Galerkin method. Initially it involves the discretisation of a aerofoil profile and the construction of a basic wing with or without end caps. Due to the high curvature at the leading edge of an aerofoil section, a uniform node spacing would not provide a good representation of the shape of aerofoil, thus a curvature dependant panelling scheme based on the work of Forehand (1998) is implemented.

Once the wing geometry is established, and the wing is set at an appropriate angle of attack to oncoming flow, the initial state of the wake vortex sheet must be defined. The spatial Kutta condition is used for this purpose and is an essential part of potential flow models which incorporate lift. The spatial Kutta condition defines a direction which the wake must take as it detaches from the trailing edge of the aerofoil and ensures that the flow leaves the aerofoil smoothly. At time $t = 0$, the wake exists, but its free edge (the bounding line of the wake surface excluding the intersection with the aerofoil trailing edge) is assumed to have zero $[\phi]$. This achieves two goals, the first being a more accurate physical representation of the sheet strength reducing to zero at the transition to open space and, secondly, to allow the Galerkin gradient method to operate on nodes internal to the wake edge from the second time step onwards. On the wake edges $[\phi]$ must be zero for the Galerkin gradient method to be applicable (see section 4.3.4) and, thus, a zero valued wake must exist at $t = 0$.

Finally, a clear algorithm for the time-stepping of the wake must be established. This is a challenging task as the Galerkin method cannot be applied to the nodes on the free edge or to the row of nodes adjacent to the aerofoil trailing edge. In addition, the velocities at those nodes that cannot be solved using the Galerkin method must be estimated using alternative means. This process is made more challenging by the true solution to the hypersingular boundary integral equation at the sheet edge being singular, even with zero μ applied there.

To provide an initial estimate of velocities at the wake edges, a basic off-surface average of the velocity induced by the wake is calculated. For each node, these velocities are taken at points above and below the wake at a specified distance in the normal direction. The quality of this estimate is discussed as part of the results. This method may also be applied to solving the velocity for the majority of the wake sheet. The trailing edge nodes must have the spatial Kutta condition applied there at the beginning of each time-step. During the intermediate stages of a time-step, the spatial Kutta condition is also applied to the nodes adjacent to the trailing edge (mainly to ensure the spatial Kutta condition is enforced throughout the time-stepping process). The velocity for the wake nodes adjacent to the trailing edge, at the beginning of each time step, must be found using the off-surface estimate as the Galerkin gradient method is not applicable here. For the second time-step and beyond, the Galerkin gradient method can be applied to valid nodes using the calculated edge and Kutta strip velocities as a 'seed' for the method, or the off-surface method can be used throughout.

To the best of the author's knowledge, this is the first attempt to apply the Galerkin gradient method of Gray *et al.* (2004b) to a vortex sheet in a quasi-potential flow. The validity of the method is discussed in response to the preliminary results obtained here. The stability of the wake sheet and the interaction of the internal nodes with those at the sheet edges is also discussed along with recommendations for improving this interaction.

6.2 Aerofoil Discretisation

To begin the process of verifying the application of the Galerkin gradient method to a wake vortex sheet, an aerofoil must be discretised and placed at an angle of attack to the flow direction within the fluid. The discretisation of that aerofoil is now discussed. The National Advisory Committee for Aeronautics (NACA) aerofoil profiles describe a wide range of commonly used aerofoil shapes. The primary reference for details of the NACA aerofoil profiles, along with extensive tables of lift and drag coefficients for various angles of attack, can be found in Abbott and von Doenhoff (1959).

Particular classes of NACA profile can be reconstructed by formula and a number of computer codes have been developed to provide ordinates for such profiles (see Ladson *et al.*, 1996). A particular program for providing the ordinates for NACA 4 and 5 digit profiles was developed by Mason (2007) and the code is available on-line. Mason's code is applied here to generate a large set of guide points which are used to position the final, more sparse, mesh points. The particular profile chosen for this simulation is a NACA 23015, which is used for a high tip speed ratio wind turbine design in Le Gourieres (1982), although it is applied to a straight wing section here. Obviously, any NACA 4 or 5 digit profile aerofoil can be described by this method and many others can be specified with other publicly available software.

With a complete set of ordinates to describe the wing, the number of nodes required for a chosen number of panels must be distributed about the profile. Due to the strong contrast in curvature between the mid-section and the leading edge of the aerofoil, a regular node spacing would not describe the surface well. Thus, a method which could place more nodes at areas of high curvature would be highly desirable. Such a method was used by (Forehand, 1998, pg. 74) as part of his free surface regridding routines and was based on an earlier method described in Tsai and Yue (1993). The same principles can be applied to the discretisation of the aerofoil.

The main premise of Forehand's method is to evenly distribute the integral of the curvature along the surface. This integral is given by

$$I(s) = \int_{s_1=0}^{s} \kappa(t)\, dt \tag{6.1}$$

where s is the arc length, κ is the curvature and t a dummy variable of integration. The total sum over the entire surface of the wing cross-section, $I(s_n)$ (where n is the number of guide points), can then be split into m equal steps, denoted by I_{step}. The arc length and curvature along the surface at the guide points must be estimated in order to calculate $I(s_n)$ and this can be achieved through various means. The approach taken for this project was to apply the method of Lewiner *et al.* (2004) to estimate both arc length and curvature at the guide points.

During the calculation of $I(s_n)$, an array containing each $I(s_i)$ for the points in the guide surface can be populated. Once $I(s_n)$ has been calculated and the desired I_{step} found by dividing $I(s_n)$ with the desired number of nodes, the guide node with the nearest $I(s)$ for each I_{step} is chosen to be the computational node. In fact, although an interpolation scheme would provide a better match to the values of I_{step}, by allowing the choice of points which lie between the existing guide points, this was unnecessary as the number of guide points used was very high in comparison to the number of computational nodes.

It was found that the method above worked well, however, it was also found to favour the areas of high curvature too much. In order to apply more nodes to areas of low curvature $I(s)$ was modified so that

$$I(s)_{\text{new}} = \left(\frac{I(s)}{I(s_n)}\right)^{\alpha} \tag{6.2}$$

and now I_{step} is given by

$$I_{\text{step}} = \frac{I(s)_{\text{new}}}{m}. \tag{6.3}$$

The value of α is arbitrary although, to increase the number of nodes in the low curvature regions, it should be chosen such that $\alpha < 1$. A value of $\alpha = 0.5$ was found to produce an adequate spacing and an example of a wing section discretised using this method can be seen in figure 6.1. Note the small chordwise node spacing at the leading edge.

The wing section seen in figure 6.1 is open at both ends and although this is

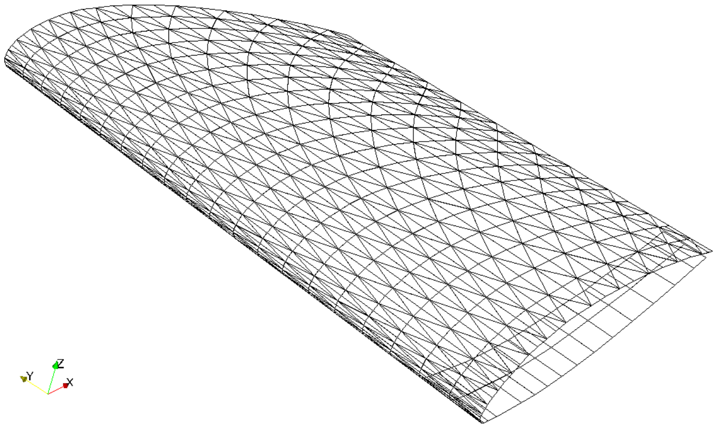


Figure 6.1: Curvature based node spacing for an aerofoil section.

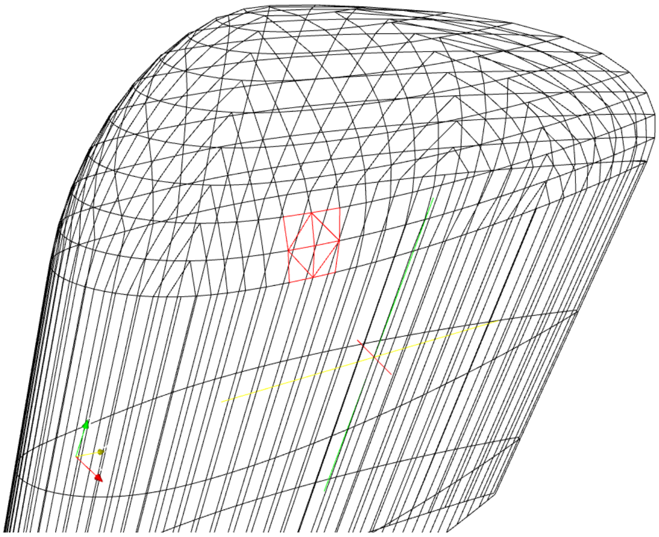


Figure 6.2: Wing section with cap produced with the diamond panel discretisation scheme. A single nine node panel on the cap is highlighted.

reasonable for verification of the numerical model, in reality the wing would be closed. The discretisation schemes developed in chapter 5 have the capability to cap the aerofoil section, specifically by applying the diamond capping scheme that was developed in section 5.2.1 to discretise the sphere. Instead of building the cap sections using circles, the aerofoil profile at the end of the wing is gradually reduced in size to converge on a point directly above the centre of mass of the aerofoil section. The radius of each section is chosen to be the minimum distance from the centre of the aerofoil section to its sides. This radius can then be reduced in an elliptical arc towards the final point at the tip of the wing, where the minor radius is chosen to be the radius of the original section and the major radius is some multiple of that value.

The results of this process are shown in figure 6.2. One of the nine node panels is highlighted in red. The number of panels on the end cap is determined by the discretisation of the straight aerofoil section. The number of panels on the aerofoil section must, in turn, be divisible by eight and greater than or equal to sixteen. This restriction means that the end caps are neglected in the low-discretisation test cases presented later in this chapter.

Note that the wing geometry used to gather the results of this chapter is ‘doubly connected’. As such, the geometry by itself is not sufficient to guarantee a unique solution to the Laplace equation within a constant; as Katz and Plotkin put it, the solution can not be found “on purely mathematical grounds” and must be “determined on the basis of physical assertions”. This problem has been previously discussed in relation to the uniqueness of lifting flows in section 2.4.3. There, it was proposed that the Kutta condition provides the physical relation required to make the solution unique. As the Kutta condition is applied for each time step of the simulations in this chapter, and as the results seem ‘reasonable’, it is concluded that the Kutta condition alleviates the uniqueness issue of the open ended geometry seen in this chapter, in a similar manner to those of a two dimensional aerofoil. Obviously, an open ended wing is not realistic, but its purpose here is to provide values to test the effectiveness of the Galerkin gradient method on the wake, not as a validation experiment. For evaluating the Galerkin gradient method, the open-ended geometry performs admirably.

6.3 Wake Generation Algorithm

This section describes the algorithm for generating and advancing the wake in time. This is a multi-stage process beginning at the trailing edge of the simulated aerofoil. These stages shall now be briefly summarised before being discussed in detail in the subsequent subsections.

Stage One: Spatial Kutta Condition

The first task is to calculate the initial state of the wake sheet; the strength of this sheet is governed by the numerical Kutta condition as described in section 3.5.2. The direction at which the Kutta strip leaves the trailing edge is also critical to enforcing the Kutta condition and this is determined analytically. The velocity vector at the trailing edge of the aerofoil can then be established by calculating the average velocity at the trailing edge and taking its component in the desired direction. This velocity, multiplied by the time-step provides the first line of panels known as the *Kutta strip*.

Stage Two: Off Wake Averaging

The position of the first line of panels leaving the trailing edge is determined without reference to the wake itself. Beyond the Kutta strip, the velocity of all the wake nodes not coincident to the aerofoil trailing edge must be determined by other means. Two processes are used to achieve this in the present method. The first of these processes is always applied at the edges of the wake sheet. This is because the Galerkin gradient method introduced in chapter 4 will not produce a sensible answer here. To overcome this a simple off-surface averaging of velocities is used to estimate the velocity of the wake sheet at the edges. In addition, as the value of μ on the trailing edge is not zero this method must also always be applied at the line of nodes adjacent to the trailing edge. Therefore, it is necessary to apply this process independently for all the nodes not on the trailing edge during the first two time-steps.

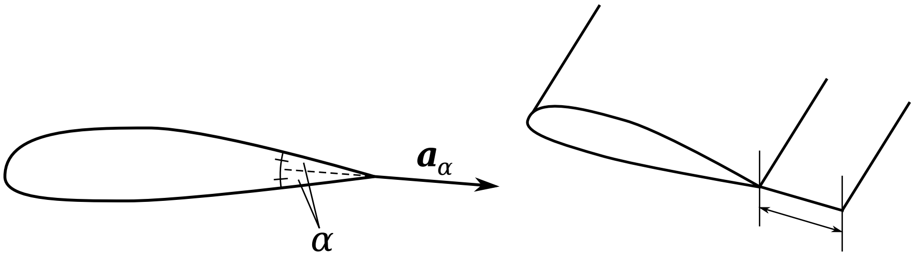
Stage Three: Galerkin Gradient Method

Once the third time-step is reached, the Galerkin gradient method developed in chapter 4 can be applied to the nodes between the edges of the wake and the line of nodes adjacent to the trailing edge. The method is supplied with the velocity calculated at the edge nodes by the averaging algorithm in order to seed the Galerkin algorithm.

6.3.1 Spatial Kutta Condition

Although the present method is essentially a free wake method (i.e. one that allows the wake to move under its own influence with no pre-assumed structure), during its generation, where the wake leaves the trailing edge, the time dependant evolution is restricted to a particular motion. This restriction is part of the Kutta condition as introduced in section 2.4.4. The aim of this section is to describe how the spatial Kutta condition is implemented in the present method.

As stated in section 2.4.4, for ease of implementation, the first group of panels leaving the trailing edge (known as the *Kutta strip*) leaves at the bisecting angle, α. A unit vector is defined that leaves the trailing edge at this angle and is perpendicular to the local wing section as seen in figure 6.3a. This vector is denoted $\mathbf{a}_\alpha$ and once it is determined, it can be used to calculate the direction and extent of the Kutta strip as illustrated in figure 6.3b. Note that, the following implementation still holds should the Kutta strip be orientated tangent to either the top or bottom surface of the aerofoil.



(a) Direction vector leaving the trailing edge at the bisecting angle.

(b) The Kutta strip formed along the trailing edge vector.

Figure 6.3: Direction and length of the Kutta strip.

Initially, at $t = 0$, a 'starting wake' is appended to the lifting body. The direction of this starting wake is determined by $\mathbf{a}_\alpha$; all that remains is to calculate its length. As the velocities on the lifting body are not known at this stage, the far field velocity, $\mathbf{U}$, is used. By projecting the velocity onto the vector $\mathbf{a}_\alpha$, the direction and length of the Kutta strip is given by

$$\left(\left(\mathbf{U} \cdot \mathbf{a}_\alpha\right) \Delta t\right) \mathbf{a}_\alpha. \tag{6.4}$$

For time $t > 0$, the Kutta strip length must be determined using the flow velocity at the trailing edge. In order to calculate this, a non-central difference numerical differentiation scheme can be used to differentiate ϕ on the top and bottom surfaces which gives velocities at the trailing edge, denoted by $\mathbf{v}_{\text{top}}$ and $\mathbf{v}_{\text{bottom}}$. These are then averaged to give the velocity at the trailing edge,

$$\mathbf{v}_{\text{TE}} = \frac{1}{2}(\mathbf{v}_{\text{top}} + \mathbf{v}_{\text{bottom}}). \tag{6.5}$$

This average velocity is then projected onto $\mathbf{a}_\alpha$, which gives the length and direction of the Kutta strip for $t > 0$ as

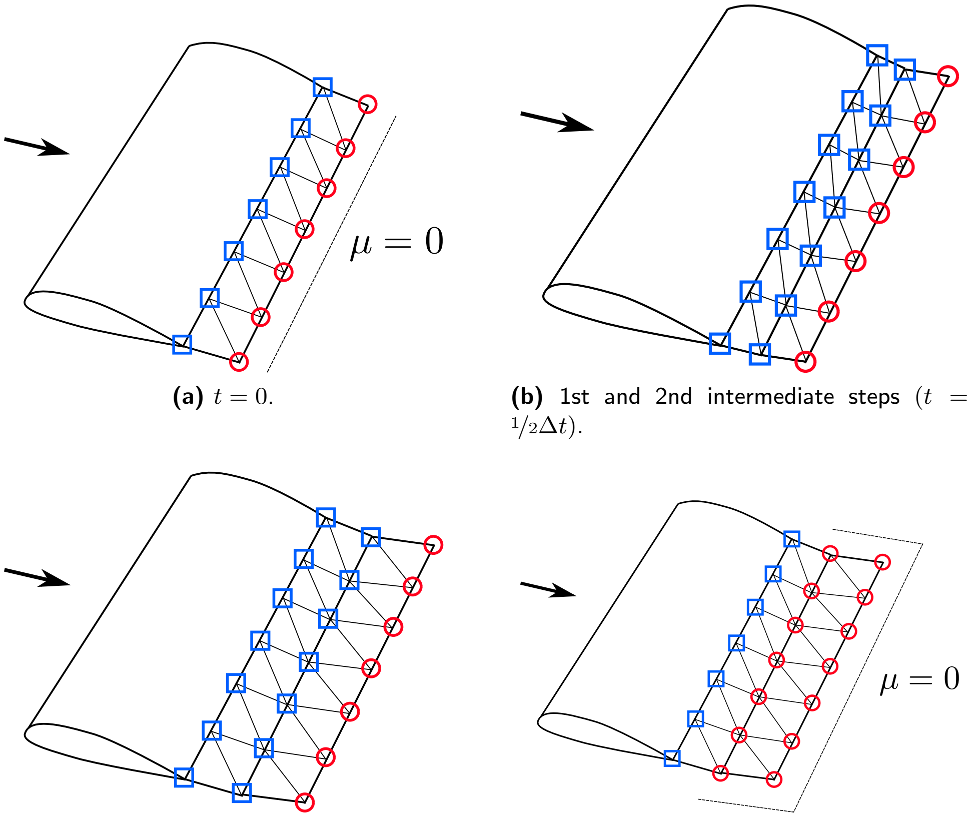
$$\left(\left(\mathbf{v}_{\text{TE}} \cdot \mathbf{a}_\alpha\right) \Delta t\right) \mathbf{a}_\alpha. \tag{6.6}$$

6.3.2 Off Wake Averaging

With the spatial Kutta condition defined, the $t = 0$ position of the wake (see figure 6.4a) is known and subsequent positions of the Kutta strip easily calculated. The next stage is to determine the wake velocities for the nodes not coincident to the trailing edge for $t > 0$. Unlike the nodes at the trailing edge, the velocity of the remaining nodes must be determined using the influences from all the surfaces in the domain, including the wake itself, determined by equation (2.74). At this stage the Galerkin gradient method is not available as the true solution at the wake edges is in fact singular (see section 4.3). Because of this, an alternative method must be used to calculate the velocity at the wake edges. These nodes are marked by red circles in figure 6.4a.

Due to time constraints, a simple and easily implemented method to calculate the edge velocities was desired and, thus, a simple off-surface average was chosen. Figure 6.5 demonstrates how this method is applied. Using a multiplier, ϵ, to the unit normal vector, $\mathbf{n}$, the points P^+ and P^- are positioned above and below the wake surface node P such that

$$P^\pm = P \pm \epsilon \mathbf{n}. \tag{6.7}$$



(a) $t = 0$.

(b) 1st and 2nd intermediate steps ($t = {}^1\!/_2\Delta t$).

(c) 3rd intermediate step ($t = \Delta t$).

(d) $t = \Delta t$.

Figure 6.4: Advancing the wake geometry for the first time step from $t = 0$ to $t = \Delta t$. Nodes with blue squares use the spatial Kutta condition to calculate velocity and those with red circles use off-surface averaging.

Equation (2.74) is applied at $P^{\pm}$ to gives the velocities at the top and bottom of the sheet, $\mathbf{v}^{+}$ and $\mathbf{v}^{-}$ respectively. These velocities can either be derived using just the velocity induced by the wake or from all of the computational surfaces and the free stream velocity. In the first case the total velocity is given by

$$\mathbf{v}_{\text{av}} = \frac{1}{2}\left(\mathbf{v}^{+} + \mathbf{v}^{-}\right) \tag{6.8}$$

and in the second case

$$\mathbf{v}_{\text{total}} = \mathbf{v}_{\text{av}} + \mathbf{v}_{\text{add}} \tag{6.9}$$

where $\mathbf{v}_{\text{add}}$ is the velocity induced by the wing and the free stream. The average velocity isolated to the influence of the wake shall be referred to as $\mathbf{v}_{\text{av}}$.

With the velocity at the wake edges defined, the time-stepping process can begin.

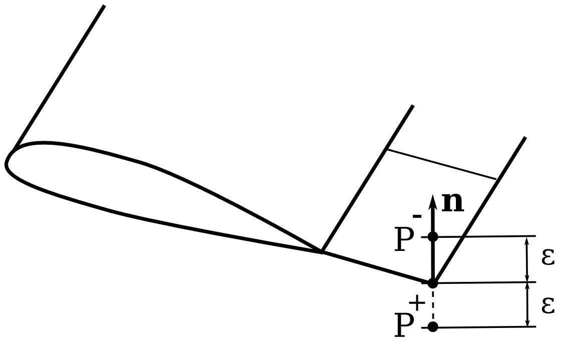


Figure 6.5: Position of points for the off-surface velocity averaging scheme applied at the wake edges.

In contrast to the free surface simulations, the wake can only be solved using the RK4 time-stepping scheme (see section 4.4). The difficulty with applying the alternative AB4 scheme is that new panels in the wake are generated for each time-step. As AB4 requires information about the prior state of the system for three previous time-steps, when new panels are spawned from the trailing edge, this information is not available for all the nodes on those panels. Hence, RK4, which does not require previous information about the wake is applicable here. Nonetheless, the RK4 time-stepping scheme must also be applied carefully in order to preserve the direction of the Kutta strip. Figure 6.4 shows the stages of RK4 for the first time-step. The blue squares show the nodes at which the Kutta strip velocity is applied and the red circles show where the wake averaging described in this section must be applied. During the intermediate stages of the RK4 method a new line of nodes is generated at the trailing edge. It is important that the previous trailing edge nodes (now adjacent to the trailing edge) continue to have the velocity set by the Kutta condition, or the Kutta strip will not be correctly oriented at the end of the step. This is the case for all intermediate stages.

Once the end of the time-step is reached, note that the line of nodes adjacent to the trailing edge, in figure 6.4d, are labelled as edge nodes. This is because the Galerkin gradient method is also incapable of solving for velocity here as the doublet strength of the trailing edge nodes is not zero. Hence, the average velocity method is applied to these nodes also. Once the second time-step is reached, the Galerkin method can be employed.

6.3.3 Galerkin Gradient Method

Beyond the $t = \Delta t$ time-step, the Galerkin gradient method can be utilised. Once again, great care is required to apply the Galerkin gradient method correctly. In particular, there is a need to truncate the domain of the Galerkin equations by feeding the *a priori* values of velocity at the wake edges. As discussed in chapter 4, the Galerkin gradient method cannot be used at the edges as the solution there is infinite. Thus, the velocity at the wake edges must be provided by other means and, for the wing, the velocities for the nodes adjacent to the trailing edge must also be computed separately.

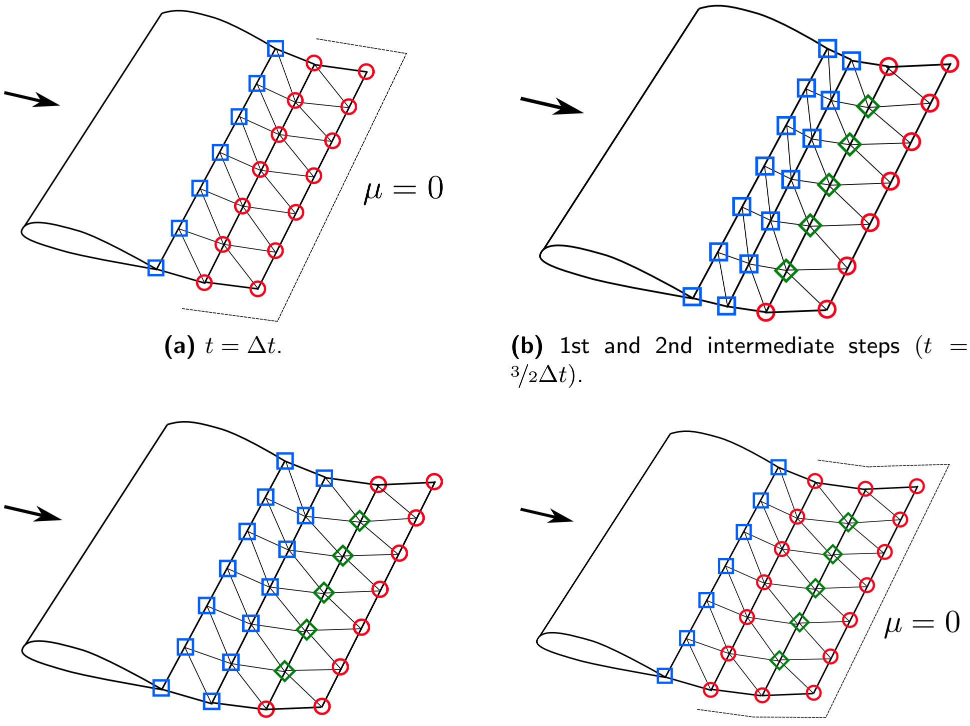
The process for taking the second time-step is illustrated in figure 6.6. The initial stage (figure 6.6a) is identical to the last stage of the first time step. During the first and second intermediate RK4 steps the line of nodes marked by green diamonds in figure 6.6b can be solved using the Galerkin gradient method. The method is only used to solve for the self-induced velocity from the wake. At the edge nodes, the average velocity of just the wake, given by $\mathbf{v}_{\mathrm{av}}$, is fed to the Galerkin method. The velocity supplied by the Kutta nodes is more complicated. Because the Kutta nodes implicitly carry the total trailing edge velocity, the velocity from the free stream and the wing must be removed. The seed velocity from the Kutta nodes is given by

$$\mathbf{v}_{\mathrm{TE\ seed}} = \mathbf{v}_{\mathrm{TE}} - \mathbf{v}_{\mathrm{add}}, \tag{6.10}$$

where, as before, $\mathbf{v}_{\mathrm{add}}$ is the velocity induced by the wing and the free-stream.

