A HIGHLY ADAPTIVE THREE DIMENSIONAL HYBRID VORTEX METHOD FOR INVISCID FLOWS AND HELICALLY SYMMETRIC VORTEX EQUILIBRIA

Daniel Lucas

A Thesis Submitted for the Degree of PhD
at the
University of St Andrews

2012

**Full metadata for this item is available in
Research@StAndrews:FullText
at:
http://research-repository.st-andrews.ac.uk/**

**Please use this identifier to cite or link to this item:
http://hdl.handle.net/10023/3091**

A HIGHLY ADAPTIVE THREE DIMENSIONAL HYBRID VORTEX METHOD FOR INVISCID FLOWS AND HELICALLY SYMMETRIC VORTEX EQUILIBRIA

Daniel Lucas
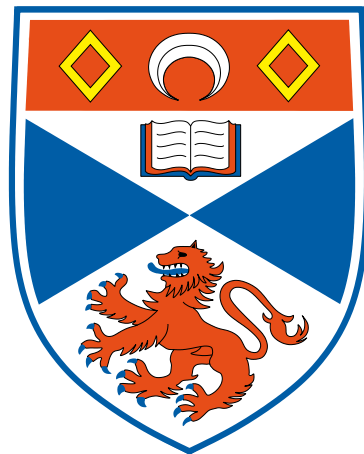
A thesis submitted for the degree of Doctor of Philosophy at the

University of St Andrews

21st December 2011

Abstract

This thesis is concerned with three-dimensional vortex dynamics, in particular the modelling of vortex structures in an inviscid context. We are motivated by the open problem of regularity of the inviscid equations, i.e. whether or not these equations possess solutions. This problem is manifest in small scales, where vortex filaments are stretched and intensify as they are drawn into increasingly thin tendrils. This creates great difficulty in the investigation of such flows. Our only means of experimentation is to perform numerical simulations, which require exceptionally high resolution to capture the small scale vortex structures.

A new numerical method to solve the inviscid Euler equations for three-dimensional, incompressible fluids is presented, with special emphasis on spatial adaptivity to resolve as broad a range of scales as possible in a completely self-similar fashion. We present a hybrid vortex method whereby we discretise the vorticity in Lagrangian filaments and perform and inversion to compute velocity on an arbitrary unstructured finite-volume grid. This allows for a two-fold adaptivity strategy. First, although naturally spatially adaptive by definition, the vorticity filaments undergo 'renoding'. We redistribute nodes along the filament to concentrate their density in regions of high curvature. Secondly the Eulerian mesh is adapted to follow high strain by increasing resolution based on local filament dimensions. These features allow vortex stretching and folding to be resolved in a completely automatic and self-similar way. The method is validated via well known vortex rings and newly discovered helical vortex equilibria

are also used to test the method.

We begin by presenting this new class of three-dimensional vortex equilibria which possess helical symmetry. Such vortices are observed in propeller and wind turbine wakes, and their equilibria shapes have until now been unknown. These vortices are described by contours bounding regions of uniform axial vorticity. Material conservation of axial vorticity enables equilibria to be calculated simply by a restriction on the helical stream function. The states are parameterised by their mean radius and centroid position. In the case of a single vortex, the parameter space cannot be fully filled by our numerical approach. We conjecture that multiply connected contours will characterise equilibria where the algorithm fails. We also consider multiple vortices, evenly azimuthally spaced about the origin. In such cases instabilities often lead to a single helical vortex.

Acknowledgements

Declaration

I, Daniel Lucas, hereby certify that this thesis, which is approximately 30000 words in length, has been written by me, that it is the record of work carried out by me and that it has not been submitted in any previous application for a higher degree.

I was admitted as a research student in October 2007 and as a candidate for the degree of PhD in October 2007; the higher study for which this is a record was carried out in the University of St Andrews between 2007 and 2011.

Date: .... **Signature of Candidate:**.............................

I hereby certify that the candidate has fulfilled the conditions of the Resolution and Regulations appropriate for the degree of PhD in the University of St Andrews and that the candidate is qualified to submit this thesis in application for that degree.

Date: .... **Signature of Supervisor:**.............................

In submitting this thesis to the University of St Andrews I understand that I am giving permission for it to be made available for use in accordance with the regulations of the University Library for the time being in force, subject to any copyright vested in the work not being affected thereby. I also understand that the title and the abstract will be published, and that a copy of the work may be made and supplied to any bona fide library or research worker, that my thesis will be electronically accessible for personal or research use unless exempt by award of an embargo as requested below, and that the library has the right to migrate my thesis into new electronic forms as required to ensure continued access to the thesis. I have obtained any third-party copyright permissions that may be required in order to allow such access and migration, or have requested the appropriate embargo below.

The following is an agreed request by candidate and supervisor regarding the electronic publication of this thesis: Embargo on both printed copy and electronic copy for the same fixed period of 2 years (maximum five) on the grounds that publication would preclude future publication.

Date: ....
Signature of Candidate:............................
Signature of Supervisor:............................

Contents

Commonly used symbols

Note: bold characters refer to vectors.

Generic symbols

$\boldsymbol{x} = (x, y, z)$	Cartesian position vector
$\boldsymbol{u}$	velocity
$\boldsymbol{\omega}$	vorticity
Re	Reynolds number
Γ	circulation
p	pressure
ρ	density
ψ	streamfunction
∇	gradient operator
Δ	Laplacian
ν	kinematic viscosity
T	kinetic energy
H	helicity
Ω	rotation rate
$J(a, b)$	Jacobian
t	time
T	time steps

4

Helical Vortex Parameters

ϵ	helical pitch
$\bar{R}$	mean core radius
d	centroid position
$\boldsymbol{h}$	helical vector
E	excess energy
J	angular impulse
N	number of vortices

VortexFoam Parameters

Note:

subscript p denotes Lagrangian filament quantities.

subscript i denotes Eulerian mesh quantities

$\boldsymbol{\psi}$	vector potential
$\boldsymbol{x}_p$	filament nodes
c_p	vortex element scaling
κ	curvature
μ	renode parameter
d	filament core radius/inter-filament distance
N	total vortex nodes
N_{cells}	total mesh cells
γ'	mesh refinement parameter
α	vortex ring/tube radius
N_{fil}	total vortex filaments
h_i	mesh cell width

<u>Abbreviations</u>

NS	Navier-Stokes equations
DNS	Direct Numerical Simulation
LES	Large Eddy Simulation
ODE	Ordinary Differential Equation
PDE	Partial Differential Equation
CASL	Contour-advective semi-Lagrangian
FEM	Finite Element Method
FVM	Finite Volume Method
FFT	Fast Fourier Transform
CFD	Computational Fluid Dynamics
ADI	Alternating Direction Implicit
2D	Two-dimensional
3D	Three-dimensional

Chapter 1

Introduction

Arguably one of the most widely utilised mathematical disciplines, fluid mechanics has applications which are far reaching in modern life. From weather and climate forecasting, wind turbine modelling to the design of planes, ships and cars, we have all benefitted from accurate mathematical models of fluid flows. However, such is its complexity that no proof exists that the equations at the heart of fluid mechanics even possess smooth solutions. This is considered such a fundamental problem in contemporary mathematics that it is one of the seven million dollar 'millennium prizes' offered by the Clay Institute (Fefferman, 2000).

1.1   Governing equations

The system of equations which describe the motion of a general viscous, Newtonian fluid are the Navier-Stokes (NS) equations named for George Gabriel Stokes and Claude-Louis Navier who independently derived them.[1]   Newton's

---

[1]Most of the credit should properly lie with Stokes, Navier having stumbled upon the correct form of the equations without a sound understanding of the processes he was describing (O'Connor and Robertson, 2011a).

second law provides the momentum equation, describing the force balance in a fluid, derived from a continuum hypothesis. In this work we will consider incompressible flows (sound waves are not permitted) and the Navier-Stokes can be written as

$$\frac{D\boldsymbol{u}}{Dt} = -\frac{\nabla p}{\rho} + \nu\nabla^2\boldsymbol{u} \tag{1.1}$$

where $\boldsymbol{u}$ is fluid velocity, p pressure, ρ density, ν the kinematic viscosity and $\frac{D}{Dt} = \frac{\partial}{\partial t} + \boldsymbol{u} \cdot \nabla$ the material derivative. The left hand side of the equation represents the inertial acceleration of the fluid, the first term on the right hand side the acceleration induced by pressure gradients and the final term the dissipation brought about by viscous shear stresses. The right hand side is often supplemented with body forces such as gravity or rotational forces.

In addition to conservation of momentum, a fluid must obey conservation of mass, for incompressible flows this reduces to

$$\nabla \cdot \boldsymbol{u} = 0 \tag{1.2}$$

i.e. the velocity field is divergence free, or *solenoidal*. Together with appropriate boundary and initial conditions, equations (1.1) and (1.2) describe the general motion of a viscous, incompressible, Newtonian fluid. (Note, pressure can be solved for by simply taking the divergence of (1.1).)

This equation set is somewhat misleading in its simplicity. It is quite clear to even a casual observer that the behaviour of fluids like air and water can be highly complex and irregular. The chaotic motions stem from the strong nonlinearity in the left hand size of (1.1) namely the convective acceleration term, $\boldsymbol{u} \cdot \nabla\boldsymbol{u}$. It is this quadratic nonlinearity in $\boldsymbol{u}$ that is responsible for the turbulent and unpredictable nature of certain fluid flows.

While the complexity is driven by the nonlinear term, there is still a part to be

played by viscosity. The form viscosity takes in an incompressible flow is diffusive and serves to regularise and damp the small scale turbulent motions. Therefore any meaningful comment about the nature of the flow regime must involve some relationship between the convective and viscous terms. First introduced by Stokes but named for Osbourne Reynolds after his famous experiments concerning the onset of turbulence in pipe flows (see figure 1.1 and (Reynolds, 1883)) , the *Reynolds number*, Re, is a dimensionless parameter defined as the ratio of the characteristic magnitudes of the inertial and viscous forces in a given flow, i.e.

$$Re = \frac{UL}{\nu}$$
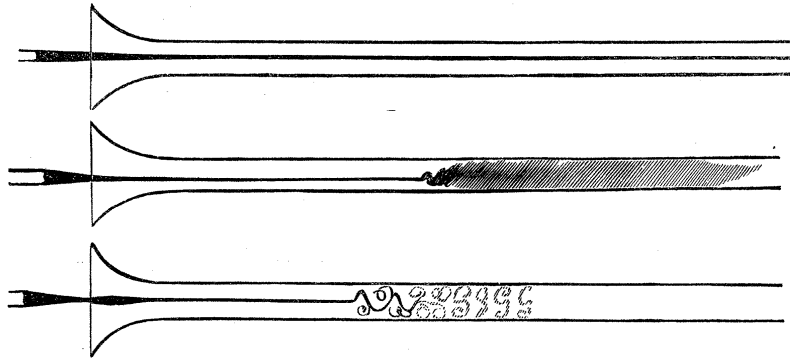
Figure 1.1: Onset of turbulence in pipe flow from (Reynolds, 1883).

where U is a characteristic speed and L is a characteristic length scale of the flow. The most challenging regimes for the numerical modeller and mathematical analyst are those for which the Reynolds number is high. The smoothness and existence problem for the NS equation can therefore be supposed to be most relevant for high Re. Indeed it may be argued that the more pertinent mathematical question concerns the regularity of the equations neglecting viscosity, corresponding to infinite Reynolds number (the inviscid equations actually predate (1.1) and were first presented by Leonard Euler after whom they are named). There is a commonly held, yet unproven belief that even a small amount of viscosity will

9

serve to maintain the regularity of the equations. Conversely, if it can be shown that even the inviscid form of the equations remain regular, the question will likely likewise be closed for the NS equations.

Thus far we have only made general statements concerning the form of the equations, yet clearly it is important to understand the mechanisms by which the nonlinear complexity is manifest.

A key concept in understanding unsteady flows is vorticity, defined as

$$\boldsymbol{\omega} = \nabla \times \boldsymbol{u} \tag{1.3}$$

which can be thought of as a rotation rate for a fluid particle (precisely twice the rotation rate). We obtain the prognostic equation for vorticity by simply taking the curl of (1.1). Thus, for an inviscid fluid of constant density, we have

$$\frac{D\boldsymbol{\omega}}{Dt} = \boldsymbol{\omega} \cdot \nabla \boldsymbol{u} \tag{1.4}$$

and

$$\nabla \cdot \boldsymbol{\omega} = 0 \tag{1.5}$$

i.e. vorticity is also solenoidal. Now taking the curl of (1.3) and using (1.2) gives the Poisson equation for velocity

$$\Delta \boldsymbol{u} = -\nabla \times \boldsymbol{\omega}.$$

It is common for this Poisson equation to be expressed using a vector potential $\boldsymbol{\psi}$, i.e.

$$\Delta \boldsymbol{\psi} = -\omega. \tag{1.6}$$

$$\boldsymbol{u} = \nabla \times \boldsymbol{\psi}$$

which enforces incompressibility and reduces derivatives on $\boldsymbol{\omega}$. For a given vorticity distribution, this equation can be inverted to compute the velocity field, $\boldsymbol{u}$,

given boundary conditions. For an unbounded flow, $\boldsymbol{u}$ may be expressed using the Biot-Savart integral

$$\boldsymbol{u} = -\frac{1}{4\pi} \int_V \frac{(\boldsymbol{x} - \boldsymbol{x}') \times \boldsymbol{\omega}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}'}{\left|\boldsymbol{x} - \boldsymbol{x}'\right|^3}. \tag{1.7}$$

The dynamics of vorticity is a rich area of research (Saffman, 1995), (Majda and Bertozzi, 2001), (Pullin and Saffman, 1998). Vorticity dynamics is relevant to a wide variety of phenomena, from the wing-tip vortices in aerodynamics, to the large swirling motions of hurricanes and cyclones in the atmosphere. Indeed vortices are nearly ubiquitous in fluid flows: the complex turbulent flows which are the source of such intense study are known to possess coherent vortex structures in even their smallest (arguably the most important) scales (see section 1.4). It is this dominance of vorticity in complex flows which motivates scientists and mathematicians to seek a thorough understanding of vortex structures and their interactions.

Figure 1.2: Vortex structures in nature. Left is a 'steam ring' vortex ring (a few metres across) produced by Mount Etna during volcanic activity in 2000 (Alean, 2011). Right is a satellite image of Hurricane Katia (of the order of hundreds of kilometres across), the remnant of which crossed the Atlantic and caused high winds and disruption across the west of the UK in September 2011 (AFP/Getty Images, 2011).

1.2 Circulation and Helmholtz's vortex laws

Despite the fact that the removal of viscosity typically leads to increased complexity in fluid flows, it leads to some elegant vortex laws first described by (Helmholtz, 1858). These laws are also justified in the context of weak viscosity (high Re), where the diffusion of vorticity is small (for a more thorough presentation see (Saffman, 1995)).

First let us define the following important scalar, circulation Γ,

$$\Gamma = \oint_C \boldsymbol{u} \cdot \mathrm{d}\boldsymbol{s} = \iint_A \boldsymbol{\omega} \cdot \boldsymbol{n} \, \mathrm{d}A \tag{1.8}$$

where C is a simple closed curve traversed anti-clockwise, $\mathrm{d}\boldsymbol{s}$ an element of the curve, A the open surface bounded by C with $\boldsymbol{n}$ its unit normal.

Also let us define the concept of vortex lines, that is space curves which are everywhere tangential to the vorticity vector, $\boldsymbol{\omega}$. A collection or bundle of vortex lines will define vortex tubes, e.g. a smoke ring (vortex ring), a tornado, etc. Let us consider a volume, V, enclosing some segment of a vortex tube and integrate the divergence of vorticity over it. Clearly this integral is zero due to (1.5), however due to Gauss' Theorem so is the integral of the flux of vorticity across the closed surface, S, of V, i.e.

$$\int_V \nabla \cdot \boldsymbol{\omega} \mathrm{d}V = \int_S \boldsymbol{\omega} \cdot \boldsymbol{n} \mathrm{d}S = 0$$

The surface integral reduces to the sum of the circulations at either end of the segment of vortex tube, with opposite sign due to the direction of $\boldsymbol{n}$:

$$\int_S \boldsymbol{\omega} \cdot \boldsymbol{n} \mathrm{d}S = \int_{A_1} \boldsymbol{\omega} \cdot \boldsymbol{n} \mathrm{d}S + \int_{A_2} \boldsymbol{\omega} \cdot \boldsymbol{n} \mathrm{d}S = 0$$

where A_1 and A_2 are the areas at either ends of the vortex tube (see figure 1.3).

This demonstrates that the vortex tube must have constant strength, or circulation along its length: $\Gamma =$constant. This is the first of Helmholtz's vortex
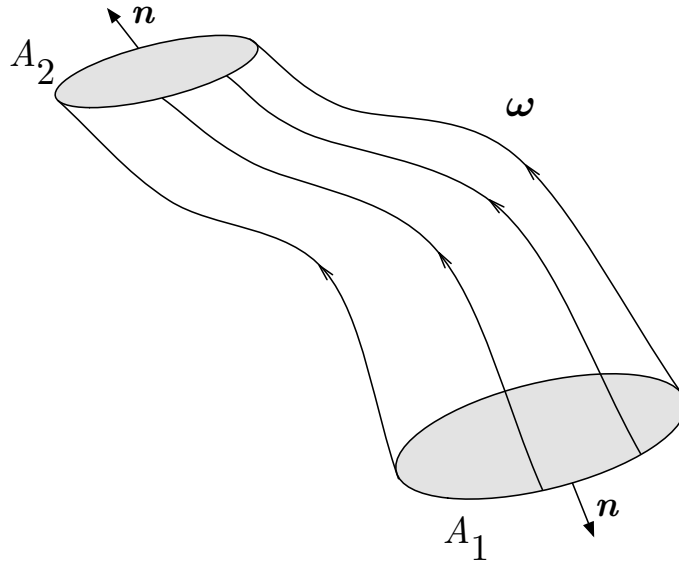
Figure 1.3: A volume V, enclosing a segment of vortex tube demonstrating Helmholtz's first vortex law.

laws.

The final law concerns the dynamics of vortex lines and states that vortex lines are also material lines. Considering the evolution of Γ,

$$\frac{d\Gamma}{dt} = \frac{d}{dt}\left(\oint_C u \cdot \mathrm{d}\boldsymbol{s}\right) = \oint_C \frac{D\boldsymbol{u}}{Dt} \cdot \mathrm{d}\boldsymbol{s} + \oint_C \boldsymbol{u} \cdot \frac{D\mathrm{d}\boldsymbol{s}}{Dt}$$

then substituting from the Euler equations (equation (1.1) minus the final term) the first term becomes

$$\oint_C \frac{D\boldsymbol{u}}{Dt} \cdot \mathrm{d}\boldsymbol{s} = \frac{1}{\rho} \oint_C \nabla p \cdot \mathrm{d}\boldsymbol{s} = 0$$

and the second term

$$\oint_C \boldsymbol{u} \cdot \frac{D\mathrm{d}\boldsymbol{s}}{Dt} = \oint_C \boldsymbol{u} \cdot \mathrm{d}\boldsymbol{s} \cdot \nabla \boldsymbol{u} = \oint_C \nabla(\boldsymbol{u} \cdot \boldsymbol{u}) \cdot \mathrm{d}\boldsymbol{s} = 0$$

since $\oint_C \nabla f \cdot \mathrm{d}\boldsymbol{s} = 0$ for any f. Hence, for any homogenous (ρ =constant), inviscid fluid,

$$\frac{D\Gamma}{Dt} = 0$$

for any material, closed contour, C. This is also known as Kelvin's circulation theorem. As a consequence one can now see that in the absence of viscosity the generation of rotational flow from irrotational flow is impossible. While rotational flows can intensify via instability and vortex stretching, the rotationality must be present for all time, and is not able to vanish.

1.3   Energy and Helicity

The invariance of Γ actually constitutes an infinite number of invariants, since the contour C is arbitrary. Alas, even with an infinite number of constraints, Euler flows have a great deal of freedom to evolve. Along with circulation, Γ, the Euler equations (which we will be primarily concerning ourselves with in the majority of the thesis) have another two invariants which prove extremely useful when discussing inviscid flows. The first of these is the kinetic energy, defined as half the bulk square of velocity magnitude, i.e.

$$T = \frac{1}{2} \int_V |\boldsymbol{u}|^2 \mathrm{d}V. \tag{1.9}$$

Conservation can be shown via the momentum equation. Removing viscous dissipation from equation 1.1 and taking the scalar product with $\boldsymbol{u}$ yields

$$\boldsymbol{u} \cdot \frac{D\boldsymbol{u}}{Dt} = -\boldsymbol{u} \cdot \nabla \frac{p}{\rho}$$

$$\text{i.e.} \quad \frac{1}{2} \frac{D}{Dt} \boldsymbol{u} \cdot \boldsymbol{u} = -\frac{1}{\rho} \nabla \cdot (p\boldsymbol{u})$$

via incompressibility (equation (1.2)). Now integrating over the entire volume of the fluid and employing Gauss' Theorem

$$\begin{aligned}
\frac{dT}{dt} &= -\frac{1}{\rho} \int_V \nabla \cdot (p\boldsymbol{u}) \mathrm{d}V \\
&= -\frac{1}{\rho} \int_{\delta A} p\boldsymbol{u} \cdot \hat{\boldsymbol{n}} \mathrm{d}S \\
&= 0
\end{aligned}$$

provided $\boldsymbol{u} \cdot \hat{n} = 0$ on the boundary, δA, of the volume, V, where $\hat{\boldsymbol{n}}$ is the unit normal to δA.

The second conserved quantity is helicity, defined as

$$H = \frac{1}{2} \int_V \boldsymbol{u} \cdot \boldsymbol{\omega} \mathrm{d}V. \tag{1.10}$$

we can determine its conservation by noting

$$\frac{D}{Dt} \boldsymbol{u} \cdot \boldsymbol{\omega} = \boldsymbol{u} \cdot \frac{D\boldsymbol{\omega}}{Dt} + \boldsymbol{\omega} \cdot \frac{D\boldsymbol{u}}{Dt}$$

Thus taking $\boldsymbol{u}\cdot$ the vorticity equation and $\boldsymbol{\omega}\cdot$ the momentum equation and using

15

equations 1.2 and 1.5 we can write

$$\frac{D}{Dt}\boldsymbol{u} \cdot \boldsymbol{\omega} = \boldsymbol{u} \cdot (\boldsymbol{\omega} \cdot \nabla \boldsymbol{u}) - \boldsymbol{\omega} \cdot \nabla \frac{p}{\rho}$$

$$= \frac{1}{2}\boldsymbol{\omega} \cdot \nabla |\boldsymbol{u}|^2 - \boldsymbol{\omega} \cdot \nabla \frac{p}{\rho}$$

$$= \boldsymbol{\omega} \cdot \nabla \phi = \nabla \cdot (\boldsymbol{\omega}\phi)$$

where $\phi = \frac{1}{2}|\boldsymbol{u}|^2 - \frac{p}{\rho}$. Thus integrating over the volume V and employing Gauss' Theorem as before

$$\frac{dH}{dt} = \int_{\delta A} \phi \boldsymbol{\omega} \cdot \hat{\boldsymbol{n}} \mathrm{d}S = 0$$

provided $\boldsymbol{\omega} \cdot \hat{\boldsymbol{n}} = 0$ on δA.