The velocities used to seed the Galerkin gradient method must be doubled as the method returns twice the average velocity as the result. Additionally, all values of μ on the wake are supplied to the Galerkin gradient method and, importantly, as stated in section 4.3.3, the method acts on values of $-\mu$, rather than μ. Also, the values of μ at the wake edges, other than the trailing edge, must be zero. With this complete information the Galerkin method will return twice the self-induced velocity at the internal nodes. Post calculation, this velocity is halved and the induced velocity from the wing plus the free stream can be added. This will provide the complete wake velocity at the internal nodes.

The process at the trailing edge nodes for the intermediate steps is similar to those for the first time-step. Care must be taken to preserve the Kutta node velocity during the intermediate stages. From the first intermediate stage of the second time-step, and



(a) $t = \Delta t$.

(b) 1st and 2nd intermediate steps ($t = {}^{3}\!/_{2}\Delta t$).

(c) 3rd intermediate step ($t = 2\Delta t$).

(d) $t = 2\Delta t$.

Figure 6.6: Advancing the wake geometry for the second time step from $t = \Delta t$ to $t = 2\Delta t$. Nodes with green diamonds use the Galerkin gradient method to calculate the surface velocity.

for all remaining time-steps, the Galerkin gradient method can be applied to all wake nodes that do not lie on the wake edge or are coincident with or adjacent to the trailing edge.

6.4 The CFL Condition for the Wake

Prior to beginning the examination of the behaviour of the wake, it is prudent to consider its stability under time-stepping. In order to define a CFL condition for the wake, it is necessary to start by rewriting the CFL condition given by equation (4.109) in terms of the free stream velocity, $\mathbf{U}$. This gives

$$\Delta t = C_n \Delta s \, |\mathbf{U}| = C_n \Delta s \, U. \tag{6.11}$$

The length Δs of the wake at the first time-step is governed by equation (6.4), i.e.

$$\Delta s = (\mathbf{U} \cdot \mathbf{a}_\alpha)\, \Delta t. \tag{6.12}$$

Substituting equation (6.12) into equation (6.11) gives

$$\begin{aligned} & \Delta t = C_n\, (\mathbf{U} \cdot \mathbf{a}_\alpha)\, \Delta t\, U \\ \Rightarrow \qquad & \Delta t\, (1 - C_n\, (\mathbf{U} \cdot \mathbf{a}_\alpha)\, U) = 0. \end{aligned} \tag{6.13}$$

Not wishing Δt to be zero, equation (6.13) implies that

$$1 - C_n\, (\mathbf{U} \cdot \mathbf{a}_\alpha)\, U = 0$$

and thus

$$C_n = \frac{1}{U\,(\mathbf{U} \cdot \mathbf{a}_\alpha)} = \frac{1}{U\,(U(\mathbf{a}_\alpha)_x)} \tag{6.14}$$

where $(\mathbf{a}_\alpha)_x$ is the x-component of the Kutta vector; the free-stream velocity being directed in the positive x-direction. Evidently, the CFL number is controlled only by the speed of the oncoming flow and is proportional to U^{-2}. This result will only alter slightly for $t > 0$ where equation (6.6) is applied at the Kutta strip, resulting in a similar conclusion based on the velocity about the wing. Thus, this form of CFL condition is not applicable to dynamic time-stepping since C_n is fixed. Some alternative metric is required and the most compelling possibility may be to define a condition based on the geometry at the the wake edges where roll-up deforms the surface greatly. However, as yet, the form of such a condition is not clear. The impact of variation of the time-step for the forthcoming numerical experiments will provide an interesting insight.

6.5 Wake Generation Without the Galerkin Gradient Method

This section will explore wake generation without the use of the Galerkin gradient method. Instead, the velocity averaging algorithm, described in section 6.3.2, will be applied to both the edge and internal nodes of the wake. The primary purpose of undertaking this test is to evaluate the effect of varying ϵ in equation (6.7). This is an important factor in the entire wake generation algorithm as the averaging algorithm provides the wake velocity at the edges and for the nodes adjacent to the trailing edge as a requirement to the Galerkin gradient method.

There exists a conflict in the choice of ϵ between taking the average too far from the surface, avoiding the true surface velocity, and approaching too near to the surface, where the solution to equation (2.74) is singular. Time constraints during the project undermined the design of an adaptive scheme for this near-singular problem and thus the averaging is likely to be extremely sensitive here.

This test will provide some insight into the effectiveness and implementation of the numerical Kutta condition developed in section 3.5.2. The angle of attack can also be varied to examine the influence on the resulting form of the wake sheet. Note that, as the main purpose of this section is to investigate wake generation, no integrated properties such as surface pressures are calculated at this stage. It was felt that, until the wake can be shown to be stable for a reasonable period, these values would not be comparable to existing results in the literature, that are generally recorded for steady state.

For qualitative comparison of the wake evolution, figure 7 of Pyo and Kinnas (1997) is reproduced here as figure 6.7. The figure shows the wake behind a lifting line with elliptic loading. Note the wake roll-up located at both the streamwise edges of the sheet and also note that the centre of the wake 'sags'. The solution Pyo and Kinnas (1997) used to calculate the wake velocity is also higher order, however, it was implemented in a different manner to the method presented here. Firstly, Pyo and Kinnas used a wake relaxation scheme rather than the time-stepping scheme used in the present method. This means that the wake in figure 6.7 was 'relaxed' to its final position from a pre-existing flat wake. In general this approach is more stable than time stepping the wake into position, but if the expected position of the wake is not well known, as is the case for tidal turbines, and if an unsteady free surface is also present, the wake relaxation method is not valid. Secondly, the Pyo and Kinnas approach calculated the velocities at panel centres and interpolated to the collocation points rather than calculating the value at the collocation points directly, as attempted within this thesis. Regardless of these differences, figure 6.7 provides an excellent guide for qualitative verification of the present method.

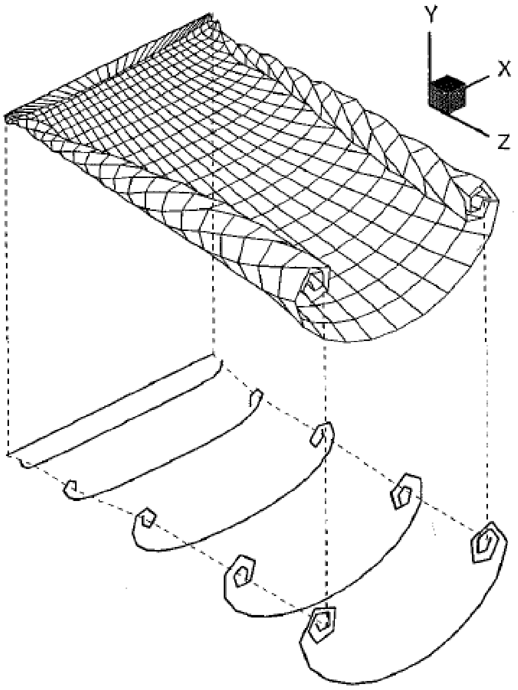


Figure 6.7: Wake roll-up behind a lifting line. Source: Pyo and Kinnas (1997, figure 7).

The foil to be tested has span = 2 and chord = 1. The wing is subject to an oncoming flow of speed 2. Varying time-steps can be used but unless otherwise stated, $\Delta t = 0.1$. The complete flow velocity for the wake is the combination of its self-induced velocity, the far field velocity and the velocity induced by the wing. In order to check if the velocity induced by the wing is feasible, the wake is initially stepped forward in time without including its self-induced velocity. As the initial velocity perturbation generated by the wing on the free stream is quite small, a time-step of $\Delta t = 0.02$ is required. The initial conditions for this simulation can be seen in figure 6.8. Figure 6.9 shows the results of the simulation after a couple of time-steps. The wake can be seen to gently return to the far field flow direction as it gets further from the wing, as expected.

The self-induced velocity of the wake can now be introduced into the simulations.

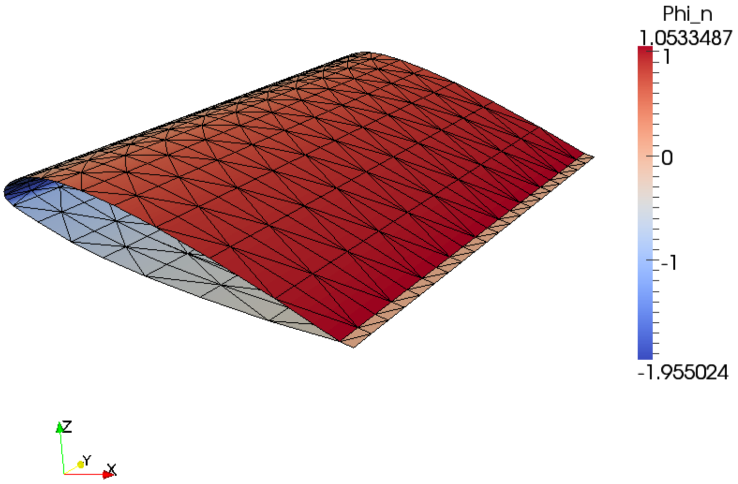


Figure 6.8: Initial ϕ_n, geometry and Kutta strip. $\alpha = 5°$, $t = 0$ & $\Delta t = 0.02$.

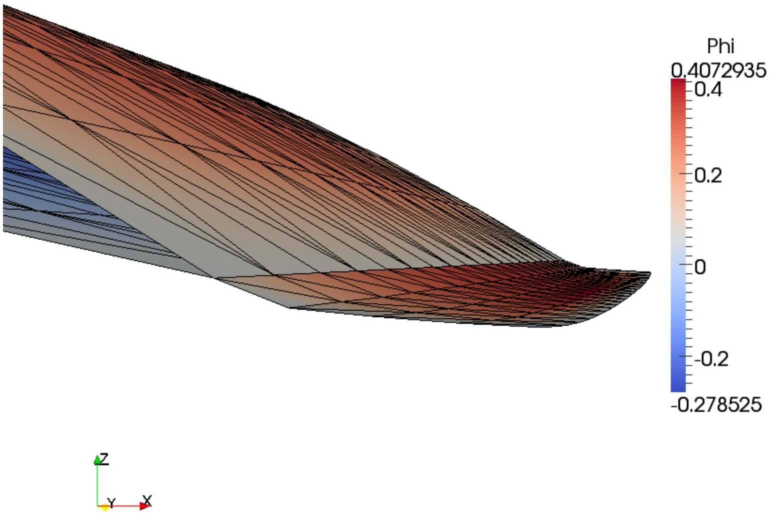


Figure 6.9: Velocity induced without wake influence. $\alpha = 5°$, $t = 0.06$ & $\Delta t = 0.02$.

Three main parameters of interest will be varied in order to gauge their importance. These are:

1. the discretisation of the wake, governed by the number of panels in the spanwise direction of the wing,
2. the choice of ϵ for the wake average velocity algorithm,
3. the angle of attack, α, of the wing.

Regarding the choice of ϵ, it was decided to set a global value which would be used over the entire wake. Similar to the adaptive quadrature metrics defined in chapter 3, ϵ must be normalised by some value. This value is chosen to be the largest edge length in the wake panels modified by a multiplier such that

$$\epsilon = C_\epsilon l_{\text{max}} \tag{6.15}$$

where l_{max} is the longest edge length in the wake and C_ϵ is a user defined constant.

To begin, the effect of discretisation, for a wing with $\alpha = 5°$, $C_\epsilon = 0.5$, was examined using three simulations with varying numbers of spanwise panels. Figure 6.10 shows the wake after $t = 1.1$ for 5 spanwise panels. Figure 6.11 shows the result for 10 spanwise panels and figure 6.12 shows the results for 20 spanwise panels. Each figure shows the wake profile in full and a section through the centre of the wake and aerofoil. A comparison of slices through the wakes taken in the x and y directions is shown in figure 6.13. In figure 6.13, and for all the other comparisons, the dashed black line shown with the x-slices plot indicates the x-coordinate of the y-slices below.

It can be seen from the comparison that the slices in the y-direction clearly shows the initial stages of roll-up of the side edges, as expected. It also reveals that the impact of increased discretisation is quite interesting. A trend can be seen for the wake to rise as the discretisation increases. With increased discretisation, the ‘bowing’ of the centre of the wake in the y-direction is less apparent. This may be due to the impact reduced discretisation has on ϵ. In fact, because the time-step is unchanging, and ϵ is set using the longest panel edge length in the wake, the value of ϵ will be unchanging across the three discretisations. Hence, the relative distance of the averaging points from the panel is increasing with increased discretisation; this may reduce the likelihood that the true velocity is being calculated. It is therefore prudent to examine the effect of varying C_ϵ.

Figure 6.14 and figure 6.15 show the wing at 5° angle of attack and 20 spanwise panels. The simulation seen in figure 6.14 has $C_\epsilon = 0.2$, whilst $C_\epsilon = 0.1$ was used for the simulation in figure 6.15. Figure 6.16 shows the comparison of the slices in the x and y directions along with the final slices from figure 6.12. In terms of the wake's deviation in the z-direction, the impact of reducing C_ϵ has the opposite effect to increasing the discretisation, in that the wake, initially anyway, descends with reducing C_ϵ. However, the trend for the wake to 'flatten' is maintained - as can be seen in the y-direction slice. This may indicate a trend that the wake should, in fact, be more flat than the example seen in figure 6.7. This may be because the loading is more regular than on the elliptically loaded example presented by Pyo and Kinnas.

The final parameter of interest is the angle of attack. Figure 6.17 and figure 6.18 show wings with angle of attack $\alpha = 10°$ and $\alpha = 20°$ respectively. Both simulations were run with $C_\epsilon = 0.5$ and 20 spanwise panels. Figure 6.19 compares the slices of these two simulations with the final slices from figure 6.12. The difference between these simulations is quite significant. As the angle of attack increases, the depth of the wake increases - as would be expected. The amount of roll-up also appears to increase. A noteworthy aspect of the results, concerning these simulations and all the previous simulations, is the lack of movement of the wake edges themselves. Because of this, a further investigation into the behaviour of the wake edges was carried out.

As the wake edges appeared to be so inert in comparison to the rest of the sheet, it was decided to decouple the values of C_ϵ for the edges from the rest of the sheet. Maintaining the previously defined values of C_ϵ for the internal nodes, the value of C_ϵ at the edges was set to zero. Figure 6.20 demonstrates the outcome of this simulation for a 5° angle of attack and 10 spanwise panels. A clear difference with the wake edges can be seen here with a much greater amount of roll-up compared to the previous simulations. The fact that C_ϵ can be chosen to be zero at all raises a number of serious issues. As discussed in chapter 4, the velocity at the wake edges should be infinite. The result seen in figure 6.20 may be a consequence of the numerical scheme allowing the value to be finite at the edges. The combination of the zero value of μ at the edges and the quadrature schemes which never give a zero value of r may be combining to produce this result. It is therefore possible that this is still not the true result at the edges of

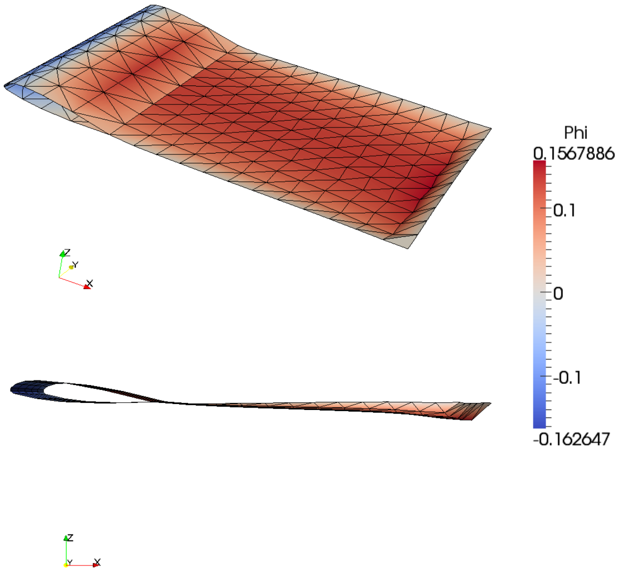


Figure 6.10: Wake profile at $t = 1.1$ for $\alpha = 5^{\circ}$, $C_{\epsilon} = 0.5$ and 5 spanwise panels.

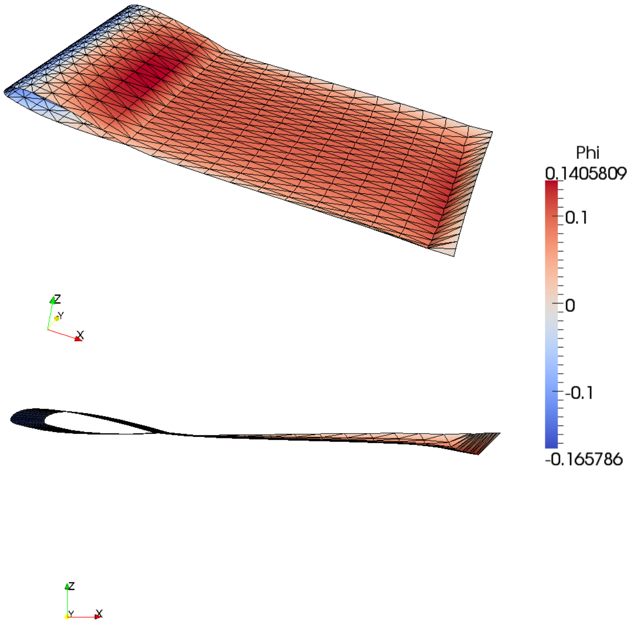


Figure 6.11: Wake profile at $t = 1.1$ for $\alpha = 5^{\circ}$, $C_{\epsilon} = 0.5$ and 10 spanwise panels.

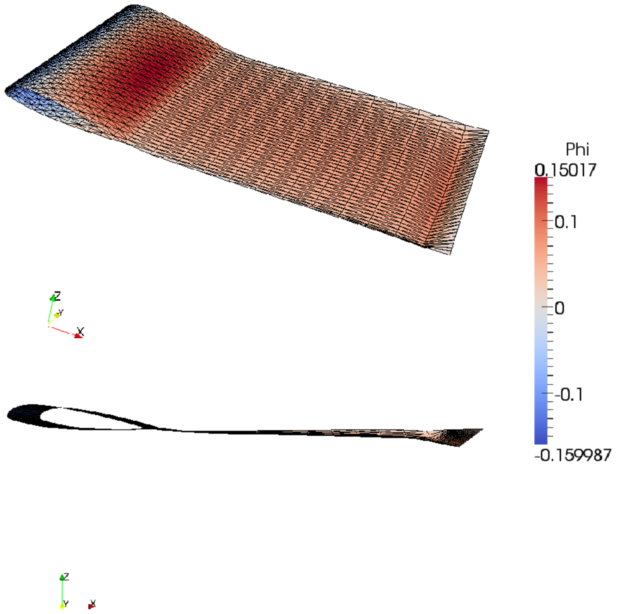


Figure 6.12: Wake profile at $t = 1.1$ for $\alpha = 5°$, $C_\epsilon = 0.5$ and 20 spanwise panels.

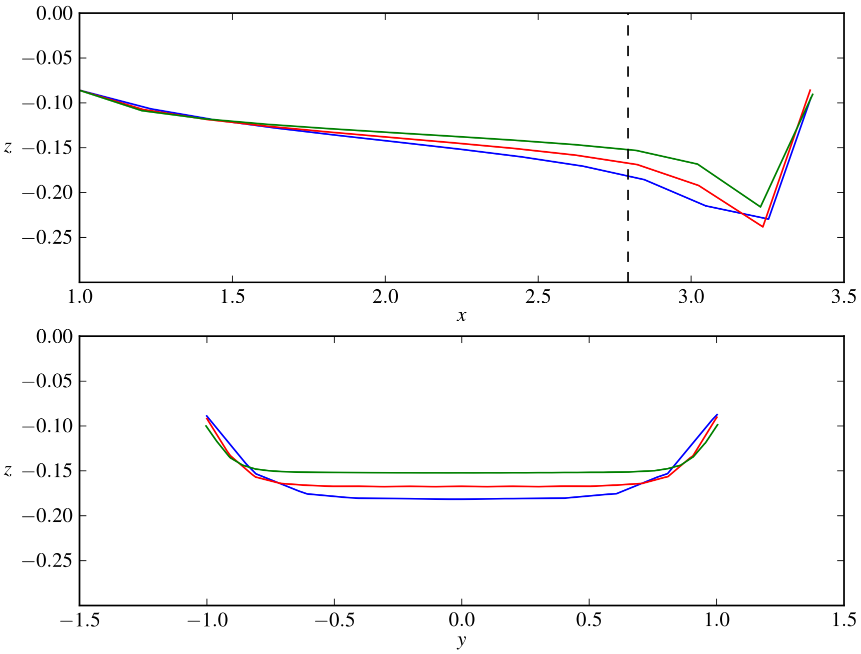


Figure 6.13: Comparison of wakes at varying discretisation for slices in the x and y directions. The blue line shows 5 spanwise panels, the red line 10 and green line shows the results for 20 spanwise panels.

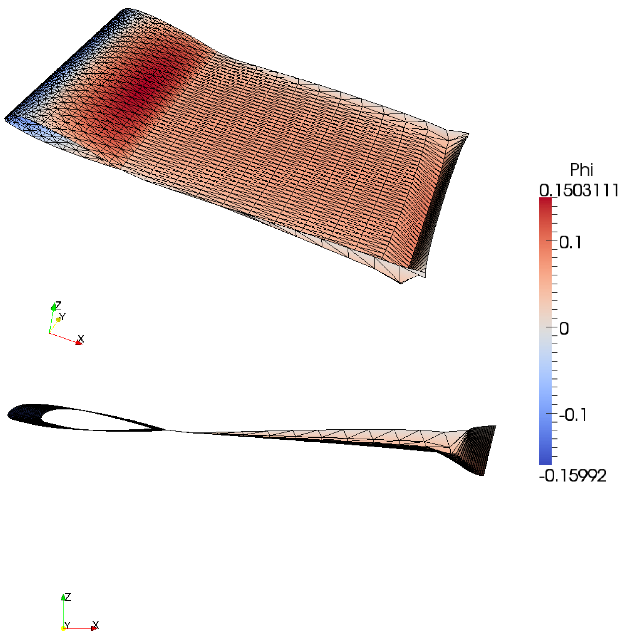


Figure 6.14: Wake profile at $t = 1.1$ for $\alpha = 5°$, $C_\epsilon = 0.2$ and 20 spanwise panels.

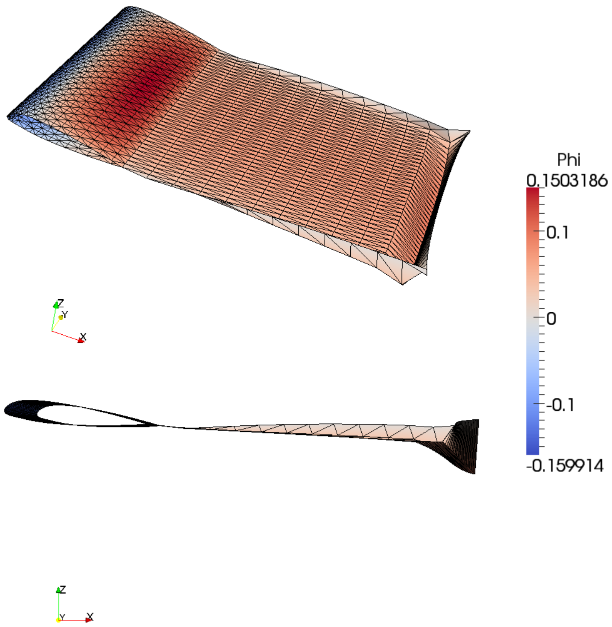


Figure 6.15: Wake profile at $t = 1.1$ for $\alpha = 5°$, $C_\epsilon = 0.1$ and 20 spanwise panels.

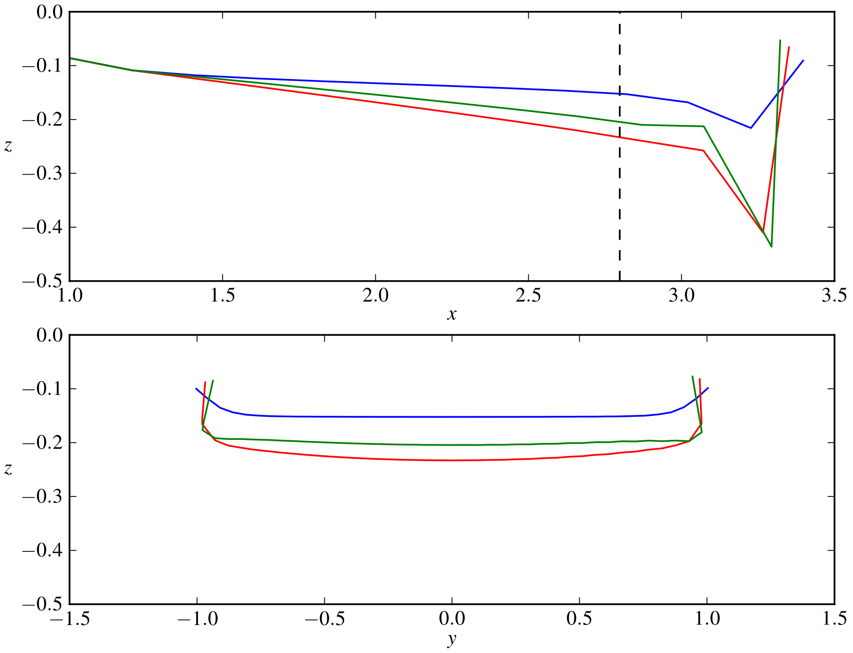


Figure 6.16: Comparison of wakes at varying C_ϵ for slices in the x and y directions. The blue line shows results for $C_\epsilon = 0.5$, the red line $C_\epsilon = 0.2$ and the green line shows results for $C_\epsilon = 0.1$.

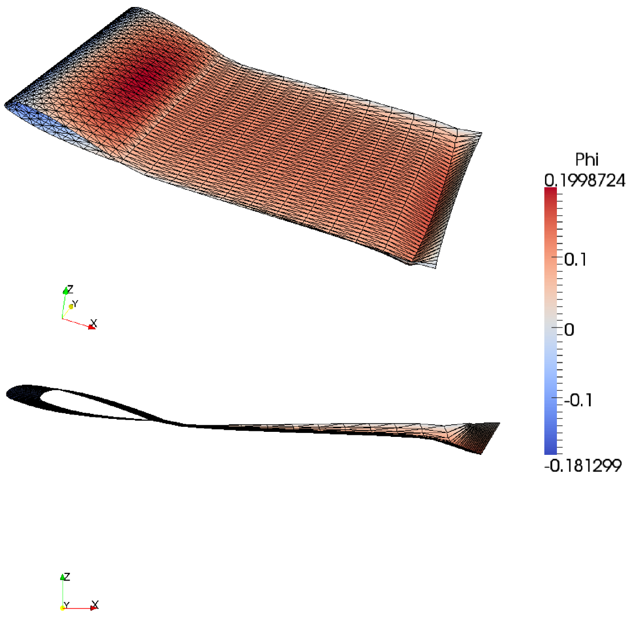


Figure 6.17: Wake profile at $t = 1.1$ for $\alpha = 10°$, $C_\epsilon = 0.5$ and 20 spanwise panels.

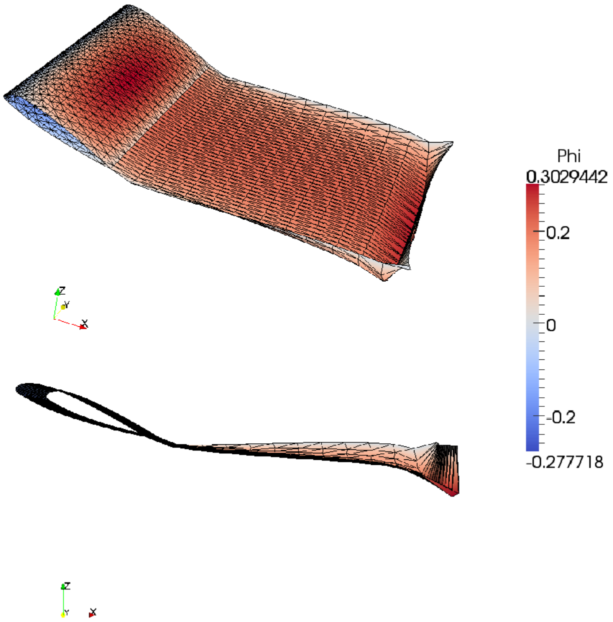


Figure 6.18: Wake profile at $t = 1.1$ for $\alpha = 20^{\circ}$, $C_{\epsilon} = 0.5$ and 20 spanwise panels.

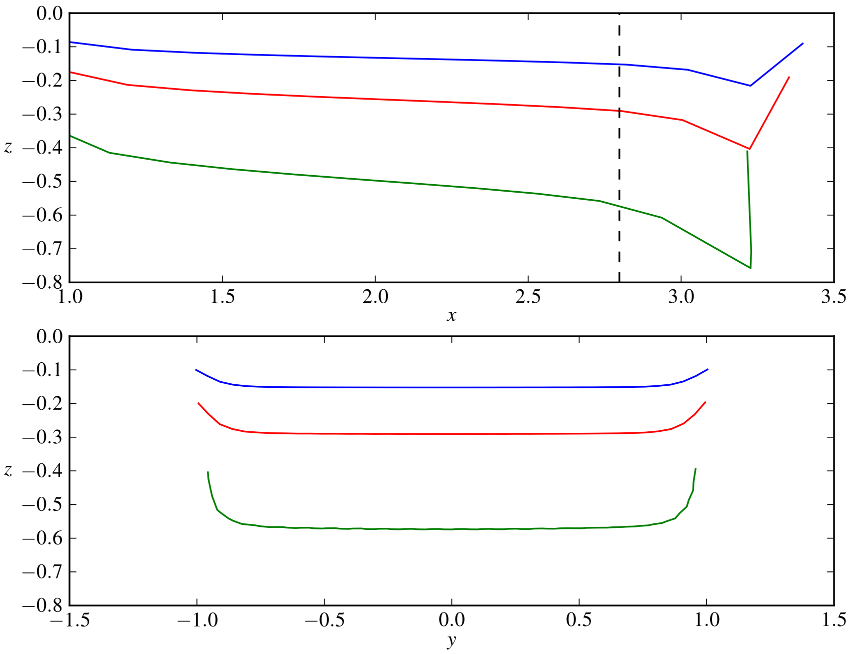


Figure 6.19: Comparison of wakes at varying α for slices in the x and y directions. The blue line shows the results for $\alpha = 5^{\circ}$, the red line $\alpha = 10^{\circ}$ and the green line shows results for $\alpha = 20^{\circ}$.

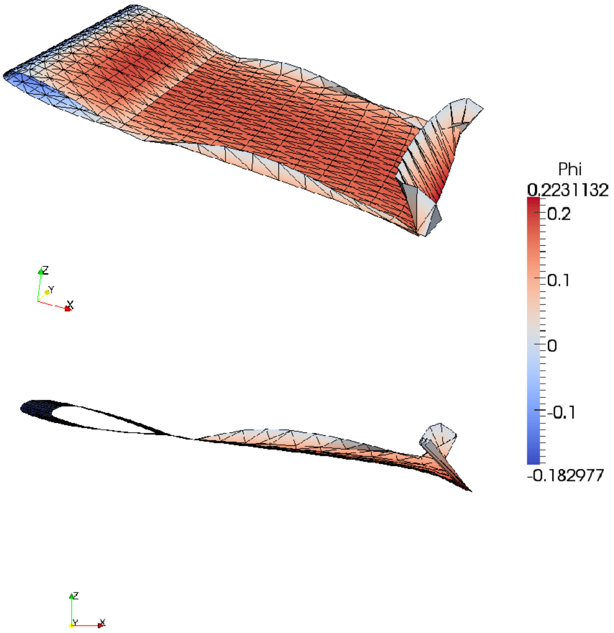


Figure 6.20: Wake profile at $t = 1.1$ for $\alpha = 5°$, $C_\epsilon = 0.5$, 10 spanwise panels and zero ϵ at the free edges.

the wake and merely a numerical artifact. In fact, the uncertainty of the results at the wake edges will have significant consequences on the next section where the Galerkin gradient method is applied.

6.6 Wake Generation With the Galerkin Gradient Method

Using the parameters established for the wake averaging algorithm, the Galerkin gradient method can now be applied to the wake. The key factors to examine with this series of tests are the stability of the algorithm and the qualitative alignment with expected results, such as those from the wake averaging algorithm and the Pyo and Kinnas example shown in figure 6.7. Although the theoretical grounding for the algorithm used here was established in chapter 4, there was insufficient time to program and test the algorithm before its application to this problem. The author is greatly indebted to Professor Leonard Gray of Oak Ridge National Laboratory for supplying the core of the code which was used in the following calculations. The only adjustments made to Professor Gray's code was the introduction of the velocity at the edges of the wake in order to circumvent the singularity there and truncate the linear system.

Similarly to the non-Galerkin wake generation, the value of C_ϵ at the edges and for the nodes adjacent to the trailing edge can be varied along with the discretisation, the angle of attack, etc. The initial set of tests investigate the effect of varying C_ϵ for a wing with $\alpha = 5°$ and 10 spanwise panels using a time-step of $\Delta t = 0.1$. Figure 6.21 compares the results for $C_\epsilon = 0.5, 0.2, 0.1$ and the zero ϵ edge case (with $C_\epsilon = 0.5$ for nodes adjacent to the trailing edge) from a top down view at $t = 0.3$. The interesting point to observe here is that the wake profile is stable, but it appears to be completely unvarying between the different simulations. Figure 6.22 shows slices taken in the x and y directions respectively. From the x-slices, note that for the $C_\epsilon = 0.5, 0.2, 0.1$ cases the Galerkin algorithm does not alter the position of the wake. Nonetheless, for the zero edge case the wake does translate slightly in the z-direction. This is a useful result as it indicates that the Galerkin algorithm does respond to the translation at the edges, which is not apparent from figure 6.21.

The slices taken in the y-direction, as seen in figure 6.22, are also very interesting. In particular, the shape which the wake takes in the y-direction is different to that seen when using the averaging routine. The wake appears to form an unlikely 'W' shape rather than the bow shape seen previously. However, this is only the case for the zero ϵ edge case and, in fact the shape is much flatter when the edge movement is not so extreme.

A serious issue, which is yet to be discussed, is that all of the simulations so far have been illustrated at $t = 0.3$. In fact, beyond this time the simulations fail. This event is illustrated in figure 6.23. In general, the interaction between the edge nodes and the nodes adjacent to the edges is quite unstable and is restricting the amount of time-steps that can be taken. This is a serious flaw with the method and is discussed in greater detail later. Prior to the introduction of methods which can be attempted to circumvent these problems, the response of the method to another angle of attack should be studied. Therefore, an identical study to the one above using $\alpha = 10°$ was undertaken. The x and y-direction slices of the results can be seen in figure 6.24. Once again a lack of stability meant that the results could only be compared at $t = 0.3$. The orange line shows the results from a test using just the wake averaging with $C_\epsilon = 0.2$. The wakes tends to be very flat, in comparison to the wake averaging case, although the zero edge case (the purple line) shows some roll-up. This case also bows slightly in the y-direction, but in the opposite direction to the wake averaging case. However, due to the lack of stability it is very hard to make any definitive statements.

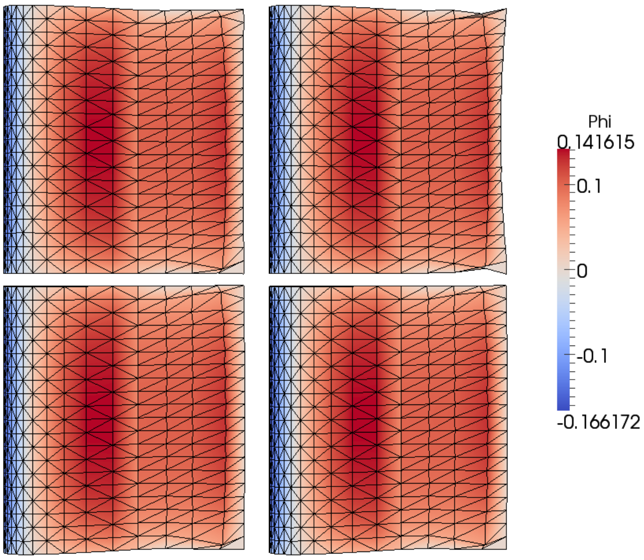


Figure 6.21: Comparison of wake profiles at varying C_ϵ for $\alpha = 5°$, viewed from positive z. Top left is $C_\epsilon = 0.5$, bottom left is $C_\epsilon = 0.2$, top right is $C_\epsilon = 0.1$, bottom right is zero ϵ edge case.

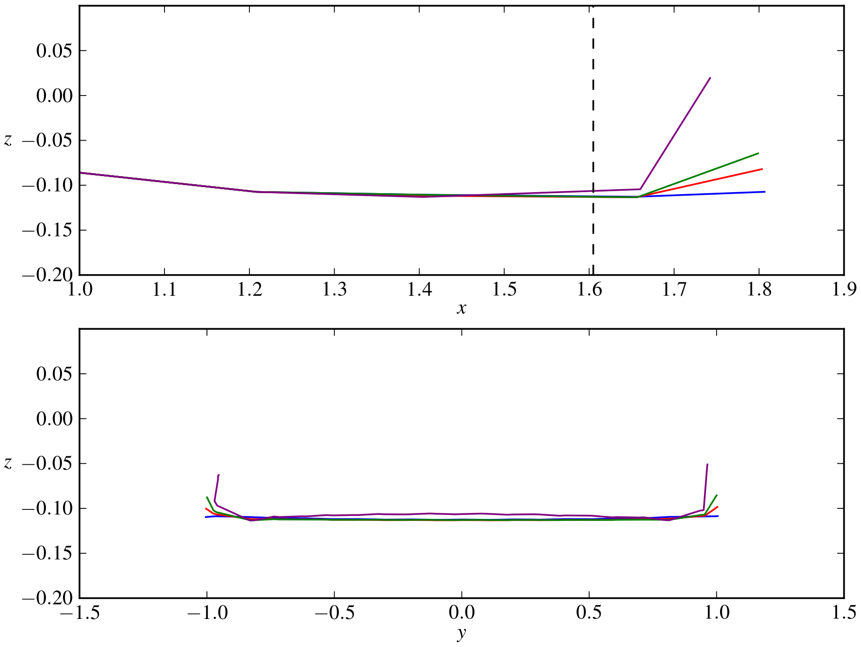


Figure 6.22: Comparison of wakes at varying C_ϵ at $\alpha = 5°$ for slices in the x and y directions at $t = 0.3$. The blue line is $C_\epsilon = 0.5$, red line is $C_\epsilon = 0.2$, the green line is $C_\epsilon = 0.1$ and the purple line is the zero ϵ edge case.

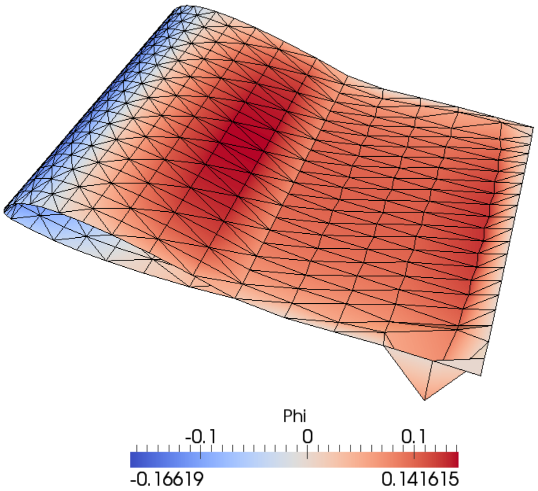


Figure 6.23: Failure of wake at $t = 0.4$ for $\alpha = 5^\circ$, $C_\epsilon = 0.5$ and 10 spanwise panels.

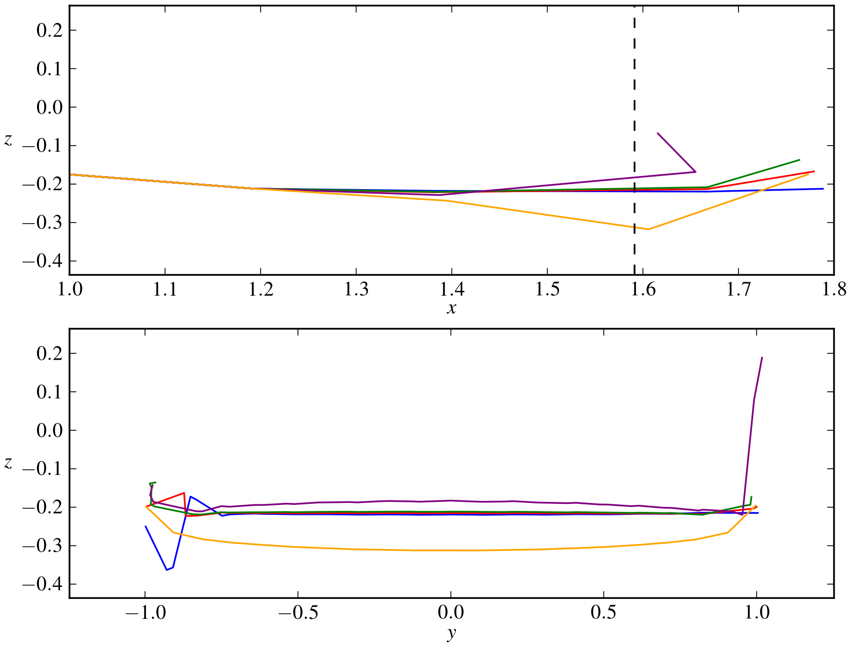


Figure 6.24: Comparison of wakes for varying C_ϵ at $\alpha = 10^\circ$ for slices in the x and y-directions at $t = 0.3$. The blue line show the results for $C_\epsilon = 0.5$, the red line $C_\epsilon = 0.2$ and the green line shows results for $C_\epsilon = 0.1$. The purple line shows the zero ϵ edge case and the orange line is the wake averaging case.

As discussed in chapter 4, the solution at the wake edges is singular. It is for this reason that the Galerkin gradient method cannot be applied at the wake edges. In fact, it turns out that this singular behaviour causes an inaccuracy in the results near to the edges also (see Gray *et al.*, 2004b, figure 9). It is possible that this inaccuracy is being encountered in the wake simulations as well. In order to attempt to overcome this issue, a second line of nodes at the wake edges can be solved with the averaging algorithm rather than the Galerkin method. An initial application of this algorithm is seen in figure 6.25 for a wing with $\alpha = 5°$. The wake remains more stable using this approach and does not fail until $t = 0.9$. A comparison between these results and a similar simulation using just the averaging algorithm with $C_\epsilon = 0.2$ together with zero edge ϵ is shown in figure 6.26. Here the marked contrast in the downwash between the averaging and the Galerkin algorithms can be seen along with the lack of a bow shape.

Unfortunately using the above double edge node algorithm does not improve the result for $\alpha = 10°$ as illustrated in figure 6.27. The averaging algorithm itself is not particularly stable in this case so it is not surprising that this induces further instability in the Galerkin algorithm. A possible solution to overcoming this instability is to damp the value of μ on the line of nodes adjacent to the line of nodes furthest from the trailing edge of the wing (i.e those nodes that are generated at $t = \Delta t$). The effect of this for $\alpha = 10°$ is seen in figure 6.28. Although this approach has a positive outcome for the averaging algorithm used at the nodes adjacent to the leading edge, the instabilities at the streamwise edges once again cause the algorithm to break down. It may be feasible to attempt to damp these edges also, but whereas damping the nodes adjacent to the leading edge could be seen as circumventing the unnatural impulsive start of the simulation, for the edge nodes there is no physical reason to do this. It would be a purely artificial means to ensure stability for a longer period of time.

In conclusion, the results seen in this section raise serious questions about the applicability of the Galerkin gradient method for the time-stepping of wakes. As can be seen, the algorithm is least stable at the edges where most of the physics, embodied by the wake roll-up, is occurring. Perhaps then the search for improved accuracy of the wake evolution is less important than assuring stability in the regions of high roll-up, particularly in a time-stepping simulation? It also remains unclear as to whether the Galerkin algorithm is reproducing the correct physics for the wake sheet. As three-dimensional open vortex sheets are less studied than closed and periodic vortex sheets

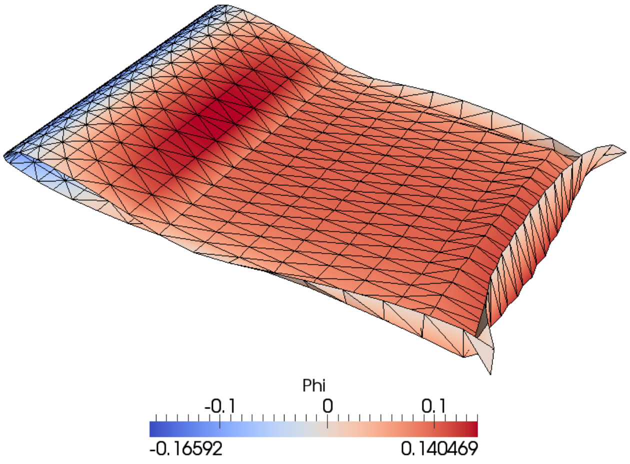


Figure 6.25: Wake profile of the double edge node algorithm at $t = 0.8$ for $\alpha = 5°$, $C_\epsilon = 0.5$ and 10 spanwise panels.

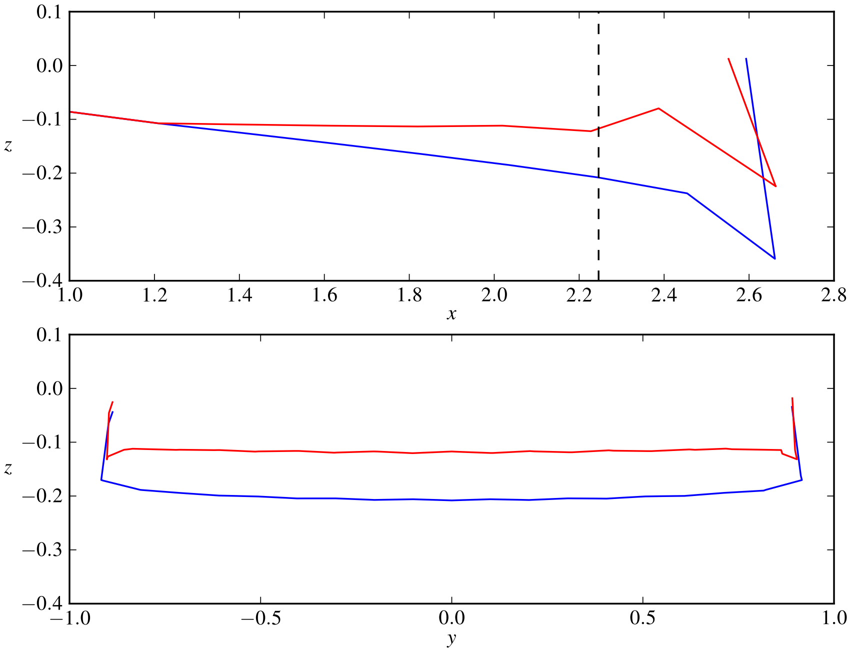


Figure 6.26: Comparison of wakes slices in the x and y directions with double edge node algorithm and the wake averaging algorithm at $t = 0.8$. The blue line is the wake averaging algorithm with $C_\epsilon = 0.2$, and the red line is double edge node Galerkin case.

it is relatively difficult to make comparisons with them in anything but a qualitative manner and due to the stability issues, comparing integrated properties for the wing is also very difficult. For more definitive proof that the Galerkin method is applicable to wake sheets, the stability issues must be addressed. Firstly, the author was unable to verify the code supplied to him by Professor Gray; it was applied to the wake problem as a 'black box' and such simulations lack a basic analytical solution with which the code can be verified. Nonetheless, it has been successfully applied to other, related, problems. More examination of the code, in addition to applying it to other fluids problems, such as closed and periodic vortex sheets, would be beneficial. Furthermore, if the theoretical analogy to the crack problem is sound, the introduction of *crack tip* or *quarter point* elements (see Bonnet, 1999, pg. 313) may prove beneficial to the stability of the sheet. A final key component to ensuring stability of the wake is the accurate calculation of the velocity at the edges. This has not been sufficiently investigated in this study and almost certainly improvements can be made over the averaging algorithm demonstrated here. One possibility may be to integrate vortex cores at the sheet edges into the model, although care must be taken not to invalidate the irrotational assumptions of potential flow.

Only if these issues can be addressed, and the Galerkin algorithm can be shown to produce the expected physical results for phenomena such as wake roll-up, will there be justification for the greater computational effort required to apply the Galerkin algorithm. It may well be that in order to ensure greater stability and improved computational time, alternative methods, such as vortex particles, may be more appropriate for the submerged lifting body problem than the use of vortex wake sheets. Yet, the Galerkin gradient method may well still prove useful in other roles, such as improving the calculation of the free surface velocity near to channel walls where differencing schemes are less accurate. The possibility of solving the entire problem using a symmetric Galerkin boundary element method (see Sutradhar *et al.*, 2008) may also prove attractive as using this approach to calculate the values of potential and normal derivative may reduce the additional computational effort required to then calculate the other gradients.

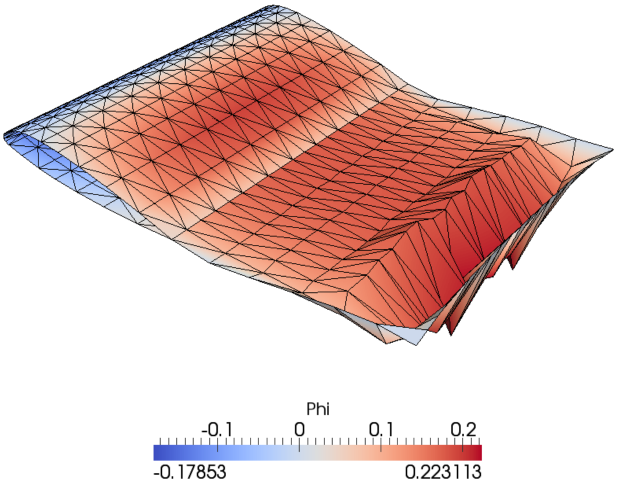


Figure 6.27: Wake profile of the double edge node algorithm at $t = 0.4$ for $\alpha = 10^\circ$, $C_\epsilon = 0.5$ and 10 spanwise panels.

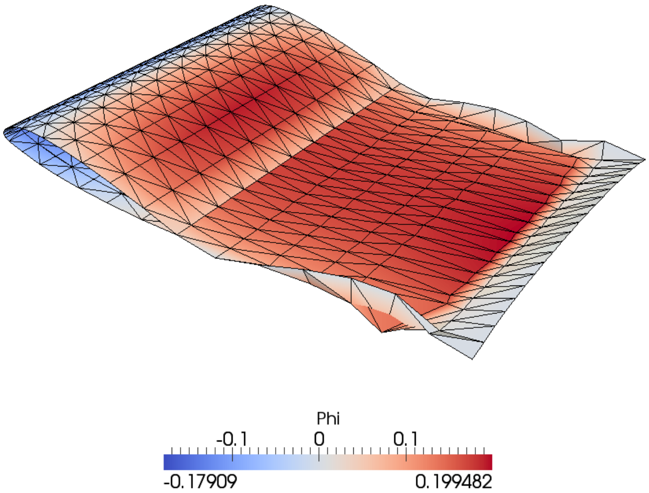


Figure 6.28: Wake profile of the double edge node algorithm at $t = 0.6$ for $\alpha = 10^\circ$, $C_\epsilon = 0.5$ and 10 spanwise panels. The nodes adjacent to the line of wake nodes furthest from the trailing edge are damped.