Thus given appropriate boundary conditions T and H are conserved for inviscid flows. Notice these boundary conditions are not restrictive since an inviscid boundary requires the free slip condition, namely

$$\boldsymbol{u} \cdot \hat{\boldsymbol{n}} = 0 \quad \& \quad \boldsymbol{\omega} = \boldsymbol{0}$$

on δA.

1.4   Turbulence and Vortex stretching

In the discussion of Reynolds number above we alighted on the complexity in turbulent flows. The problem of turbulence is one which has plagued physicist, mathematician and engineer alike (Frisch, 1995), (Davidson, 2004). One of the main reasons for this level of interest is due to the ubiquitous nature of turbulent flows. From flow over planes, cars and even buildings, to large scale atmospheric motions, turbulence is perhaps the most important barrier to our understanding of these flows (Davidson, 2004), (Moffatt, 1981), (Batchelor, 1967). The drag

associated with turbulent motions can be associated with a substantial proportion of human energy consumption as we force ourselves and our commodities through the fluids of the air and ocean to transport them. The accuracy of weather and climate forecasting is influenced by our understanding of turbulence, and thus so is our ability to predict adverse or dangerous weather events.

Having satisfied ourselves with the importance of turbulence, we turn to gathering some understanding of certain properties of turbulent motions. The notion of a turbulent fluid is a difficult one to quantify. Leading scientists in the subject today still debate appropriate definitions of turbulence.[2] One inescapable truth is that turbulence is a chaotic process, both in the intuitive physical sense and in the rigorous mathematical one. Observing all of the complexity in an energetic mixing fluid, there is a clear sense of unpredictability. Research over many years has shown this is exactly the case, as fluid motions are highly sensitive to their initial conditions and therefore are typically not reproducible. It is for this reason that much of turbulence theory is interested in the *statistical* properties of turbulence, i.e. what *type* of phenomena the flow is likely to exhibit.

Another commonly-held view in the classification of turbulence is the range of scales of motion that are encompassed. These scales are best characterised by the sizes of the vortices or eddies in the flow. Richardson (Richardson, 1926) and Kolmogorov (Kolmogorov, 1941) popularised the idea of a turbulent cascade, whereby energy in the flow is transferred from large to small scales via a succession of inertial instabilities. The effect of viscosity is to act where shear stress is highest, that being where the eddies are smallest. This is why the cascade is an inertial process, driven entirely by the action of the eddies on themselves. The scale at which viscosity eventually steps in to dissipate the energy is commonly

---

[2] *There has been a long standing tradition in turbulence of studiously avoiding any formal definition of what we mean by a 'turbulent eddy', or for that matter turbulence,* see page 52 of (Davidson, 2004)

known as the Kolmogorov microscale, and beyond this scale the motions are effectively suppressed. The Kolmogorov microscale scales with an inverse power of Reynolds number, and hence the higher the Reynolds number the smaller the scale at which viscosity damps the motion. This implies a fundamental difficulty in modelling turbulence; at physically appropriate Reynolds numbers the range of scales present in the flow may be (and often is) prohibitively large. An example given in (Davidson, 2004) shows that for a wind tunnel experiment at $Re = 1000$ with characteristic length scale $L = 1$cm the Kolmogorov microscale is 0.06mm. i.e. motions from 1cm down to hundredths of a mm require representing, this is 3-4 orders of magnitude!

The mechanisms at work in this cascade are the subject of intense study. However the phenomenology of Richardson suggests a dominating role is played by the process of vortex stretching. Vortex tubes or sheets are intensified by the background strain, and as they are stretched they also must thin due to incompressibility (and the lack of action by viscosity). This process of stretching, combined with the folding of vortex lines, serves to transfer energy from the mean flow into thin coherent vortex structures (Moffat and Kida, 1994), (Vincent, 1994). Ignoring viscosity means circulation is conserved and vortex lines are material lines; they are *frozen-in* to the fluid.

Empirical observations of the inertial range (the range of scales from the characteristic length scale to the Kolmogorov microscale) indicate that turbulent flows are in fact highly intermittent at high Re, that is vorticity is concentrated in exactly the thin, coherent filaments and sheets that one might expect from the vortex stretching process (see section 8.9 of (Frisch, 1995)). These vortex structures in general only occupy a small fraction of the volume in a given region (Chorin, 1994). Given this, it is not surprising, though somewhat frustrating that the vorticity field is a particularly challenging one to measure experimentally. This is one motivation for performing numerical simulations (see section
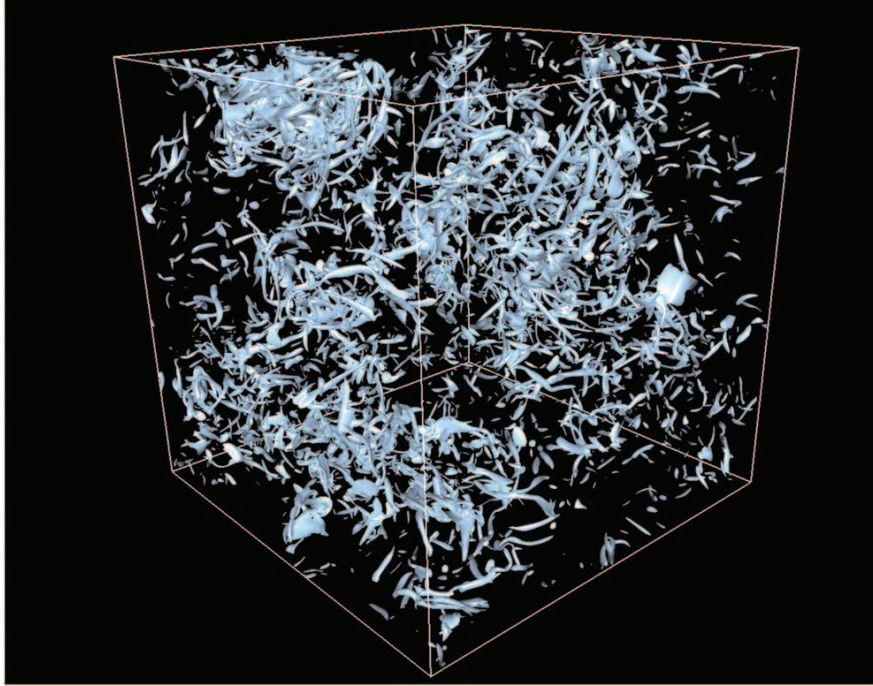
Figure 1.4: Isolevels of the magnitude of vorticity from a high-resolution pseudospectral simulation of homogeneous isotropic forced turbulence. Notice the prevalence of coherent vortex structures (Schumacher et al., 2008).

1.5). However the modelling of turbulence is itself inherently difficult due to the large range of scales which need to be resolved. Without viscosity, one expects this range to extend indefinitely to small scales. This cascade is central to the regularity of the Euler equations, but is there any hope in modelling it?

For even moderate Reynolds numbers, modelling the range of scales required for an accurate representation of the entire flow in a *direct numerical simulation* (DNS) is often acutely computationally expensive. Given that most flows of significance (aerodynamics, atmosphere/ocean flows) have Reynolds numbers on the order of tens or hundreds of thousands it is common practise to parameterise the unresolved scales in the flow. This requires turbulence models to estimate the average influence of the sub-grid processes on the mean flow. This is in part physically justified since the energy cascade is largely direct, in that the large

19

scales feed into the small, without the small scales having much influence on the large (or so it is believed by many). This type of method is known as a *large eddy simulation* (LES) and is useful provided the primary focus is on the larger scale motions, for example in producing a forecast of atmospheric conditions (LES was first proposed in 1963 by Smagorisky in the context of an atmospheric general circulation model (Smagorinsky, 1963)).

The more 'academic' questions surrounding inertial range processes require efficient numerical algorithms to provide insight. This is particularly important for simulating completely inviscid flow where there is no viscous diffusion; the Reynolds number is infinite, and there is no microscale lower bound in resolution. In addition the only method of experimentation at our disposal for an inviscid fluid is numerical. As was mentioned previously, the open question of regularity of the fluid dynamical equations is perhaps even more pointed in the case of zero viscosity.

While weak solutions to the Navier-Stokes equations exist (Fefferman, 2000), weak solutions to the Euler equations have been found to be non-unique (Shnirelman, 1996) . This difficulty is indicative of the problem facing the community in confirming the nature of the solutions to the ideal equations. We have, however, been furnished with various conditional regularity results, perhaps the most important of which is the so called BKM (Beale, Kato, Majda (Beale et al., 1984)) theorem. In simple terms it states that the regularity is controlled by the infinity (maximum) norm of vorticity; solutions will exist provided, that, for any $T > 0$

$$\int_0^T \|\boldsymbol{\omega}(t)\|_\infty \mathrm{d}t < \infty.$$

This result means that a numerical modeller need only observe the growth in vorticity maximum to establish finite-time blow-up evidence. This theorem, combined with the vortex picture of turbulence, provides certain weight to the importance of vortex structures and their stability, the process of vortex stretching and

20

the observed intermittency in turbulent flows. The finite-time blow-up problem, the numerical evidence for it and further conditional regularity results based on the geometry of the vorticity field are discussed further in chapter 4.

1.5   Numerical methods

The numerical simulation of fluid flows plays an extremely important role in the study and analysis of their behaviour. While physical experiments continue to provide important results, the computer age has allowed for numerical solutions to be carried out without the expense and time involved in constructing elaborate laboratories. In addition numerical calculations allow for a greater variety of variables and diagnostics to be monitored and stored during an experiment. There are also instances where physical experiments are impractical, for example large scale atmosphere/ocean scenarios as applicable to weather and climate forecasting. Unfortunately the computation of solutions to (1.1) have their own unique set of challenges.

The development of such numerical calculations in the context of fluid flows has its roots in the finite-difference calculations performed by (Richardson, 1922). These calculations were performed by hand in order to demonstrate the possibility of modelling and predicting the state of the atmosphere and thereby provide a forecast of weather conditions. In 1922 this idea was far ahead of its time, indeed Richardson realised it would require 60,000 people to perform calculations sufficiently quickly to enable a forecast to be made in time (O'Connor and Robertson, 2011b). However his pioneering work initiated the branch of modern applied mathematics now know as numerical analysis, and since the advent of computers, the field has become one of the most important in all of mathematics, with the numerical solution of PDEs utilised in an enormous number of industrial and academic applications.

There now exist a plethora of numerical techniques for the solution of fluid flow problems, each with their own set of advantages and disadvantages. For completeness a brief summary of a selection of methods is given to provide some context for the description of the numerical code to follow in chapter 3.

1.5.1 Grid based methods

The prototype example for a numerical method is the one used by Richardson known as finite difference, whereby the equation of interest is discretised on some grid of points (methods which make use of some fixed grid of points are known as Eulerian methods). Gradients are then approximated by a difference equation across adjacent grid points. Over the entire domain a linear system of equations is obtained for the spatial discretisation, which is then inverted, generally by some iterative linear solver. This results in a system of ODEs in time which can then be integrated with an ODE solver (e.g. Runge-Kutta, Adams-Bashforth), often referred to as time-stepping. The numerical scheme is then constrained by stability criteria determined by the stencil shape, the size of the time step and the grid length. A necessary but not sufficient stability criterion was developed by Courant, Friedrich and Lewy, and is known as the CFL condition, where the Courant number, a ratio of time step to grid length dictates stability.

There are time integration schemes available which avoid the CFL constraint. An example, which has garnered popularity in the geophysical setting is the semi-Lagrangian scheme, whereby variables are updated on grid points given the Lagrangian position of the fluid element at the previous time-step. This requires, in general, some interpolation which has been shown to be dissipative (see the review (Staniforth, 1991) and (Dritschel et al., 1999)).

The finite difference methods remain popular largely due to their ease of implementation; a more sophisticated grid based method (in some respects a

generalisation of finite difference) is the finite element method (FEM). FEM is popular in engineering applications where mesh generality and adaptation are advantageous (or indeed essential). In this case the discrete set of equations is obtained from the weak form of the equations, having multiplied first by some test function and subsequently integrated by parts to reduce the order of the derivatives. The space of test functions is extremely general. Most often piecewise polynomials make up the finite elements over the discretised space. The generality of this approach allows for a range of spatial adaptations to be employed, commonly classified as one of three types. Refinement which adjusts the mesh topology without changing the number of computational elements is known as *r-refinement*; increasing resolution by means of an increase (or decrease) of nodes in the mesh is known as *h-refinement*; and finally an adjustment of the order of the element polynomials is denoted *p-refinement*. The elaborate nature of this method means it is more favoured by modellers involved with complex geometries and/or moving boundaries rather than those studying large problems where many scales require to be discretised.

Fluid mechanics, particularly in an industrial setting, favours another method which shares certain similarities with FEM, the finite volume method (FVM). The discretisation is again very general, allowing for h and r-refinement; mesh nodes are centred on small computational control volumes, over which the PDE is integrated, allowing divergence terms to be converted to surface integrals via Gauss' Theorem. This makes for a conservative scheme as fluxes across the control volumes are explicitly accounted for. Many commercial computational fluid mechanics (CFD) codes favour this method due to its mesh generality and relatively low memory cost (see figure 1.5 for an example of a complex FVM mesh used in an industrial context). FVM will be discussed in more detail in appendix A (for further reading see (Versteeg and Malalasekera, 2007)).

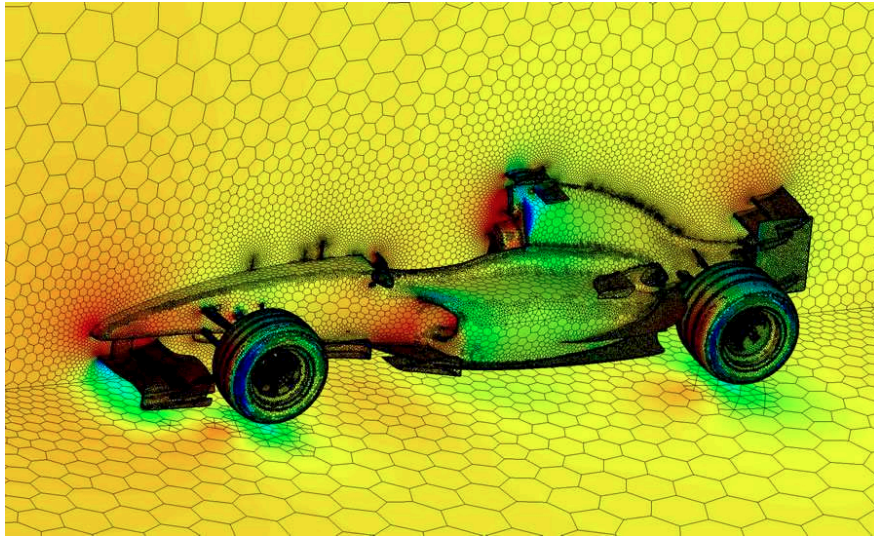Another Eulerian method which can be compared to the FEM most closely

Figure 1.5: Adapted unstructured polyhedral mesh with pressure distribution on an F1 car using ANSYS CFD FVM Software (Analysis, 2011).

is the spectral method. In this case the discretisation utilises a global basis of Fourier modes, i.e. a discrete and finite series in wavenumber space, but with each Fourier component continuous over the whole domain (as opposed to a small subset). The 'Fast Fourier Transform' (FFT) algorithm allows for the transfer between physical and spectral spaces, and derivatives are now easily performed as wavenumber multiplication (see e.g. (Orszag, 1971)). A variation of the spectral method is the pseudospectal method where a gridded discretisation is transferred between physical space, where nonlinear products are carried out, and spectral space, where derivatives and time-stepping take place. These methods are extremely high order (for sufficiently smooth fields) and give superior convergence compared to finite-difference based methods (see (Hussaini, 1987) for a review of the use of these methods in fluid mechanics). However, due to the global support of the basis functions it is restricted to uniform resolution (it does not allow for any local refinement) and is limited to periodic boundary conditions in general. Certain extensions have been developed for e.g. complex geometries (Orszag, 1980) and a combination of FEM and Fourier bases has resulted in the spectral

24

element method (Henderson, 1995).

It is also possible to develop finite-difference based methods with some h-refinement. One such example is the ADI (alternating direction implicit) method (Douglas Jr et al., 1963), whereby one iterates across clustered patches of the domain in each direction, at each refinement level, applying boundary conditions from the level above. This method is simple to implement as each direction will yield a tridiagonal system which is quick to solve via the Thomas algorithm. A much more common approach is to implement a multigrid method (Briggs et al., 2000), where nested patches of refinement are cycled over, where in each pass iterations are performed (relaxation) with corrections obtained from residuals on the refined grid. This is known as V-cycling; one ascends from fine to coarse, computing residuals (of residuals...) and then descends correcting and relaxing at each level. Multigrid has gained much popularity in numerical analysis in a purely algebraic setting, i.e. algebraic multigrid (AMG), as a method of solving any linear system.

1.5.2   Vortex methods

Thus far all of the methods discussed have been purely Eulerian in that they involve computing the solution on a grid which is fixed in space. While r-refinement allows for a certain degree of mesh movement, and FEM can handle complex and moving boundaries, the frame of reference for the solution of the PDE is still fixed. All of these methods will suffer, to some degree or another, from CFL type stability constraints. Lagrangian methods on the other hand integrate the equations via computational elements which are not fixed to a grid but are free in space to move, with the flow, independently of each other.

Perhaps the most widely used grid-free methods are the vortex methods. These methods generally employ a finite number of computational elements to

represent the vorticity field. This can potentially give a computational advantage as vortex elements are only required on the subset of the domain where vorticity is non-zero. The elements are then advected with the flow field, computed by the inversion of an elliptic problem. In standard vortex methods this inversion is given by the Biot-Savart integral (1.7). The advection must also be supplemented in 3D with an update of the circulations of the elements, given by the local strain field. Due to this Lagrangian approach, vortex methods can be thought of as being self adaptive; vortex elements will naturally tend to cluster where computational effort is required.

This is a broad overview of the general vortex method, but as with most numerical methods there is much diversity as research has innovated and applied them to various problems (see (Cottet and Koumoutsakos, 2000)). One main area of innovation has been to increase the efficiency of the algorithms which were traditionally slow due to the solution of essentially an N-body problem for the vortex elements (computations scale as $O(N^2)$ for N vortex elements). Domain decomposition and multipole expansions have seen this expense greatly reduced (see appendix B of (Cottet and Koumoutsakos, 2000) for example). Another strategy in reducing this computational burden is to implement a so-called hybrid method, whereby an underlying Eulerian grid is employed to invert, for example (1.6), and interpolation transfers variables between the grid and the elements. This type of method is often referred to as particle or vortex-in-cell (Christiansen, 1973). In this setting it is possible to grasp some of the further advantages of the vortex method; where a standard grid-based method will have a representation which terminates at the resolution of the grid, vortex-in-cell methods have a representation which can continue far beneath that scale. Indeed some workers go further still to correct the sub-grid vorticity due to its local interactions (Anderson, 1986).

There is also a wide variety of approaches to the treatment of the vortex ele-

ments themselves. Most often the elements are simply point vortices with some regularising (or interpolating) kernel. In two dimensions this is somewhat simplified due to the material conservation of the scalar vorticity. An alternative strategy which has seen great success in 2D and 'quasi-geostrophic' regimes has been the use of contours as vorticity elements, to create a piecewise-constant vorticity representation (see figure 1.6) (Dritschel, 1988b) (Dritschel and Ambaum, 1997) (Fontane and Dritschel, 2009).
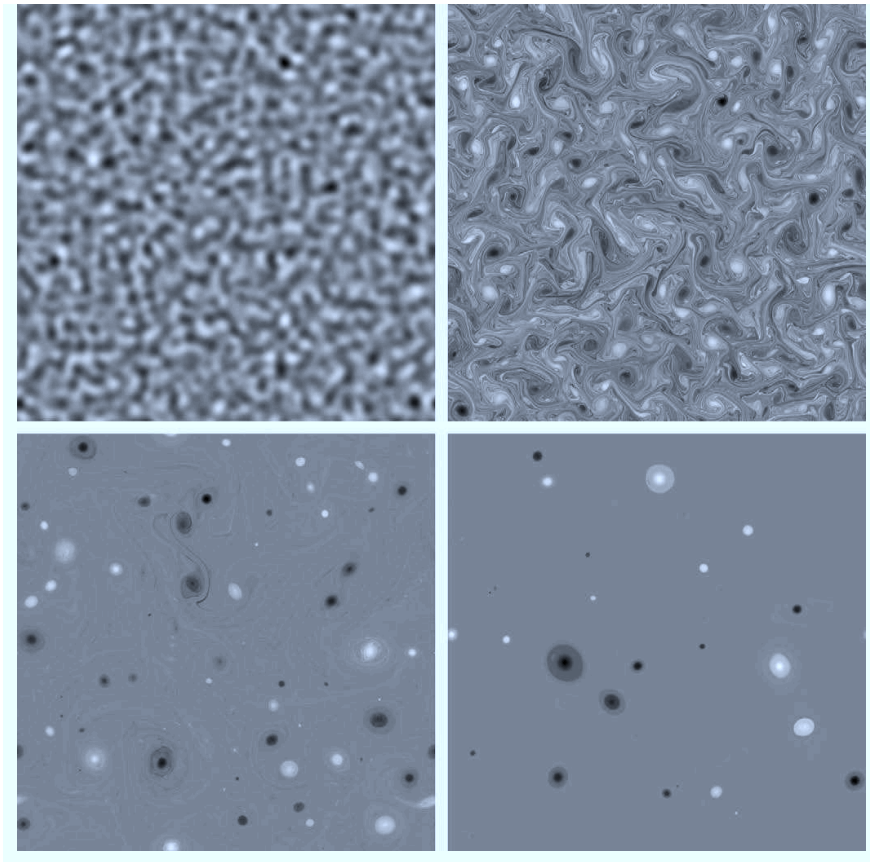

Figure 1.6: Time series of potential vorticity (PV) from a two-dimensional, unforced, inviscid, quasi-geostrophic turbulence simulation using the 'hyperCASL' algorithm which employs a number of vortex method features, including vorticity (here PV) contours, point vortices and an Eulerian grid for various elements of the computation (Fontane and Dritschel, 2009).

The situation is complicated somewhat in three dimensions as vorticity is no longer a conserved scalar. Then Helmholtz/Kelvin circulation theorem states that circulation is conserved, implying that vortex lines are materially advected for inviscid flow. In computations this requires the circulations of each vortex element be updated due to vortex stretching. An extension of the particle method is to create vortex filaments, with each vortex element representing a segment of some piecewise linear space curve representing the centre of a vortex filament having a core width given by some regularising or interpolation kernel (Cottet and Koumoutsakos, 2000). This means vortex stretching can be implicitly accounted for simply by the separation of the adjacent nodes on the space curve ((Leonard, 1985) is a good review of a variety of these methods). For particle methods where the elements are subjected to high strain, it is necessary to regrid the particles onto a uniform array to maintain the overlap requirement for convergence of the method (Cottet et al., 1999) (Beale and Majda, 1982). A significant advantage of the filament method is that the connectivity of the elements allows for easy spatial adaptation of the Lagrangian representation and hence removes the need for regridding which inherently introduces some dissipation into the method.

1.5.3   Validation of numerics

Any new numerical method must be properly validated to convince the user of its accuracy and robustness. While convergence results provide a rigorous mathematical foundation for a particular scheme, the process of writing the code is a minefield of bugs and blind alleys. For these reasons it is extremely important that a thorough validation is performed. Simulations of well documented flows are the best way of doing this. However, few comprehensive mathematical results are available to benchmark a new method, particularly one concerned with 3D vortex structures.

One class of results which performs this task very well consists of vortex steady states or equilibria. In two dimensions these are manifold, e.g. Kirchhoff's ellipses (Kirchhoff, 1876) and the more general results of (Dritschel, 1985). However, in three-dimensions the results are far more sparse. The seminal work of (Norbury, 1973) presents a set of axisymmetric vortex rings with a vorticity profile proportional to radius r, parameterised by a mean core radius (cross-sectional area). These results combined with the extensive literature on the steady translation and stability of vortex rings (Widnall et al., 1974),(Saffman, 1978),(Shariff and Leonard, 1992) make them the natural choice for validating a new 3D vortex method.

Relying purely on a single example is, however, bad practise, particularly one with a strong symmetry like the vortex ring, regardless of how extensive the literature is.

1.5.4 Synopsis

In chapter 2 we address the shortage of 3D vortex equilibria and present a new class which have helical symmetry (this work is published in (Lucas and Dritschel, 2009)). Helical vortices, while having previously no known steady states, do have a rich literature describing their stability (Widnall, 1972),(Okulov, 2004), (Walther et al., 2007) as their existence in rotor wakes has made them of considerable interest to engineers, particularly in recent years in the context of wind turbine wakes (Okulov and Sørensen, 2007a).

Having motivated the importance of the process of vortex stretching in section 1.4, chapter 3 details a new hybrid vortex method which combines a vortex filament approach with renoding ideas from the contour methods of Dritschel (Dritschel, 1988b) and an inversion carried out using a FVM code taking advantage of h-refinement. The goal is to model complex vortex structures in a

completely automatic and self-similar way over as broad a range of scales as possible.

Chapter 2

Helical Vortex Equilibria

The following chapter consists of a study published in The Journal of Fluid Mechanics (Lucas and Dritschel, 2009) and the work is reproduced here with minor modifications.

2.1   Introduction

The generation of helical tip vortices in rotor wakes is of major significance in the study of many applications of propeller and wind turbine flow dynamics. Recent research has highlighted the importance of such wakes trailing wind turbines and the subsequent consequences of strong tip vortices interacting with other turbines in the wind farm; (Okulov and Sørensen, 2007b), (Walther et al., 2007). These studies model the wake by a system of N tip vortices, which are infinitely long, slender helical vortices, equally azimuthally spaced and an additional axial hub vortex.

(Widnall, 1972) was the first to consider the linear stability of a helical vortex filament by calculating the self-induced velocities due to sinusoidal perturbations of the filament. Studies presented by Okulov and co-workers (Okulov, 2004),

(Fukumoto and Okulov, 2005), (Okulov and Sørensen, 2007b) have continued this procedure of evaluating the induced velocity field and have extended it to a multiplicity of vortices to address the problem of the rotor wake. Such research has remained predominately asymptotic and has been concerned almost exclusively with vortex filaments of small core radius e.g. $R_{core} \ll d_{cen}$, where d_{cen} is the distance from the centre of the filament from the central axis.

Despite this recent interest in helical vortex stability, there has been no general theory to enable a complete description of helical vortex equilibria of arbitrary size. In a classical paper in fluid dynamics, Norbury (Norbury, 1973) computed, numerically, equilibria for the axisymmetric problem of a vortex ring. This class of equilibria is parameterised by a mean core radius, and each equilibrium state consists of a single closed contour bounding a distribution of azimuthal vorticity which is proportional to r, the distance from the axis of symmetry of the ring. Axisymmetric vortex rings are amongst the most widely studied fluid dynamical structures and have a multitude of applications, both in physical problems and as mathematical constructs.

This work aims to adapt Norbury's approach and apply it to that of helical vortices. We make use of helical symmetry to compute equilibria of constant pitch ($\epsilon = 1$) by considering a cross-section of constant height, z, and parameterising the family of equilibria not only by a mean radius, $\bar{R}$, of arbitrary size, but also by a centroid position, d, which can also be considered as the radius of the vortex system (in applications to wind turbine wakes, d is the rotor radius). We find that for such helically symmetric flows, and for a special choice of the velocity component parallel to vortex lines, we have material conservation of axial vorticity, ω [1] in an absolute frame of reference. We may thus consider

---

[1]Such conservation was first noticed in (Dritschel, 1991) but ω was not identified with axial vorticity. Another interesting study (Alekseenko et al., 1999) also mentions this point and is worthy of mention here

a contour bounding a uniform distribution of ω for which a constant helical stream function, ψ, upon this boundary in a rotating frame of reference implies an equilibrium state. The two main problems posed by this approach are in inverting a linear helical operator $\mathscr{L}\psi$ to compute velocities upon the contour and also in parameterising the contour itself. The former is performed via a combination of spectral methods and finite differences and the parameterisation uses a particularly effective "time-travel" coordinate.

In the first instance we consider the case of single vortex equilibrium states and then extend our method to N equally spaced vortices in azimuthal angle.

2.2   Helical symmetry

Let us consider an unbounded, inviscid, incompressible fluid which possesses helical symmetry (Landman, 1990), (Dritschel, 1991)), i.e. invariant to rotation and translation. We are able to reduce the problem to two dimensions by considering a cylindrical polar description (r, θ, z) and introducing the helical coordinate $\phi = \theta + \epsilon z$. Here ϵ is the pitch of the helix taken in our equilibria computations to be $\epsilon = 1$ without loss of generality ($\epsilon = 0$ recovers two-dimensional flow, $\epsilon \to \infty$ the axisymmetric case, and $\epsilon = 1$ represents a helix which performs a single twist in an axial distance of 2π, see figure 2.1). We now introduce the helical vector $\boldsymbol{h}$ defined as orthogonal to ∇r and $\nabla \phi$

$$
\begin{aligned}
\boldsymbol{h} &= h^2(\boldsymbol{e}_z - \epsilon r \boldsymbol{e}_\theta) \\
h^2 &= (1 + \epsilon^2 r^2)^{-1}
\end{aligned}
$$

Helical symmetry now implies that velocity, vorticity and pressure do not vary with $\boldsymbol{h}$ and $\boldsymbol{h} \cdot \nabla$ applied to any scalar function of r, ϕ and time, t, is zero. We may then define the unit vector for the helical coordinate ϕ by
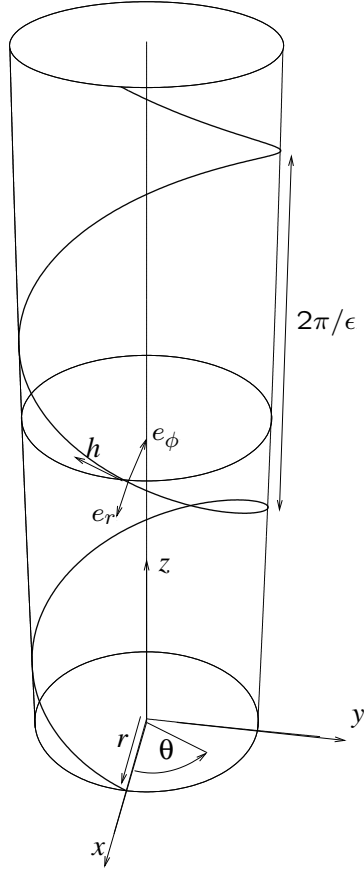
33

Figure 2.1: Schematic of the helical coordinate system. Note that the vectors $\boldsymbol{h}$ and $\boldsymbol{e}_\phi$ lie on the surface of the cylinder $r = $ constant. For the special class of flows considered, the vorticity $\boldsymbol{\omega}$ is everywhere tangent to $\boldsymbol{h}$.

$$\boldsymbol{e}_\phi = h^{-1}\boldsymbol{h} \times \boldsymbol{e}_r = h(\boldsymbol{e}_\theta + \epsilon r \boldsymbol{e}_z)$$

which reduces the gradient operator to $\nabla = \boldsymbol{e}_r \partial/\partial r + \boldsymbol{e}_\phi (rh)^{-1} \partial/\partial \phi$. We are now able to decompose the velocity and vorticity fields in terms of helical scalar functions

$$\boldsymbol{u} = \boldsymbol{h} \times \nabla \psi + \boldsymbol{h}\upsilon \tag{2.1}$$

$$\boldsymbol{\omega} = \boldsymbol{h} \times \nabla \chi + \boldsymbol{h}\zeta \tag{2.2}$$

which automatically enforces the incompressibility and divergence-free vorticity conditions. The individual velocity components can be expressed as

$$u_r = -\frac{1}{r}\frac{\partial \psi}{\partial \phi} \tag{2.3}$$

$$u_\theta = h^2 \left(\frac{\partial \psi}{\partial r} - \epsilon r \upsilon \right) \tag{2.4}$$

$$u_z = h^2 \left(\upsilon + \epsilon r \frac{\partial \psi}{\partial r} \right) \tag{2.5}$$

The definition of vorticity $\boldsymbol{\omega} = \nabla \times \boldsymbol{u}$ furnishes us with the well known helical equation coupling vorticity and velocity (Landman, 1990), (Dritschel, 1991),

$$\mathscr{L}\psi \equiv \frac{1}{r}\frac{\partial}{\partial r}\left(rh^2\frac{\partial \psi}{\partial r} \right) + \frac{1}{r^2}\frac{\partial^2 \psi}{\partial \phi^2} = \omega + 2\epsilon h^4 \upsilon \tag{2.6}$$

where $\omega = \boldsymbol{h} \cdot \boldsymbol{\omega} = h^2 \zeta$. We must also have that $\chi = -\upsilon$.

We seek steadily rotating vortex solutions in a flow which is irrotational as $r \to \infty$ (in fact the flow will be rotational only within one or several helical vortex tubes). To this end it is convenient to express the flow in a rotating frame of reference (ω_R, υ_R, ψ_R etc.)

35

$$\boldsymbol{\omega} \;=\; \boldsymbol{\omega}_R + 2\boldsymbol{\Omega}$$

$$\boldsymbol{u} \;=\; \boldsymbol{u}_R + \boldsymbol{\Omega} \times \boldsymbol{x}$$

where $\boldsymbol{\Omega} = \Omega \boldsymbol{e}_z$ is the rotation rate (to be chosen so that the flow is steady in this frame). Taking the scalar product of $\boldsymbol{h}$ with these equations gives

$$\boldsymbol{h} \cdot \boldsymbol{\omega} \;\equiv\; \omega = \omega_R + 2\Omega h^2 = \omega_R + \bar{\omega}(r)$$

$$\boldsymbol{h} \cdot \boldsymbol{u} \;=\; h^2 v = h^2 v_R + \boldsymbol{h} \cdot (\boldsymbol{\Omega} \times \boldsymbol{x}) = h^2 v_R - \epsilon r^2 h^2 \Omega$$

i.e. $v = v_R - \epsilon r^2 \Omega = v_R + \bar{v}(r)$.

Similarly we can express the stream function as $\psi = \psi_R + \bar{\psi}(r)$ and use (2.6) to derive $\bar{\psi}(r)$;

$$\mathscr{L} \bar{\psi}(r) = \bar{\omega}(r) + 2\epsilon h^4 \bar{v}(r) = 2\Omega h^4$$

and solving for $\bar{\psi}(r)$ gives

$$\bar{\psi}(r) = \frac{1}{2}\Omega r^2 \tag{2.7}$$

given that we require bounded velocities.

Now turning to the full nonlinear dynamical equations in the rotating frame we have from (Dritschel, 1991)

$$\frac{\partial v_R}{\partial t} + J(\psi_R, v_R) = -2\epsilon\Omega \frac{\partial \psi_R}{\partial \phi} \tag{2.8}$$

$$\frac{\partial \omega_R}{\partial t} + J(\psi_R, \omega_R) = 2\epsilon\Omega h^2 \frac{\partial v_R}{\partial \phi} + 2\epsilon h^4 \left(J(\psi_R, v_R) - \epsilon v_R \frac{\partial v_R}{\partial \phi} \right) \tag{2.9}$$

where $J(f, g)$ is the Jacobian, defined in helical coordinates as

$$J(f, g) = \frac{1}{r}\left(\frac{\partial f}{\partial r}\frac{\partial g}{\partial \phi} - \frac{\partial g}{\partial r}\frac{\partial f}{\partial \phi} \right).$$

and the terms proportional to Ω in (2.8) and (2.9) account for the effect of background rigid rotation.

Rewriting (2.8) and (2.9) in terms of the absolute frame variables υ and ω, we find

$$\frac{\partial \upsilon}{\partial t} + J(\psi_R, \upsilon) - \frac{1}{r}\frac{\partial \psi_R}{\partial \phi}2\epsilon\Omega r = -2\epsilon\Omega\frac{\partial \psi_R}{\partial \phi}$$

$$\frac{\partial \upsilon}{\partial t} + J(\psi_R, \upsilon) = 0 \tag{2.10}$$

and (2.9) becomes

$$\frac{\partial \omega}{\partial t} + J(\psi_R, \omega) - \frac{1}{r}\frac{\partial \psi_R}{\partial \phi}4\epsilon^2\Omega rh^4 = \tag{2.11}$$

$$2\epsilon\Omega h^2\frac{\partial \upsilon}{\partial \phi} + 2\epsilon h^4\left(J(\psi_R, \upsilon) - 2\epsilon\Omega\frac{\partial \psi_R}{\partial \phi} - \epsilon(\upsilon - \bar{\upsilon})\frac{\partial \upsilon}{\partial \phi}\right) \tag{2.12}$$

$$\frac{\partial \omega}{\partial t} + J(\psi_R, \omega) = 2\epsilon\Omega h^2\frac{\partial \upsilon}{\partial \phi}(1 - \epsilon^2 r^2 h^2) + 2\epsilon h^4\left(J(\psi_R, \upsilon) - \epsilon\upsilon\frac{\partial \upsilon}{\partial \phi}\right) \tag{2.13}$$

$$= 2\epsilon h^4\left(J(\psi_R, \upsilon) - \epsilon\upsilon\frac{\partial \upsilon}{\partial \phi}\right) \tag{2.14}$$

Hence, for the special case $\upsilon = $ constant, considered henceforth, it follows that the quantity ω is *materially conserved*

$$\frac{D\omega}{Dt} \equiv \frac{\partial \omega}{\partial t} + J(\psi_R, \omega) = 0. \tag{2.15}$$

For a general helically symmetric flow with bounded momentum, it can be shown that υ must in fact be equal to the conserved circulation multiplied by $-\epsilon/2\pi$. This is a non-trivial result and will be discussed in further detail in section 2.3. Since $\chi = -\upsilon = constant$ in (2.2), we can express the full vorticity field in terms of ω

$$\boldsymbol{\omega} = \omega\boldsymbol{e}_z - \epsilon r\omega\boldsymbol{e}_\theta. \tag{2.16}$$

Note that $\boldsymbol{\omega}$ is everywhere tangent to $\boldsymbol{h}$. It is the conservation of ω in (2.15) which provides us with a means to compute equilibria consisting of a contour

37

bounding a uniform distribution of ω in a constant z cross section and irrotational flow in the absolute frame away from the vortex. Each equilibrium state must then satisfy $J(\psi_R, \omega) = 0$ i.e. for a single contour, the quantity $\psi_R = \psi - \frac{1}{2}\Omega r^2$ must be constant on the contour.

2.3 Inverting $\mathscr{L}\psi$

Having determined the prescription for an equilibrium state in terms of the helical stream function, ψ, we are faced with the task of accurately inverting the helical operator $\mathscr{L}\psi$. In this section we present a numerical method based on both Fourier transforms and finite differences to perform this inversion for a general distribution of axial vorticity. In addition we verify the method against an analytic solution for a particular distribution of ω.

The first step is to express ψ and ω as Fourier series in ϕ

$$\psi = \sum_m \hat{\psi}_m(r)e^{im\phi} \qquad \& \qquad \omega = \sum_m \hat{\omega}_m(r)e^{im\phi}$$

allowing us to rewrite (2.6) as

$$\tilde{\mathscr{L}}_m\hat{\psi}_m = \frac{d}{dr}\left(rh^2\frac{d\hat{\psi}_m}{dr}\right) - \frac{m^2}{r}\hat{\psi}_m = r\hat{\omega}_m \qquad (2.17)$$

for $m > 0$. Numerically we are able to use the fast Fourier transform algorithm (with 512 wavenumbers) to transfer between physical and spectral space in this way, and hence solve (2.17) for each azimuthal wavenumber $m > 0$ by approximating the equation by central differences (we address the $m = 0$ case later). In doing this we obtain a symmetric tridiagonal system of difference equations, easily inverted via the Thomas algorithm. To solve for large radii a novel approach is employed whereby we divide the radial grid in two parts, with a uniform grid

close to the origin and a stretched grid, scaled by h, for large radius, enabling increased accuracy for smaller r. Specifically we use the coordinates

$$s = \begin{cases} r & r < r_m \\ b + crh & r > r_m \end{cases}$$

where r_m is the grid transition radius, chosen such that the vorticity is fully contained within this radius. We integrate (2.17) up to $s = s_N = c + b$ corresponding to $rh = 1$, i.e. $r = \infty$. The constants c and b are given by continuity conditions at r_m

$$\left. \frac{ds}{dr} \right|_{r=r_m} = 1 \quad \Rightarrow \quad c = h_m^{-3}$$

$$s_m = r_m \quad \Rightarrow \quad b = r_m(1 - ch_m) = -\epsilon r_m^3$$

so $s = h_m^{-3}rh - \epsilon r_m^3$. In the calculations to follow a total of 400 intervals in r and s are used and a comparison with an analytic test solution is presented in section 2.3.1.

We now turn attention to the problem of the axisymmetric ($m = 0$) mode, ψ_{axi}, and first note that we are able to decompose this mode into a v-independent part, $\hat{\psi}_0$ and a v-dependent part, ψ_v

$$\psi_{axi} = \hat{\psi}_0 + \psi_v$$

where ψ_v satisfies

$$\frac{d}{dr}\left(rh^2 \frac{d\psi_v}{dr} \right) = 2\epsilon v r h^4$$

which we can solve directly to give

$$\psi_v = \frac{1}{2}v\epsilon r^2 \qquad (2.18)$$

given that we require bounded velocities at the origin.

Now for large r we can rewrite our expression for the axial velocity, (2.5), purely in terms of the axisymmetric mode, (since the contribution from all other modes tends to zero in this limit)

$$\lim_{r\to\infty} u_z = \lim_{r\to\infty} \left\{ v + \epsilon r h^2 \frac{d\hat{\psi}_0}{dr} \right\}.$$

In order for the axial momentum of the system to remain bounded we require that this limit goes to zero and we can satisfy this constraint by setting

$$v = -\lim_{r\to\infty} \epsilon r h^2 \frac{d\hat{\psi}_0}{dr}; \qquad (2.19)$$

a constant. It is possible to relate this constant to another conserved quantity, namely circulation, defined as

$$\Gamma \equiv \int\int_A \omega \, \mathrm{d}A = \oint_C \boldsymbol{u}.d\boldsymbol{x}$$

(this is conserved due to material conservation of ω and incompressibility). Here C is a contour sufficiently large to bound all the vorticity. Substituting in the velocity components (2.3) & (2.4) and considering a contour for constant z we can express circulation as

$$\Gamma = \oint_C rh^2 \left(\frac{\partial\psi}{\partial r} - \epsilon rv \right) \mathrm{d}\phi - \oint_C \frac{1}{r}\frac{\partial\psi}{\partial\phi}\mathrm{d}r$$

Consider now a circular contour of radius $r \to \infty$. Noting that for large r the stream function ψ becomes independent of ϕ, since in this region $\omega = 0$, we find

40

$$\Gamma = 2\pi \lim_{r \to \infty} \left(rh^2 \frac{d\hat{\psi}_0}{dr} \right) = -\frac{2\pi\upsilon}{\epsilon} \quad \Rightarrow \quad \upsilon = -\frac{\epsilon\Gamma}{2\pi}$$

using (2.19). Considering now the calculation for the complete axisymmetric stream function we first make the following observations. From (2.17) for $m = 0$ we can write

$$rh^2 \frac{d\hat{\psi}_0}{dr} = \int_0^r r'\hat{\omega}_0(r')\mathrm{d}r' \qquad (2.20)$$

which tends to $\Gamma/2\pi$ for $r \to \infty$. We can now add the contribution from ψ_υ, having substituted for υ into (2.18), and rewrite (2.20) as the difference of two integrals (from 0 to ∞ minus from r to ∞) giving

$$
\begin{aligned}
rh^2 \frac{d\psi_{axi}}{dr} &= \frac{\Gamma}{2\pi}\left(1 - \epsilon^2 r^2 h^2\right) - \int_r^\infty r'\hat{\omega}_0(r')\mathrm{d}r' \\
\Rightarrow \quad \frac{d\psi_{axi}}{dr} &= \frac{\Gamma}{2\pi r} - \frac{1}{rh^2}\int_r^\infty r'\hat{\omega}_0(r')\mathrm{d}r'
\end{aligned}
$$

showing that $d\psi_{axi}/dr$ tends to 0 as $r \to \infty$ like $\Gamma/2\pi r$. Numerically, we use the equivalent expression obtained by adding (2.20) and $d\psi_\upsilon/dr$:

$$\frac{d\psi_{axi}}{dr} = S_0(r)$$

where

$$S_0(r) = \frac{1}{rh^2}\int_0^r r'\hat{\omega}_0(r')\mathrm{d}r' - \frac{\epsilon^2 r\Gamma}{2\pi}$$

which we have already shown tends to $\Gamma/2\pi r$ as $r \to \infty$. Integrating S_0 with respect to r produces a logarithmic singularity as $r \to \infty$. To avoid this numerically, the singularity is treated explicitly by adding and subtracting the function

$$f(r) \equiv \frac{\epsilon^2 r h^2 \Gamma}{2\pi}$$

from the right hand side, using

$$f(r) = -\frac{\Gamma}{2\pi} \frac{d \log h}{dr}$$

leading to

$$\psi_{axi} = \frac{\Gamma}{2\pi} \log h + \int_r^\infty (f(r') - S_0(r')) \, dr'$$

Now both the integrand and the integral are finite for all radii. We proceed by this technique in our numerical inversion, computing the integral via the trapezoidal rule.