Chapter 7

Conclusions and Further Work

The interaction of a free surface with the wake and blades of a tidal current turbine is an important and as yet unquantified phenomenon. The presence of a free surface is one of the main aspects which differentiates the tidal turbine from the similar technology of wind turbines.

In summary of this project, its main aim was to attempt to develop a numerical method for simulating the flow past a tidal turbine in an open channel. In addition, the developed model would also be designed to model accurately the free-surface and the turbine wake, as well as the interaction between them, the turbine and the boundaries of the channel. This was an ambitious goal, requiring the combination and development of various complex methods which had not been previously brought together. After extensive review of all the available literature relating to the topic, the author decided upon developing a panel method to model the quasi-potential flow within the channel; the flow is assumed to be irrotational everywhere except for the wakes of the turbine blades, which are approximated by vortex sheets. As a base method, the high-order panel method of Xü (1992) was adopted. This method was originally developed to model 3D free-surface flows. Many of the aspects of this model have been enhanced and developed here, partly due to the lack of some details in Xü (1992). To this method the perturbation potential formulation was added, which enabled reduced discretisation for pumped flow problems.

The order of lifting surface solutions using boundary element methods is often lower than those used for free surface solutions, such as in Xü (1992). This is due, in part, to the requirement to solve the hypersingular boundary integral equation on the vortex sheet representing the wake. A new method, originally developed for the field of fracture mechanics to find the gradient of the potential jump across cracks, was identified as capable of solving the hypersingular boundary integral equation for velocity at the edge and corner points of boundary elements without the requirement for C^2 continuity at their interfaces. This method, known as the Galerkin boundary element method, was developed by Gray *et al.* (2004b) and for the first time it was applied here to the evolution of wakes. The Galerkin method is currently only applicable to linear panels and, therefore, three node triangular panels were added to the base method for use on the turbine wakes. The power of the Galerkin method is that most of the steps are carried out analytically and, therefore, exactly; only the final stages require numerical approximation.

In the following text, the chapters of this thesis are summarised in more detail and the important issues are highlighted. After this summary, conclusions are drawn and further work is suggested.

At the start of chapter 2, the boundary conditions for the simulation, including the free surface, were established using two different schemes. The first scheme, known as the full potential method, requires all domain boundaries to be discretised and the far field flow is enforced by Neumann boundary conditions on the inlet and outlet boundaries. In the second scheme, called the perturbation potential formulation, the far field velocity is implicit in the solution and, as a result, it is no longer necessary to discretise the inlet and outlet boundaries. In section 2.4.3 and section 2.4.4, the boundary condition for the strength and evolution of the wake, known as the *Kutta condition*, was derived. Prior to considering full lifting solutions, the boundary integral equations required for the solution of the Laplace equation were determined for a simple channel with a free surface and a uniform, steady flow. In section 2.6, the hypersingular boundary integral equation for the calculation of velocity on a domain boundary was introduced. Finally, in chapter 2, alternative schemes for the discretisation and solution of the integral equations, known as the collocation and Galerkin methods, were introduced prior to their application to the standard boundary integral and hypersingular boundary integral equations problems.

To provide a basis for the present method, a high order boundary element method was sought that was shown to solve three-dimensional free surface problems accurately. The method developed by Xü (1992) has been shown to produce excellent results for wave motions up to the point of breaking. The concepts underpinning Xü's method, which uses iso-parametric quadratic boundary elements, have been applied to the present method, as detailed in chapter 3. Alterations to Xü's original adaptive scheme, where parameters were not clearly defined, have been made in section 3.3. The original parameters for the adaptive control of the self doublet influence, given by equation (3.48), were retained as no analytical result for an isolated panel could be evaluated to compare against. This fact may have an impact on the results of the method and is discussed further later.

The method was extended in section 3.4 to include linear triangular elements for use on the wake sheets of any lifting bodies submerged in the flow. The full discretised linear system of equations was illustrated in section 3.5, first for non-lifting flows and then including the numerical Kutta condition for defining wake strengths. Section 3.6 introduced the use of reflective kernels to reduce the required amount of discretisation of the flow channel, and the application of double nodes at surface intersections to ensure continuity of ϕ.

The order of convergence for this method was examined in chapter 5. In section 5.2, the order of convergence was shown to be similar to that of Xü's method (QBEM) for the Neumann problem of flow passed a sphere. The present method did not achieve the exact order of convergence of QBEM as can be seen in table 5.1. This may be due to the difficulties in producing a test for the self doublet influence of an isolated panel. The accuracy of the routines to calculate this influence could only be tested for a global problem, and thus the verification of this part of the program was not as rigorous as for the other influences. Also, instead of using an iterative solver for the linear system, as in Xü (1992), a commercial matrix inverting routine was employed here. Finally, as the discretisation of the sphere itself was not discussed in great detail in Xü (1992), it is possible that the discretisation schemes for the sphere, developed in section 5.2.1, were not as uniform as those used by Xü. This possibility is supported by figure 5.5 which shows that the type of discretisation used on the sphere (and the accuracy of the initial conditions) has a significant impact on the accuracy and order of convergence of the method.

In section 5.3, additional convergence studies were carried out for steady uniform flow through a simple channel with a free surface. These tests were performed in order to evaluate the accuracy of the method once a Dirichlet surface has been introduced. The tests also facilitated a comparison between the performance of the standard Green's function and the Green's function including the reflective bottom and right hand side channel wall boundaries. The results presented in figure 5.8 initially seem to be very poor; on closer inspection however, the low order of convergence is actually a result of the method exceeding its target accuracy, even at the lowest level of discretisation. Despite this fact, figure 5.8 clearly indicates that the adaptive numerical differentiation is working as the maximum relative error never exceeds 1×10^{-5}.

The patterns of errors illustrated in figure 5.7c and figure 5.7d are also highly revealing. Here, when comparing the reflective Green's function solution to that of the non-reflective, the impact of the channel wall boundaries on the errors becomes clear. Although the error caused by the proximity of one surface to another surface is consistent, where three or more surfaces meet, these errors combine (as is expected for a linear problem) in both destructive and constructive manners. Thus, to reduce the maximum errors seen in these regions, reducing the number of discretised surfaces is highly beneficial. An alternative approach may be to further enhance the adaptive numerical scheme to better cope with the proximity of nearby surfaces.

In chapter 4, a numerical differentiation scheme was developed for the free surface boundary in order to evaluate the surface velocities, which can then be used for time-stepping. In chapter 5, the examples of single and counterpropagating solitary waves (see Craig *et al.*, 2006) were used to evaluate this numerical differentiation method and to assess the accuracy and stability of the time integration and dynamic time-stepping schemes developed in section 4.4. The results of section 5.4.2, for the single soliton, indicate that if the chosen CFL number for the dynamic stepping was low enough, the simulations were stable without the need for smoothing. In these cases, the often witnessed sawtooth instability on the free surface did not appear. The counterpropagating soliton study, undertaken in section 5.4.3, showed reasonable agreement with the recent study of Craig *et al.* (2006) in some of the cases, although for the larger wave height the simulation was not evolved long enough for the residual waves to separate entirely from the solitons, post collision. As is shown in figure 5.16, a similar simulation of greater length would be a useful addition, providing more accurate asymptotes to the motion of the resulting solitons than those extrapolated from figure 5.14b. For all the examples, an improved method to track the peaks of the waves would be beneficial, as they do not always coincide with a computational node. This may improve the agreement to the results of Craig *et al.* for the counterpropagating case. In general, the soliton tests illustrate that the numerical differentiation, time integration and dynamic time-stepping schemes have been successfully implemented.

Within section 4.3, the Galerkin gradient method of Gray *et al.* (2004b) was derived in full and the analogy of finding the velocity on the wake of an aerofoil to that of calculating the gradient of the potential difference across a crack in fracture mechanics was established. Many of the details of the Galerkin gradient method which were omitted from Gray *et al.* (2004b) for space reasons have been re-derived and presented here for completeness. In section 4.3.4, it was stressed that care must be taken at the wake edges and that an alternative solution scheme must be applied at these locations, otherwise the Galerkin method will return the singularities known to be present there. It was shown that the Galerkin method could, instead, be truncated using an alternative solution scheme to find the velocity at the wake edges and that the Galerkin method could only be applied at nodes directly adjacent to the edges if the strength, μ, on the wake edges was zero. If this was not the case, such as at the trailing edge of the aerofoil, then the Galerkin algorithm can only be applied at the second line of nodes adjacent to the edge.

An alternative method for calculating the velocity at the wake edges, and for the nodes adjacent to the trailing edge, was derived in section 6.3.2. This algorithm takes an average of velocities above and below the wake sheet at a predefined distance, ϵ. The velocity of the nodes at the trailing edge is provided by the spatial Kutta condition, which fixes the direction of the first line of panels leaving the wake, as illustrated in figure 6.3. Care must be taken to preserve this direction during the first time-step and the required algorithm (represented by figure 6.4) was carefully described. In section 6.4, it was found that the CFL condition for the wake is determined by the speed of the oncoming flow, as a consequence of the spatial Kutta condition. This removed the ability to apply the same dynamic time-stepping schemes used for the free surface problems to those involving the wakes. It was postulated that an alternative scheme could be formulated, based on the roll-up of the wake edges, although this has been reserved for further work.

To evaluate the effectiveness of the wake averaging velocity algorithm, the wake simulation was initially run without the use of the Galerkin method. Initially, a simulation with no self-induced velocity from the wake was undertaken to examine the influence of the wing on the wake sheet. As can be seen from figure 6.9, the influence is very small but a slight 'bowing' can be detected. The remainder of section 6.5 examined the impact of choosing various values of ϵ for the averaging scheme applied to every node on the wake sheet, apart from those at the trailing edge. The results are qualitatively compared to those of Pyo and Kinnas (1997) for an elliptically loaded lifting line, as can be seen in figure 6.7, paying particular attention to the bowing of the wake and 'roll-up' of the wake edges. It was found that the bowing of the wake reduces with increased discretisation on the wake when using a fixed ϵ. This is because the relative length of ϵ to the panel size is increasing as the panel sizes are reduced. When the discretisation is fixed and ϵ is reduced the bowing is seen to increase. Furthermore, it was found that the 'downwash' of the wakes increased as the angle of attack increased. Some roll-up was witnessed at the wake edges although this was very little at first. The amount of roll-up was increased when the value of ϵ at the wake edge was reduced to zero. Although the true solution is singular for zero ϵ at the wake edge, the algorithm necessarily produces a finite result which appears credible. This is likely to be due to the application of numerical quadrature combined with the zero value of μ at the wake edges producing a pseudo near-singular problem. In this case the wake edges exhibit significantly more roll-up than previously seen, although it is still unclear as to whether this is the correct solution and further investigation into how to properly deal with the wake edges is an important part of any further work, with or without the application of the Galerkin method. Some alternative solutions may be present in the work of Pyo (1995). Potentially a vortex core model, similar to a 'Rankine vortex' (see Acheson, 1990, page 16), could be applied to the wake edges, although this would invalidate the irrotational flow assumption.

The final stage of this work concerned the application of the Galerkin gradient method to the wake sheet. Again, care was required in the time-stepping of the method, and the method cannot be applied until the second time-step. The same conditions for maintaining the direction of the Kutta strip must also be observed. These issues are illustrated in figure 6.6. The truncation of the Galerkin algorithm at the wake edges was provided by the wake averaging scheme. Initially, for the simulations undertaken in section 6.6, it was found that the Galerkin method calculated internal nodes correctly for the first time-step with which it was applied. Beyond that, unfortunately, instabilities began to occur at the wake edges. Attempts were made to reduce these instabilities by using the averaging algorithm for an additional line of nodes adjacent to the edges. Figure 6.27 shows that this improved the stability of the method. However, the interaction between the edges and the internal nodes of the wake, once again, produced a catastrophic instability before the simulation was completed in all cases. Indeed, the problems grew more severe as the angle of attack was increased. For the limited time that the wake was stable, the shape of the wake calculated by the Galerkin method could be examined. It did not exhibit the bowing that was expected from previous examples and, in fact, its motion in the vertical direction was very slight, if at all noticeable. It is not clear if this is the correct behaviour.

It was also difficult to determine whether the wake edge roll-up was being simulated correctly due to the lack of stability. This is a severe difficulty with testing the applicability of the Galerkin method to vortex wakes and a better solution at the wake edges must be found. A similar instability is witnessed in the crack problem for fracture mechanics and is improved with the use of quarter point elements at the edges. Further investigation into whether this approach is applicable to vortex wakes may prove fruitful.

Unfortunately, there is not currently enough information to determine whether the Galerkin gradient method correctly models the physics it has been applied to within this thesis. As further work, alternative methods of wake generation and velocity calculation schemes should be investigated. This will give a clearer picture as to the accuracy of both the wake averaging method and the Galerkin gradient method for calculating the wake velocity. Using alternative, and possibly less complicated, methods to generate wakes will also allow the global method to be validated against real world solutions, which will provide a better baseline for the accuracy of more advanced methods, such as the Galerkin gradient method. The conclusion of such a study may be that the increased accuracy offered by the Galerkin algorithm for solving at the collocation points is outweighed by the additional instability inherent to the method. Along with the additional computation cost of inverting the Galerkin matrix, if the stability issues can not be solved then the Galerkin method may offer little benefit to lifting flow applications. Indeed, in the case of tidal turbines, where so little is known about the wake structure, it may be that modelling the wake as a vortex sheet is not a robust enough approach and, perhaps, a discrete vortex method or similar may be more durable, if not as accurate. Nonetheless, the Galerkin method may still prove useful for accurate calculation of the velocity on the free surface, particularly in the regions where walls are present and skewed differencing schemes must be used. This represents another viable, and achievable, course for further work.

In closing, although the wake velocity evaluation methods failed to produce stable, and thus comparable, results, the method as a whole, developed throughout this thesis, has been implemented with a reasonable amount of success. In particular, as the results in chapter 5 attest, the free surface solver of the method is accurate and stable. Therefore, there are many extensions for this method that go beyond tidal turbines and for which it could benefit the marine energy community. In particular, in combination with work such as Forehand (1998), boundary element methods have excellent potential to be used as numerical wave tanks, including the generation of waves by paddles and the interaction with fixed and floating structures. Where experimental facilities exist, there is the opportunity to undertake excellent validation studies of simulations which mimic those facilities. This method has also demonstrated that current can be applied in a boundary element method, and this may also be of benefit to the wave energy community – the impact of wave and current interactions on wave energy converters is, as yet, poorly understood. At the very least, the present method may provide the basis for a model of greater complexity, as it is made available to the marine energy research community as a whole.

Appendix A

Review of the Kutta Condition

A.1 Kutta-Joukowski Hypothesis

In the early twentieth century the development of flight lit the imaginations of politicians, entrepreneurs and academics alike. Joukowski (or Zhukovski or Zhukovsky), the renowned Russian mathematician, provided groundbreaking techniques for applying complex number theory to the modelling of the flow around an aerofoil. The Joukowski transformation could be used to transform a circle into an aerofoil shape that had both a sharp trailing edge and a blunt nose. These transformations are known as Joukowski aerofoils (see Milne-Thomson, 1968). The techniques, although useful, have less bearing given today's computational power, where three-dimensional simulations are becoming compulsory and analytical results may not be available for complex geometries. The theoretical observations that Joukowski made, however, have had enduring importance to aeronautical modelling.

The Kutta-Joukowski hypothesis is best described by figure 2.2. The flow about an aerofoil, when considered to be irrotational and have no circulation, must 'turn the corner' around the aerofoil trailing edge causing a singularity to be formed there. This would violate the irrotational assumption as well as be unphysical for a aerofoil that was working normally. Joukowski hypothesised that the correct amount of circulation in the flow would move the stagnation point (seen on the top surface of the foil in figure 2.2(a)) to the trailing edge which would produce a finite velocity there. The hypothesis was originally attributed to Joukowski from his papers published in 1906. However, the German mathematician Martin Wilhelm Kutta had made similar observations in his 1902 thesis. Thus history has credited both men with the work.

A.2 The Kutta-Joukowski Theorem

The main outcome of the Kutta-Joukowski hypothesis was the connection of the circulation about an aerofoil to the generation of lift. Both Kutta and Joukowski were able to prove a proportional relationship between the circulation around the aerofoil and the lifting force (and direction) generated. A proof, using complex velocities, is given in Milne-Thomson (1968). An important distinction to make about the Kutta-Joukowski theorem is that it is derived for steady state motion. As described earlier, the starting of circulation about the aerofoil would violate Kelvin's circulation theorem (see Katz and Plotkin, 2001) making the flow irrotational. Kutta and Joukowski circumvented this problem by concentrating on the steady state problem where the circulation had already been set up. Physically, in order for the circulation to start about the aerofoil, a vortex sheet must be shed into the wake behind the aerofoil that has a circulation which is equal and opposite to that about the aerofoil. It is the generation of the said wake sheet that is of great interest to aerodynamicists.

A.3 The Kutta Condition

The *Kutta condition* is merely the mathematical interpretation of the Kutta-Joukowski hypothesis, effectively providing a condition at the sharp trailing edge of an aerofoil. Although the Kutta condition seems to be derived directly from the Kutta-Joukowski hypothesis, Joukowski's name appears to have been lost from this often used expression. From a numerical modelling perspective, the Kutta condition is invaluable, in that it provides a means of calculating the vorticity shed into the wake of a aerofoil and it converts a multivalued problem into a unique one.

Katz and Plotkin (2001) provide a good argument for the uniqueness of potential theory based lifting solutions where the domain is no longer simply connected. It is shown that if the circulation is not explicitly known then the potential is not unique. Thus the Kutta condition provides a physical relationship that is vital to the application of boundary element modelling. However, the correct implementation of the Kutta condition, particularly for unsteady and three dimensional flows, remains a topic of great interest.

During the late 1960s and early 1970s the use of boundary element methods to numerically calculate potential flows were becoming more popular, with advances in computing power especially available to industries like aviation. Early work on lifting potential flow solutions (such as the work of Hess (1972)) pioneered the ideas and mathematics that would go on to become commonly known as panel methods. Hess (1972) provides a good description of the types of numerical Kutta condition and cites some of the early references. Hess, however, did not formulate his panel method with the now commonly used Green's identity; instead, he chose to use the more physical, induced velocities of constant strength panels. As a consequence, Hess failed to explicitly define the Kutta condition as many numerical analysts know it today.

The most cited work regarding the Kutta condition is that of Morino and Kuo (1974). Morino and Kuo were the first to formulate the singularities of the panel method using Green's identity. In the same paper they introduced the numerical Kutta condition for Green's identity based panel methods. Morino and Kuo chose to interpret the Kutta condition as a 'no pressure difference' condition at the trailing edge and, in order to enforce this, set the potential difference in the trailing edge panel (commonly referred to as the Kutta strip) to the difference of the potentials on the upper and lower panels connected to the trailing edge. Equation (2.59) describes this process mathematically. As mentioned earlier, this elegant relationship provides both uniqueness and a means of calculating the wake strength. It also has the numerical advantage of easy integration into non-lifting boundary element codes. In recent times the issue has arisen as to whether the Morino type Kutta condition actually describes the physics it was intended to. Also, there continues to be debate as to the extension of the condition to both unsteady and three dimensional simulations.

It has often been noted that the Morino type Kutta condition does not give the most accurate representation of the Joukowski-Kutta hypothesis. Hess (1990) noted that physically specifying equality of pressure, in terms of velocity at the trailing edge, worked more effectively than the relationship given by equation (2.59). The reasons for this discrepancy and the methods for dealing with it are quite varied. Hess points out that pressure is quadratic when written in terms of velocity yet the Morino formulation is only linear. He also states that accurate evaluation of the Kutta condition in a steady simulation is critical to the accuracy of the result. It is unclear at the moment if an unsteady simulation is as sensitive.

Kinnas and Hsin (1992) were aware of the problems associated with the Morino formulation. They attempted to improve the accuracy of the dipole strength on the wake panels of their simulation by observing that although the Kutta condition is derived from the mid-points of constant panels, its application to the wake actually occurs at the trailing edge. Thus, they derived a linear method with the strengths of the panels determined at the panel edges rather than at the centroids. Still, as is the case for the Morino type condition, Kinnas and Hsin used the centroids of the upper and lower trailing edge panels to calculate the dipole strength.

Kinnas and Hsin found that even with the improved representation of the wake, the pressure distribution, particularly at the trailing edge where the pressure should be equal, was not accurate and thus a further iterative scheme was developed by Kinnas and Hsin (1992) to adjust the potentials on the aerofoil surface in order to enforce the no pressure difference condition on the trailing edge. This raises interesting questions as to whether the basic Morino type Kutta condition is sufficient. Should the present method be developed for application to turbine design, the correct implementation of the Kutta-Joukowski hypothesis may be key to ensuring rigorous validation.

To address the question of why the Morino formulation does not work as well as expected, Morino reexamined the assumptions he made 26 years earlier. Morino and Bernardini (2001) focused their attention on the physics of the Kutta-Joukowski hypothesis and in particular on the calculation of the velocity at the trailing edge when using a Green's function based panel method. In order to examine this, Morino and Bernardini call upon the work of Epton (1992) which examines the hypersingular boundary integral equation (HBIE). The HBIE is formulated by taking the derivative of the boundary integral equation for potential, which results in equation (2.74).

Epton applies a Stokes' theorem based transformation of the HBIE to isolate two singular line integrals. The first is the common '$\ln r$' term that produces singular behaviour in the ordinary boundary integral equation. The second line integral contains the hypersingular part. Epton (and Morino and Bernardini) refer to these line integrals as the "edge-jet" and "line-vortex" terms, respectively. Morino and Bernardini attempt to relate this representation of velocity to the Kutta condition. Interestingly, the Kutta condition as Morino and Kuo first formulated it takes care of the more difficult hypersingular term. However, the edge-jet term remains. Thus, Morino and Bernardini formulated new equations to enforce a Kutta Condition on the trailing edge that would remove both singularities. These equations were implemented in Bernardini's thesis using a double node at the trailing edge.

Alternative literature suggests that in order to properly enforce the Kutta Condition, potential based solutions must be turned away from completely. Bassanini *et al.* (1999) reject the potential approach in favour of a velocity formulation based on the Poincaré identity. Bassanini *et al.* list the difficulties of properly formulating the Kutta condition in potential theory and also some of the difficulties encountered with using the HBIE for finding velocities as good reasons for exploring this alternative approach. Overcoming some of the difficulties associated with the HBIE are explored in the current work.

Despite all of the criticism of the Morino and Kuo (1974) formulation of the Kutta condition it remains heavily used in potential based lifting solutions. Katz and Plotkin (2001), Politis (2004) and Zhu *et al.* (2006) have all used the Morino Kutta condition in 3D unsteady simulations. Even though he has criticised it's accuracy, Kinnas has also continued to apply the technique to his work such as in Kinnas and Fine (1993) and Lee and Kinnas (2005a). In the present study the ease of use of Morino Kutta condition is hard to ignore and, as a 'first attempt', will be sufficient to produce a unique and physical solution. In the interest of further accuracy, the application of Bernardini's modification described in Morino and Bernardini (2001) would be an appealing extension.

A.4 The Wake Shape and the Kutta Condition

The successful design of a quasi-potential boundary element method (i.e. one containing a wake modelled as a vortex sheet) requires that the vortex sheet leave the trailing edge with the correct orientation. There is some debate as to how important the orientation is to the successful implementation of the Kutta-Joukowski hypothesis but sheet orientation is often described as the Kutta condition and Hess (1972) lists it as an alternative to the no pressure difference or finite velocity condition. As an example, consider the flow leaving the trailing edge of a flat plate. It would seem intuitive that any discontinuity of velocity on the plate should leave parallel to the surface. Indeed, this observation is described as the Kutta condition in the work by Tsai and Yue (1993). Unfortunately, the correct orientation for a sheet leaving the trailing edge of a thick wing is less intuitive.

For the thick wing case, Katz and Plotkin (2001) simply choose the bisecting angle of the trailing edge to orientate the first panel of the vortex sheet as it separates. It is the author's opinion that, although it is often not explicitly stated, many other simulations follow the same principle. Mangler and Smith (1970), Morino and Bernardini (2001) and indeed, Bassanini *et al.* (1999) suggest that this does not correctly represent the physics.

Mangler and Smith (1970) were interested in the extension of boundary element methods to lifting flows and the incorporation of vortex sheets. They set out to prove how such a vortex sheet should be oriented downstream of the trailing edge. They examined the five possible cases for the orientation of the vortex sheet listed as

1. the wake passes above the upper tangent of the trailing edge,
2. the wake is tangent to the upper tangent of the trailing edge,
3. the wake passes between the upper and lower tangents of the trailing edge,
4. the wake is tangent to the lower tangent of the trailing edge,
5. the wake passes below the lower tangent of the trailing edge.

Mangler and Smith deduced that cases 1 and 5 are unphysical as they would immediately violate the Kutta condition of finite velocity at the trailing edge. Morino and Bernardini (2001) state that such an orientation would also generate a Kondrat'ev and Oleinik singularity and must be avoided.

With cases 2 to 4 remaining, the surprising result is that case 3 is the exception. Mangler and Smith found that, dependant on the velocity tangential to the spanwise direction of the trailing edge, the wake would either resemble cases 2 or 4. They found that case 3 only occurred in limited scenarios, such as at the centre of a symmetric wing, and they showed that the sweep of the wing had an important role in determining the orientation of the wake.

Morino and Bernardini (2001) confirmed the result of the Mangler and Smith analysis, but both papers concentrated only on the steady state solution. Bassanini *et al.* (1999) were able to apply their formulation of the spatial Kutta condition to the same problem but this time considering unsteady flows. In examining unswept wings they found that, once again, the bisecting angle (intermediate) case was the exception and that the wake should align with either the upper or lower tangent to the trailing edge. The bisecting case was shown only to occur when there is no oncoming flow or at the centre of a symmetric wing. Bassanini *et al.* found that the extension of their analysis to swept wings agreed with that of Mangler and Smith when the limit was taken as $t \to \infty$.

The pertinent literature demonstrates that the basic trailing edge bisecting angle vortex wake orientation of Katz and Plotkin (2001) may not properly describe either the Kutta condition or the flow of interest. Mangler and Smith (1970) and later Bassanini *et al.* (1999) found that the bisecting case is in fact the exception rather than the rule and showed that the wake tends to align to either the upper or lower tangent of the trailing edge. In the present work, the wake leaves the trailing edge at the bisecting angle. For further work, provided a stable wake can be achieved, dynamic wake orientation could potentially provide more accurate results.

Appendix B

Derivatives of Shape Functions

For the nine node quadratic elements the s and t derivatives of Q are

$$\begin{aligned} Q_s = \frac{\partial Q}{\partial s} &= \left(\frac{\partial x_q}{\partial s}, \frac{\partial y_q}{\partial s}, \frac{\partial z_q}{\partial s} \right) \\ &= \left(\sum_{j=1}^{9} \frac{\partial \psi_j}{\partial s} x_j, \sum_{j=1}^{9} \frac{\partial \psi_j}{\partial s} y_j, \sum_{j=1}^{9} \frac{\partial \psi_j}{\partial s} z_j \right) \end{aligned} \tag{B.1}$$

and

$$\begin{aligned} Q_t = \frac{\partial Q}{\partial t} &= \left(\frac{\partial x_q}{\partial t}, \frac{\partial y_q}{\partial t}, \frac{\partial z_q}{\partial t} \right) \\ &= \left(\sum_{j=1}^{9} \frac{\partial \psi_j}{\partial t} x_j, \sum_{j=1}^{9} \frac{\partial \psi_j}{\partial t} y_j, \sum_{j=1}^{9} \frac{\partial \psi_j}{\partial t} z_j \right) . \end{aligned} \tag{B.2}$$

The differentiated interpolation functions are given by

$$\left\{ \begin{aligned} \frac{\partial \psi_j(s,t)}{\partial s} &= \frac{1}{4}(2s + s_j)t(t + t_j) && j = 1, 3, 5, 7, \\ \frac{\partial \psi_j(s,t)}{\partial s} &= \frac{s_j}{2} \left(1 - {t_j}^2 s^2 - {s_j}^2 t^2\right)(1 + 2s_j\, s) && \\ &\quad - {t_j}^2 s \left(t_j\, t\,(1 + t_j\, t) + s_j\, s\,(1 + s_j\, s)\right) && j = 2, 4, 6, 8, \\ \frac{\partial \psi_9(s,t)}{\partial s} &= -\, 2s(1 - t^2) && \end{aligned} \right. \tag{B.3}$$

and

$$\begin{cases} \dfrac{\partial \psi_j(s,t)}{\partial t} = \dfrac{1}{4}(2t+t_j)s(s+s_j) & j=1,3,5,7, \\ \dfrac{\partial \psi_j(s,t)}{\partial t} = \dfrac{t_j}{2}\left(1-{t_j}^2s^2-{s_j}^2t^2\right)(1+2t_j\,t) & \\ \qquad\qquad -\,{s_j}^2t\,(t_j\,t\,(1+t_j\,t)+s_j\,s\,(1+s_j\,s)) & j=2,4,6,8, \\ \dfrac{\partial \psi_9(s,t)}{\partial t} = -\,2t(1-s^2). & \end{cases} \tag{B.4}$$

For the three node triangular linear elements, the s derivatives of Q is

$$\begin{aligned} Q_s = \frac{\partial Q}{\partial s} &= \left(\frac{\partial x_q}{\partial s}, \frac{\partial y_q}{\partial s}, \frac{\partial z_q}{\partial s}\right) \\ &= \left(\sum_{j=1}^{3}\frac{\partial \tilde{\psi}_j}{\partial s}x_j, \sum_{j=1}^{3}\frac{\partial \tilde{\psi}_j}{\partial s}y_j, \sum_{j=1}^{3}\frac{\partial \tilde{\psi}_j}{\partial s}z_j\right), \end{aligned} \tag{B.5}$$

with the t derivative following similarly. The s and t derivatives of of $\tilde{\psi}$ are given by

$$\frac{\partial \tilde{\psi}_1}{\partial s} = -\frac{1}{2}, \quad \frac{\partial \tilde{\psi}_2}{\partial s} = \frac{1}{2}, \quad \frac{\partial \tilde{\psi}_3}{\partial s} = 0, \tag{B.6}$$

and

$$\frac{\partial \tilde{\psi}_1}{\partial t} = -\frac{\sqrt{3}}{6}, \quad \frac{\partial \tilde{\psi}_2}{\partial t} = -\frac{\sqrt{3}}{6}, \quad \frac{\partial \tilde{\psi}_3}{\partial t} = \frac{\sqrt{3}}{3}. \tag{B.7}$$

Appendix C

Derivation of Equation (3.38)

Equation (3.38) can be proved by demonstrating equality of the left hand and right hand sides. To begin consider the triangle in figure C.1 with vertices p_1, p_2, p_3 and the relationships

$$\begin{aligned}\mathbf{a} &= p_2 - p_1 \\ \mathbf{b} &= p_3 - p_1.\end{aligned} \tag{C.1}$$

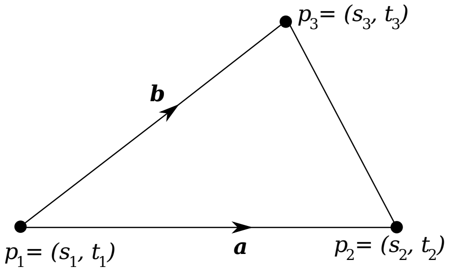


Figure C.1: Triangle.

The area of the triangle is given by $\frac{1}{2}|\mathbf{a} \times \mathbf{b}|$ where

$$\begin{aligned}\mathbf{a} \times \mathbf{b} &= \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ s_2 - s_1 & t_2 - t_1 & 0 \\ s_3 - s_1 & t_3 - t_1 & 0 \end{vmatrix}, \\ &= \mathbf{k}\left((s_2 - s_1)(t_3 - t_1) - (s_3 - s_1)(t_2 - t_1)\right), \\ &= \mathbf{k}\left(s_2t_3 - s_1t_3 - s_2t_1 - s_3t_2 + s_1t_2 + s_3t_1\right).\end{aligned} \tag{C.2}$$

Hence, the right hand side of equation (3.38) can be written as

$$2\rho_1 A_\mu = \rho_1 \left| s_2 t_3 - s_1 t_3 - s_2 t_1 - s_3 t_2 + s_1 t_2 + s_3 t_1 \right| . \tag{C.3}$$

Now, s and t can be represented in (ρ_1, ρ_2) parameter space by using equation (3.30) and the three vertices in figure C.1. This results in

$$\begin{aligned} s &= (1-\rho_1)s_1 + \rho_1(1-\rho_2)s_2 + \rho_1\rho_2 s_3, \\ t &= (1-\rho_1)t_1 + \rho_1(1-\rho_2)t_2 + \rho_1\rho_2 t_3, \end{aligned} \tag{C.4}$$

and the partial derivatives given by

$$\begin{aligned} \frac{\partial s}{\partial \rho_1} &= -s_1 + (1-\rho_2)s_2 + \rho_2 s_3, & \frac{\partial s}{\partial \rho_2} &= -\rho_1 s_2 + \rho_1 s_3, \\ \frac{\partial t}{\partial \rho_1} &= -t_1 + (1-\rho_2)t_2 + \rho_2 t_3, & \frac{\partial t}{\partial \rho_2} &= -\rho_1 t_2 + \rho_1 t_3. \end{aligned} \tag{C.5}$$

The Jacobian, J_μ, can be calculated using equation (3.9) such that

$$J_\mu(\rho_1, \rho_2) = \left| \frac{\partial s}{\partial \rho_1}\frac{\partial t}{\partial \rho_2} - \frac{\partial s}{\partial \rho_2}\frac{\partial t}{\partial \rho_1} \right| . \tag{C.6}$$

By substituting the results of equations (C.5) into equation (C.6) observe that

$$\begin{aligned} J_\mu(\rho_1, \rho_2) &= \rho_1 \left| (-s_1 + (1-\rho_2)s_2 + \rho_2 s_3)(t_3 - t_2) - \right. \\ &\qquad \left. (-t_1 + (1-\rho_2)t_2 + \rho_2 t_3)(s_3 - s_2) \right| \\ &= \rho_1 \left| s_2 t_3 - s_1 t_3 - s_2 t_1 - s_3 t_2 + s_1 t_2 + s_3 t_1 \right| . \end{aligned} \tag{C.7}$$

The above result is the same as equation (C.3) and hence the left and right hand sides of equation (3.38) are equal.

Appendix D

Wall Boundary Correction

Forehand (1998) and Grilli *et al.* (1989) point out the possible existence of singularities in the solution in the case of an accelerating Neumann boundary intersecting a Dirichlet boundary. Although the case of a fixed boundary is not so severe (nor does a singularity present itself in simulations) it is still useful for the free surface to "track" the side walls. This is in preference to interpolating the nodes back to the side walls or simply translating them in the y-direction. To implement such a condition correctly extra conditions must be enforced within the linear system.

A new boundary condition must be implemented for the free surface at the nodes which intersect the side walls. For the left channel wall, the desired behaviour is enforced by setting

$$\mathbf{v}_f \cdot \mathbf{n}_l = 0 \tag{D.1}$$

where $\mathbf{v}_f$ is the velocity of the free surface at the left channel wall and $\mathbf{n}_l$ is the normal to the wall (pointing out of the domain). $\mathbf{n}_l$ always points in the positive y direction, and so the challenge is to represent $\mathbf{v}_f$. To achieve this, consider the parametric directions on the free surface. $\mathbf{x}_t$ represents the t parametric direction tangent to the free surface at x and $\mathbf{x}_s$ represents the s parametric direction. These are illustrated in figure D.1. Now consider the case when the free surface is still and level. Here, $\mathbf{x}_t$ and $\mathbf{x}_s$ will be perpendicular to each other and to $\mathbf{n}_f$ and thus the velocity at $\mathbf{x}$ has the equation

$$\mathbf{v}_f = \phi_t \mathbf{x}_t + \phi_s \mathbf{x}_s + \phi_n \mathbf{n}_f \tag{D.2}$$

where ϕ_s and ϕ_t are the derivatives of ϕ in the s and t parametric directions.

Equation (D.2) is perfectly adequate while the free surface remains flat and the

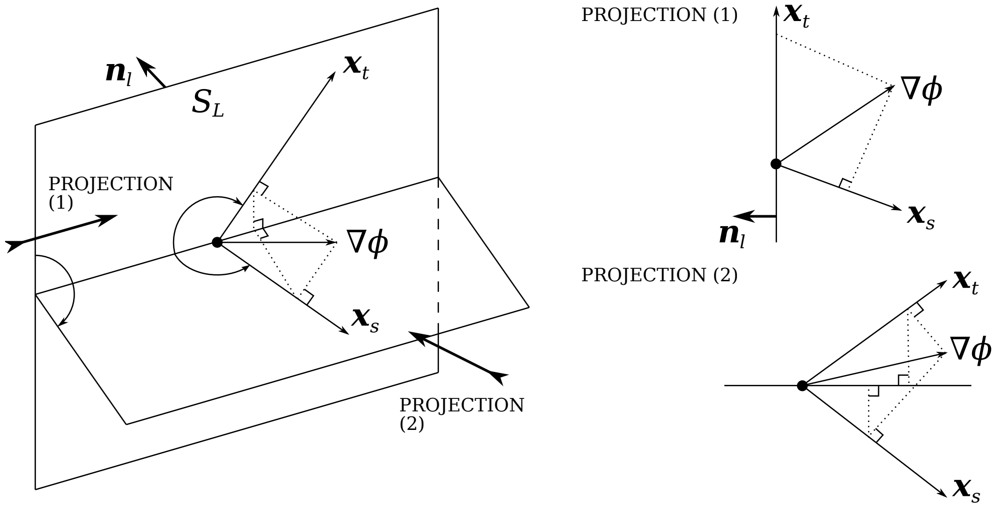


Figure D.1: Velocity vectors tangent to the free surface in the parametric directions. Projections are shown which are (1) tangent to the channel wall in positive x and (2) tangent to the positive normal to the wall.

vectors $\mathbf{x}_t$, $\mathbf{x}_s$ and $\mathbf{n}_f$ form an orthogonal system. However, once the surface begins to deform these vectors are no longer orthogonal and thus equation (D.2) is invalid. To compensate, a new orthogonal set of principle vectors $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ must be defined using the Gram-Schmidt process which will generate a velocity vector on the free surface of the form

$$\mathbf{v}_f = a\mathbf{e}_1 + b\mathbf{e}_2 + c\mathbf{e}_3. \tag{D.3}$$

For convenience, $\mathbf{e}_3$ can be chosen to be $\mathbf{n}_l$. The plane formed by $\mathbf{x}_t$ and $\mathbf{x}_s$ is (by definition) orthogonal to $\mathbf{n}_l$, but the vectors are not orthogonal to each other. The next stage is to choose one of these two vectors for $\mathbf{e}_1$, say $\mathbf{x}_t$. It then remains to find the remaining vector $\mathbf{e}_2$ using Gram-Schmidt. Begin by choosing

$$\mathbf{u}_1 = \mathbf{x}_t, \qquad \mathbf{v}_1 = \mathbf{x}_t, \qquad \mathbf{v}_2 = \mathbf{x}_s. \tag{D.4}$$

Then define

$$\begin{aligned}\mathbf{u}_2 &= \mathbf{x}_s - \mathrm{proj}_{\mathbf{u}_1}\mathbf{x}_s \\ &= \mathbf{x}_s - \left[\frac{\mathbf{x}_t \cdot \mathbf{x}_s}{\mathbf{x}_t \cdot \mathbf{x}_t}\right].\end{aligned} \tag{D.5}$$

Finally,

$$\mathbf{e}_2 = \frac{\mathbf{u}_2}{|\mathbf{u}_2|}. \tag{D.6}$$

Thus, the velocity at the free surface can now be written as

$$\mathbf{v}_f = \phi_t \mathbf{x}_t + b\mathbf{e}_2 + \phi_n \mathbf{n}_f. \tag{D.7}$$

As can be seen from equation (D.7) in order to fully calculate the velocity the coefficient b must be evaluated. A related piece of information is the projection of $\mathbf{v}_f$ onto $\mathbf{x}_s$. This must give

$$\mathbf{v}_f \cdot \mathbf{x}_s = \phi_s. \tag{D.8}$$

Now, substituting equation (D.7) into equation (D.8) gives

$$\begin{aligned} \mathbf{v}_f \cdot \mathbf{x}_s &= (\phi_t \mathbf{x}_t + b\mathbf{e}_2 + \phi_n \mathbf{n}_f) \cdot \mathbf{x}_s \\ \Rightarrow \qquad \phi_s &= (\phi_t \mathbf{x}_t + b\mathbf{e}_2 + \phi_n \mathbf{n}_f) \cdot \mathbf{x}_s \\ &= \phi_t \left(\mathbf{x}_t \cdot \mathbf{x}_s\right) + b\left(\mathbf{e}_2 \cdot \mathbf{x}_s\right) + \phi_n \left(\mathbf{n}_f \cdot \mathbf{x}_s\right) \\ &= \phi_t \left(\mathbf{x}_t \cdot \mathbf{x}_s\right) + b\left(\mathbf{e}_2 \cdot \mathbf{x}_s\right) \end{aligned} \tag{D.9}$$

as $(\mathbf{n}_f \cdot \mathbf{x}_s) = 0$. Thus,

$$b = \frac{\phi_s - \phi_t \left(\mathbf{x}_t \cdot \mathbf{x}_s\right)}{\left(\mathbf{e}_2 \cdot \mathbf{x}_s\right)}. \tag{D.10}$$

Hence, all of the components of $\mathbf{v}_f$ are given by equation (D.7) and equation (D.10). What remains is to apply the boundary condition described by equation (D.1). As the channel walls are vertical $\mathbf{n}_l = (0, 1, 0)$ and thus the y component of equation (D.7) must be set to zero. Thus

$$(\mathbf{v}_f)_y = (\phi_t (\mathbf{x}_t)_y + b(\mathbf{e}_2)_y + \phi_n (\mathbf{n}_f)_y) = 0 \tag{D.11}$$

$$\Rightarrow \qquad b(\mathbf{e}_2)_y + \phi_n (\mathbf{n}_f)_y = 0 \tag{D.12}$$

where $(\ )_y$ implies the y component of the vector. In order to include equation (D.11) within the linear system it must be discretised. For this, define a 5 node differencing scheme such as in figure D.2. Node 1 is the node at which $\mathbf{v}_f$ is to be evaluated. A central differencing scheme is applied in the $\mathbf{x}_t$ direction, but because the node lies at the surface edge an 'unbalanced' differencing scheme must be used in the $\mathbf{x}_s$ direction.

In figure D.2 nodes 6,7,1,8 and 9 are used in the $\mathbf{x}_t$ direction and nodes 1,2,3,4 and 5 are used in the $\mathbf{x}_s$ direction. Thus, the differencing schemes the derivations of ϕ are

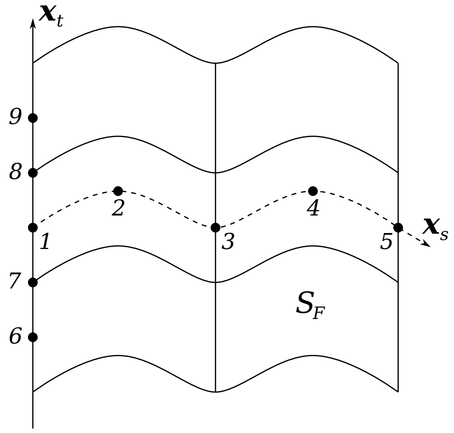


Figure D.2: Five node differentiation scheme for a free surface point coincident to the channel wall in the s and t parametric directions.

written as

$$\phi_s|_1 = \approx c_1\phi_1 + c_2\phi_2 + c_3\phi_3 + c_4\phi_4 + c_5\phi_5, \tag{D.13}$$

$$\phi_t|_1 = \approx d_6\phi_6 + d_7\phi_7 + d_1\phi_1 + c_8\phi_8 + c_9\phi_9, \tag{D.14}$$

where c and d are known coefficients (see Forehand, 1998, appendix D) as are all the values of ϕ in equation (D.13) apart from ϕ_1. All of the values of ϕ on the free surface/wall boundary are to be corrected and are thus unknowns. Finally, substituting equations (D.10), (D.13) and (D.14) into equation (D.11) produces

$$\begin{aligned}(c_1 - (\mathbf{x}_t \cdot \mathbf{x}_s)\, d_1)\, \phi_1 - (\mathbf{x}_t \cdot \mathbf{x}_s)\, (d_6\phi_6 + d_7\phi_7 + d_8\phi_8 + d_9\phi_9) \\ + \frac{(\mathbf{n}_f)_y\, (\mathbf{e}_2 \cdot \mathbf{x}_s)}{(\mathbf{e}_2)_y}\phi_{n,1} = -\left(c_2\phi_2 + c_3\phi_3 + c_4\phi_4 + c_5\phi_5\right).\end{aligned} \tag{D.15}$$

To implement equation (D.15) the explicit enforcement of continuity of ϕ on the side wall double node, $I2$, is relaxed and the equation for this node is replaced by equation (D.15). The continuity of ϕ is then re-enforced once the solution is computed.

Appendix E

Transformations of the Coincident Galerkin Integral

E.1 Application of the First Polar Transformation to the Distance Vector

The first section of this appendix demonstrates the effect of the polar transformation defined by equations (4.12) and (4.13) on the distance vector **R**, defined by equation (4.8). Begin by considering the triangle shape functions, ψ_l and ψ_j, as two vectors such that

$$\psi_l = \begin{pmatrix} -\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2} \\ \\ \frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2} \\ \\ \frac{1}{3}\sqrt{3}t \end{pmatrix} \tag{E.1}$$

and

$$\psi_j = \begin{pmatrix} -\frac{1}{2}s^* - \frac{1}{6}\sqrt{3}t^* + \frac{1}{2} \\ \\ \frac{1}{2}s^* - \frac{1}{6}\sqrt{3}t^* + \frac{1}{2} \\ \\ \frac{1}{3}\sqrt{3}t^* \end{pmatrix}. \tag{E.2}$$

Now define the three corner points of the triangular element $p_{1,2,3}$ as

$$p_1 = \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix}, \qquad p_2 = \begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix} \qquad p_3 = \begin{pmatrix} x_3 \\ y_3 \\ z_3 \end{pmatrix}. \tag{E.3}$$

The point P lying on element Σ_P is given by $\sum_{l=1}^{3} p_l \psi_l$ such that

$$P(s,t) = \\ \begin{pmatrix} \left(-\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) x_1 + \left(\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) x_2 + \left(\frac{1}{3}\sqrt{3}t\right) x_3 \\ \left(-\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) y_1 + \left(\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) y_2 + \left(\frac{1}{3}\sqrt{3}t\right) y_3 \\ \left(-\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) z_1 + \left(\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) z_2 + \left(\frac{1}{3}\sqrt{3}t\right) z_3 \end{pmatrix}. \tag{E.4}$$

Also, $P + \epsilon\mathbf{N}$ becomes

$$\begin{pmatrix} \left(-\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) x_1 + \left(\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) x_2 + \left(\frac{1}{3}\sqrt{3}t\right) x_3 + \epsilon N_1 \\ \\ \left(-\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) y_1 + \left(\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) y_2 + \left(\frac{1}{3}\sqrt{3}t\right) y_3 + \epsilon N_2 \\ \\ \left(-\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) z_1 + \left(\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) z_2 + \left(\frac{1}{3}\sqrt{3}t\right) z_3 + \epsilon N_3 \end{pmatrix}. \tag{E.5}$$

Similarly, for the point $\mathbf{Q}$,

$$Q(s^*,t^*) = \\ \begin{pmatrix} \left(-\frac{1}{2}s^* - \frac{1}{6}\sqrt{3}t^* + \frac{1}{2}\right) x_1 + \left(\frac{1}{2}s^* - \frac{1}{6}\sqrt{3}t^* + \frac{1}{2}\right) x_2 + \left(\frac{1}{3}\sqrt{3}t^*\right) x_3 \\ \left(-\frac{1}{2}s^* - \frac{1}{6}\sqrt{3}t^* + \frac{1}{2}\right) y_1 + \left(\frac{1}{2}s^* - \frac{1}{6}\sqrt{3}t^* + \frac{1}{2}\right) y_2 + \left(\frac{1}{3}\sqrt{3}t^*\right) y_3 \\ \left(-\frac{1}{2}s^* - \frac{1}{6}\sqrt{3}t^* + \frac{1}{2}\right) z_1 + \left(\frac{1}{2}s^* - \frac{1}{6}\sqrt{3}t^* + \frac{1}{2}\right) z_2 + \left(\frac{1}{3}\sqrt{3}t^*\right) z_3 \end{pmatrix}. \tag{E.6}$$

Applying the transformation defined by equations (4.12) and (4.13) to equation (E.6) and then subtracting equation (E.5), $\mathbf{R}^+$ is now given by

$$\mathbf{R}^+ = Q - (P + \epsilon\mathbf{N}) = \begin{pmatrix} (A_1\cos(\theta) + B_1\sin(\theta))\,\rho - \epsilon N_1 \\ \\ (A_2\cos(\theta) + B_2\sin(\theta))\,\rho - \epsilon N_2 \\ \\ (A_3\cos(\theta) + B_3\sin(\theta))\,\rho - \epsilon N_3 \end{pmatrix} \tag{E.7}$$

where, for convenience, the following quantities are defined as

$$A_1 = \left(-\frac{1}{2}x_1 + \frac{1}{2}x_2\right), \tag{E.8}$$

$$B_1 = \left(-\frac{1}{6}\sqrt{3}x_1 - \frac{1}{6}\sqrt{3}x_2 + \frac{1}{3}\sqrt{3}x_3\right), \tag{E.9}$$

with A_2, A_3, B_2 and B_3 having identical form baring the 'x's being exchanged for 'y's and 'z's respectively. Equation (E.7) can now be written

$$\mathbf{R}^+ = \begin{pmatrix} (A_1\cos(\theta) + B_1\sin(\theta))\,\rho - \epsilon N_1 \\ \\ (A_2\cos(\theta) + B_2\sin(\theta))\,\rho - \epsilon N_2 \\ \\ (A_3\cos(\theta) + B_3\sin(\theta))\,\rho - \epsilon N_3 \end{pmatrix}. \tag{E.10}$$

Similarly, the alternative limit defined by $\mathbf{R}^- = Q - (P - \epsilon\mathbf{N})$ is given by

$$\mathbf{R}^- = \begin{pmatrix} (A_1\cos(\theta) + B_1\sin(\theta))\,\rho + \epsilon N_1 \\ \\ (A_2\cos(\theta) + B_2\sin(\theta))\,\rho + \epsilon N_2 \\ \\ (A_3\cos(\theta) + B_3\sin(\theta))\,\rho + \epsilon N_3 \end{pmatrix}. \tag{E.11}$$

This can be further simplified giving

$$\mathbf{R}^{\pm} = \begin{pmatrix} a_1\rho \mp \epsilon N_1 \\ \\ a_2\rho \mp \epsilon N_2 \\ \\ a_3\rho \mp \epsilon N_3 \end{pmatrix}. \tag{E.12}$$

The shape function $\psi_j(s^*, t^*)$ must also be transformed. Applying the transformation defined by equations (4.12) and (4.13) to equation (E.2), the coefficients take the form

$$\psi_j(s, t, \rho) = \begin{pmatrix} c_{1,0}(s,t) + c_{1,1}(\theta)\rho \\ \\ c_{2,0}(s,t) + c_{2,1}(\theta)\rho \\ \\ c_{3,0}(s,t) + c_{3,1}(\theta)\rho \end{pmatrix} \tag{E.13}$$

where

$$\begin{aligned} c_{1,0} &= -\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}, & c_{1,1} &= -\frac{1}{2}\cos(\theta) - \frac{1}{6}\sqrt{3}\sin(\theta), \\ c_{2,0} &= \frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}, & c_{2,1} &= \frac{1}{2}\cos(\theta) - \frac{1}{6}\sqrt{3}\sin(\theta), \\ c_{3,0} &= \frac{1}{3}\sqrt{3}t, & c_{3,1} &= \frac{1}{3}\sqrt{3}\sin(\theta). \end{aligned} \tag{E.14}$$

E.2 Variable Substitution and Second Polar Transformation

This section demonstrates various impacts from the substitution and second polar transformation demonstrated in Gray *et al.* (2004a,b). These techniques are initially demonstrated using the coefficients $c_{j,1}$ and a_k. Restating $c_{j,1}$, it's form after the first polar transform is

$$c_{j,1}(\theta) = \begin{pmatrix} -\frac{1}{2}\cos(\theta) - \frac{1}{6}\sqrt{3}\sin(\theta) \\ \\ \frac{1}{2}\cos(\theta) - \frac{1}{6}\sqrt{3}\sin(\theta) \\ \\ \frac{1}{3}\sqrt{3}\sin(\theta) \end{pmatrix}. \tag{E.15}$$

Subsequently, the substitution given by equation (4.24) is performed followed by the polar transformation defined by equation (4.25). The effect of equations (4.24) and (4.25) on $\sin(\theta)$ is shown to be

$$\begin{aligned} \sin(\theta) &= \sin\left(-\frac{\pi}{2} + \tan^{-1}\left(\frac{\tau - s}{t}\right)\right) \\ &= -\left(1 + \frac{(\tau - s)^2}{t^2}\right)^{-\frac{1}{2}} \\ &= -\left(1 + \frac{((\Lambda\cos(\Psi) + s) - s)^2}{(\Lambda\sin(\Psi))^2}\right)^{-\frac{1}{2}} \\ &= -\left(1 + \frac{cos^2(\Psi)}{\sin^2(\Psi)}\right)^{-\frac{1}{2}} \\ &= -\sin(\Psi). \end{aligned} \tag{E.16}$$

Similarly $\cos(\theta) = \cos(\Psi)$. Thus, equation (E.15) is transformed such that

$$c_{j,1}(\Psi) = \begin{pmatrix} -\frac{1}{2}\cos(\Psi) + \frac{1}{6}\sqrt{3}\sin(\Psi) \\ \\ \frac{1}{2}\cos(\Psi) + \frac{1}{6}\sqrt{3}\sin(\Psi) \\ \\ -\frac{1}{3}\sqrt{3}\sin(\Psi) \end{pmatrix}, \tag{E.17}$$

and the effect on a_k is

$$a_k(\theta) = a_k(\Psi) = A_k\cos(\Psi) - B_k\sin(\Psi). \tag{E.18}$$

maintaining the values of the coefficients defined in section E.1.