2.3.1 Analytic Solution

It is possible to construct analytic solutions to equation (2.17) via the use of Green's functions for the case of circular, compact distributions of $\hat{\omega}$, corresponding to $\omega = \hat{\omega}(r)e^{im\phi}$. Following the general theory of Green's functions we seek a solution of the form

$$\hat{\psi}(r) = \int_0^\infty G(r; r_0) r_0 \hat{\omega}(r_0) dr_0$$

where G is Green's function and satisfies the homogeneous equation, $\tilde{\mathcal{L}}G = 0$ (see (2.17), here we take $\epsilon = 1$ consistent with the results presented). It can be shown that Green's function has the form

$$G(r; r_0) = \begin{cases} rr_0 I'_m(mr) K'_m(mr_0) & r < r_0 \\ rr_0 K'_m(mr) I'_m(mr_0) & r > r_0 \end{cases}$$

where I_m and K_m are the modified Bessel functions of the first and second kind, of order m, and a prime denotes a derivative with respect to the argument.

Consider an idealised axial vorticity defined by

$$\hat{\omega}(r) = \begin{cases} 1 & r < R \\ 0 & r > R \end{cases}$$

42

for a circular cross-section, radius R. Then the stream function becomes

$$
\hat{\psi} =
\begin{cases}
-rK'_m(mr)\int_0^r r_0^2 I'_m(mr_0)\mathrm{d}r_0 - rI'_m(mr)\int_r^R r_0^2 K'_m(mr_0)\mathrm{d}r_0 & r < R \\
-rK'_m(mr)\int_0^R r_0^2 I'_m(mr_0)\mathrm{d}r_0 & r > R
\end{cases}
$$
(2.21)

This analytic solution for $\hat{\psi}$ allows the numerical inversion to be accurately verified for this particular distribution of axial vorticity.

Figure 2.2 displays plots showing the difference between the analytically computed solution (e.g. (2.21) with integrals computed via a midpoint rule), and the numerically computed solution computed on the two part grid as described in section 2.3. We have very good agreement and suffer partial loss in accuracy across the boundary of the circular vortex patch ($r = R$) since the piecewise constant distribution can only be approximately represented on a discrete grid.

2.4 Computing equilibrium states

Having discussed the method of computing the helical stream function we are now able to address the problem of computing the equilibrium states. As mentioned in section 2.2 the equilibrium states are those for which $\psi - \frac{1}{2}\Omega r^2$ is constant upon a contour bounding a uniform distribution of ω. We now consider the problem in terms of a contour with $z = 0$, i.e. a vertical cross-section and consider such a contour bounding a region of uniform axial vorticity, $\omega = 1$.

We expand the constraint on ψ about the previous iteration or 'basic state' and linearise, computing both the contour perturbation and angular velocity Ω at each iteration, while preserving area and centroid position. This process is repeated with the updated contour becoming the basic state until the maximum correction falls inside a prescribed threshold (10^{-12}) compared to the mean radius.
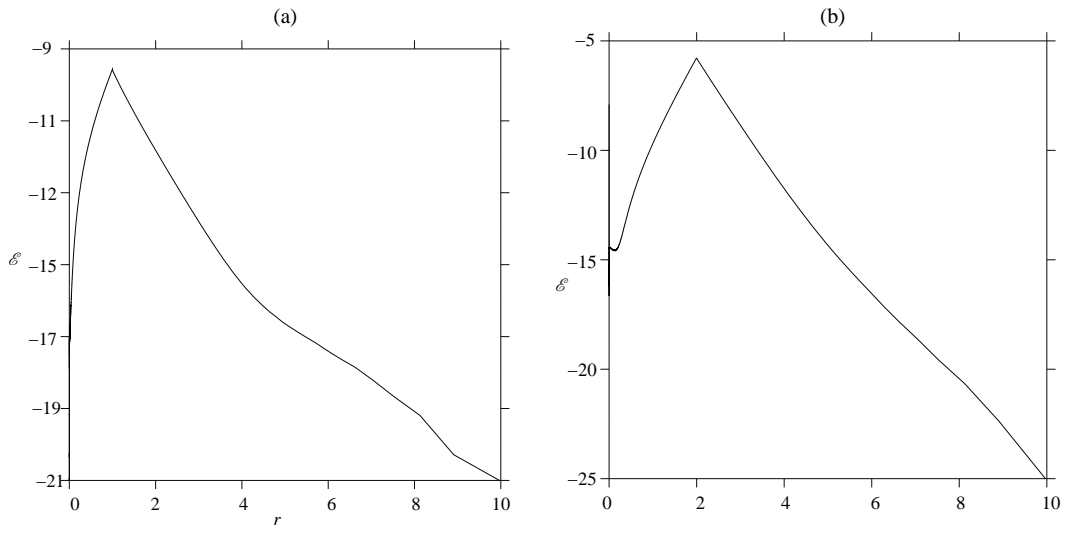
Figure 2.2: Plots showing difference in the numerical calculation of $\hat{\psi}$ and the analytic solution (2.21) ($\mathscr{E} = \log_{10} |\hat{\psi}_{ana} - \hat{\psi}_{num}|$) for a circular distribution with radius $R = 1$, wave number $m = 2$ (a) and radius $R = 2$, wave number $m = 3$ (b).

The states are thus parameterised by the mean radius $\bar{R}$ and centroid d and the numerical algorithm spans families of states for a given $\bar{R}$ or d.

Let the new contour be $\boldsymbol{x} = \boldsymbol{x}_0 + \boldsymbol{x}'$ where $\boldsymbol{x}_0$ is the previous state and $\boldsymbol{x}'$ is a small correction. Likewise let $\Omega = \Omega_0 + \Omega'$. Linearising $\psi - \frac{1}{2}\Omega r^2 = C$ yields

$$x'v_0 - y'u_0 - \Omega_0\left(x_0 x' + y_0 y'\right) - \frac{1}{2}\Omega'\left(x_0^2 + y_0^2\right) = C - \tilde{\psi}(x_0, y_0) \qquad (2.22)$$

where

$$u_0 = -\frac{\partial}{\partial y}\psi\left(x_0, y_0\right) \qquad v_0 = \frac{\partial}{\partial x}\psi\left(x_0, y_0\right)$$

$$\tilde{\psi}(x_0, y_0) = \psi(x_0, y_0) - \frac{1}{2}\Omega_0\left(x_0^2 + y_0^2\right).$$

To define the contour in terms of a single variable and allow the calculation of a single perturbation quantity, we parameterise the contour following the method of (Dritschel, 1995) (Appendix B) whereby a normal displacement multiplied by a 'perturbation function' constitutes the disturbance, i.e.

$$\boldsymbol{x} = \boldsymbol{x}_0 + \boldsymbol{x}' = \boldsymbol{x}_0(\alpha) + \frac{\{dy_0/d\alpha, -dx_0/d\alpha\}}{(dx_0/d\alpha)^2 + (dy_0/d\alpha)^2}\eta(\alpha) \qquad (2.23)$$

where η is a perturbation function. In general α can be any parameterisation of the contour but here we take it to be the "travel-time coordinate", i.e. proportional to the time taken for a fluid parcel to travel a fixed distance along the contour (for convenience $\alpha = 2\pi$ is taken as a full circuit of the contour). This parameterisation simplifies the problem significantly and circumvents difficulties which arise for particularly elongated contours when using a polar type parameterisation.

The basic state velocity on the contour in the rotating frame is given by,

$$\tilde{\boldsymbol{u}}_0 = \frac{\partial \boldsymbol{x}_0}{\partial \alpha}\frac{d\alpha}{dt}$$

45

and $d\alpha/dt$ is defined by our definition for α to be

$$\frac{d\alpha}{dt} = \bar{\Omega}$$

where $\bar{\Omega}$ is defined as the frequency of the rotation of the fluid elements around the boundary $\boldsymbol{x}_0$. This is independent of α and implies that in a given increment of time all locations along the contour will move through the same increment of α. Thus the parameterisation reduces the first three terms of (2.22) i.e.

$$
\begin{aligned}
x'v_0 - y'u_0 - \Omega_0\left(x_0 x' + y_0 y'\right) &= \tilde{v}_0 x' - \tilde{u}_0 y' \\
&= \bar{\Omega}\eta(\alpha)
\end{aligned}
$$

Now (2.22) gives

$$\eta(\alpha) = \frac{1}{\bar{\Omega}}\left(C - \tilde{\psi}(x_0, y_0) + \frac{1}{2}\Omega'\left(x_0^2 + y_0^2\right)\right). \tag{2.24}$$

So we have an equation for the perturbation function in terms of the previous state, $\boldsymbol{x}_0$, C and Ω'. We calculate C and Ω' from the equations for area and centroid of the contour. In terms of a contour integral we can express the area as

$$A = \frac{1}{2}\oint x\mathrm{d}y - y\mathrm{d}x$$

Substituting our expansion $\boldsymbol{x} = \boldsymbol{x}_0 + \boldsymbol{x}'$, linearising to first order and substituting in our time-travel coordinate parameterisation (2.23) gives

$$A = A_0 + \int_0^{2\pi}\eta\mathrm{d}\alpha \tag{2.25}$$

where A_0 is the area from the previous iteration. Substituting for η from (2.24) we have

$$A = A_0 + \frac{1}{\bar{\Omega}}\int_0^{2\pi}\left(C - \tilde{\psi}(x_0, y_0) + \frac{1}{2}\Omega'\left(x_0^2 + y_0^2\right)\right)\mathrm{d}\alpha. \tag{2.26}$$

The contour integral formulation for the x centroid position is given by

$$d = \frac{1}{3A} \oint x \left(x \mathrm{d}y - y \mathrm{d}x \right)$$

As with the area we expand in perturbation terms, linearise to first order and substitute in our parameterisation (2.23) yielding

$$Ad = \frac{1}{3} \oint x_0 \left(x_0 \mathrm{d}y_0 - y_0 \mathrm{d}x_0 \right) + \int_0^{2\pi} x_0 \eta \mathrm{d}\alpha,$$

and substituting for η from (2.24) gives

$$Ad = \frac{1}{3} \oint x_0 \left(x_0 \mathrm{d}y_0 - y_0 \mathrm{d}x_0 \right) + \frac{1}{\bar{\Omega}} \int_0^{2\pi} x_0 \left(C - \tilde{\psi}(x_0, y_0) + \frac{1}{2}\Omega' \left(x_0^2 + y_0^2 \right) \right) \mathrm{d}\alpha$$
$$(2.27)$$

We are now able to calculate Ω' and C from (2.26) and (2.27) (effectively a 2×2 system) which in turn enables us to update the perturbation function η and thus update the contour. Numerically these integrals are evaluated using a trapezoidal rule on 400 contour points.

2.5 Results

Having parameterised the states in terms of a mean radius $\bar{R}$ and centroid d, we compute equilibrium states by fixing $\bar{R}$ or d and incrementing the other. Circles are provided as a first guess and convergence is generally achieved within around 20 iterations. Subsequent computations use the previous state, or an extrapolation of 2 previous states, as a starting point for the routine.

2.5.1 The single vortex, $N = 1$

Figure 2.3 shows a sample of different equilibrium states from across parameter space. These plots show the projection of the contour to form the 3-

dimensional helix. In particular we note that while the contour in the plane $z = 0$ can be quite elongated and curved for large d, the core of the vortex, viewed from a plane perpendicular to $\boldsymbol{h}$ (or $\boldsymbol{\omega}$), is close to circular. For smaller d we find the contours in the $z = 0$ plane are themselves closer to circular, this being nearer to the core projection. Attention is also drawn to (a) and (b) in the figure which show states of identical mean radius $\bar{R} = 2.20$ and exceptionally close centroid, $d = 0.695$ and $d = 0.696$. Despite their proximity in parameter space these states are quite distinct. Turning now to figure 2.4 we are able to observe where the different states lie within parameter space. Of particular importance is the region where the code fails to converge (shown as a bold line), the states mentioned above being at the lower boundary of this region. These limiting states indicate that multiply connected contours may describe the equilibria inside this region. The states on the left hand edge of the boundary are cusp-like, for example state (a) in figure 2.3, while those on the right hand edge are more curved and elongated, e.g. (c) in figure 2.3. We conjecture that states inside the left edge will have a small hole, pinched off at this cusp, and those inside the right edge will have a larger hole, where the edges of the curved contour have 'coalesced'. These conjectures are substantiated by simulations carried out in section 2.5.4.

2.5.2 Multiple vortices, $N > 1$

Having examined parameter space in the case of single vortex equilibria we extend the method to consider N evenly azimuthally spaced vortices. Clearly this implies a geometric restriction on the mean radius of the vortices dependent on the centroid radius d. We show results for the limiting states in this regime in figure 2.5. The contours computed show quite a degree of variation over parameter space and for small d in the cases of 2 and 3 vortices these states take on the configuration of two-dimensional corotating vortices computed by
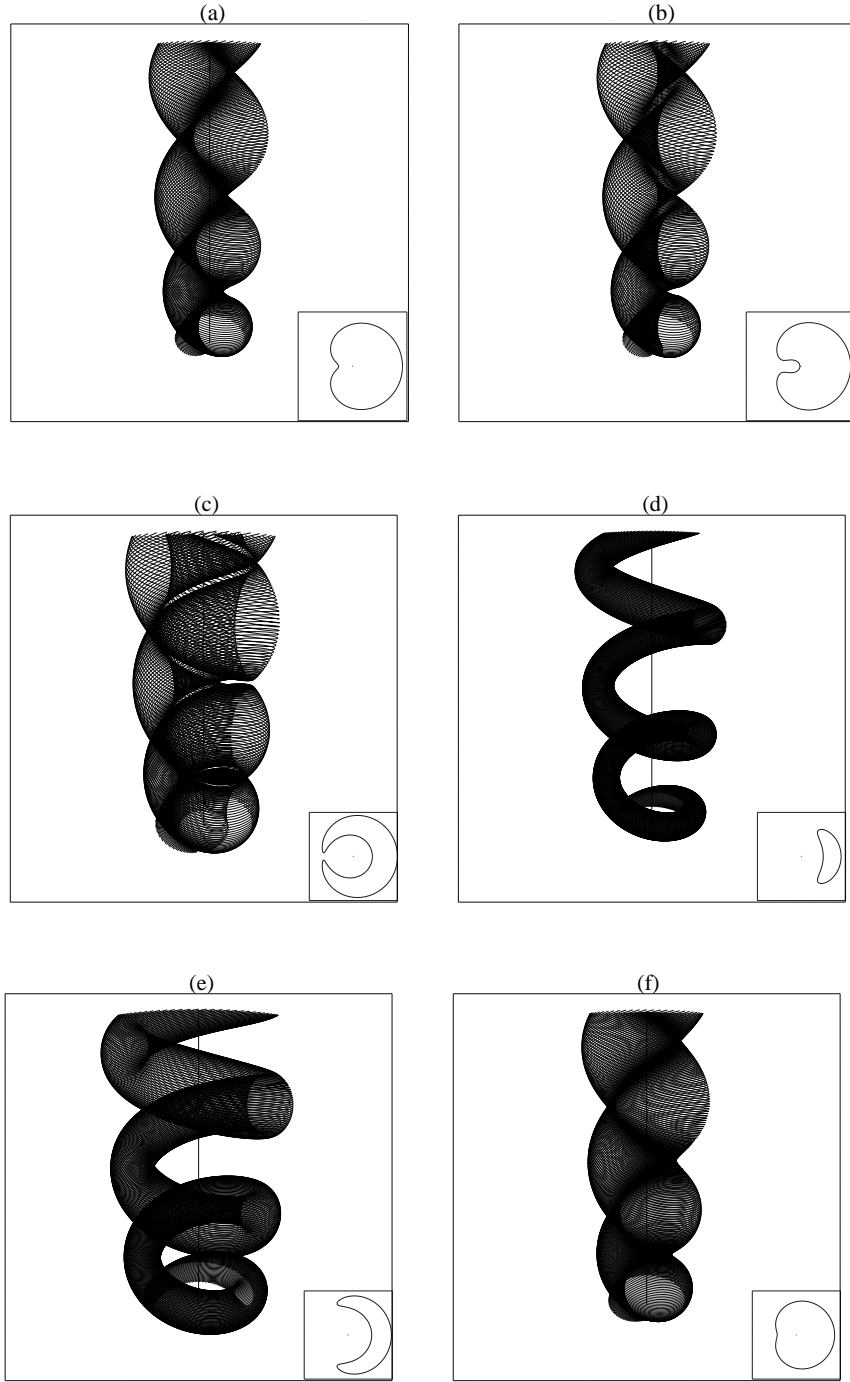
48

Figure 2.3: Plots of equilibrium states for a sample of parameter space: (a) $\bar{R} = 2.20$ $d = 0.695$, (b) $\bar{R} = 2.20$ $d = 0.696$, (c) $\bar{R} = 2.75$ $d = 0.85$, (d) $\bar{R} = 1.50$ $d = 2.50$, (e) $\bar{R} = 2.50$ $d = 2.50$, (f) $\bar{R} = 2.55$ $d = 0.6$. Inserts show the contours along with the full 3D helical vortex, viewed from a perspective of $30°$ elevation and 20 units from the origin. 3 spirals of the helix are shown here.
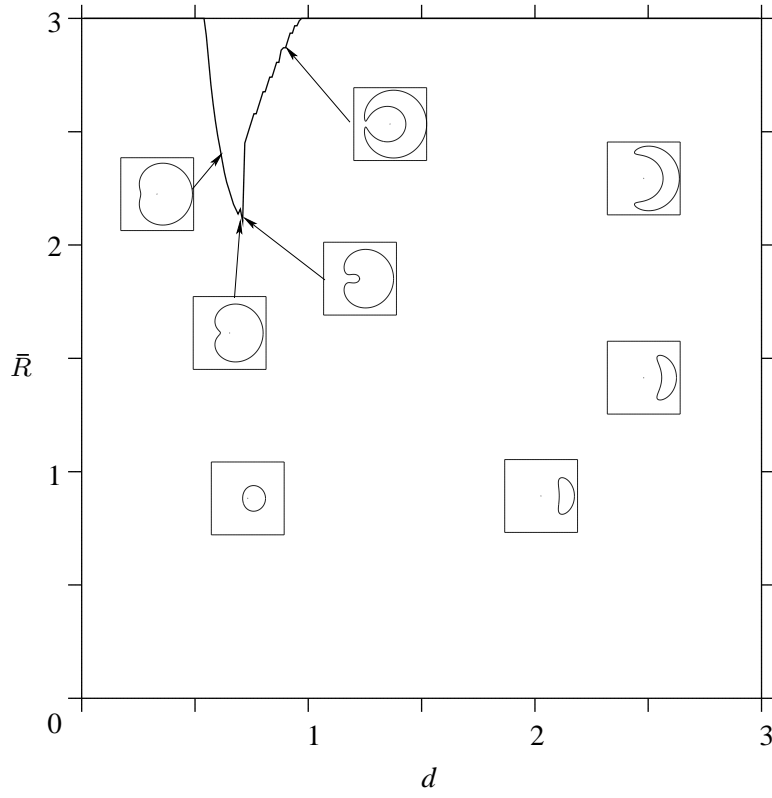
Figure 2.4: Plot of parameter space, for a single vortex, showing the boundary where convergence is not achieved with insets of equilibrium states.

(Dritschel, 1985) where the limiting behaviour is for the vortices to pinch toward the origin. As d increases we see helical effects becoming more pronounced. The contours becoming increasingly curved and elongated and the vortices close up around the circle radius d. As with the single vortex case we see contours which are closer to circular for smaller d.

Figure 2.6 shows parameter space for the cases of $N = 2$ and $N = 3$ vortices, the transition from the two-dimensional type states is notable by a distinct change in the curve showing the boundary of the limit of convergence. The states in this transition region neither close together at the origin nor around the circle, rather they flatten and form an edge facing inwards.

2.5.3   Diagnostics

In addition to axial vorticity and circulation, there are two further invariants for these flows, namely 'excess' energy E and angular impulse J per unit axial length (the axial momentum is proportional to J). The angular impulse is

$$ J = \iint \omega r^2 \mathrm{d}x \mathrm{d}y = \frac{\omega}{4} \oint \left(x^2 + y^2 \right) \left(x \mathrm{d}y - y \mathrm{d}x \right) $$

for a vortex patch of constant vorticity ω.