The transformation given by equation (4.25) will impact upon the shape functions of the P element, ψ_l. Rewriting equation (E.1) in terms of the new coordinate system produces

$$\psi_l(s,\Lambda) = \begin{pmatrix} -\frac{1}{2}s + \frac{1}{2} - \frac{1}{6}\sqrt{3}\Lambda\sin(\Psi) \\ \\ \frac{1}{2}s + \frac{1}{2} - \frac{1}{6}\sqrt{3}\Lambda\sin(\Psi) \\ \\ \frac{1}{3}\sqrt{3}\Lambda\sin(\Psi) \end{pmatrix} = \begin{pmatrix} k_{1,0} + k_{1,1}\Lambda \\ \\ k_{2,0} + k_{2,1}\Lambda \\ \\ k_{3,0} + k_{3,1}\Lambda \end{pmatrix}. \tag{E.19}$$

Finally, to demonstrate the effect of the change of variables followed by the transformation on the variables of integration, observe that

$$\frac{\mathrm{d}\,\theta}{\mathrm{d}\tau} = \frac{t}{t^2 + \tau^2 - 2\,\tau\,s + s^2}. \tag{E.20}$$

Applying the transformation given by equation (4.25) the left hand side will become

$$\frac{\Lambda\,\sin(\Psi)}{\Lambda^2\,(\sin(\Psi))^2 + \Lambda^2\,(\cos(\Psi))^2} = \frac{\sin(\Psi)}{\Lambda}\,. \tag{E.21}$$

Therefore,

$$dt\;d\theta = \frac{\sin(\Psi)}{\Lambda}\;dt\;d\tau = \sin(\Psi)\;d\Psi\;d\Lambda. \tag{E.22}$$

To allow the limit of $\epsilon \to 0$ in equation (4.41) the substitution and transformation described by equations (4.24) and (4.25) must be applied once more. The result for $c_{j,0}$ is as follows:

$$c_{j,0}(s,\Lambda) = \begin{pmatrix} -\frac{1}{2}\,s + \frac{1}{2} - \frac{1}{6}\sqrt{3}\,\Lambda\sin(\Psi) \\ \\ \frac{1}{2}\,s + \frac{1}{2} - \frac{1}{6}\sqrt{3}\,\Lambda\sin(\Psi) \\ \\ \frac{1}{3}\,\Lambda\sin(\Psi)\,\sqrt{3} \end{pmatrix} \equiv \psi_l(s,\Lambda). \tag{E.23}$$

Appendix F

Reduction of Coincident Integrals to One Dimension

This appendix describes the reduction of equations (4.20) and (4.21) and their additional outer integrals to one dimensional integrals which can be solved numerically. This is achieved using the technique presented by Gray *et al.* (2006) which swaps the last two variables of integration in order a provide an innermost integral which can be solved analytically.

F.1 Integrals for Equation (4.21)

Equation (4.35) represents the result of equation (4.21), in addition to its outer s and t integrals, reduced to two integrals and then having the s integration variable swapped to become innermost. Before beginning to evaluate equation (4.35) the the effect on the limits of Ψ on splitting to integration domain at $s = 0$ must be considered. In addition, the domain of integration (derived in appendix E) must be transformed. Thus, when $s > 0$ observe that

$$\begin{aligned}\Psi_1 = \tan^{-1}\left(\frac{\sqrt{3}\,(1-|s|)}{1-s}\right) &= \tan^{-1}\left(\frac{\sqrt{3}\,(1-s)}{1-s}\right) \\ &= \tan^{-1}\left(\sqrt{3}\right) \\ &= \frac{\pi}{3}.\end{aligned} \tag{F.1}$$

Similarly when $s < 0$,

$$\Psi_2 = \frac{2\pi}{3}. \tag{F.2}$$

Thus, with $s > 0$ the domain of integration is

$$\begin{aligned} 0 < \Psi < \pi/3; \quad & \Lambda_L = (1-s)/\cos(\Psi), \\ \pi/3 < \Psi < \Psi_s; \quad & \Lambda_L = \sqrt{3}(1-s)/\sin(\Psi), \\ \Psi_s < \Psi < \pi; \quad & \Lambda_L = -(1+s)/\cos(\Psi), \end{aligned} \tag{F.3}$$

where

$$\Psi_s = \pi - \tan^{-1}\left(\frac{\sqrt{3}\,(1-s)}{1+s}\right). \tag{F.4}$$

When $s < 0$ the domain of integration is

$$\begin{aligned} 0 < \Psi < \Psi s; \quad & \Lambda_L = (1-s)/\cos(\Psi), \\ \Psi_s < \Psi < 2\pi/3; \quad & \Lambda_L = \sqrt{3}(1-s)/\sin(\Psi), \\ 2\pi/3 < \Psi < \pi; \quad & \Lambda_L = -(1+s)/\cos(\Psi), \end{aligned} \tag{F.5}$$

where

$$\Psi_s = \tan^{-1}\left(\frac{\sqrt{3}\,(1+s)}{1-s}\right). \tag{F.6}$$

Considering each triangle in turn, the effect of the resulting integration on each coefficient is analysed. Thus, for the triangle defined by $0 < \Psi < \Psi_1$ when $s < 0$ it follows that

$$\int_{-1}^{0} ds \int_0 \Psi_1 d\Psi = \int_0^{\pi/3} d\Psi \int_{\alpha}^{0} ds, \tag{F.7}$$

where α is defined by equation (4.36).

Starting with the coefficient $k_{1,0}$, applying the new limits of integration to each kernel produces

$$\begin{aligned} \int_{\alpha}^{0} \Lambda_1 k_{1,0}\, ds &= \int_{\alpha}^{0} \left(\frac{1-s}{\cos(\Psi)}\right)\left(-\frac{1}{2}s + \frac{1}{2}\right) ds \\ &= \int_{\alpha}^{0} \frac{1}{2}\,\frac{(s-1)^2}{\cos(\Psi)} ds \\ &= \left[\frac{1}{6}\,\frac{(s-1)^3}{\cos(\Psi)}\right]_{\alpha}^{0} \\ &= -\frac{1}{6}\,\frac{\beta^3 + 3\,\beta^2 + 3\,\beta - 7}{(\beta+1)^3\cos(\Psi)}. \end{aligned} \tag{F.8}$$

Similarly, the result for $k_{2,0}$ is

$$\begin{aligned}
\int_{\alpha}^{0} \Lambda_1 k_{2,0}\, ds &= \int_{\alpha}^{0} \left(\frac{1-s}{\cos(\Psi)}\right)\left(\frac{1}{2}s+\frac{1}{2}\right) ds \\
&= \int_{\alpha}^{0} -\frac{1}{2}\,\frac{-1+s^2}{\cos(\Psi)} ds \\
&= \left[\frac{1}{6} - \frac{1}{6}\,\frac{s\left(-3+s^2\right)}{\cos(\Psi)}\right]_{\alpha}^{0} \\
&= -\frac{1}{3}\,\frac{(\beta-1)\left(\beta^2+4\,\beta+1\right)}{(\beta+1)^3\cos(\psi)}.
\end{aligned} \tag{F.9}$$

To illustrate that the results of equation (F.8) and equation (F.9) are non-singular over the domain of the first triangle, the function $f(\Psi) = (1+\beta)^3 \cos(\Psi)$ is plotted over the range of the first triangle $(0 < \Psi < \pi/3)$ in figure F.1. It can be seen that the equations will have no singularities over the given range of Ψ.

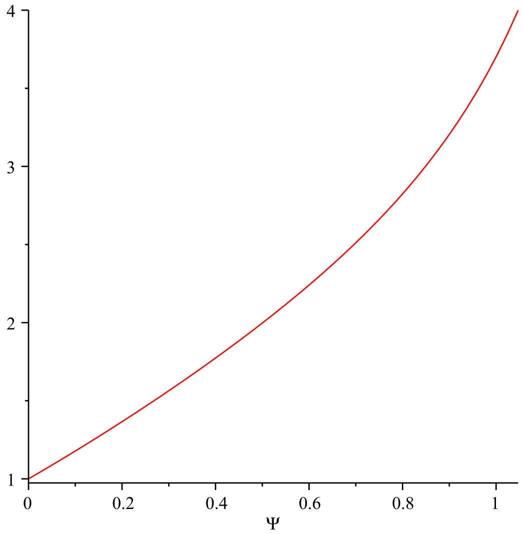


Figure F.1: $f(\Psi) = \cos(\Psi)\left(1+3^{-1/2}\tan\Psi\right)^3 \quad \forall \quad 0 < \Psi < \pi/3.$

Moving on, $k_{3,0}$ is zero so now consider $k_{1,1}$, the result being

$$\begin{aligned}\int_\alpha^0 \Lambda_1^2 k_{1,1}\,ds &= \int_\alpha^0 \left(\frac{1-s}{\cos(\Psi)}\right)^2 \left(-\frac{1}{6}\sqrt{3}\sin(\Psi)\right) ds\\ &= \int_\alpha^0 -\frac{1}{6}\,\frac{(-1+\eta)^2\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2} ds\\ &= \left[-\frac{1}{18}\,\frac{(-1+\eta)^3\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2}\right]_\alpha^0\\ &= \frac{1}{18}\,\frac{\sqrt{3}\sin(\Psi)\left(\beta^3+3\beta^2+3\beta-7\right)}{(\beta+1)^3(\cos(\Psi))^2}.\end{aligned} \tag{F.10}$$

The above result is identical for $k_{2,1}$.

The outcome for $k_{3,1}$ is

$$\begin{aligned}\int_\alpha^0 \Lambda_1^2 k_{3,1}\,ds &= \int_\alpha^0 \left(\frac{1-s}{\cos(\Psi)}\right)^2 \left(\frac{1}{3}\sqrt{3}\sin(\Psi)\right) ds\\ &= \int_\alpha^0 \frac{1}{3}\,\frac{(-1+\eta)^2\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2} ds\\ &= \left[\frac{1}{9}\,\frac{(-1+\eta)^3\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2}\right]_\alpha^0\\ &= -\frac{1}{9}\,\frac{\sqrt{3}\sin(\Psi)\left(\beta^3+3\beta^2+3\beta-7\right)}{(\beta+1)^3(\cos(\Psi))^2}.\end{aligned} \tag{F.11}$$

As can be seen from figure F.2 the denominators of the result of equations (F.10) and (F.11) will not be singular over the range of the first triangle. Thus, the integration over first triangle for $s<0$ is now reduced to a single integral in Ψ. Now consider the domain of the first triangle with $s>0$.

As can be seen from figure 4.4(b) the reordering of integration is

$$\int_0^1 ds \int_0^{\pi/3} d\Psi = \int_0^{\pi/3} d\Psi \int_0^1 ds. \tag{F.12}$$

Similar calculations to those for $s<0$ must now be completed for $s>0$. The results of $k_{1,0}$ and $k_{2,0}$ integrations are

$$\begin{aligned}\int_0^1 \Lambda_1 k_{1,0}\,ds &= \int_0^1 \frac{1}{2}\,\frac{(s-1)^2}{\cos(\Psi)} ds\\ &= \frac{1}{6\cos(\Psi)},\end{aligned} \tag{F.13}$$

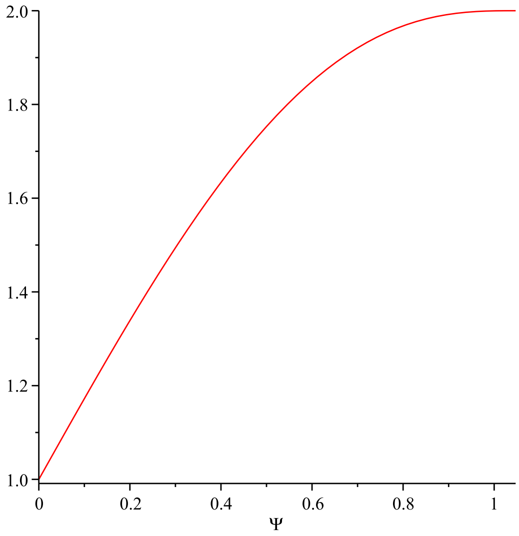


Figure F.2: $f(\Psi) = \cos^2(\Psi)\left(1 + 3^{-1/2}\tan\Psi\right)^3 \qquad \forall \quad 0 < \Psi < \pi/3.$

$$\begin{aligned}\int_0^1 \Lambda_1 k_{2,0}\, ds &= \int_0^1 -\frac{1}{2}\,\frac{-1+s^2}{\cos(\Psi)}ds \\ &= \frac{1}{3\cos(\Psi)}.\end{aligned} \tag{F.14}$$

As $\cos(\Psi)$ is positive over $0 < \Psi < \pi/3$ equations (F.13) and (F.14) exhibit no singularities.

Moving on to the $k_{1,1}$ and $k_{3,1}$ terms (recalling that $k_{3,0}$ is zero and that $k_{2,1}$ is identical to $k_{1,1}$) the results are

$$\begin{aligned}\int_0^1 \Lambda_1^2 k_{1,1}\, ds &= \int_0^1 -\frac{1}{6}\,\frac{(-1+\eta)^2\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2}ds \\ &= -\frac{1}{18}\,\frac{\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2},\end{aligned} \tag{F.15}$$

$$\begin{aligned}\int_0^1 \Lambda_1^2 k_{3,1}\, ds &= \int_0^1 \frac{1}{3}\,\frac{(-1+\eta)^2\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2}ds \\ &= \frac{1}{9}\,\frac{\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2}.\end{aligned} \tag{F.16}$$

Once more, the denominators of the results of equations (F.15) and (F.16) are non zero over the range $0 < \Psi < \pi/3$.

The integral over the domain of the second triangle can now be analysed. Note that from figure 4.4(b) both $s < 0$ and $s > 0$ require that the domain be split into two.

Starting with the $s < 0$ case, the reordering of integration produces

$$\int_{-1}^{0} ds \int_{\Psi_1}^{2\pi/3} d\Psi = \int_{0}^{\pi/3} d\Psi \int_{-1}^{\alpha} ds + \int_{\pi/3}^{2\pi/3} d\Psi \int_{-1}^{0} ds. \tag{F.17}$$

Beginning with the first part, examine the results for each coefficient. Looking at $k_{1,0}$ and $k_{2,0}$, the results are:

$$\begin{aligned} \int_{-1}^{\alpha} \Lambda_2 k_{1,0}\, ds &= \int_{-1}^{\alpha} -\frac{1}{2} \frac{\sqrt{3}\left(-1+s^2\right)}{\sin(\Psi)} ds \\ &= \frac{2}{3} \frac{\sqrt{3}\beta^2 (\beta+3)}{(\beta+1)^3 \sin(\Psi)}, \end{aligned} \tag{F.18}$$

$$\begin{aligned} \int_{-1}^{\alpha} \Lambda_2 k_{2,0}\, ds &= \int_{-1}^{\alpha} \frac{1}{2} \frac{\sqrt{3}\left(1+s\right)^2}{\sin(\Psi)} ds \\ &= \frac{4}{3} \frac{\sqrt{3}\beta^3}{(\beta+1)^3 \sin(\Psi)}. \end{aligned} \tag{F.19}$$

Now examining the $k_{1,1}$ and $k_{3,1}$ terms observe that

$$\begin{aligned} \int_{-1}^{\alpha} \Lambda_2^2 k_{1,1}\, ds &= \int_{-1}^{\alpha} -\frac{1}{2} \frac{\left(1+s\right)^2\sqrt{3}}{\sin(\Psi)} ds \\ &= -\frac{4}{3} \frac{\beta^3\sqrt{3}}{(\beta+1)^3 \sin(\Psi)}, \end{aligned} \tag{F.20}$$

$$\begin{aligned} \int_{-1}^{\alpha} \Lambda_2^2 k_{3,1}\, ds &= \int_{-1}^{\alpha} \frac{\left(1+s\right)^2\sqrt{3}}{\sin(\Psi)} ds \\ &= \frac{8}{3} \frac{\beta^3\sqrt{3}}{(\beta+1)^3 \sin(\Psi)}. \end{aligned} \tag{F.21}$$

The $\sin(\Psi)$ in the denominators of equations (F.18), (F.19), (F.20) and (F.21) can be cancelled by the loose $\sin(\Psi)$ in the outer integrand of equation (4.35). The remaining denominator is plotted in figure F.3, over the range $0 < \Psi < \pi/3$, and it can be seen that the resulting integrals will not produce singularities in the remaining Ψ integration.

The second part of the integration for triangle two is over the range $-1 < s < 0$. The results for $k_{1,0}$ and $k_{2,0}$ are

$$\begin{aligned} \int_{-1}^{0} \Lambda_2 k_{1,0}\, ds &= \int_{-1}^{0} -\frac{1}{2} \frac{\sqrt{3}\left(-1+s^2\right)}{\sin(\Psi)} ds \\ &= \frac{1}{3} \frac{\sqrt{3}}{\sin(\Psi)}, \end{aligned} \tag{F.22}$$

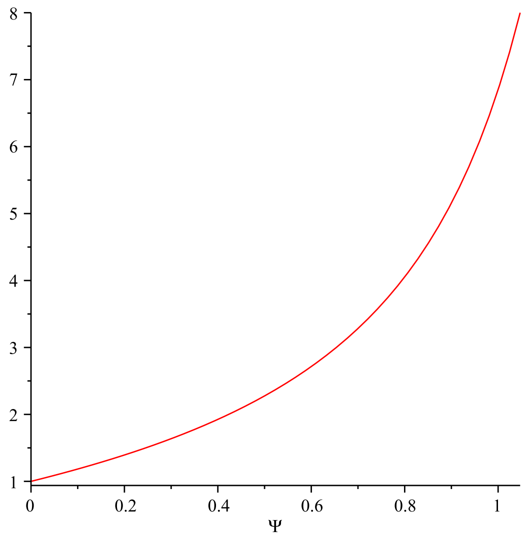


Figure F.3: $f(\Psi) = \left(1 + 3^{-1/2}\tan(\Psi)\right)^3 \qquad \forall \quad 0 < \Psi < \pi/3.$

$$\begin{aligned}\int_{-1}^{0} \Lambda_2 k_{2,0}\, ds &= \int_{-1}^{0} \frac{1}{2}\frac{\sqrt{3}\,(1+s)^2}{\sin(\Psi)} ds \\ &= \frac{1}{6}\frac{\sqrt{3}}{\sin(\Psi)}.\end{aligned} \tag{F.23}$$

Now examining the $k_{1,1}$ and $k_{3,1}$ terms observe that

$$\begin{aligned}\int_{-1}^{0} \Lambda_2^2 k_{1,1}\, ds &= \int_{-1}^{0} -\frac{1}{2}\frac{(1+s)^2\sqrt{3}}{\sin(\Psi)} ds \\ &= -\frac{1}{6}\frac{\sqrt{3}}{\sin(\Psi)},\end{aligned} \tag{F.24}$$

$$\begin{aligned}\int_{-1}^{0} \Lambda_2^2 k_{3,1}\, ds &= \int_{-1}^{0} \frac{(1+s)^2\sqrt{3}}{\sin(\Psi)} ds \\ &= \frac{1}{3}\frac{\sqrt{3}}{\sin(\Psi)}.\end{aligned} \tag{F.25}$$

For equations (F.22)–(F.25) the $\sin(\Psi)$ in the denominator will simply cancel with the loose $\sin(\Psi)$ indicating that there will be no singularities from these equations.

Moving onto the $s > 0$ part of the second triangle the reordering of the integration produces

$$\int_0^1 ds \int_{\Psi_1}^{2\pi/3} d\Psi = \int_{\pi/3}^{2\pi/3} d\Psi \int_0^1 ds + \int_{2\pi/3}^{\pi} d\Psi \int_{\alpha}^{1} ds. \tag{F.26}$$

The results of the $k_{1,0}$ and $k_{2,0}$ integrations for $0 < s < 1$ are

$$\begin{aligned}\int_0^1 \Lambda_2 k_{1,0}\, ds &= \int_0^1 \frac{1}{2} \frac{\sqrt{3}\,(-1+s)^2}{\sin(\Psi)} ds \\ &= \frac{1}{3} \frac{\sqrt{3}}{\sin(\Psi)},\end{aligned} \tag{F.27}$$

$$\begin{aligned}\int_0^1 \Lambda_2 k_{2,0}\, ds &= \int_0^1 -\frac{1}{2} \frac{\sqrt{3}\,(-1+s^2)}{\sin(\Psi)} ds \\ &= \frac{1}{3} \frac{\sqrt{3}}{\sin(\Psi)}.\end{aligned} \tag{F.28}$$

Moving on to the $k_{1,1}$ and $k_{3,1}$ terms (recalling that $k_{3,0}$ is zero and that $k_{2,1}$ is identical to $k_{1,1}$) the results are

$$\begin{aligned}\int_0^1 \Lambda_2^2 k_{1,1}\, ds &= \int_0^1 -\frac{1}{2} \frac{(-1+s)^2\sqrt{3}}{\sin(\Psi)} ds \\ &= -\frac{1}{6} \frac{\sqrt{3}}{\sin(\Psi)},\end{aligned} \tag{F.29}$$

$$\begin{aligned}\int_0^1 \Lambda_2^2 k_{3,1}\, ds &= \int_0^1 \frac{(-1+s)^2\sqrt{3}}{\sin(\Psi)} ds \\ &= \frac{1}{3} \frac{\sqrt{3}}{\sin(\Psi)}.\end{aligned} \tag{F.30}$$

Similarly, the $\sin(\Psi)$ in the denominators of equations (F.27)–(F.34) will cause no difficulties.

Thus, it remains to show the results for the integrals over the range, $\alpha \leq s < 1$. Note that for $s > 0$, α is now provided by equation (4.37). Subsequently, the results for $k_{1,0}$ and $k_{2,0}$ are

$$\begin{aligned}\int_\alpha^1 \Lambda_2 k_{1,0}\, ds &= \int_\alpha^1 \frac{1}{2} \frac{\sqrt{3}\,(-1+s)^2}{\sin(\Psi)} ds \\ &= \frac{4}{3} \frac{\sqrt{3}\beta^3}{(\beta-1)^3 \sin(\Psi)},\end{aligned} \tag{F.31}$$

$$\begin{aligned}\int_\alpha^1 \Lambda_2 k_{2,0}\, ds &= \int_\alpha^1 -\frac{1}{2} \frac{\sqrt{3}\,(-1+s^2)}{\sin(\Psi)} ds \\ &= \frac{2}{3} \frac{\sqrt{3}\beta^2(\beta-3)}{(\beta-1)^3 \sin(\Psi)}.\end{aligned} \tag{F.32}$$

Executing the analytical integration for the $k_{1,1}$ and $k_{3,1}$ terms then produces

$$\begin{aligned}\int_{\alpha}^{1} \Lambda_2^2 k_{1,1}\, ds &= \int_{\alpha}^{1} -\frac{1}{2}\,\frac{(-1+s)^2\sqrt{3}}{\sin(\Psi)}ds \\ &= -\frac{4}{3}\,\frac{\beta^3\sqrt{3}}{(\beta-1)^3\sin(\Psi)},\end{aligned} \tag{F.33}$$

$$\begin{aligned}\int_{\alpha}^{1} \Lambda_2^2 k_{3,1}\, ds &= \int_{\alpha}^{1} \frac{(-1+s)^2\sqrt{3}}{\sin(\Psi)}ds \\ &= \frac{8}{3}\,\frac{\beta^3\sqrt{3}}{(\beta-1)^3\sin(\Psi)}.\end{aligned} \tag{F.34}$$

Once again the $\sin(\Psi)$ in the denominator of equations (F.31)–(F.34) can be cancelled. Figure F.4 shows the graph of the remaining denominator plotted over the range $2\pi/3 < \Psi < \pi$. It can be seen to be non-zero over this range and therefore causes no difficulties in the subsequent Ψ integration.

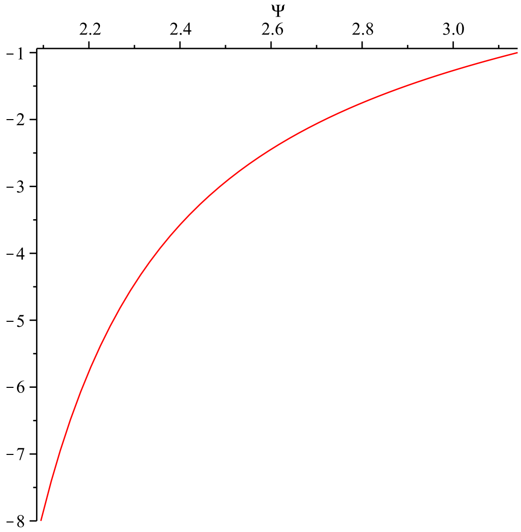


Figure F.4: $f(\Psi) = \left(3^{-1/2}\tan(\Psi) - 1\right)^3 \quad \forall \quad 2\pi/3 < \Psi < \pi.$

For the third and final triangle the required reordering for $s < 0$ is

$$\int_{-1}^{0} ds \int_{2\pi/3}^{\pi} d\Psi = \int_{2\pi/3}^{\pi} d\Psi \int_{-1}^{0} ds. \tag{F.35}$$

Executing the integration for triangle three, over the range $-1 < s < 0$, the results for $k_{1,0}$ and $k_{2,0}$ are

$$\begin{aligned}\int_{-1}^{0} \Lambda_3 k_{1,0}\, ds &= \int_{-1}^{0} \frac{1}{2}\,\frac{-1+s^2}{\cos(\Psi)} ds\\ &= -\frac{1}{3}\,(\cos(\Psi))^{-1}\,,\end{aligned} \tag{F.36}$$

$$\begin{aligned}\int_{-1}^{0} \Lambda_3 k_{2,0}\, ds &= \int_{-1}^{0} -\frac{1}{2}\,\frac{(1+s)^2}{\cos(\Psi)} ds\\ &= -\frac{1}{6}\,(\cos(\Psi))^{-1}\,.\end{aligned} \tag{F.37}$$

Now examining the $k_{1,1}$ and $k_{3,1}$ terms observe that

$$\begin{aligned}\int_{-1}^{0} \Lambda_3^2 k_{1,1}\, ds &= \int_{-1}^{0} -\frac{1}{6}\,\frac{(1+s)^2\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2} ds\\ &= -\frac{1}{18}\,\frac{\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2},\end{aligned} \tag{F.38}$$

$$\begin{aligned}\int_{-1}^{0} \Lambda_3^2 k_{3,1}\, ds &= \int_{-1}^{0} \frac{1}{3}\,\frac{(1+s)^2\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2} ds\\ &= \frac{1}{9}\,\frac{\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2}.\end{aligned} \tag{F.39}$$

As $\cos(\Psi)$ is never zero over $2\pi/3 < \Psi < \pi$ the above results cause no problems in the subsequent numerical integration.

For the $s > 0$ part of the third triangle the reordering is given by

$$\int_{0}^{1} ds \int_{\pi-\Psi 2}^{\pi} d\Psi = \int_{2\pi/3}^{\pi} d\Psi \int_{0}^{\alpha} ds. \tag{F.40}$$

The integrations over the range $0 < s \leq \alpha$ are now calculated. For $k_{1,0}$ and $k_{2,0}$,

$$\begin{aligned}\int_{0}^{\alpha} \Lambda_3 k_{1,0}\, ds &= \int_{0}^{\alpha} \frac{1}{2}\,\frac{-1+s^2}{\cos(\Psi)} ds\\ &= \frac{1}{3}\,\frac{(\beta+1)\left(\beta^2-4\,\beta+1\right)}{(\beta-1)^3\cos(\Psi)},\end{aligned} \tag{F.41}$$

$$\begin{aligned}\int_{0}^{\alpha} \Lambda_3 k_{2,0}\, ds &= \int_{0}^{\alpha} -\frac{1}{2}\,\frac{(1+s)^2}{\cos(\Psi)} ds\\ &= \frac{1}{6}\,\frac{7+\beta^3-3\,\beta^2+3\,\beta}{(\beta-1)^3\cos(\Psi)}.\end{aligned} \tag{F.42}$$

For the $k_{1,1}$ and $k_{3,1}$ terms,

$$\begin{aligned}\int_{\alpha}^{1} \Lambda_3^2 k_{1,1}\, ds &= \int_{\alpha}^{1} -\frac{1}{6}\frac{(1+s)^2\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2} ds \\ &= \frac{1}{18}\frac{\sqrt{3}\sin(\Psi)\left(7+\beta^3-3\beta^2+3\beta\right)}{(\beta-1)^3(\cos(\Psi))^2},\end{aligned} \tag{F.43}$$

$$\begin{aligned}\int_{\alpha}^{1} \Lambda_3^2 k_{3,1}\, ds &= \int_{\alpha}^{1} \frac{1}{3}\frac{(1+s)^2\sqrt{3}\sin(\Psi)}{(\cos(\Psi))^2} ds \\ &= -\frac{1}{9}\frac{\sqrt{3}\sin(\Psi)\left(7+\beta^3-3\beta^2+3\beta\right)}{(\beta-1)^3(\cos(\Psi))^2}.\end{aligned} \tag{F.44}$$

The graph of the denominator of equations (F.41) and (F.42) over the range $2\pi/3 < \Psi < \pi$ is illustrated in figure F.5. For equations (F.43) and (F.44) the graph of the denominator is illustrated in figure F.6. From both graphs it can be seen that the results from the above equations do not generate new singularities within the numerical integration.

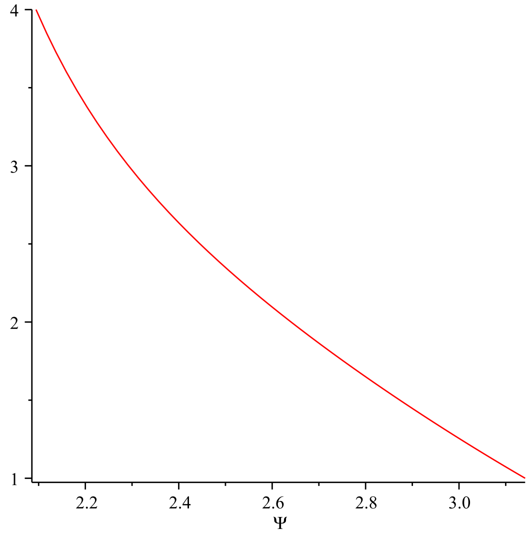


Figure F.5: $f(\Psi) = \left(3^{-1/2}\tan(\Psi) - 1\right)^3 \quad \forall \quad 2\pi/3 < \Psi < \pi.$

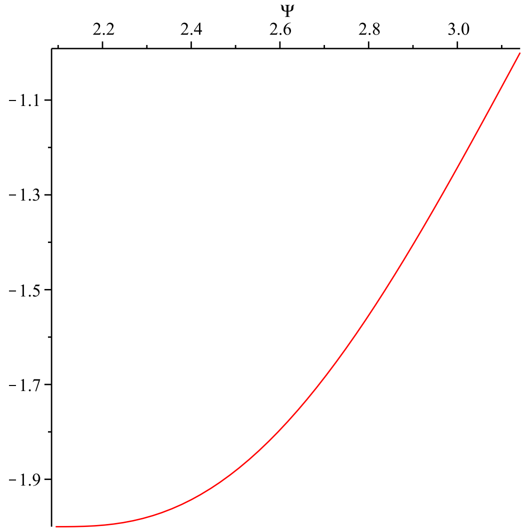


Figure F.6: $f(\Psi) = \left(3^{-1/2}\tan(\Psi) - 1\right)^3 \qquad \forall \quad 2\pi/3 < \Psi < \pi.$

F.2 Integrals for Equation (4.20)

Equation (4.48) is the result of the analytical integration of equation (4.20). To reduce the integral to one dimensional integrals which can be numerically integrated, the reordering of integration variables and final analytical integration is carried out in a similar manner to the previous section. The reordering is started, once again, with the first triangle of the three formed by the second polar transformation for which Ψ has the range $0 < \Psi < \Psi_1$. It is necessary to split the integral into $s < 0$ and $s > 0$ parts, respectively. Thus for $s < 0$ the reordering is described by equation (F.7). To examine the effect of this reordering on $k_{k,0}^2$ observe that just two terms need be considered:

$$\begin{aligned} K_1 &= k_{1,0}^2 = \frac{1}{4}(1-s)^2, \\ K_2 &= k_{2,0}^2 = \frac{1}{4}(1+s)^2. \end{aligned} \tag{F.45}$$

Consequently, the evaluated integrals are

$$\begin{aligned} \int_\alpha^0 K_1\, ds &= \int_\alpha^0 \frac{1}{4}(1-s)^2\, ds \\ &= \left[-\frac{1}{12}(1-s)^3\right]_\alpha^0 \\ &= -\frac{1}{12}\,\frac{-7+3\beta+3\beta^2+\beta^3}{(1+\beta)^3}, \end{aligned} \tag{F.46}$$

and

$$\begin{aligned}\int_{\alpha}^{0} K_2 \, ds &= \int_{\alpha}^{0} \frac{1}{4} (1+s)^2 \, ds \\ &= \left[\frac{1}{12} (1+s)^3 \right]_{\alpha}^{0} \\ &= -\frac{1}{12} \, \frac{-1 - 3\,\beta - 3\,\beta^2 + 7\,\beta^3}{(1+\beta)^3} .\end{aligned} \tag{F.47}$$

For the first triangle with $s > 0$ the reordering is given by equation (F.12). The results for this reordering are

$$\int_{0}^{1} K_1 \, ds = \left[-\frac{1}{12} (1-s)^3 \right]_{0}^{1} = \frac{1}{12} \, , \tag{F.48}$$

$$\int_{0}^{1} K_2 \, ds = \left[\frac{1}{12} (1+s)^3 \right]_{0}^{1} = \frac{7}{12} \, . \tag{F.49}$$

The reordering for the second triangle with $s < 0$ is given by equation (F.17). The results for the first $-1 < s \leq \alpha$ part are

$$\begin{aligned}\int_{-1}^{\alpha} K_1 \, ds &= \left[-\frac{1}{12} (1-s)^3 \right]_{-1}^{\alpha} \\ &= \frac{2}{3} \, \frac{\beta \left(3 + 3\,\beta + \beta^2\right)}{(1+\beta)^3} ,\end{aligned} \tag{F.50}$$

and

$$\begin{aligned}\int_{-1}^{\alpha} K_2 \, ds &= \left[\frac{1}{12} (1+s)^3 \right]_{-1}^{\alpha} \\ &= \frac{2}{3} \, \frac{\beta^3}{(1+\beta)^3} .\end{aligned} \tag{F.51}$$

The results for the second part are

$$\int_{-1}^{0} K_1 \, ds = \left[-\frac{1}{12} (1-s)^3 \right]_{-1}^{0} = \frac{7}{12} \, , \tag{F.52}$$

$$\int_{-1}^{0} K_2 \, ds = \left[\frac{1}{12} (1+s)^3 \right]_{-1}^{0} = \frac{1}{12} \, . \tag{F.53}$$

With $s > 0$ for the second triangle, the required reordering is given by equation (F.26), being careful to note that α must be redefined at this stage. The $0 < s < 1$ integrals have already been evaluated. The results for the first part of this rearrangement are

$$\begin{aligned} \int_{\alpha}^{1} K_1 \, ds &= \left[-\frac{1}{12} (1-s)^3 \right]_{\alpha}^{1} \\ &= \frac{2}{3} \frac{\beta^3}{(\beta-1)^3}, \end{aligned} \tag{F.54}$$

and

$$\begin{aligned} \int_{\alpha}^{1} K_2 \, ds &= \left[\frac{1}{12} (1+s)^3 \right]_{\alpha}^{1} \\ &= \frac{2}{3} \frac{\beta \left(\beta^2 - 3\,\beta + 3\right)}{(\beta-1)^3}. \end{aligned} \tag{F.55}$$

Finally, for the third triangle when $s < 0$ the reordering is given by equation (F.35) and for $s > 0$, it is given by equation (F.40). The integrals for $-1 < s < 0$ are already known so all that remains is to evaluate the integrals for $0 < s \leq \alpha$. The results are

$$\begin{aligned} \int_{0}^{\alpha} K_1 \, ds &= \left[-\frac{1}{12} (1-s)^3 \right]_{0}^{\alpha} \\ &= -\frac{1}{12} \frac{7\,\beta^3 + 3\,\beta^2 - 3\,\beta + 1}{(\beta-1)^3}, \end{aligned} \tag{F.56}$$

and

$$\begin{aligned} \int_{0}^{\alpha} K_2 \, ds &= \left[\frac{1}{12} (1+s)^3 \right]_{0}^{\alpha} \\ &= -\frac{1}{12} \frac{7 + \beta^3 - 3\,\beta^2 + 3\,\beta}{(\beta-1)^3}. \end{aligned} \tag{F.57}$$

The preceding equations, along with those listed in section F.1 complete the coincident integration. All remaining integrals are one dimensional and calculated numerically.

Appendix G

Transformations for the Edge Adjacent Galerkin Integral

G.1 Application of the First Polar Transform to the Distance Vector

The domain for the first polar transformation of the edge adjacent integral is illustrated in figure 4.5; the vector Q remains unchanged from equation (E.6), however, the P vector is now rewritten as

$$P(s,t) = \begin{pmatrix} \left(-\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) x_2 + \left(\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) x_1 + \left(\frac{1}{3}\sqrt{3}t\right) x_4 \\ \left(-\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) y_2 + \left(\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) y_1 + \left(\frac{1}{3}\sqrt{3}t\right) y_4 \\ \left(-\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) z_2 + \left(\frac{1}{2}s - \frac{1}{6}\sqrt{3}t + \frac{1}{2}\right) z_1 + \left(\frac{1}{3}\sqrt{3}t\right) z_4 \end{pmatrix}. \quad \text{(G.1)}$$

The Q vector must now be transformed into the (s,t) parameter space. To achieve this, the polar transformation described by equation (4.49) is applied. Some manipulation shows that

$$\mathbf{R}^{\pm} = \begin{pmatrix} \left(C_1\cos(\theta) + D_1\sin(\theta)\right)\rho + E_1 t \mp \epsilon N_1 \\ \\ \left(C_2\cos(\theta) + D_2\sin(\theta)\right)\rho + E_2 t \mp \epsilon N_2 \\ \\ \left(C_3\cos(\theta) + D_3\sin(\theta)\right)\rho + E_3 t \mp \epsilon N_3 \end{pmatrix}, \tag{G.2}$$

where

$$C_1 = \frac{1}{2}\left(x_2 - x_1\right), \qquad D_1 = \frac{1}{6}\sqrt{3}\left(-x_1 - x_2 + 2x_3\right), \tag{G.3}$$

and

$$E_1 = \frac{1}{6}\sqrt{3}\left(x_2 - 2x_4 + x_1\right). \tag{G.4}$$

Similar results hold for C_2, D_2, E_2, C_3, D_3 and E_3, by replacing the 'x's with 'y's and 'z's respectively.

It follows from equation (G.2) that the form of r^2 is different dependent on whether the internal or external limit is taken. For R^- the result is

$$r_-^2 = \epsilon^2 + 2\left(\mathbf{N}\cdot\mathbf{R}^0\right)\epsilon + b_{01}(\theta)\rho^2 + b_{11}(\theta)\rho t + b_{10}t^2 \tag{G.5}$$

where

$$\mathbf{N}\cdot\mathbf{R}^0 = b_{20}(\theta)\rho + b_{02}t. \tag{G.6}$$

and $\mathbf{R}^0 = Q - P$ or, otherwise, $\mathbf{R}^{\pm}$ with ϵ set to zero. The coefficients are

$$\begin{aligned} b_{01} &= \left(C_1^2 + C_2^2 + C_3^2\right)\cos^2(\theta) + 2\left(C_1D_1 + C_2D_2 + C_3D_3\right)\cos(\theta)\sin(\theta) + \\ &\quad \left(D_1^2 + D_2^2 + D_3^2\right)\sin^2(\theta), \\ b_{11} &= 2\left[\left(C_1E_1 + C_2E_2 + C_3E_3\right)\cos(\theta) + \left(D_1E_1 + D_2E_2 + D_3E_3\right)\sin(\theta)\right], \\ b_{10} &= E_1^2 + E_2^2 + E_3^2, \\ b_{20} &= \left(N_1C_1 + N_2C_2 + N_3C_3\right)\cos(\theta) + \left(N_1D_1 + N_2D_2 + N_3D_3\right)\sin(\theta), \\ b_{02} &= N_1E_1 + N_2E_2 + N_3E_3. \end{aligned} \tag{G.7}$$

Similarly, the result for r_+^2 is

$$r_+^2 = \epsilon^2 - 2\left(\mathbf{N}\cdot\mathbf{R}^0\right)\epsilon + b_{01}(\theta)\rho^2 + b_{11}(\theta)\rho t + b_{10}t^2. \tag{G.8}$$

G.2 Application of the Second Polar Transform to the Shape Functions and Distance Vector

The influence of the two transformations on the shape functions is as follows. For ψ_j the result is

$$\psi_j(s,\theta,\psi,\Lambda) = \begin{pmatrix} c_{1,0}(s) + c_{1,1}(\theta,\psi)\Lambda \\ \\ c_{2,0}(s) + c_{2,1}(\theta,\psi)\Lambda \\ \\ c_{3,0}(s) + c_{3,1}(\theta,\psi)\Lambda \end{pmatrix}, \tag{G.9}$$

where

$$\begin{aligned} c_{1,0} &= \frac{1}{2} + \frac{1}{2}s, \quad & c_{1,1} &= -\frac{1}{2}\cos(\Psi)\cos(\theta) - \frac{1}{6}\sqrt{3}\cos(\Psi)\sin(\theta), \\ c_{2,0} &= \frac{1}{2} - \frac{1}{2}s, \quad & c_{2,1} &= \frac{1}{2}\cos(\Psi)\cos(\theta) - \frac{1}{6}\sqrt{3}\cos(\Psi)\sin(\theta), \\ c_{3,0} &= 0, & c_{3,1} &= \frac{1}{3}\sqrt{3}\cos(\Psi)\sin(\theta), \end{aligned} \tag{G.10}$$

and for ψ_l,

$$\psi_l(s,\psi,\Lambda) = \begin{pmatrix} k_{1,0}(s) + k_{1,1}(\psi)\Lambda \\ \\ k_{2,0}(s) + k_{2,1}(\psi)\Lambda \\ \\ k_{3,0}(s) + k_{3,1}(\psi)\Lambda \end{pmatrix}, \tag{G.11}$$

where

$$\begin{aligned} k_{1,0} &= \frac{1}{2} + \frac{1}{2}s, \quad & k_{1,1} &= -\frac{1}{6}\sqrt{3}\sin(\Psi), \\ k_{2,0} &= \frac{1}{2} - \frac{1}{2}s, \quad & k_{2,1} &= -\frac{1}{6}\sqrt{3}\sin(\Psi), \\ k_{3,0} &= 0, & k_{3,1} &= \frac{1}{3}\sqrt{3}\sin(\Psi). \end{aligned} \tag{G.12}$$

Subject to the second polar transformation, $\mathbf{R}^{\pm}$ now has the form

$$\mathbf{R}^{\pm} = \begin{pmatrix} \left(\left(C_1 \cos(\theta) + D_1 \sin(\theta)\right) \cos(\Psi) + E_1 \sin(\Psi)\right) \Lambda \mp \epsilon\, N_1 \\ \\ \left(\left(C_2 \cos(\theta) + D_2 \sin(\theta)\right) \cos(\Psi) + E_2 \sin(\Psi)\right) \Lambda \mp \epsilon\, N_2 \\ \\ \left(\left(C_3 \cos(\theta) + D_3 \sin(\theta)\right) \cos(\Psi) + E_3 \sin(\Psi)\right) \Lambda \mp \epsilon\, N_3 \end{pmatrix} = \begin{pmatrix} d_1(\theta, \Psi)\Lambda \mp \epsilon\, N_1 \\ \\ d_2(\theta, \Psi)\Lambda \mp \epsilon\, N_2 \\ \\ d_3(\theta, \Psi)\Lambda \mp \epsilon\, N_3 \end{pmatrix}. \tag{G.13}$$

Appendix H

Collocation Method at the Wake Edges

A collocation based gradient method is formulated in Gray *et al.* (2004b, sec. 5). In that paper it is proved that the method, although applicable to a general Laplace problem, is not applicable to the crack problem. The difficulty arises from an additional constant in the shape functions for the most singular point. However, the zero value of $[\phi]$ here will be shown to render this problem void. This method was also considered for use in order to truncate the Galerkin method, however after discussions with Professor Gray, it was theorised that it would still produce a singular result at the wake edges. Nevertheless, the derivation of the equations for the crack with zero $[\phi]$ at the edges is presented here.

The development of the collocation method is very similar to that of the edge adjacent integral, expect that only one polar transformation is available. To begin, consider the domain in figure H.1. Note that the normal, $\mathbf{n}$, to the element Σ_Q, containing Q, is pointing into the external domain and that the calculated normal at point P, denoted as $\mathbf{L}$, may not be that same as $\mathbf{n}$. In the parameter space, the same triangular linear element in used, with shape functions defined by equation (3.56). The point P is positioned at $s^* = -1, t^* = 0$ denoted p_1.

Using the collocation method, a direct solution to equation (2.74) is sought. Additionally, the degree of singularity can be reduced if the difference between the internal and external limits are taken as illustrated by equation (2.80). The effect of differencing the limits is that only the singular panels contribute to solution. To apply this analysis to the wake, ϕ and ϕ_n can be exchanged with $[\phi]$ and $[\phi_n]$. Once applied to the wake

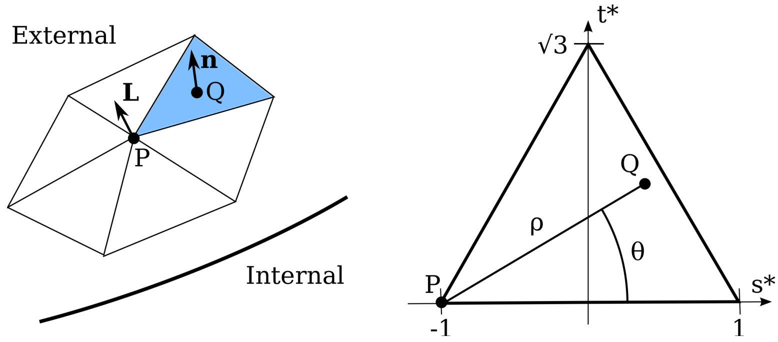


Figure H.1: Domain for collocation method in Cartesian and parameterised space.

problem, the $[\phi_n]$ term will vanish. Thus, for the domain in figure H.1 the solution for hypersingular integral of ϕ for the highlighted singular panel is demonstrated.

In the parameterised space, the governing equation to be solved is

$$\left.\frac{\partial \phi(P)}{\partial \mathbf{E}_k}\right|_{\Sigma_Q} = -\frac{1}{4\pi}\sum_{j=1}^{3}\phi(Q_j)\int_{-1}^{1}\int_{0}^{\sqrt{3}(1-|s^*|)} \psi_j(s^*,t^*)J_Q\frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}\,dt^*\,ds^*. \tag{H.1}$$

The parameterised vector for the point Q is identical to equation (E.6). (s^*, t^*) is transformed with a polar transformation centred about $(-1, 0)$, defined by

$$\begin{aligned} s^* &= \rho\cos(\theta) - 1, \\ t^* &= \rho\sin(\theta), \end{aligned} \tag{H.2}$$

where $0 \le \rho \le Q_R$, $0 \le \theta \le \pi/3$. The resulting polar scheme is illustrated in figure H.1

Following the transformation, the distance vector $\mathbf{R}^{\pm} = Q - (P \pm \epsilon\mathbf{L})$ is

$$\begin{pmatrix} (A_1\cos(\theta) + B_1\sin(\theta))\,\rho \mp \epsilon L_1 \\ \\ (A_2\cos(\theta) + B_2\sin(\theta))\,\rho \mp \epsilon L_2 \\ \\ (A_3\cos(\theta) + B_3\sin(\theta))\,\rho \mp \epsilon L_3 \end{pmatrix} = \begin{pmatrix} a_1\rho \mp \epsilon L_1 \\ \\ a_2\rho \mp \epsilon L_2 \\ \\ a_3\rho \mp \epsilon L_3 \end{pmatrix}, \tag{H.3}$$

where A_k, B_k $(k = 1, 2, 3)$ are defined using equations (E.8) and (E.9). Subsequently,

$$r_{\pm}^2 = a^2\rho^2 \mp b_1\rho\epsilon + \epsilon^2, \tag{H.4}$$

where $a^2 = a_1^2 + a_2^2 + a_3^2$ and $b_1 = 2\sum_{n=1}^{3} L_n a_n$.

Under the transformation defined in equation (H.2) the shape functions become

$$\psi_j(\rho,\theta) = \begin{pmatrix} 1 - \dfrac{1}{2\sqrt{3}}\rho\left(\sqrt{3}\cos(\theta) + \sin(\theta)\right) \\ \\ \dfrac{1}{2\sqrt{3}}\rho\left(\sqrt{3}\cos(\theta) - \sin(\theta)\right) \\ \\ \rho\dfrac{\sin(\theta)}{\sqrt{3}} \end{pmatrix} = \begin{pmatrix} 1 + c_1\rho \\ \\ c_2\rho \\ \\ c_3\rho \end{pmatrix}. \tag{H.5}$$

Subsequently, there are two types of integral to be solved. These have the form

$$-\frac{J_Q}{4\pi}\int_0^{\pi/3}\int_0^{Q_R} \rho^{\alpha}\frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}(P,Q)\,d\rho\,d\theta, \tag{H.6}$$

where $\alpha = 1$ for the first term of ψ_1 and $\alpha = 2$ for the second term of ψ_1 together with ψ_2 and ψ_3.