The 'excess' energy (minus an infinite constant proportional to the square of circulation) is

$$ E \equiv -\frac{1}{2} \iint \omega \psi r \mathrm{d}r \mathrm{d}\phi $$

(Dritschel, 1985), conservation of which can be derived in the standard way by multiplying (2.15) by ψ and integrating. To convert this into a contour integral, we use Stokes' Theorem repeatedly to give
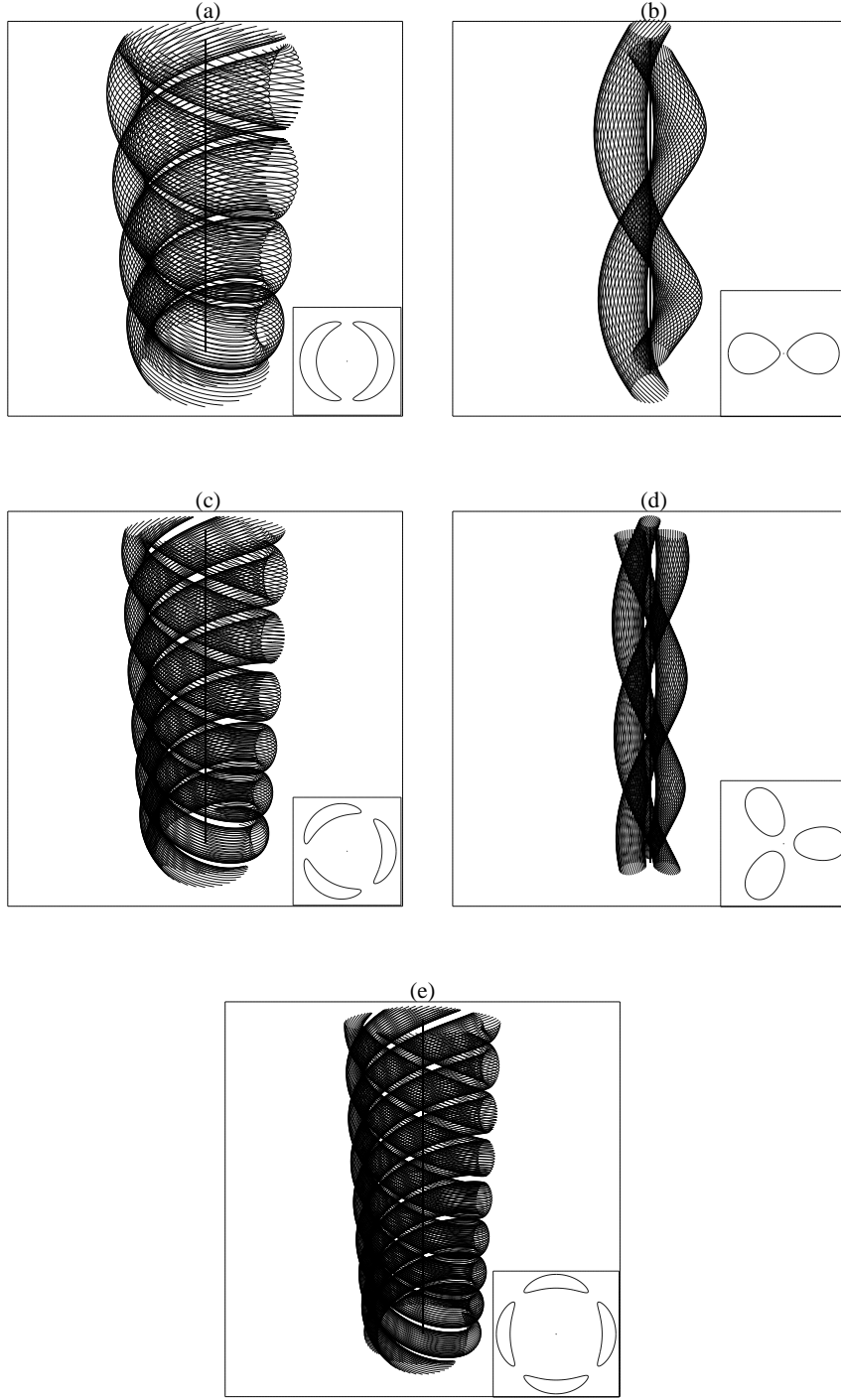
Figure 2.5: Plots of equilibrium states showing typical limiting states for $N = 2, 3$ and 4 vortices: (a) $N = 2$, $\bar{R} = 1.34$, $d = 2$; (b) $N = 2$, $\bar{R} = 0.355$, $d = 0.50$; (c) $N = 3$, $\bar{R} = 0.82$, $d = 2$; (d) $N = 3$ $\bar{R} = 0.2$ $d = 0.36$; (e) $N = 4$, $\bar{R} = 0.6$, $d = 2$. Contours are shown along with the full 3D helical vortex viewed from the same perspective as in figure 2.3, with plots (a), (c) and (e) showing 2 full twists and (b) and (d) showing a single twist of the helix.
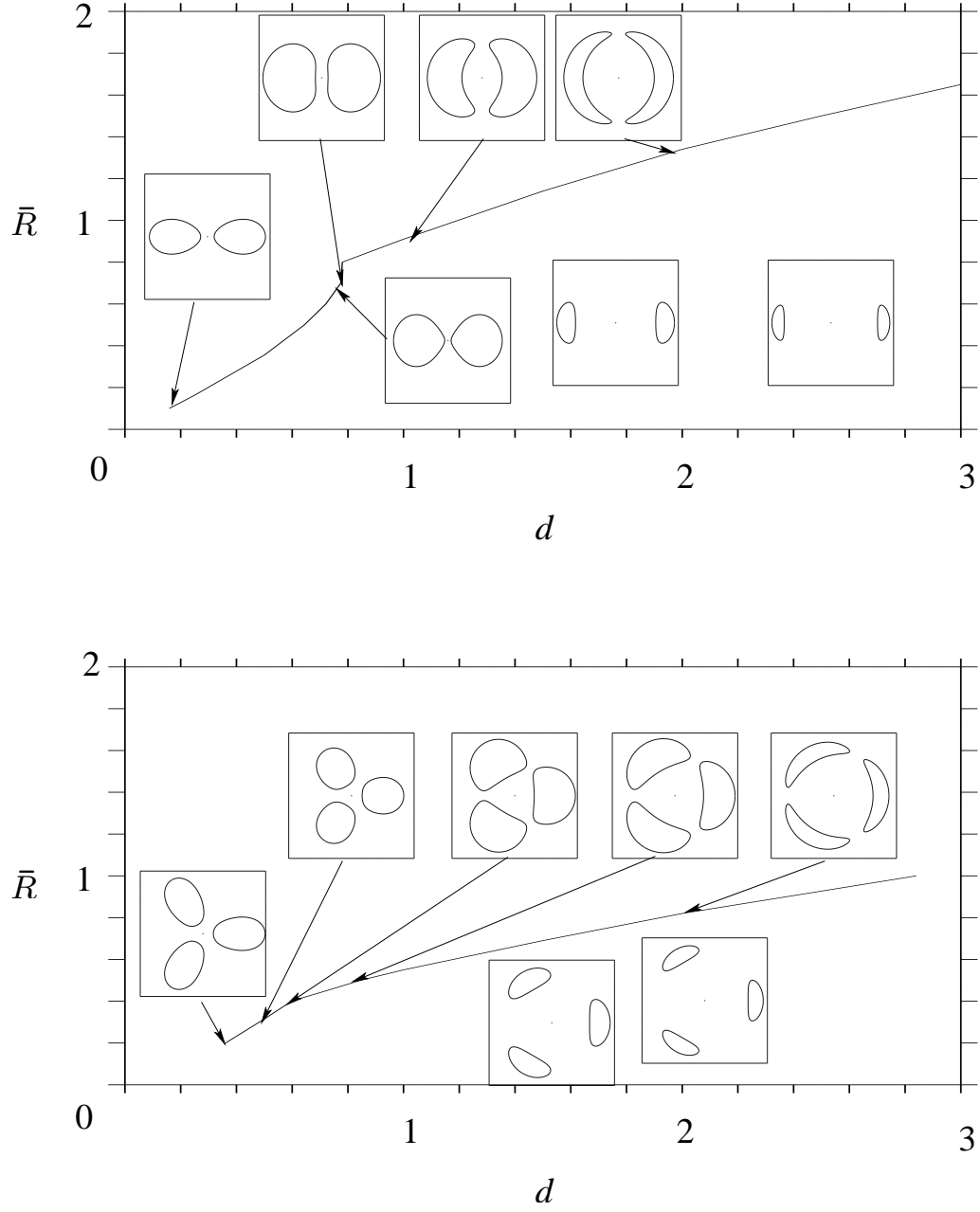
Figure 2.6: Plots of parameter space for $N = 2$ (upper) and $N = 3$ (lower) vortices showing the boundary where convergence is not achieved with insets of equilibrium states.

$$-E = \frac{1}{4}\omega \oint \left[\psi - \frac{1}{r}gh^2\frac{\partial \psi}{\partial r} + \frac{\omega}{8}\left(r^2 + \frac{1}{3}\epsilon^2 r^4\right)\right] r^2 \mathrm{d}\phi + \frac{1}{4}\omega \oint \frac{g}{r}\frac{\partial \psi}{\partial \phi}\mathrm{d}r$$

where

$$g(r) = \frac{1}{2}r^2 + \frac{1}{4}\epsilon^2 r^4.$$

These integrals are computed using cubic interpolation of the contours together with a two-point Gaussian quadrature method in each of the 400 intervals. This enables an accurate estimation of J and E over the entire parameter space. The calculations of energy and angular impulse were verified by noting that for a circular patch of constant vorticity the expressions reduce to $E = \frac{1}{4}R^4\pi(\frac{1}{4}-\log R)$ and $J = \frac{1}{2}\pi R^2$ (independent of ϵ, where R is the radius of the patch).

The energy, angular impulse and rotation rate of the single vortex states are shown in figure 2.7. These quantities confirm that there is a jump in the nature of the states over the gap in parameter space. This is not unexpected as we have already observed a clear distinction in the shapes of the limiting states, c.f. figure 2.4.

Validation of the numerics has been carried out to check the computed values for rotation rate by considering small amplitude waves on an axisymmetric helical vortex in section 2.6. It is worth mentioning that the dispersion relationship is not that of (Kelvin, 1880), who considered a Rankine vortex with only axial vorticity. Here helical symmetry implies we have an additional azimuthal component of vorticity (see equation (2.16)), and hence an associated axial flow. Further corroboration has been sought by comparing with the asymptotic approximation derived by (Widnall, 1972). This comparison however suffers from several limitations. First the asymptotic approximation considers filaments of small core radius and small pitch, a regime in which our numerical computations are most sensitive to numerical error. Second, the approximation contains a correction term which requires much extra work to compute and which is simply read

from the plot in (Widnall, 1972) in more recent work (see e.g. (Ricca, 2006)). The result is poor agreement. For $d = 0.5$, ignoring the correction term (a small negative contribution for $\epsilon = 1$), we have at $\bar{R} = 0.12$, $\Omega = -0.0091$ compared with Widnall's approximation $\Omega_W = -0.0085$, while for $\bar{R} = 0.18$, $\Omega = -0.018$ compared with $\Omega_W = -0.015$ (where Ω_W is the rotational frequency in (Widnall, 1972). A more convincing validation is given in section 2.6.

Figure 2.7 also shows energy contours in the lower half of parameter space (bottom right panel). It has previously been observed in 2D flows (Dritschel, 1995) that an extremum of energy and angular impulse in parameter space may coincide with the margin of stability for vortex equilibria. In particular this extremum must occur over states which have equivalent circulation, in this case, for a given $\bar{R}$. Figure 2.7 shows a shallow minimum for increasing $\bar{R}$. These is no corresponding extremum in J, by contrast to 2D flows, and direct numerical simulations of perturbed equilibria have yet to indicate any instability.

In the case of multiple vortices, we compare states with equal total circulation, Γ, in order to compare like states. This requires that if the mean radius of the single vortex equilibrium is $\bar{R}_1$ then the N vortex case has the individual mean radii scaled by $1/\sqrt{N}$ so that the total area of the patches is equal (i.e. $\bar{R}_N = \bar{R}_1/\sqrt{N}$). Of particular interest in figure 2.8 is a reversal in the rotation rate for small d, $\bar{R}_1$ when $N > 1$. Hence there are equilibrium states for which Ω is zero. We also observe that increasing the circulation brings the trend for the single vortex closer to that of the multiple vortex case, where we have a noticeable steepening for small d.

We have attempted to compare our results for Ω with the asymptotic slender-vortex results derived by (Okulov, 2004). Again comparisons are difficult due to the small core size but also due to limitations in the asymptotics. (Okulov, 2004) tabulates values of non-dimensional rotation rate minus a constant containing a logarithmic singularity as the core radius goes to zero. This constant term
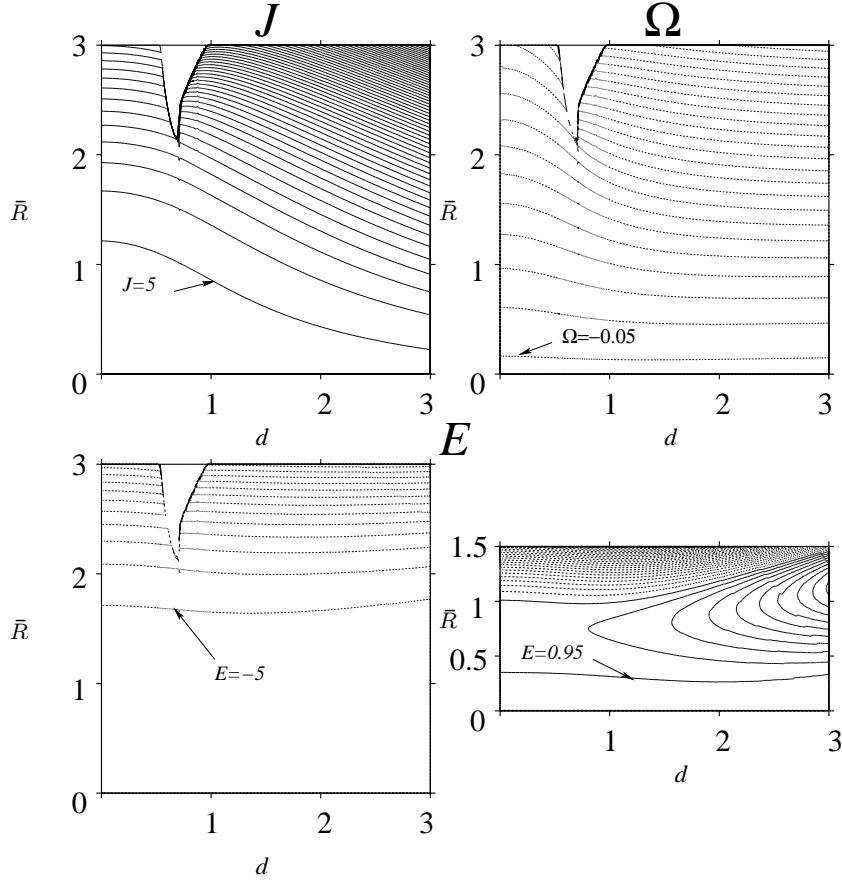
Figure 2.7: Contours of diagnostics for $N = 1$. Upper left panel shows angular impulse J, upper right angular velocity Ω and lower plots excess energy E. Contour intervals are $\Delta J = 10$, $\Delta\Omega = 0.1$ and $\Delta E = 10$ for global energy plot (bottom left) and $\Delta E = 0.1$ for lower half plane plot (bottom right)

is actually of leading order and compromises the accuracy of any comparison. Nonetheless comparisons can be made; premultiplying Ω with $-4\pi d^2/\Gamma$ to be consistent with the non-dimensionalisation in (Okulov, 2004) and considering $N = 2$, $d = 1$, and $\bar{R} = 0.15$, we find $\Omega = 0.97$ for the dimensionless rotation rate compared with $\Omega_O = 1.097$ in (Okulov, 2004) after reintroducing the logarithmic term. Similarly considering $\bar{R} = 0.12$ with other parameters fixed we find $\Omega = 1.04$ compared with $\Omega_O = 1.02$. In both examples the logarithmic term is $O(1)$.

2.5.4   CASL simulations

This section presents results from dynamical simulations carried out using an adapted CASL algorithm whereby contours of ω are advected by the flow field computed by inverting $\mathscr{L}\psi$ as discussed in section 2.3 of this paper. The reader is referred to (Dritschel and Ambaum, 1997) and (Macaskill and Padden, 2003) for a more detailed description of the model. Although this model limits one to the purely helical evolution of vorticity, we are able to deduce the validity of our equilibrium states and their stability (to spatial perturbations) under helical symmetry (and for $\upsilon = $ constant). In addition we are able to validate the conjectures made regarding the gap in parameter space when $N = 1$.

Simulations are carried out with a time step $\Delta t = 0.025$ (the CASL algorithm uses a standard fourth order Runge-Kutta time integration) together with 400 radial intervals and 512 azimuthal intervals. The equilibria are perturbed by displacing all the x coordinates of the contour nodes by some increment Δd. All figures shown in this section are in the rotating frame of the equilibrium state.

Figure 2.9 shows a generic single vortex equilibrium state, $\bar{R} = 2$, $d = 2$ perturbed by a small amount $\Delta d = 0.01$. Virtually no effect is felt by the vortex, even at late times. The only dynamical response is a slight change in the rotation rate.
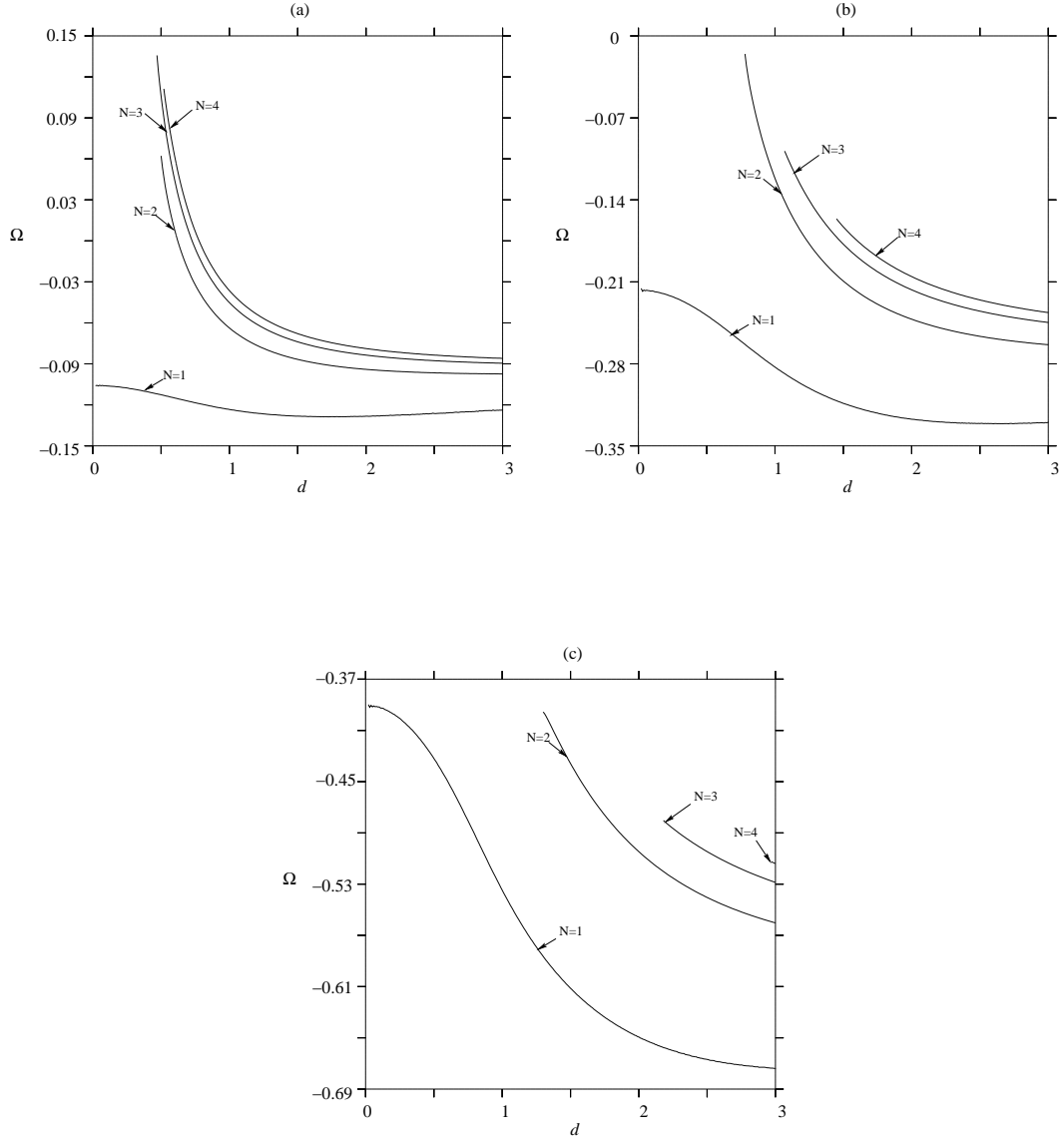
Figure 2.8: Plots of rotation rate Ω for fixed circulation or total area A_1 with curves for $N = 1, 2, 3$ & 4; (a) $A_1 = \pi/4$ ($\bar{R}_1 = 1/2$); (b) $A_1 = \pi$ ($\bar{R}_1 = 1$) and (c) $A_1 = 9\pi/4$ ($\bar{R}_1 = 3/2$)

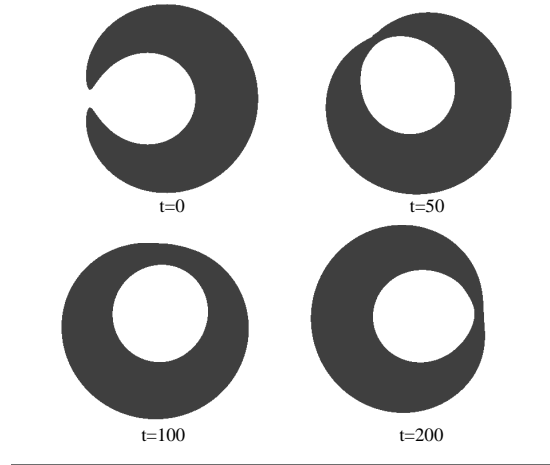Figure 2.9: CASL simulation for the $N = 1$, $\bar{R} = 2$, $d = 2$ state with perturbation $\Delta d = 0.01$

Figure 2.10: CASL simulation for the $N = 1$, $\bar{R} = 2.64$, $d = 0.8$ state with perturbation $\Delta d = -0.03$
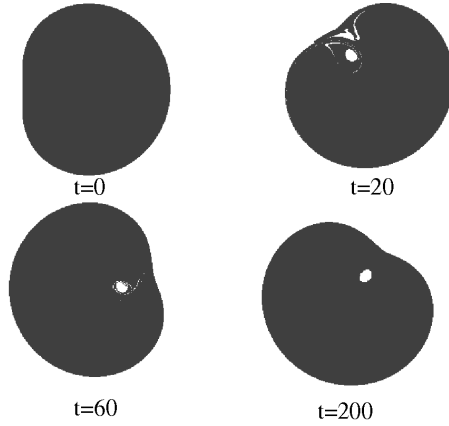
t=0   t=20

t=60   t=200

Figure 2.11: CASL simulation for the $N = 1$, $\bar{R} = 2.2$, $d = 0.6$ state with perturbation $\Delta d = 0.1$

Figures 2.10 and 2.11 show simulations for $N = 1$ where limiting states at the boundary in parameter space are perturbed in the direction of the gap. Figure 2.10 shows an elongated contour from the right edge of the gap, closing over and cutting off a region of zero vorticity within the vortex edge. Hence the equilibrium evolves into a multiply connected state. Figure 2.11 shows the entrainment of filamentary structures of irrotational fluid into the vortex. For large time, the filaments congregate into a small region of zero vorticity inside the vortex. These simulations provide compelling evidence that multiply-connected equilibrium states exist in the gap. Notably, despite the distinct change in the shape of the contour, the vortices remain coherent and robust. In the case shown in figure 2.11 filaments entrained inside the vortex coalesce into a single patch of zero vorticity inside the vortex. The changes in topology occur through the removal of extremely thin filaments by 'surgery' (Dritschel, 1988b) at a tenth of the inner radial spacing of the grid. This allows the vortex to relax to a quasi-steady state while only very weakly dissipating angular impulse and energy.