Equation (4.7) describes the kernel of equation (H.6). For this particular case, $\mathbf{n}\cdot\mathbf{E}_k = \mathbf{n}_k$ and

$$\begin{aligned} \mathbf{n}\cdot\mathbf{R}^{\pm} &= \mp(\mathbf{n}\cdot\mathbf{L})\epsilon, \\ \mathbf{E}_k\cdot\mathbf{R}^{\pm} &= a_k\rho \mp L_k\epsilon. \end{aligned} \tag{H.7}$$

For $\alpha = 2$ the internal limit is

$$-\frac{J_Q c_j}{4\pi}\int_0^{\pi/3}\int_0^{Q_R} \rho^2\left(\frac{n_k}{r_-^3} - 3\frac{(\mathbf{n}\cdot\mathbf{L})\epsilon\,(a_k\rho + L_k\epsilon)}{r_-^5}\right)d\rho\,d\theta \tag{H.8}$$

and for the external limit,

$$-\frac{J_Q c_j}{4\pi}\int_0^{\pi/3}\int_0^{Q_R} \rho^2\left(\frac{n_k}{r_+^3} + 3\frac{(\mathbf{n}\cdot\mathbf{L})\epsilon\,(a_k\rho - L_k\epsilon)}{r_+^5}\right)d\rho\,d\theta. \tag{H.9}$$

The difference in external and internal limits is now taken between equations (H.8) and (H.9) and then integrated. This results in

$$-\frac{J_q c_j}{4\pi a^2}\left(\frac{32a^2(\mathbf{n}\cdot\mathbf{L})(L_k b_1 - 2a_k)}{\left(4a^2 - b_1^2\right)^2} - \frac{4n_k b_1}{\left(4a^2 - b_1^2\right)} + \frac{n_k}{a}\ln\left(\frac{-b_1 + 2a}{b_1 + 2a}\right)\right). \tag{H.10}$$

Subsequently, the outer integral for equation (H.10) can be numerically integrated to give the result for $j = 2, 3$. For the value at $j = 1$, the $\alpha = 1$ case for equation (H.6) must be evaluated. As stated in Gray *et al.* (2004b, pg. 306), the result in this case is not forthcoming. Fortunately, as mentioned previously, the $j = 1$ case also corresponds to a wake edge node where, once calculating for $[\phi]$, the value will be zero. Hence, this case is of no consequence.

Bibliography

I. H. Abbott and A. E. von Doenhoff. *Theory of Wing Sections.* Dover Publications, 1959.

D. J. Acheson. *Elementary Fluid Dynamics.* Oxford University Press, 1990.

A. A. Afjeh and T. G. Keith, Jr. A simplified free wake method for horizontal-axis wind turbine performance prediction. *Journal of Fluids Engineering*, 108:400–406, 1986.

R. M. Ardito Marretta, G. Davi, G. Lombardi, and A. Milazzo. Hybrid numerical technique for evaluating wing aerodynamic loading with propeller interference. *Computers & Fluids*, 28:923–950, 1999.

G. R. Baker. Generalized vortex methods for free-surface flows. In R. E. Meyer, editor, *Waves on Fluid Interfaces: proceedings of a symposium.* Academic Press, 1983.

G. R. Baker, D. I. Meiron, and S. A. Orszag. Generalized vortex methods for free-surface flow problems. *Journal of Fluid Mechanics*, 123:477–501, 1982.

S. Bal and S. A. Kinnas. A BEM for the prediction of free surface effects on cavitating hydrofoils. *Computational Mechanics*, 28:260–274, 2002.

S. Bal and S. A. Kinnas. A numerical wave tank model for cavitating hydrofoils. *Computational Mechanics*, 32:259–268, 2003.

J. Balăs, J. Sládek, and V. Sládek. *Studies in Applied Mechanics 23: Stress Analysis in Boundary Element Methods.* Elsevier, 1989.

J. Baltazar and J. A. C. Falcão de Campos. Hydrodynamic analysis of a horizontal axis marine current turbine with a boundary element method. In *Proceedings of the ASME 27th International Conference on Offshore Mechanics and Arctic Engineering*, Estoril, Portugal, June 2008.

P. Bassanini, C. M. Casciola, M. R. Lancia, and R. Piva. On the trailing edge singularity and Kutta condition for 3D airfoils. *European Journal of Mechanics, B/Fluids*, 15: 809–830, 1996.

P. Bassanini, C. M. Casciola, M. R. Lancia, and R. Piva. Edge singularities and Kutta condition in 3D aerodynamics. *Meccanica*, 34:199–229, 1999.

D. J. Bernasconi and P. M. Richelsen. A higher-order potential flow method for thick bodies, thin surfaces and wakes. *International Journal for Numerical Methods in Engineering*, 73:706–727, 2008.

M. Bonnet. *Boundary Integral Equation Methods for Solids and Fluids.* John Wiley and Sons Ltd, 1999.

D. A. Bourgoyne, J. M. Hamel, S. L. Ceccio, and D. R. Dowling. Time-averaged flow over a hydrofoil at high Reynolds number. *Journal of Fluid Mechanics*, 496:365–404, 2003.

M. Brady, A. Leonard, and D. I. Pullin. Regularised vortex sheet evolution in three dimensions. *Journal of Computational Physics*, 146:520–545, 1998.

C. A. Brebbia. Weighted residual classification of approximate methods. *Applied Mathematical Modelling*, 2(3):160–164, 1978.

S. R. Breit. The potential of a Rankine source between parallel planes and in a rectangular cylinder. *Journal of Engineering Mathematics*, 25(2):151–163, May 1991.

J. Broeze, E. F. G. van Daalen, and P. J. Zandbergen. A three-dimensional panel method for nonlinear free surface waves on vector computers. *Computational Mechanics*, 13(1-2):12–28, November 1993.

A. H.-D. Cheng and D. T. Cheng. Heritage and early history of the boundary element method. *Engineering Analysis with Boundary Elements*, 29:268–302, 2005.

J.-K. Choi and S. A. Kinnas. Prediction of unsteady effective wake by a Euler solver/vortex-lattice coupled method. *Journal of Ship Research*, 47(2):131–144, 2003.

CCC. *Building a low-carbon economy - the UK's contribution to tackling climate change.* Committee on Climate Change, December 2008. URL http://www.theccc.org.uk/reports/.

M. Cooker. *The Interaction Between Steep Water Waves and Coastal Structures.* PhD thesis, University of Bristol, 1990.

H. S. M. Coxeter. *Introduction to Geometry.* Wiley, 1989.

J. Craig, H. Rudd, P. Michael, S. Couch, and J. Hartley. *Tidal Power in the UK - R5 - UK Case Studies.* Sustainable Development Commission, October 2007. URL http://www.sd-commission.org.uk/publications.php?id=616.

W. Craig, P. Guyenne, J. Hammack, D. Henderson, and C. Sulem. Solitary water wave interactions. *Physics of Fluids*, 18:057106, 2006.

DEFRA. *Implementing the Climate Change Act 2008.* Department for Environmental Food and Rural Affairs, June 2008. URL http://www.defra.gov.uk/environment/climate/legislation/.

DUKES. *Digest of United Kingdom energy statistics (DUKES).* Department of Energy and Climate Change, 2009. URL http://www.decc.gov.uk/en/content/cms/statistics/publications/dukes/.

R. H. Djojodihardjo and S. E. Widnall. A numerical method for the calculation of nonlinear, unsteady lifting potential flow problems. *AIAA Journal*, 7(10):2001–2009, 1969.

J. W. Dold. An efficient surface-integral algorithm applied to unsteady gravity waves. *Journal of Computational Physics*, 103:90–105, 1992.

J. W. Dold and D. H. Peregrine. An efficient boundary-integral method for steep unsteady water waves. In K. Morton and M. Baines, editors, *Numerical methods for fluid dynamics II*, pages 671–679. Clarendon Press, 1986.

D. G. Dommermuth, D. K. P. Yue, W. M. Lin, R. J. Rapp, E. S. Chan, and W. K. Melville. Deep-water plunging breakers: a comparison between potential theory and experiments. *Journal of Fluid Mechanics*, 189:423–442, 1988.

D. A. Dunavant. High degree efficient symmetrical Gaussian quadrature rules for the triangle. *International Journal for Numerical Methods in Engineering*, 21(6):1129–1148, 1985.

E. P. N. Duque, M. D. Burklund, and W. Johnson. Navier-Stokes and comprehensive analysis performance predictions of the NREL phase VI experiment. *Journal of Solar Energy Engineering*, 125:457–467, 2003.

M. A. Epton. Integration-by-parts formulas for boundary-element methods. *AIAA Journal*, 30(2):496–504, 1992.

S. N. Fata. Fast Galerkin BEM for 3D-potential theory. *Computational Mechanics*, 42: 417–429, 2008.

C. Fochesato and F. Dias. A fast method for nonlinear three-dimensional free-surface waves. *Proceedings of the Royal Society A*, 462:2715–2735, 2006.

D. I. M. Forehand. *Numerical Prediction of Free-Surface Flows Caused by Body/Fluid Interaction.* PhD thesis, The University of Edinburgh, 1998.

M. Garzon, D. Adalsteinsson, L. Gray, and J. A. Sethian. A coupled level set-boundary integral method for moving boundary simulations. *Interfaces and Free Boundaries*, 7:277–302, 2005.

A. Gharakhani. A regularised Galerkin boundary element method (RGBEM) for simulating potential flow about zero thickness bodies. Technical Report SAND99-2578, Sandia National Laboratories, Albuquerque, New Mexico, October 1999.

A. Gharakhani and A. F. Ghoniem. BEM solution of the 3D internal Neumann problem and a regularized formulation for the potential velocity gradients. *International Journal for Numerical Methods in Fluids*, 24:81–100, 1997.

J. P. Giesing and A. M. O. Smith. Potential flow about two-dimensional hydrofoils. *Journal of Fluid Mechanics*, 28(1):113–129, 1967.

E. Graciani, V. Mantič, F. París, and J. Cañas. A critical study of hypersingular and strongly singular boundary integral representations of potential gradient. *Computational Mechanics*, 25:542–559, 2000.

L. J. Gray. Symmetric Galerkin fracture analysis. *Engineering Analysis with Boundary Elements*, 15:103–109, 1995.

L. J. Gray and T. Kaplan. 3D Galerkin integration without Stokes' theorem. *Engineering Analysis with Boundary Elements*, 25:289–295, 2001.

L. J. Gray, D. Maroudas, and M. N. Enmark. Galerkin boundary integral method for evaluating surface derivatives. *Computational Mechanics*, 22:187–193, 1998.

L. J. Gray, J. M. Glaeser, and T. Kaplan. Direct evaluation of hypersingular Galerkin surface integrals. *SIAM Journal on Scientific Computing*, 25(5):1534–1556, 2004a.

L. J. Gray, A.-V. Phan, and T. Kaplan. Boundary integral evaluation of surface derivatives. *SIAM Journal on Scientific Computing*, 26(1):294–312, 2004b.

L. J. Gray, A. Salvadori, A.-V. Phan, and V. Mantič. Direct evaluation of hypersingular Galerkin surface integrals. II. *Electronic Journal of Boundary Elements*, 4(3):105–130, 2006.

G. I. Gretton, T. Bruce, and D. M. Ingram. Hydrodynamic modelling of a vertical axis tidal current turbine using CFD. In *Proceedings of the 8th European Wave and Tidal Energy Conference*, 2009.

S. Grilli, P. Guyenne, and F. Dias. Numerical computation of three-dimensional overturning waves. In *Proceedings of the 15th International Workshop on Water Waves and Floating Bodies*, Dan Caesarea, Israel, February 2000.

S. T. Grilli and I. A. Svendsen. Corner problems and global accuracy in the boundary element solution of nonlinear wave flows. *Engineering Analysis with Boundary Elements*, 7(4):178–195, 1990.

S. T. Grilli, J. Skourup, and I. A. Svendsen. An efficient boundary element method for nonlinear water waves. *Engineering Analysis with Boundary Elements*, 6:97–107, 1989.

S. T. Grilli, P. Guyenne, and F. Dias. A fully non-linear model for three-dimensional overturning waves over an arbitrary bottom. *International Journal for Numerical Methods in Fluids*, 35(7):829–867, 2001.

S. Grilli and R. Subramanya. A computer program for transient wave runup. Technical Report 93-02, Center for Applied Coastal Research, University of Delaware, 1993.

S. T. Grilli and Z. Hu. A higher-order hypersingular boundary element method for the modeling of vortex sheet dynamics. *Engineering Analysis with Boundary Elements*, 21:117–129, 1998.

M. Guiggiani. Hypersingular boundary integral equations have an additional free term. *Computational Mechanics*, 16:245–248, 1995.

M. Guiggiani. Formulation and numerical treatment of boundary integral equations with hypersingular kernels. In J. Sládek and V. Sládek, editors, *Singular Integrals in Boundary Element Methods*, chapter 3. Computational Mechanics, Inc, 1998.

M. Guiggiani, G. Krishnasamy, T. J. Rudolphi, and F. J. Rizzo. A general algorithm for the numerical solution of hypersingular boundary integral equations. *Journal of Applied Mechanics*, 59:604–614, September 1992.

J. L. Hess. Panel methods in computational fluid dynamics. *Annual Review of Fluid Mechanics*, 22:255–274, January 1990.

J. L. Hess. Calculation of potential flow about arbitrary three-dimensional lifting bodies. Technical Report MDC J5679-01, Douglas Aircraft Company, Long Beach, California, October 1972.

J. L. Hess. The use of higher-order surface singularity distributions to obtain improved potential flow solutions for two-dimensional lifting airfoils. *Computer Methods in Applied Mechanics and Engineering*, 5:11–35, 1975.

J. L. Hess and A. M. O. Smith. Calculation of non-lifting potential flow about arbitrary three-dimensional bodies. Technical Report E.S. 40622, Douglas Aircraft Company, March 1962.

T. Y. Hou, V. G. Stredie, and T. Y. Wu. A 3D numerical method for studying vortex formation behind a moving plate. *Communications in Computational Physics*, 1(2): 207–229, 2006.

M. Jaccard. *Sustainable Fossil Fuels.* Cambridge University Press, 2005.

M. A. Jaswon. Integral equation methods in potential theory. I. In *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 1963.

W. Johnson. *Helicopter Theory.* Dover Publications, 1994.

G. Karami and D. Derakhshan. An efficient method to evaluate hypersingular and supersingular integrals in boundary integral equations and analysis. *Engineering Analysis with Boundary Elements*, 23:317–326, 1999.

J. Katz and A. Plotkin. *Low-Speed Aerodynamics, Second Edition.* Cambridge University Press, 2001.

O. D. Kellogg. *Foundations of Potential Theory.* Verlag Von Julius Springer, 1929.

B. Kim and Y. S. Shin. A NURBS panel method for three-dimensional radiation and diffraction problems. *Journal of Ship Research*, 47(2):177–186, 2003.

S. A. Kinnas and N. E. Fine. A numerical nonlinear analysis of the flow around two- and three-dimensional partially cavitating hydrofoils. *Journal of Fluid Mechanics*, 254:151–181, 1993.

S. A. Kinnas and C.-Y. Hsin. Boundary element method for the analysis of the unsteady flow around extreme propeller geometries. *AIAA Journal*, 30(3):688–696, 1992.

V. A. Kondrat'ev and O. A. Oleinik. Boundary-value problems for partial differential equations in non-smooth domains. *Russian Mathematical Surveys*, 38(2):3–76, 1983.

C. L. Ladson, C. W. Brooks, Jr., A. S. Hill, and D. W. Sproles. Computer program to obtain ordinates for NACA airfoils. Technical Report NASA TM-4741, National Aeronautics and Space Administration, Langley Research Center, Hampton, Virginia, 1996.

D. Le Gourieres. *Wind Power Plants, Theory and Design.* Pergamon, 1982.

H. Lee and S. A. Kinnas. Application of a boundary element method in the prediction of unsteady blade sheet and developed tip vortex cavitation on marine propellers. *Journal of Ship Research*, 48(1):15–30, March 2004.

H. Lee and S. A. Kinnas. Unsteady wake alignment for propellers in nonaxisymmetric flows. *Journal of Ship Research*, 49(3):176–190, September 2005a.

H. S. Lee and S. A. Kinnas. A BEM for the modeling of unsteady propeller sheet cavitation inside of a cavitation tunnel. *Computational Mechanics*, 37(1):41–51, December 2005b.

T. Lewiner, J. D. Gomes, Jr., H. Lopes, and M. Craizer. Arc-length based curvature estimator. In *Proceedings of the Computer Graphics and Image Processing, XVII Brazilian Symposium on*, pages 250–257. IEEE Computer Society, 2004.

H.-B. Li, G.-M. Han, and H. A. Mang. A new method for evaluating singular integrals in stress analysis of solids by the direct boundary element method. *International Journal for Numerical Methods in Engineering*, 21(11):2071–2098, 1985.

Y. Li and S. M. Çalişa. A discrete vortex method for simulating a stand-alone tidal-current turbine: Modeling and validation. *Journal of Offshore Mechanics and Arctic Engineering*, 132(3):031102–1–9, 2010.

Y.-H. Liu and X.-S. Lu. Polar coordinate transformation approach for treatment of singular integrals in boundary element methods. *Applied Mathematics and Mechanics*, 9(10):959–967, October 1988.

M. S. Longuet-Higgins and E. D. Cokelet. The deformation of steep surface waves on water. I. A numerical method of computation. In *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, volume 350, pages 1–26, July 1976.

M. S. Longuet-Higgins. Capillary rollers and bores. *Journal of Fluid Mechanics*, 240: 659–679, 1992.

R. G. Lueck and Y. Lu. The logarithmic layer in a tidal channel. *Continental Shelf Research*, 17(14):1785–1801, 1997.

M. S. Lynn and W. P. Timlake. The numerical solution of singular integral equations of potential theory. *Numerische Mathematik*, 11:77–98, 1968.

K. W. Mangler and J. H. B. Smith. Behaviour of the vortex sheet at the trailing edge of a lifting wing. *The Aeronautical Journal of the Royal Aeronautical Society*, 74: 906–908, 1970.

V. Mantič and F. París. Existence and evaluation of the two free terms in the hypersingular boundary integral equation of potential theory. *Engineering Analysis with Boundary Elements*, 16:253–260, 1995.

V. Mantič, E. Graciani, and F. París. A simple local smoothing scheme in strongly singular boundary integral representation of potential gradient. *Computer Methods in Applied Mechanics and Engineering*, 178:267–289, 1999.

J. F. Manwell, J. G. McGowan, and A. L. Rogers. *Wind Energy Explained*. John Wiley and Sons Ltd, 2002.

W. H. Mason. *Software for Aerodynamics and Aircraft Design*. Virginia Tech, March 2007. URL http://www.aoe.vt.edu/~mason/Mason_f/MRsoft.html.

T. McCombes, C. Johnstone, and A. Grant. Unsteady 3D wake modelling for marine current turbines. In *Proceedings of the 8th European Wave and Tidal Energy Conference*, 2009.

D. E. Medina and J. A. Liggett. Three-dimensional boundary-element computation of potential flow in fractured rock. *International Journal for Numerical Methods in Engineering*, 26:2319–2330, 1988.

L. M. Milne-Thomson. *Theoretical Hydrodynamics*. MacMillan & Co Ltd, 1968.

L. Morino and C.-C. Kuo. Subsonic potential aerodynamics for complex configurations - A general theory. *AIAA Journal*, 12(2):191–197, 1974.

L. Morino and G. Bernardini. Singularities in BIEs for the Laplace equation; Joukowski trailing-edge conjecture revisited. *Engineering Analysis with Boundary Elements*, 25: 805–818, 2001.

L. Morino and Z. Kaprielian, Jr. Free wake analysis of helicopter rotors. *Vertica*, 9: 127–140, 1985.

L. E. Myers and A. S. Bahaj. Scale reproduction of the flow field for tidal energy converters. In *Proceedings of the 10th World Renewable Energy Congress*, Glasgow, July 2008.

J. N. Newman. The Green function for potential flow in a rectangular channel. *Journal of Engineering Mathematics*, 26(1):51–59, February 1992.

A. K. Otta, I. A. Svendsen, and S. T. Grilli. Unsteady free surface waves in a region of arbitrary shape. Technical Report CACR-92-10, Centre for Applied Coastal Research, University of Delaware, 1992.

G. H. Paulino and A. Sutradhar. The simple boundary element method for multiple cracks in functionally graded media governed by potential theory: a three-dimensional Galerkin approach. *International Journal for Numerical Methods in Engineering*, 65:2007–2034, 2006.

G. K. Politis. Simulation of unsteady motion of a propeller in a fluid including free wake modeling. *Engineering Analysis with Boundary Elements*, 28:633–653, 2004.

F. L. Ponta and P. M. Jacovkis. A vortex model for Darrieus turbine using finite element techniques. *Renewable Energy*, 24(1):1–18, 2001.

C. Pozrikidis. Theoretical and computational aspects of the self-induced motion of three-dimensional vortex sheets. *Journal of Fluid Mechanics*, 425:335–366, 2000.

S. Pyo. *Numerical modeling of propeller tip flows with wake sheet roll-up in three dimensions*. PhD thesis, Massachusetts Institute of Technology, July 1995.

S. Pyo and S. A. Kinnas. Propeller wake sheet roll-up modeling in three dimensions. *Journal of Ship Research*, 41(2):81–92, June 1997.

P. J. Roache. *Verification and Validation in Computational Science and Engineering*. Hermosa Publishers, 1998.

V. Rokhlin. Rapid solution of integral equations of classical potential theory. *Journal of Computational Physics*, 60:187–207, 1983.

T. Sarpkaya. Computational methods with vortices – The 1988 Freeman scholar lecture. *Journal of Fluids Engineering*, 111:5–52, 1989.

H. Schlichting. *Boundary-Layer Theory*. McGraw Hill Book Company, 1960.

F. J. Simoes and J. M. R. Graham. Application of a free vortex wake model to a horizontal axis wind turbine. *Journal of Wind Engineering and Industrial Aerodynamics*, 39:129–138, 1992.

V. Sladek and J. Sladek. Non-singular boundary integral representation of potential field gradients. *International Journal for Numerical Methods in Engineering*, 33: 1181–1195, 1992a.

V. Sladek and J. Sladek. Non-singular boundary integral representation of stresses. *International Journal for Numerical Methods in Engineering*, 33:1481–1499, 1992b.

A. M. O. Smith and J. Pierce. Exact solution of the Neumann problem. Calculation of non-circulatory plane and axially symmetric flows about or within arbitrary boundaries. Technical Report ES 26988, Douglas Aircraft Company, April 1958.

S. Solomon, D. Qin, M. Manning, Z. Chen, M. Marquis, K. B. Averyt, M. Tignor, and H. L. Miller, editors. *Contribution of Working Group I to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change.* Cambridge University Press, 2007.

J. N. Sørensen and W. Z. Shen. Numerical modelling of wind turbine wakes. *Journal of Fluids Engineering*, 124:393–399, 2002.

J. J. Stoker. *Water Waves.* Interscience, 1957.

A. Sutradhar, G. H. Paulino, and L. J. Gray. *Symmetric Galerkin Boundary Element Method.* Springer Verlag, 2008.

A. Sutradhar, G. H. Paulino, and L. J. Gray. On hypersingular surface integrals in the symmetric Galerkin boundary element method: application to heat conduction in exponentially graded materials. *International Journal for Numerical Methods in Engineering*, 62:122–157, 2005.

I. A. Svendsen and I. G. Jonsson. *Hydrodynamics of Coastal Regions.* Den Private Ingeniørfond, University of Denmark, Lyngby, Denmark, 1980.

J. L. Swedlow and T. A. Cruse. Formulation of boundary integral equations for three-dimensional elasto-plastic flow. *International Journal of Solids and Structures*, 7: 1673–1683, 1971.

G. T. Symm. Integral equation methods in potential theory. II. In *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 1963.

M. Tanaka. The stability of solitary waves. *Physics of Fluids*, 29(3):650–655, March 1986.

W.-T. Tsai and D. K. P. Yue. Interactions between a free surface and a vortex sheet shed in the wake of a surface-piercing plate. *Journal of Fluid Mechanics*, 257:691–721, 1993.

W.-T. Tsai and D. K. P. Yue. Computation of nonlinear free-surface flows. *Annual Review of Fluid Mechanics*, 28:249–278, January 1996.

J. Whale, C. G. Anderson, R. Bareiss, and S. Wagner. An experimental and numerical study of the vortex structure in the wake of a wind turbine. *Journal of Wind Engineering and Industrial Aerodynamics*, 84(1):1–21, 2000a.

J. Whale, C. J. Fisichella, and M. S. Selig. Correcting inflow measurements from wind turbines using a lifting-surface code. *Journal of Solar Energy Engineering*, 122:196–202, 2000b.

A. J. Wilde and H. Aliabadi. Direct evaluation of boundary stresses in the 3D BEM of elastostatics. *Communications in Numerical Methods in Engineering*, 14:505–517, 1998.

A. J. Wilde and H. Aliabadi. A 3-D BEM formulation for the analysis of crack growth. *Computational Mechanics*, 23:250–257, 1999.

M. L. Williams. Stress singularities resulting from various boundary conditions in angular corners of plates in extension. *Journal of Applied Mechanics*, 19(4):526–528, 1952.

G. Xu and L. N. Sankar. Computational study of horizontal axis wind turbines. *Journal of Solar Energy Engineering*, 122:35–39, 2000.

H. Xü. *Numerical Study of Fully Nonlinear Water Waves in Three Dimensions*. PhD thesis, Massachusetts Institute of Technology, 1992.

M. Xue, H. B. Xu, Y. M. Liu, and D. K. P. Yue. Computations of fully nonlinear three-dimensional wave-wave and wave-body interactions. Part 1. Dynamics of steep three-dimensional waves. *Journal of Fluid Mechanics*, 438:11–39, July 2001.

A. Young. A single-domain boundary element method for 3-D elastostatic crack analysis using continuous elements. *International Journal for Numerical Methods in Engineering*, 39:1265–1293, 1996.

Y. L. Young and S. A. Kinnas. Analysis of supercavitating and surface-piercing propeller flows via BEM. *Computational Mechanics*, 32:269–280, 2003.

Y. L. Young and S. A. Kinnas. Performance prediction of surface-piercing propellers. *Journal of Ship Research*, 28(4):288–304, December 2004.

Q. Zhu, Y. Liu, and D. K. P. Yue. Dynamics of a three-dimensional oscillating foil near the free surface. *AIAA Journal*, 44(12):2997–3009, December 2006.