To gain a more comprehensive picture of the interior states, simulations were

Figure 2.12: Final frame for CASL simulation, $t = 200$, left hand plot shows the $\bar{R} = 2.64$, $d = 0.8$ state with perturbation $\Delta d = -0.105$ and the right the $\bar{R} = 2.64$, $d = 0.59$ state with perturbation $\Delta d = 0.105$
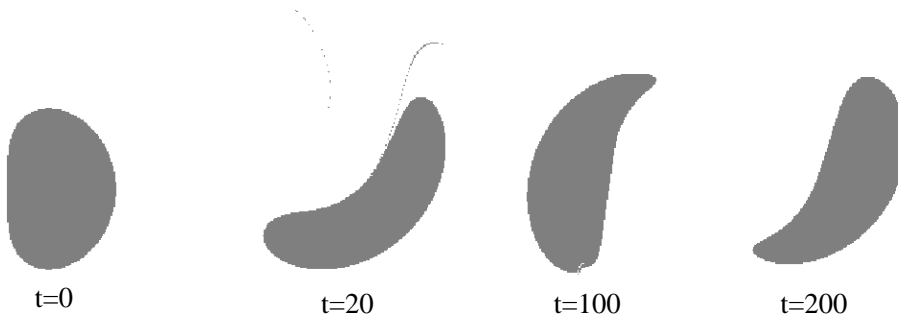

t=0   t=20   t=100   t=200

Figure 2.13: CASL simulation for the $N = 1$, $\bar{R} = 1$, $d = 1$ state with perturbation $\Delta d = 1$

carried out perturbing the edge states from either side of the gap by an equal and opposite amount. This gives the evolution of two different states of the same mean radius at the same centroid location, d. Figure 2.12 shows the final frame for large time of these simulations for the limiting states at $\bar{R} = 2.64$ and suggests that there is the possibility that more than one equilibria state exists for a particular point of parameter space inside the gap. It is conceivable that multiply-connected equilibria exist anywhere in parameter space, not solely in this region where we were unable to find singly-connected states.
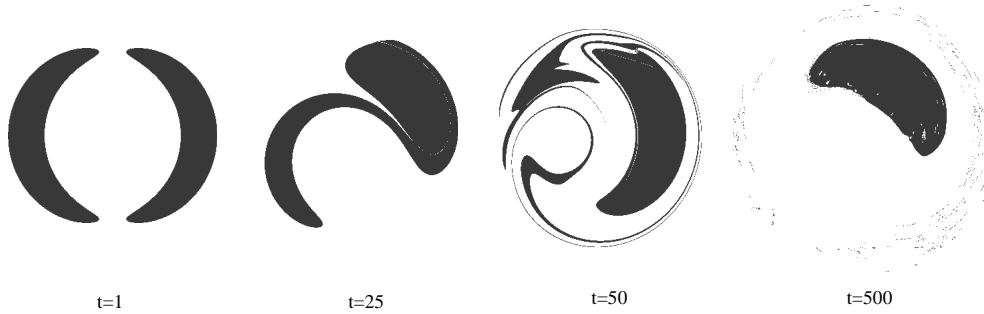
t=1   t=25   t=50   t=500

Figure 2.14: CASL simulation for the $N = 2$, $d = 2$, $\bar{R} = 1.34$ state with perturbation $\Delta d = 0.01$

As previously suggested, these single-vortex states are remarkably stable to finite-amplitude perturbations. To demonstrate this, we performed a simulation where the equilibrium for $\bar{R} = 1$ and $d = 1$ was perturbed by $\Delta d = 1$, i.e. a full doubling of the centroid. Figure 2.13 shows the flow evolution. Although the contour deforms strongly and has a substantial anomaly in its angular velocity, it remains coherent after only tiny filaments have been expelled.

In contrast to the strong stability of the single-vortex states, the multiple-vortex states present more varied results. Generally it is found that the vortex states for $\bar{R} \ll d$ are more robust than those for which $\bar{R} \approx d$. Perturbing states in which the vortices are larger and more closely spaced, we observe that the rotation rates of the individual patches diverge from the rotation rate of the system, causing the patches to move together and interact. Figures 2.14, 2.15 and 2.16 show this for $N = 2$, 3 & 4 vortices respectively. The long time behaviour, in particular for the two vortex case, shows the vorticity tending towards a single patch type configuration, as we have observed to be stable in helical dynamics.
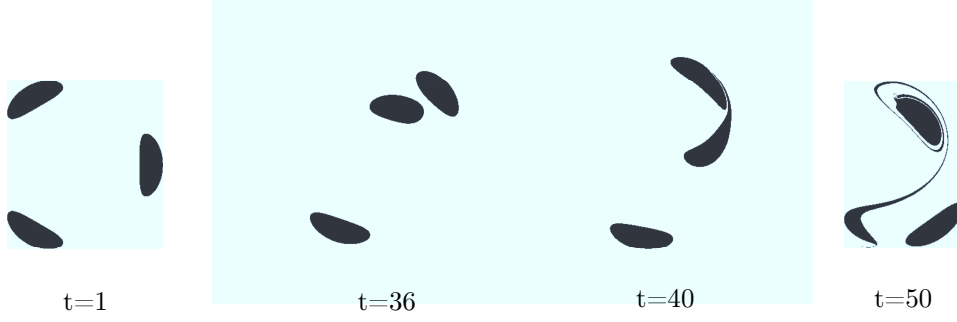
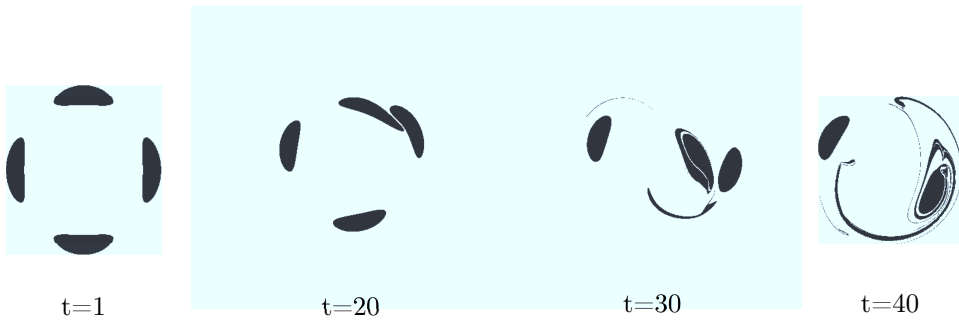Figure 2.15: CASL simulation for $N = 3$, $d = 2$, $\bar{R} = 0.5$ state with perturbation $\Delta d = 0.01$

Figure 2.16: CASL simulation for $N = 4$, $d = 2$, $\bar{R} = 0.5$ state with perturbation $\Delta d = 0.01$

2.6 Dispersion relation for a columnar vortex with helical symmetry

Over a century ago Lord Kelvin calculated the dispersion relationship for infinitesimal linear perturbations of a uniform vortex column (Kelvin, 1880). The non-axisymmetric azimuthal disturbances with azimuthal wavenumber $m = 1$ are known as bending modes and correspond to helical disturbances of the vortex. This work was generalised in (Moore and Saffman, 1972) to include an axial flow which subsequently became the starting point for many asymptotic studies of a helical vortex filament, e.g. (Ricca, 2006), (Kuibin and Okulov, 1998). Linear stability of a Rankine vortex with a discontinuous axial flow and the addition of swirl has been considered in (Loiseleux et al., 1998) where frequencies for the helical bending modes were computed. It is, however, possible to generalise this linear theory yet further by considering an axisymmetric columnar vortex with helical symmetry.

Starting from linear theory we expand about the basic state. We note that, due to the presence of azimuthal vorticity ($\boldsymbol{\omega} = \omega \boldsymbol{e}_z - \epsilon r \omega \boldsymbol{e}_\theta$), there is a parabolic axial flow within the undisturbed vortex given by $\bar{u}_z = \frac{1}{2}\omega\epsilon(r^2 - R^2)$. Note also that $\bar{u}_z = 0$ outside the vortex. Due to material conservation of ω, (2.15), we can restrict attention to deformations of the vortex edge in the plane $z = 0$. This implies that the perturbation vorticity $\omega' = 0$ both inside and outside the vortex boundary. The linearised version of equation (2.6) then gives simply $\mathscr{L}\psi' = 0$, and expressing $\psi' = \Re(\sum_m \hat{\psi}_m(r)e^{im\theta - i\sigma t})$ we have solutions to the homogeneous problem as given in section 2.3.1, namely

$$\hat{\psi}_m = AG(r; R) \tag{2.28}$$

where G is Green's function and A is a constant to be determined by matching

the velocity components at the boundary.

Defining the perturbed vortex edge as $r = R + \eta(\theta, t)$, where R is the radius of the undisturbed vortex, we use the fact that η evolves materially according to

$$\frac{Dr}{Dt} = u_r \quad \Rightarrow \quad \frac{D\eta}{Dt} = u_r \quad \Rightarrow \quad \frac{\partial \eta}{\partial t} + \frac{u_\theta}{r}\frac{\partial \eta}{\partial \theta} = u_r.$$

Rewriting η as $\eta = \Re(\sum_m \hat{\eta}_m e^{im\theta - i\sigma t})$ we obtain after linearisation

$$(\frac{1}{2}\omega m - \sigma)\hat{\eta}_m = -\frac{m}{R}\hat{\psi}_m(R) = -mA\frac{G(R;R)}{R}, \qquad (2.29)$$

given the simple form of the velocity field for an axisymmetric uniform helical vortex.

We now require the velocity field to be continuous on the perturbed boundary $r = R + \eta$. The radial component consists only of a perturbed part and is continuous over r by the continuity of $\hat{\psi}_m$. The azimuthal component has a mean part whose shear jumps across the boundary. Continuity of the full azimuthal component then implies a jump in the perturbed part, i.e. $[u'_\theta] = \omega\eta$, at $r = R$, upon linearisation. This implies

$$\omega\hat{\eta}_m = h^2(R)\left[\frac{d\hat{\psi}_m}{dr}\right] \quad \Rightarrow \quad \omega\hat{\eta} = \frac{A}{R} \qquad (2.30)$$

from (2.28) and section 2.3.1. Continuity of axial velocity yields the same relationship.

Substituting (2.30) into (2.29) leads to the dispersion relation

$$\Omega_m = -\frac{\sigma}{m\omega} = -\frac{1}{2} + (\epsilon R)^2 I'_m(m\epsilon R)K'_m(m\epsilon R) \qquad (2.31)$$

where we have now restored the dependence on pitch. In our analysis ϵ is the direct analogue of the axial wavenumber, k, and ϵR is a dimensionless axial wavenumber (denoted ka in (Saffman, 1995)) The dispersion relation (2.31) is not

65

the same as the one derived by (Kelvin, 1880) who assumed a two-dimensional (non-helical) basic state with $\boldsymbol{\omega} = \omega \boldsymbol{e}_z$ in contrast to $\boldsymbol{\omega} = \omega \boldsymbol{e}_z - \epsilon r \omega \boldsymbol{e}_\theta$ in helical flows.

This expression for Ω_m furnishes us with a further validation of our numerics. For $m = 1$, (2.31) gives an approximation for the rotation rate Ω (for $\omega = 1$), in the regime $d \ll 1$. Figure 2.17 shows Ω_1 from (2.31) versus the numerically computed Ω (for $d = 0.02$), together with the rotational frequency obtained by (Kelvin, 1880) against dimensionless axial wavenumber. Notice that despite considering a more complex flow, the dispersion relation derived here is much simpler than Kelvin's, allowing the precessional frequency of the vortex to be directly calculated without having to solve a transcendental equation. The curves for Ω_1 and Ω indicate very close agreement of the numerical computations and the linear theory. These curves diverge from Kelvin's as axial wavenumber increases and the vortex lines become less rectilinear.

The angular frequencies of the linear helical modes, $\Omega_m(\epsilon R)$ for $m = 1$ to 5 are plotted in figure 2.18 (for $\omega = 1$). For $\epsilon R \ll 1$, these tend to the two-dimensional values $\frac{1}{2}(m-1)/m$, while for $\epsilon R \gg 1$, these tend to $\frac{1}{2}\epsilon R/m$. The existence of these modes implies that there are finite-amplitude non-axisymmetric helical vortex equilibria also for $m > 1$ centred on the z-axis.

2.7 Conclusions

This work has presented a numerical method for computing equilibrium states for helically-symmetric vortices. A novel approach using Fourier transforms and finite differences on a two-part grid was employed to invert the linear operator $\mathscr{L}\psi$ for a general distribution of axial vorticity. A property which has proved crucial in allowing the computation of these equilibrium states is that helically symmetric flows materially conserve axial vorticity when the flow parallel to vor-
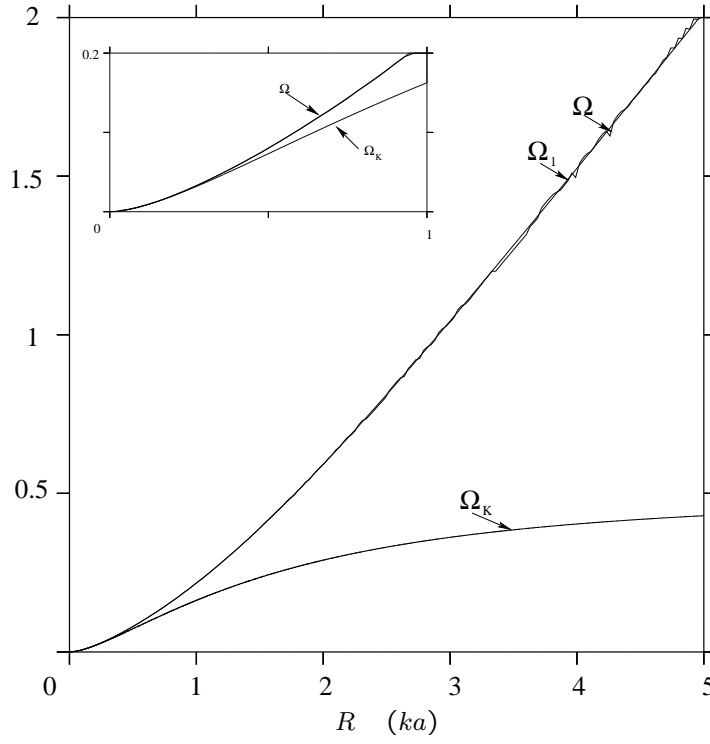
Figure 2.17: Curves showing the numerically computed rotation rate, Ω (for $d = 0.02$) and the analytical bending-mode frequencies, Ω_1 and Ω_K, of a perturbed axisymmetric helical and rectilinear vortex, respectively, against non-dimensional wavenumber, ϵR ($= R$ since we take $\epsilon = 1$) or ka.
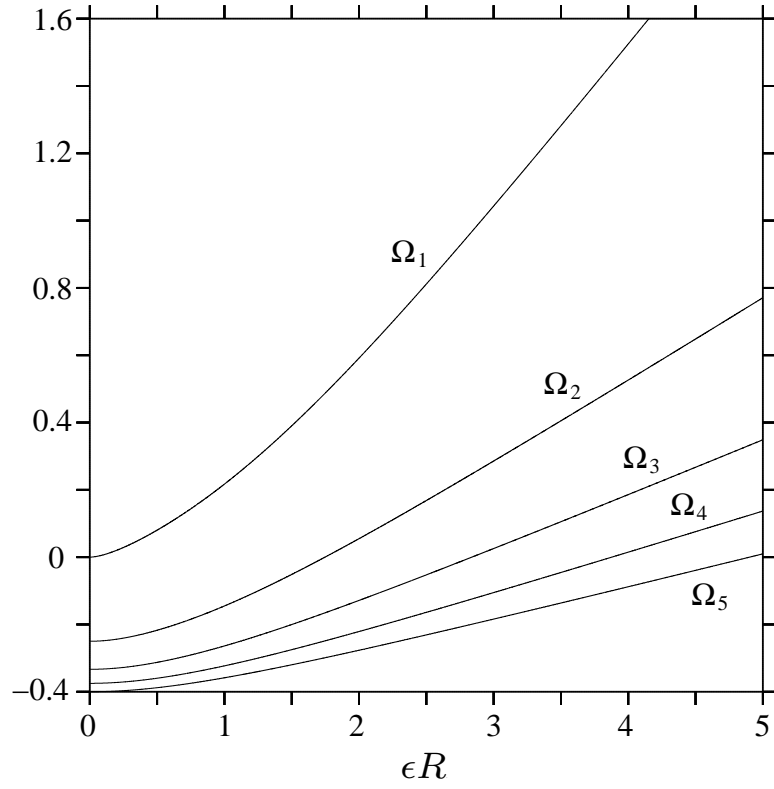
Figure 2.18: Curves showing the analytical linear mode frequencies, Ω_1 to Ω_5, of a perturbed axisymmetric helical vortex against non-dimensional wavenumber, ϵR.

tex lines is proportional to $(1 + \epsilon^2 r^2)^{-1/2}$. Moreover, the constant of proportionality is uniquely determined by the integral of axial vorticity over a cross section $z = $ constant (the circulation), by the requirement of bounded momentum per unit length. Equivalently, this conservation property follows from restricting the vorticity to be everywhere tangent to the helical vector $\boldsymbol{h}$.

Here we have computed equilibria described by contours bounding regions of uniform axial vorticity $\omega = 1$. The equilibria depend only on their radius and centroid position. The contours are parameterised by a time-travel coordinate to allow arbitrary distortions. In the single vortex case we have been able to compute equilibrium states over all of parameter space apart from a specific region where we conjecture only multiply-connected states exist. CASL simulations support this conjecture and also demonstrate the remarkable stability of the states. It should be noted, however, that this stability is for purely helical dynamics and previous asymptotic results (Widnall, 1972) suggest that instability would likely occur in the full 3D dynamics.

In the case of multiple vortices we have computed equilibrium states over parameter space and determine where geometric constraints prevent equilibria. We see a distinction between the states for small radial spacing d, where we have a configuration analogous to the two-dimensional vortex equilibria of (Dritschel, 1985), and the states for moderate or large radial spacing, where helical effects become important. Simulations show helical instability for larger, more closely spaced configurations, with the long time evolution tending towards a single patch configuration.

When considering equilibrium vortex configurations, one must entertain the possibility that there exists a multiplicity of states for a fixed location in parameter space. In the case of a single two-dimensional Rankine vortex patch we now know that there is an infinite multiplicity of states arising from the linear displacement modes proportional to $e^{im\theta}$ (see (Saffman, 1995)). When $m = 2$,

these states are elliptical (Kirchhoff, 1876). These elliptical states have a self-induced rotation, $\Omega = \omega\lambda/(1+\lambda)^2$, associated with them which is dependent upon the aspect ratio, λ, of the patch. Analogous states almost certainly exist also in helical flows, for vortices centred on the z-axis (see section 2.6). In helical flows an additional rotation is induced by the curved shape of the vortex as it twists around the z-axis. If we now consider, as in this work a single vortex *not* centred on the z-axis but displaced from it by a small distance d, an equilibrium in some rotating frame of reference requires that the self-induced rotation (for $d = 0$) nearly matches the rotation of the curved vortex about the z-axis. This additional requirement for helical flows suggests that the vortex shape is unique for a given dimensionless mean radius $\epsilon\bar{R}$ and displacement ϵd. We hence conjecture that there are no other single vortex equilibria for $\epsilon d > 0$ than the ones we have found here.

We motivated this work by the recent studies on rotor wake flows (Okulov and Sørensen, 2007b), (Walther et al., 2007) and many parameters commonly associated with such applications can easily be represented by the parameters used here (noting first that setting $\epsilon = 1$ sets the length scale of the system). Parameters such as rotor radius, d, and pitch, ϵ, are readily available and the circulation of each vortex has been discussed in section 2.3 and is proportional to $\bar{R}^2$. The advance ratio, commonly utilised in aerodynamics, defined as the ratio of the velocity of the flow over the rotor to the angular velocity of the rotor, can be expressed here as simply $1/d\epsilon$ (velocity of flow over the rotor can be expressed as $2\pi/\epsilon T$ and angular velocity $2\pi d/T$ where T is the period of rotation). Although the problem of the rotor wake is not directly addressed in this work, care has been taken to ensure that the size, strength and configurations of vortices considered are of such a general nature as to be applicable to a variety of applications.

The problem of the rotor wake can be more fully addressed in future work by adapting this method to include a central hub vortex consistent with the rotor

wake configuration. This introduces two additional parameters to the problem, namely the strength and size of the inner vortex. Also of interest is to extend the method to consider multiply connected patches of vorticity.

The equilibrium states found in this study can be adapted for use as non-trivial test cases in the simulation of the 3D Euler equations and are a step more complex than Norbury's vortex rings. We plan to use these states to test a new hybrid Lagrangian-Eulerian algorithm for the 3D Euler equations.

Chapter 3

VortexFOAM

3.1 Introduction

Since the earliest point vortex approximations of (Rosenhead, 1931), many authors concerned with the simulation of incompressible flows have found numerous benefits in formulating the problem using Lagrangian 'particles' or elements to represent the vorticity field. The primary benefit of this seemingly benign transformation is to reduce the solution of a highly nonlinear hyperbolic PDE (Navier-Stokes or Euler equations) to that of a set of ODE's for the trajectories of each element plus an inversion of a linear elliptic PDE. An immediate consequence of this formulation is the avoidance of the usual stability constraints which limit the efficiency of a direct discretisation of the equations. In conjunction with this computational saving, vortex methods can be considered to be self-adaptive; the computational elements defining the vorticity field are required only where vorticity is nonzero and their motion, defined by the flow-map, concentrates computational effort where it is most required. These facts mean that vortex methods are very well suited for studying high Reynolds number or inviscid flows where vortical motions dominate the dynamics and small scales present a challenge to

traditional methods.

In this study we present a new method which seeks to combine several beneficial advances in the field in an effort to apply additional adaptation and hence computational savings to resolve vortex stretching in a consistent and completely automatic way. Firstly, in the Lagrangian description of vorticity we combine a vortex filament approach (Greengard, 1986) (Leonard, 1985) with a variable vortex blob method (Cottet et al., 2000) (Hou, 1990). We consider space curves represented by a series of points, or nodes, to define centres of vortex tubes or filaments, and define vorticity contributions using smoothing lengths for segments of the filaments which are dynamically updated to resolve the small scales generated by vortex stretching while maintaining the overlap criteria required for the method to converge. In addition, we add further refinement to the method by employing a 'renoding' procedure to redistribute points along the filament, increasing the concentration in regions of high strain.

Kelvin's circulation theorem means that for an ideal flow, vortex lines are frozen into the fluid, therefore we materially advect the vortex filament having first calculated the induced velocity field. Traditionally this inversion is carried out by some regularised Biot-Savart type calculation (Leonard, 1985), (Beale and Majda, 1982), although this proves particularly computationally expensive, being $O(N^2)$ in operations for N nodes. Efforts to reduce this computational burden include domain decomposition procedures, where the N-body problem is replaced with contributions from near and far fields (see appendix B in (Cottet and Koumoutsakos, 2000) for a review of such methods). Hybrid methods replace the Biot-Savart integral with a discretised version of the original Poisson equation, $\Delta \boldsymbol{u} = -\nabla \times \boldsymbol{\omega}$, on a fixed Eulerian grid, and employ a grid based solver to compute velocities (Christiansen, 1973). Having designed the Lagrangian component of the method to concentrate effort in regions of interest, we continue this strategy in the Eulerian portion by employing adaptive mesh refinement

(AMR) in locations where the velocity gradients, responsible for straining the vortex filaments, require careful resolving. To this end we take advantage of the OpenFOAM CFD library (`http://www.openfoam.com`, 2011) which uses the finite-volume discretisation and allows for great generality in mesh topology, refinement criteria, linear solvers etc. A method with various similarities, including AMR and vortex filaments was presented in (Almgren et al., 1994). This method, however, is significantly different to that presented here. The treatment of the inversion problem included the separation of the near and far fields (Method of Local Corrections) where a Poisson equation solved for the far field and an expensive N-body problem the near field. Their Lagrangian adaptation is comparatively primitive, vortex line segments are split as they stretch rather than a full renoding algorithm which accounts for the well known folding (hairpin vortices) associated with vortex stretching (Chorin, 1988).

The final component of the method concerns the communication between Lagrangian vorticity elements and the Eulerian grid. Vortex-in-Cell (VIC) methods require an interpolation procedure to convert vorticity values to the grid and velocities from the grid to the elements. Traditionally this is achieved using piecewise polynomials, e.g. B-splines (Monaghan, 1985); however, the application of these interpolation kernels is restricted to uniform grids. Instead we choose interpolation weights based on a vortex-blob concept, noting that interpolation to the grid is directly analogous to providing some core profile to the elements, while maintaining a partition of unity to conserve circulation.

3.1.1 Adaptivity and self-similarity

The Euler equations (and indeed their viscous counterpart the Navier-Stokes equations) remain one of the most intensely studied sets of equations in applied mathematics. Their role in fluid modelling of a wide range of commercial ap-

74

plications has motivated much research, and their strong nonlinearity poses an onerous challenge to their analysis and simulation. The subject of yet more fervent debate surrounds the unproven regularity of the equations. However it is clear the main challenge for numerical methods is in resolving the vast range of scales which their turbulent evolution will support. It has also been found that the smallest scales exhibit coherent vortex filaments, and it is their geometry that is proving to be a fruitful area of study in tackling the problem of finite-time blow-up in the Euler equations. (Deng et al., 2006), (Deng et al., 2005), (Hou and Li, 2006).

The adaptivity of the method presented, and its inherent self-similar nature, combined with the emphasis on resolving vorticity and its stretching, is motivated predominately by these observations. The Lagrangian vortex elements present themselves as an obvious candidate for diagnosing the geometry of the flow in locations of intense vorticity production.

3.2   The Method

We seek to calculate solutions to the incompressible Euler equations, written here in their vorticity-velocity form

$$\frac{D\boldsymbol{\omega}}{Dt} = \boldsymbol{\omega}.\nabla\boldsymbol{u} \tag{3.1}$$

$$\boldsymbol{\omega} = \nabla \times \boldsymbol{u} \tag{3.2}$$

$$\nabla.\boldsymbol{u} = \nabla.\boldsymbol{\omega} = 0 \tag{3.3}$$

where $\boldsymbol{u}$ and $\boldsymbol{\omega}$ denote fluid velocity and vorticity respectively. We discretise these two constituents separately, with an emphasis placed upon resolving vorticity; velocity by definition is supported by a narrower range of scales.

Vortex filaments are defined by space curves (piecewise linear for the purposes of advection and vorticity interpolation), denoted by a sequence of N points, $\boldsymbol{x}_p$.

Since these are discrete approximations to vortex lines, we insist that they form closed loops as is required in an ideal fluid in the absence of solid boundaries. The vorticity is inferred from the filaments, by considering each segment as a discrete element, whose vorticity is proportional to its length, i.e.

$$\boldsymbol{\omega}_p = c_p \boldsymbol{\ell}_p \qquad (3.4)$$

where $\boldsymbol{\ell}_p = \boldsymbol{x}_{p+1} - \boldsymbol{x}_p$. The vorticity is defined at the midpoint between $\boldsymbol{x}_{p+1}$ and $\boldsymbol{x}_p$ and here c_p a strength scale factor. Kelvin's circulation theorem then allows us to materially advect these points,

$$\frac{\mathrm{d}\boldsymbol{x}_p}{\mathrm{d}t} = \boldsymbol{u}(\boldsymbol{x}_p, t)$$

which is performed numerically using the fourth-order Runge-Kutta time stepping scheme, given velocities on the elements (this can be considered as conservation of circulation for the filament). An adaptive time step is chosen based on the peak vorticity to maintain accuracy

$$\delta t = \frac{\pi}{10\|\boldsymbol{\omega}\|_\infty}.$$

Velocities are calculated from the vector potential $\boldsymbol{\psi}$ and the Poisson equation

$$\Delta \boldsymbol{\psi} = -\boldsymbol{\omega} \qquad (3.5)$$

$$\boldsymbol{u} = \nabla \times \boldsymbol{\psi} \qquad (3.6)$$

using the OpenFOAM CFD library, which in turn uses the finite-volume method (FVM), on an arbitrary polyhedral mesh. A number of other options were explored (e.g. an ADI method (Douglas Jr et al., 1963), (Ziegler, 2004), a mulitgrid method (Briggs et al., 2000) etc.) before deciding on the OpenFOAM solver due to its simple implementation and generality. We employ periodic boundaries in

one space variable, which we define to be z and maintain free-slip boundary conditions in x and y. The vector potential is used to avoid derivative boundary conditions and to enforce incompressibility. To keep the vorticity distribution far enough away from the horizontal boundaries we scale our domain such that the horizontal to vertical aspect ratio $a = \frac{L_h}{L_z} = 3$. Since $\boldsymbol{u} \to 0$ like $1/r^2$, we will set $L_z = \pi$ and consider an initial vortex length scale and circulation of unity, then this aspect ratio allows sufficient decay without the necessity to solve a large domain. The boundary conditions are

$$\boldsymbol{\psi}(x, y, L_z) = \boldsymbol{\psi}(x, y, -L_z)$$
$$\boldsymbol{\psi}(L_h, y, z) = \boldsymbol{\psi}(-L_h, y, z) = 0 \qquad (3.7)$$
$$\boldsymbol{\psi}(x, L_h, z) = \boldsymbol{\psi}(x, -L_h, z) = 0$$

with the nodes $\boldsymbol{x}_p$ of the vortex filaments periodically shifted at L_z and $-L_z$.

A global mesh is generated initially with $a^2 16^3$ equal hexahedral cells, which are subjected to a mesh refinement procedure outlined in section 3.2.3. Standard FVM in OpenFOAM is used in the discritisation of (3.5) with a diagonal incomplete-Cholesky preconditioned conjugate gradient linear solver.

While creating vortex filaments as closed loops will only approximately satisfy the solenoidality property for vorticity, it is not, in fact, necessary to perform any additional correction when computing velocities since any corrected component is not involved in the advection of vorticity. If we assume the uncorrected vorticity is $\boldsymbol{\omega}^*$ then

$$\boldsymbol{\omega}^* = \boldsymbol{\omega} + \nabla \chi \qquad \Rightarrow \qquad \nabla.\boldsymbol{\omega}^* = \Delta \chi$$

by a Helmholtz decomposition. Thus the corrected vorticity field can be expressed as

$$\boldsymbol{\omega} = \boldsymbol{\omega}^* - \nabla \Delta^{-1} \nabla.\boldsymbol{\omega}^*.$$

Equation (3.5) requires taking the curl of $\boldsymbol{\omega}$, and so the correction term will not contribute. For any detailed discussion of the vorticity field to be performed this correction will be necessary in post processing.

3.2.1   Interpolation

Having described the model set up for the evolution of vorticity and the calculation of velocities upon the grid, the next step is to discuss the communication of information between element and grid. The process to which we shall give most consideration is that of assigning vorticity values from the elements to the grid. As has already been alluded to, the nonuniform nature of our grid presents additional challenges in determining an accurate and consistent interpolation kernel. This must, however, be tempered with the efficiency requirement of the algorithm, consequently we employ some local interpolation procedure such that only a small subset of the grid is in direct communication with the elements.

Following the convention in (Cottet and Koumoutsakos, 2000), we denote gridded quantities by subscript i and particle variables by subscript p, so for example $\boldsymbol{x}_i$ are the cell centres of the control volumes in the Eulerian mesh and $\bar{\boldsymbol{x}}_p$ are the locations of the 'particle' centres. We introduce an interpolation kernel for each segment, p, and denote it as $\phi_p(\boldsymbol{x}) = \phi(\bar{\boldsymbol{x}}_p - \boldsymbol{x})$. Our assignment of vorticity to the grid follows from

$$\boldsymbol{\omega}_i = \frac{1}{V_i} \sum_{p}^{N} v_p \boldsymbol{\omega}_p \phi_p(\boldsymbol{x}_i) \qquad (3.8)$$

where the V_i are the volumes of the cells, and v_p are the particle volumes (see below). For a uniform grid, appropriate choices for ϕ_p have been established as tensor products of one dimensional B-splines. These kernels are limited to second order accuracy, but have favourable smoothness properties. (Monaghan, 1985) was able to increase the accuracy of B-spline kernels using Richardson extrapo-

lation, and his M_4' kernel is a standard method for vortex method interpolation (Cottet et al., 1999). Unfortunately, these basis functions are dependent on uniform mesh widths and the arbitrary unstructured mesh paradigm employed in OpenFOAM does not permit mapping to a uniform Cartesian mesh.

In order to determine a mesh independent interpolating function, we must consider its smoothing properties. Too much smoothing and we smear out the vorticity gradients, too little and we overestimate the gradients, resulting in a much too noisy vorticity field. One can consider any interpolation procedure for vortex-in-cell methods to be analogous to smearing a blob of vorticity of size proportional to the smoothing length, onto the grid. We are able to draw a certain comparison to the vortex blob method (Hou, 1990), (Leonard, 1980) and consider our interpolation to be providing some core vorticity profile to the Lagrangian elements.

We have designed our method to be able to follow the direct cascade to small scales *automatically*, throughout the process attempting to resolve the strong straining of vortex filaments as accurately as possible. Previous methods which utilise a fixed uniform grid, and therefore smoothing lengths, effectively employ vortex blobs of fixed width and thus suffer from a upper bound in resolution which is fixed *a priori*. We therefore propose an interpolation method that dynamically adjusts the smoothing provided to each element precisely dependent upon the vortex stretching which drives the descent to small scales.

Several authors have made great use of vortex methods employing a Gaussian profile for vortex elements (Leonard, 1985), (Barba et al., 2005), (Cottet et al., 2000) as it allows vorticity and its derivatives to decay quickly enough at infinity for the method to satisfy the convergence criteria (Cottet and Koumoutsakos, 2000). This choice is also practical for our purposes as it allows a mesh independent smoothing of compact support to be determined.

79

To maintain overlap between segments and have consistent particle 'sizes' during our simulations we use an ellipsoidal Gaussian of the form

$$\phi_p(\hat{\boldsymbol{x}}) \propto e^{-\frac{\hat{x}^2+\hat{y}^2}{\sigma_p^2}-\frac{\hat{z}^2}{\tau_p^2}}$$

where $\hat{\boldsymbol{x}} = (\hat{x}, \hat{y}, \hat{z})$ form a local Cartesian coordinate system centred on the particle with the $\hat{z}$-axis tangential to the filament, such that σ_p is the smoothing parameter perpendicular to segment p and τ_p is the smoothing parameter parallel to it. We can now define a particle volume by

$$v_p = \frac{4}{3}\pi\sigma_p^2\tau_p \tag{3.9}$$

which remains constant over the lifespan of the element (section 3.2.2 describes updating of filaments). Now since vortex stretching is accounted for simply via the separation of the points $\boldsymbol{x}_p$ in the filament, we are able to maintain overlap by carefully updating σ_p and τ_p . We initialise by

$$\sigma_p = d \tag{3.10}$$

$$\tau_p = \ell_p \tag{3.11}$$

where we define ℓ_p the length of the element and d the inter-filament spacing. τ_p is updated as element end points are time-stepped and incompressibility allows σ_p to be calculated from (3.9).

The most important property of ϕ_p is that it is conservative, i.e. that the precise amount of vorticity possessed by the element is distributed to the grid, which we can express as

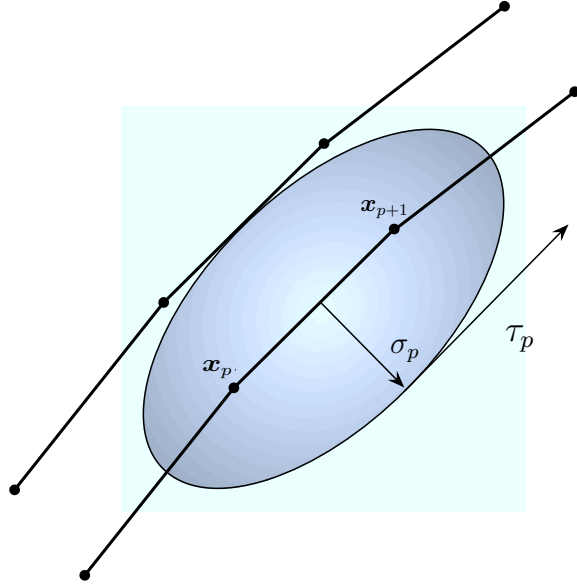$$\sum_i^{n_p} \phi_p(\boldsymbol{x}_i) \equiv 1.$$

80

Figure 3.1: Schematic showing smoothing lengths relative to filaments and segments demonstrating overlapping properties.

where n_p is the number of cells inside the support of segment p. This is often referred to as a 'partition of unity' for the interpolation kernel. Therefore to satisfy partition of unity, the discrete form for our Gaussian interpolation kernel is

$$\phi_p(\hat{\boldsymbol{x}}) = \frac{V_i}{\pi^{3/2}\sigma_p^2\tau_p}e^{-\frac{\hat{x}^2+\hat{y}^2}{\sigma_p^2}-\frac{\hat{z}^2}{\tau_p^2}} \tag{3.12}$$

which satisfies $\sum\limits_{i}^{n_p}\phi_i \approx 1$ since

$$\frac{1}{\pi^{3/2}\sigma_p^2\tau_p}\iiint_V e^{-\frac{\hat{x}^2+\hat{y}^2}{\sigma_p^2}-\frac{\hat{z}^2}{\tau_p^2}}\mathrm{d}V = 1.$$

This interpolation procedure is easily verified in one dimension for a uniform distribution of point sources. Let σ be the smoothing length, such that the volumes are $v_p = \sigma$ and the separation of the points $d = \sigma$. Therefore the

81

equivalent expression to (3.8) with the one dimensional form of (3.12) reads

$$\omega(x_i) = \frac{1}{V_i} \sum_p^N \sigma \omega_p \frac{V_i}{\sqrt{\pi}\sigma} e^{\frac{-(x_i - x_p)^2}{\sigma}}$$

$$= \frac{1}{\sqrt{\pi}} \sum_p^N \omega_p e^{\frac{-(x_i - x_p)^2}{\sigma}}$$

where ω_p is the strength of each particle. Figure 3.2 shows a plot of such a profile for 10 point sources uniformly spaced at integer intervals (i.e. $d = v_p = \sigma = 1$) with uniform strengths ($\omega_p = 1$, for all p). The overlap is such that a constant profile is maintained with a smoothly varying edge.

Note that conservation is only weakly enforced since we consider only cells within some shell (6 times σ or τ in the appropriate direction) of each segment to lie within the support of ϕ_p. The more cells included the better the conservation. In fact the cell density within a smoothing length distance has more influence on conservation that the size of the support. However we are faced with a trade off; if n_p, the number of cells in the support of segment p, tends toward the total number of vortex nodes, N, then the hybrid method becomes redundant and we approach a calculation on the order of the N-body full Lagrangian problem. In section 3.2.3 we describe mesh refinement criteria based upon σ and τ which ensures $n_p \ll N$.

It is also possible to explicitly force conservation by exchanging (3.12) with

$$\phi_p(\hat{\boldsymbol{x}}) = \frac{W_p(\hat{\boldsymbol{x}})}{\sum_i^{n_p} W_p(\hat{\boldsymbol{x}})}, \qquad W_p(\hat{\boldsymbol{x}}) = V_i e^{-\frac{\hat{x}^2 + \hat{y}^2}{\sigma_p^2} - \frac{\hat{z}^2}{\tau_p^2}}$$

however this can lead to undesirable noise if n_p is small.

Velocity interpolation from the grid to the filament nodes is (in the first instance) a linear interpolation from neighbouring cells. Notice this interpolation is distinct from the vorticity interpolation; vorticity interpolation requires discrete 'packets' of circulation to be distributed to the grid in a conservative way,
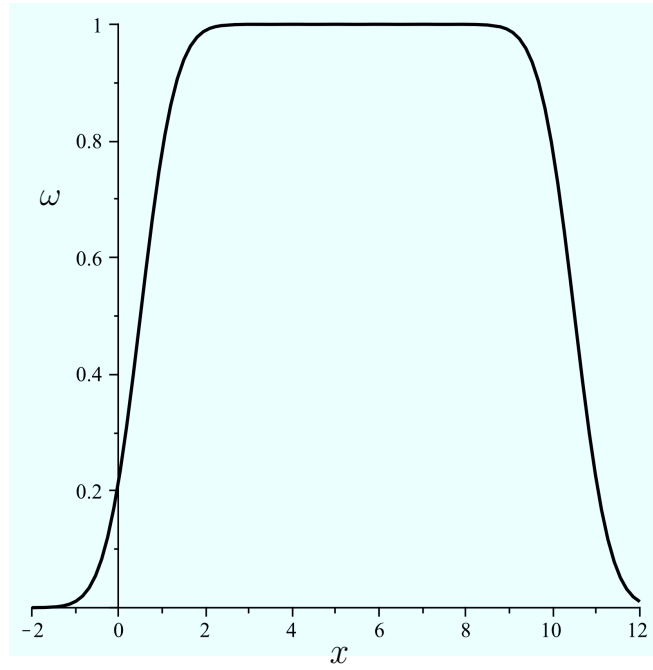
82

Figure 3.2: Plot showing vorticity profile for a uniform distribution of point sources given the interpolation formulae outlined. 10 particles are placed at integer intervals starting from $x = 1$ with uniform unit strength.

whereas velocity interpolation at the filament nodes requires only the sampling of the continuous field defined upon the grid. We employ the *cellPointFace* function contained in the OpenFOAM library which decomposes the parent cell into tetrahedra with one vertex at the cell centre and the rest on the cell faces, one of which is a face centre. The sample node is located within a particular tetrahedra and the velocity is linearly interpolated from the vertices (face values being interpolated from cell centres). Further choices for smoothing velocity interpolation are discussed in section 3.3.2.

3.2.2   Renoding

To motivate an appropriate refinement strategy for both the Lagrangian and Eulerian parts of the method an appreciation of vortex stability and 3D turbulence is necessary. The filament approach that has been devised takes advantage of Kelvin's circulation theorem to employ an implicit handling of the dominant nonlinear vortex stretching term in equation (3.1) which allows a certain relaxation in the numerical stability of the method. The high strain this term imposes on the vortex filaments is associated with the well known cascade to small scales and the associated regularity question for inviscid flows. From the point of view of a vortex filament, small scales are generated through formation of hairpin vortices, as the strained filaments are curved back on themselves. Given this, it is clear that resolution is required in regions of high strain and curvature of vortex lines.

A common practise in filament methods is to allow the segments to subdivide once they exceed some threshold in length, and likewise merge adjacent segments when their sizes are small (Almgren et al., 1994). It was found, however, that this strategy does not take fully into account the bending and folding of filaments associated with vortex stretching (Chorin, 1988). The result was irregular

filaments, with acute angles at the nodes.

A promising alternative to this traditional filament adaptivity of splitting and merging, is to perform a renoding procedure similar to that used in the contour algorithms of Dritschel (Dritschel, 1988b), (Dritschel and Ambaum, 1997). This method redistributes nodes, adjusting their total where necessary, to increase their concentration in regions of high curvature or strain. It is particularly advantageous to accurately represent high curvature regions since a degree of curvature has been discovered to be essential in order to generate near singular conditions (Deng et al., 2006).

Since space curves form a 3D analogue of planar contours we can follow (Dritschel, 1988b) in defining a node density function, ρ, defined along the filament which when integrated gives an updated number of nodes required to resolve the filament. Instead of using curvature, we define ρ to be proportional to the local velocity gradient; velocity gradients are nonlocal and will account for developing deformations of the filament due to the induced velocity of all filaments, not purely the deformations due to the self induced velocity gradients for which curvature is responsible.

To allow a continuous definition of the space curve we perform cubic interpolation between the nodes along the filament. We start with a local orthonormal coordinate basis for each segment p:

$$\boldsymbol{\xi}_p = \frac{\boldsymbol{\ell}_p}{\ell_p}, \tag{3.13}$$

$$\boldsymbol{n}_p = \frac{\boldsymbol{x}_{p+1} \times \boldsymbol{x}_p}{|\boldsymbol{x}_{p+1} \times \boldsymbol{x}_p|}, \tag{3.14}$$

$$\boldsymbol{b}_p = \boldsymbol{\xi}_p \times \boldsymbol{n}_p \tag{3.15}$$

where $\ell_p = |\boldsymbol{\ell}_p|$. Notice, while we use the symbols $\boldsymbol{n}$ and $\boldsymbol{b}$ and the terms 'normal' and 'binormal', these vectors are distinct from the Frenet-frame, which would break down in regions of zero curvature (in the unlikely scenario that $\boldsymbol{n}_p$ we can

use spherical polar coordinates $\boldsymbol{e}_R$, $\boldsymbol{e}_\theta$, $\boldsymbol{e}_\phi$). $\boldsymbol{\xi}_p$ is thus the local Lagrangian unit vorticity vector. The space curve is interpolated by

$$\boldsymbol{x}(q) = \boldsymbol{x}_p + q\boldsymbol{\ell}_p + \eta(q)\boldsymbol{n}_p + \xi(q)\boldsymbol{b}_p \tag{3.16}$$

$$\eta(q) = a_p q + b_p q^2 + c_p q^3 \tag{3.17}$$

$$\xi(q) = \alpha_p q + \beta_p q^2 + \gamma_p q^3 \tag{3.18}$$

where $q \in [0, 1]$ is a parameterisation between the nodes p and $p+1$ (see figure 3.3) . The coefficients $a_p, b_p, c_p, \alpha_p, \beta_p, \gamma_p$ are found by fixing $\boldsymbol{x}(0) = \boldsymbol{x}_p$, $\boldsymbol{x}(1) = \boldsymbol{x}_{p+1}$ and matching the curvatures at $\boldsymbol{x}_p$ and $\boldsymbol{x}_{p+1}$ in the $\boldsymbol{n}_p$ and $\boldsymbol{b}_p$ directions.

Curvature contributions at the points $\boldsymbol{x}_p$ are found in the planes defined by $\boldsymbol{n}_p$ and $\boldsymbol{b}_p$, consistent with the spherical contour surgery approach in (Dritschel, 1988a). We define the 'curvature vector'

$$\boldsymbol{\kappa}_p = \frac{2(\boldsymbol{\ell}_{p-1} \times \boldsymbol{\ell}_p)}{\left|\boldsymbol{\ell}_p \ell_{p-1}^2 + \boldsymbol{\ell}_{p-1} \ell_p^2\right|}$$

and therefore for a given segment p we define the projections of $\boldsymbol{\kappa}_p$

$$\hat{\kappa}_p = \boldsymbol{n}_p \cdot \boldsymbol{\kappa}_p, \qquad \hat{\kappa}_{p+1} = \boldsymbol{n}_p \cdot \boldsymbol{\kappa}_{p+1} \tag{3.19}$$

$$\tilde{\kappa}_p = \boldsymbol{b}_p \cdot \boldsymbol{\kappa}_p, \qquad \tilde{\kappa}_{p+1} = \boldsymbol{b}_p \cdot \boldsymbol{\kappa}_{p+1} \tag{3.20}$$

and match these to the equivalent expressions from our interpolation parameterisation (3.16),

$$\hat{\kappa}(q) = \frac{\eta''(q)\ell_p}{\left(\ell_p^2 + \eta'(q)^2 + \xi'(q)^2\right)^{\frac{3}{2}}} \tag{3.21}$$

$$\tilde{\kappa}(q) = \frac{-\xi''(q)\ell_p}{\left(\ell_p^2 + \eta'(q)^2 + \xi'(q)^2\right)^{\frac{3}{2}}} \tag{3.22}$$

where primes denote derivatives with respect to q. As with (Dritschel, 1988a), nodes are distributed such that the deviations of $\boldsymbol{x}(q)$ from the straight line, i.e.
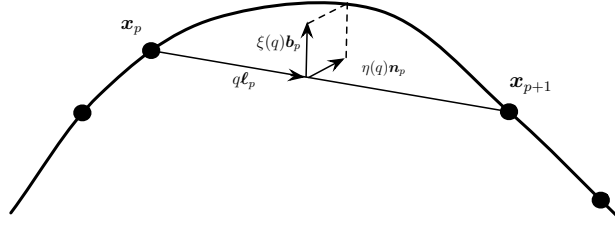
Figure 3.3: Diagram showing the interpolation contributions for renoding

$\eta(q)$ and $\xi(q)$, are small compared to ℓ_p. This means we can neglect the nonlinear terms in equations (3.21) and (3.22) and curvature becomes piecewise linear along the contour:

$$\tilde{\kappa}(q) = \frac{\mathrm{d}^2\eta}{\mathrm{d}q^2}\ell_p^{-2}$$

$$\hat{\kappa}(q) = -\frac{\mathrm{d}^2\xi}{\mathrm{d}q^2}\ell_p^{-2}$$

This results in coefficients remarkably similar to those in (Dritschel, 1988b) and (Dritschel, 1988a)

$$a_p = -\frac{\ell_p^2}{6}\left(2\tilde{\kappa}_p + \tilde{\kappa}_{p+1}\right) \qquad\qquad \alpha_p = \frac{\ell_p^2}{6}\left(2\hat{\kappa}_p + \hat{\kappa}_{p+1}\right)$$

$$b_p = \frac{\ell_p^2}{2}\tilde{\kappa}_p \qquad\qquad \beta_p = -\frac{\ell_p^2}{2}\hat{\kappa}_p$$

$$c_p = \frac{\ell_p^2}{6}\left(\tilde{\kappa}_{p+1} - \tilde{\kappa}_p\right) \qquad\qquad \gamma_p = -\frac{\ell_p^2}{6}\left(\hat{\kappa}_{p+1} - \hat{\kappa}_p\right)$$

It is possible to show that the convergence for this interpolation is precisely the same fourth order as found in (Dritschel, 1988b). Considering an area RMS error defined as
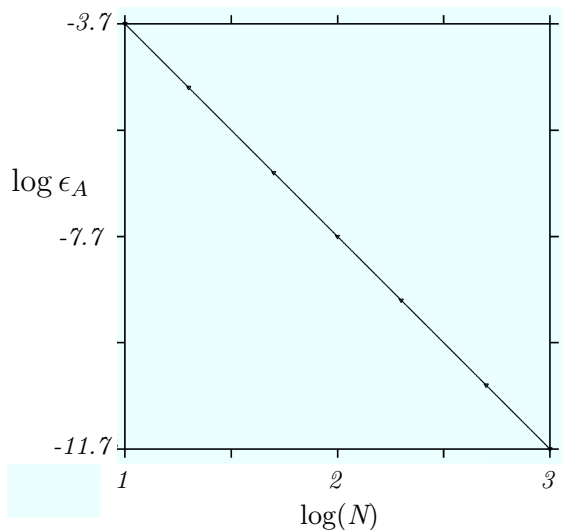
Figure 3.4: Plot showing $\log \epsilon_A$ vs $\log N$ for a single helical filament

$$\epsilon_A = \left(\sum_{p=1}^{N} \frac{\ell_p^2}{N} \int_0^1 |\tilde{\boldsymbol{x}}(q) - \boldsymbol{x}_p(q)|^2 \, \mathrm{d}q \right)^{\frac{1}{2}}$$

where $\tilde{\boldsymbol{x}}(q)$ represents some predefined exact expression for a space curve for which the $\boldsymbol{x}_p$ denote the piecewise linear segment end points. For error analysis purposes a helical filament was chosen so as to have both components, η and ξ, in its correction. Figure 3.4 contains a log-log plot of this error and shows the $O(N^{-4})$ dependence.

We follow (Dritschel, 1988b) and (Dritschel and Ambaum, 1997) to use curvature averages to construct a smooth and consistent node density function ρ.

88

First let

$$\bar{\kappa}_p = \frac{\ell_p}{\mu} \left(\kappa_p + \kappa_{p+1} \right)$$

$$\nu_p = \left(L^{-2} + \bar{\kappa}_p^2 \right)^{\frac{1}{2}}$$

The form for $\bar{\kappa}_p$ is dimensionless and is chosen to preserve the self similarity of the method; the goal is that for a given value of μ, the dimensionless node density parameter, a circle will be represented by a similar number of points, regardless of radius. This is generally a typical segment length on the unit circle, i.e. from the initial condition $\mu = 2\pi/N_s$ for N_s segments. The parameter L is included to ensure an appropriate node density on straight filament sections where κ is small. This serves to place a lower bound on node density. It is desirable for this bound to be related to the resolution of the Eulerian mesh; segments should not exceed some proportion of the largest cell which they encounter. We set

$$L = 2l_0$$

where l_0 is the largest cell width in the mesh (the global mesh width).

Further averaging is performed such that four nodal curvature values will contribute to the node density function, ρ_p

$$\bar{\nu}_p = \frac{\ell_{p-1}\nu_{p-1} + \ell_p\nu_p}{\ell_{p-1} + \ell_p}$$

$$\nu'_p = \frac{1}{2} \left(\bar{\nu}_{p+1} + \bar{\nu}_p \right)$$

$$\rho_p = \frac{\ell_p}{2L^2\mu} \nu_p'^{\frac{1}{4}}$$

This form has sufficient non-locality and smoothness to prevent abrupt changes in node density. Various forms were investigated for ρ_p, including using nodal values for strain in place of circulation, and extensive tests performed. Strain

was found to perform no better than curvature as a node density function, but required additional computations. The exponent of $\frac{1}{4}$ in ρ_p was found necessary for stability. Considering values of μ and L which are physically relevant, it was discovered higher powers on ν'_p result in oscillations in node density and number on sequential passes of the renode subroutine. For example with a single unit radius circular filament with $N = 128$ nodes, i.e. $\mu = 2\pi/128$, the form of ρ_p above gives $N = 127$ new nodes on the filament on the first call of renode and every call thereafter. With this initial condition but now changing $\mu = 2\pi/256$, the first pass yields $N = 297$, second pass $N = 246$ and on the third pass $N = 256$ and this is stable on all subsequent calls. Similarly $\mu = 2\pi/64$ gives $N = 57$, $N = 60$, $N = 64$. This convergence is not so readily achievable for other choices of ρ_p. e.g. setting $\bar{\rho}_p = \frac{2\ell_p}{13L^2\mu}\nu'_p$ then the sequence of filament nodes for the same initial condition and $\mu = 2\pi/128$ is $N = 71, 120, 75, 114$. Setting $\mu = 2\pi/256$ the sequence is $N = 263, 138, 245, 146$. Similar results are obtained until the exponent of ν'_p is reduced sufficiently to damp node density variations. Clearly it is possible that alternative forms for a node density function could be obtained and function equally as successfully as this one.

Renoding thus proceeds filament by filament, fixing the first node on each and distributing nodes by the variation of ρ_p, with the number of nodes on each filament given by the integer part of the sum of all ρ_p. The process of fixing the first node was found to initiate oscillations at the join due to the first node never having the ability to relax relative to the interpolated second and final nodes. The fix was simply to adjust the ordering of the nodes along each filament such that node 1 was incremented each time the renode subroutine was called.

Certain quantities associated with the filament segments require updating at renoding. To maintain consistent volume and vorticity strength conservation from each filament, we require to update v_p and c_p at every renode call. The new nodes naturally create new segments which require some volume. Volumes are

interpolated linearly from the original filament via the parameter q (q provides a natural partition of unity). This automatically conserves their total and has the effect of relaxing the widths σ_p to respond to the small deviations in length introduced by perturbations η and ξ. We now have a new set of segments upon a given filament, with new lengths determined by ρ. Since vorticity is proportional to length (equation (3.4)) and volumes have been determined from the original filament, the contribution to the full vorticity field from each segment needs to be updated to match the value before renoding (c.f. equation (3.8)). In other words the precise vorticity from the original filament needs redistributing to the new one. This can also be performed during renoding by linear interpolation in q, now aimed at conserving $\sum v_p c_p \boldsymbol{l}_p$, i.e. for each new segment, p'

$$ g_{p'} = \sum_p w(q) v_p c_p \boldsymbol{\ell}_p $$

where $w(q)$ are the linear interpolation weights (for example if an entire segment is included $w(q) = 1$) and the sum is over all overlapping segments on the original filament (note this sum could contain a single term). After renoding is complete the c_p are updated:

$$ c_{p'} = \frac{g_{p'}}{v_p \ell_p} $$

3.2.3   Mesh adaptivity

To perform dynamic mesh refinement in OpenFOAM it is necessary to define a suitable scalar function from which cells are flagged for refinement. Appropriate choices seem to present themselves in the form of the interpolation smoothing lengths σ and τ. To enable accurate velocity gradients to be computed and transferred to the filament nodes, sufficient resolution in the grid must be present. The physical small scales inevitably generated in the flow can be attributed to the stretching and folding of vortex lines, as already discussed. Therefore to resolve such structures the grid should be adapted where σ and τ are small, i.e. where
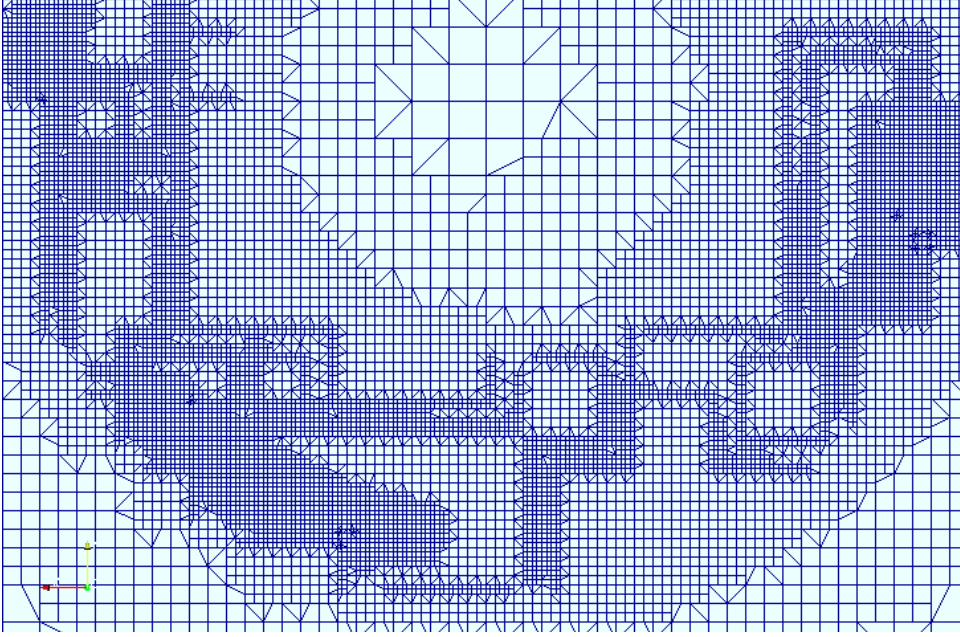
Figure 3.5: Mesh configuration on simulating a vortex ring with refinement criterion given in (3.23)

stretching is more intense and (or) filaments require more resolution via renoding. Various forms have been trialled for the refinement function γ. The two used in the following results are as follows

$$\gamma_1(\boldsymbol{x}_i) = \max_p \left(\frac{h_i}{\min_p(\tau_p, \sigma_p)} \right) \tag{3.23}$$

i.e. γ_1 returns the largest ratio of cell width, h_i, to smoothing length. The support over which the maximum is sought is proportional to the cell width, i.e. any segment within 6 cell widths is considered. In this way, small scales are tracked by the mesh via the shrinking smoothing lengths. This was found to be appropriately smooth for uniform filaments; however, when employed for more dynamic simulations, it generated noisy mesh configurations, by which we mean a mesh containing refinement patterns which do not vary smoothly, see figure 3.5. To address this problem, we introduce a different refinement criterion using

Gaussian smoothing, with a smoothing length given by some proportion of the cell width,

$$\gamma_2(\boldsymbol{x}_i) = \max_p \left(\frac{V_i e^{-\frac{|\boldsymbol{x}_p - \boldsymbol{x}_i|^2}{h_i^2}}}{\min_p(\tau_p, \sigma_p)} \right). \tag{3.24}$$

This form allows for better smoothness in the refinement, and avoids large and abrupt changes in cell size which brings about undesirable non-uniformity (see section 3.3.2).

We then flag cells satisfying $\gamma > \gamma'$, where γ' is the refinement parameter (see Appendix A for some details of the mesh refinement class and flagging cells in OpenFOAM). This refinement strategy ensures that the filament will be continually and automatically well resolved based on the scales represented by the filament segments themselves, with very little computational effort expended acquiring γ.

For computational efficiency we do not perform adaptation (mesh or renoding) every single time step. It was found 2-3 time steps gave sufficient adaptivity, and we renode on every third time step and resmesh every second.

3.3   Results

We validate the method by comparison with well known three dimensional vortex problems. There are surprisingly few examples of steady 3D vortex structures from which we are able to judge the performance of a new method; vortex rings are a very well studied class of flows, which are the most obvious candidate. Their steady behaviour and instability are well documented (Norbury, 1973), (Widnall and Tsai, 1977), (Leonard, 1985).

First we compare vortex ring propagation speeds with those given by Norbury's steady states (Norbury, 1973) and from Kelvin's 1867 paper (see e.g.

93

(Lamb Sir, 1993)). We then demonstrate the method's capability to capture complex unsteady vortical motions by considering the Widnall instability of the thin core ring (Widnall and Tsai, 1977). This tests both the spatial adaptation techniques: renoding and mesh refinement. We also verify the helical vortex equilibria presented in chapter 2. In all cases we calculate the known invariants, energy

$$T = \frac{1}{2} \int_V |\boldsymbol{u}|^2 \mathrm{d}V$$

and helicity

$$H = \int_V \boldsymbol{\omega} \cdot \boldsymbol{u} \, \mathrm{d}V$$

(circulation is implicitly conserved by the method).

3.3.1   Norbury's ring translation

(Norbury, 1973) considered a class of steady axisymmetric vortex rings, computing contours bounding a vorticity distribution proportional to radial distance, parameterised by a mean core radius. These steady states represent an ideal test case for numerical methods such as the one presented here, their speed of propagation being accurately known.

For Norbury's vortex rings, nodes are placed lying inside the symmetric core boundary δA defined by a cosine series with coefficients given by table 1 in (Norbury, 1973). These contours for various values of mean core radius, α, are shown in figure 3.6. We initialise the filaments on a cylindrical grid, placing an array of nodes in the rz-plane and project each filament azimuthally. Inside the boundary the (purely azimuthal) vorticity is given by $\omega_\phi = ar$, and therefore we scale the r grid to increase filament density with increasing r i.e. $r = \sqrt{i_r \Delta r}$ where $\Delta r = \pi^2 / n_r$ and i_r varies between 0 and n_r. While the profile we obtain doing this is approximating the radial structure, the discrepancies turn out to be negligible in the translation velocity after we non-dimensionalise with circulation Γ.
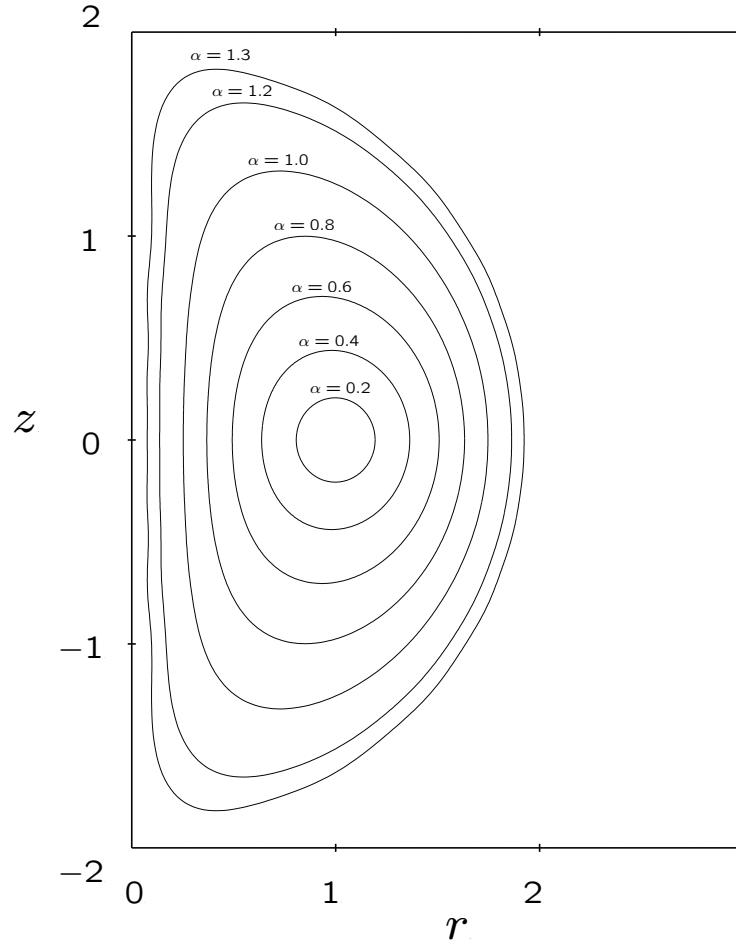
Figure 3.6: Contours in the rz-plane representing the boundary δA of the steady states computed by (Norbury, 1973) for various mean core radii α (the $\alpha = \sqrt{2}$ limit represents Hill's spherical vortex (Hill, 1894))

In all cases we also offset the filaments relative to the grid to reduce the effect of grid generated symmetries on the solution. The filament resolution, given by the rz-plane grid spacing, is chosen such that the core of the ring is represented by a similar number of filaments regardless of its size. The motivation for this is to attempt to construct self-similar vorticity profiles across all sizes of ring. We do, however, perform simulations with a variety of such resolutions to gauge the accuracy and efficiency of the method.

To test performance we are able to compare the speed of propagation of the vortex ring to those given by Norbury and Kelvin. Kelvin found for a uniform vortex ring of radius R and core radius α' that the speed is given by

$$V_K = \frac{\Gamma}{4\pi R} \left(\log \frac{8R}{\alpha'} - \frac{1}{4} \right). \tag{3.25}$$

We demonstrate the steady translation of the vortex ring by computing the z centroid of the vorticity distribution, i.e.

$$\bar{z}(t) = \frac{\iiint_V \omega_\phi z \, \mathrm{d}V}{\iiint_V \omega_\phi \, \mathrm{d}V}.$$

The translation rate is a particularly good metric for determining the accuracy and stability of the method, but we can also observe the invariants of the flow, energy and helicity. For these rings, helicity is zero and Norbury published results for the energy of the rings to which we can compare. We non-dimensionalise the states spatially via the ring radius $R = 1$ and in time by setting circulation $\gamma = 1$. The easiest way to compute Γ is to evaluate the integral

$$\Gamma = \iint_A \boldsymbol{\omega} \cdot \boldsymbol{n} \, \mathrm{d}A$$

from the gridded vorticity. However due to the variations in mesh configuration, it is more appropriate to determine Γ from the Lagrangian description of vorticity;

hence since the ring is by definition axisymmetric we set the circulation for a vortex ring to be

$$\Gamma \approx \frac{\sum v_p c_p l_p}{2\pi} \equiv 1$$

by initialising

$$c_p = c = \frac{2\pi}{\sum v_p l_p}$$

for all p.

To compare to Norbury's non-dimensionalisation we must rescale the diagnostic quantities (energy, helicity and propagation speed). Norbury sets $\Gamma = aR^3\alpha^2\bar{\Gamma}$ where $\bar{\Gamma}$ is given in (Norbury, 1973) and table 3.1. Thus the translation speed

$$\bar{V} = \frac{V}{aR^2\alpha^2} = \frac{R\bar{\Gamma}V}{\Gamma}$$

and the energy is

$$\bar{T} = \frac{T}{a^2R^7\alpha^4} = \frac{R\bar{\Gamma}^2T}{\Gamma^2}.$$

As a consequence results for $\bar{V}$ and $\bar{T}$ are sensitive to our estimate for Γ, the circulation for the ring and how well it is represented upon the grid. Thus we can read from the gridded vorticity a value for a, compute V and T and compare to the expected results from (Norbury, 1973) (table 3.1).

α	a	$\bar{V}_N$	V_K	$\bar{T}_N$	$\bar{\Gamma}$
0.2	7.966	0.8488	0.8589	9.85	3.1385
0.4	1.991	0.6586	0.6830	6.58	3.1262
0.6	0.8995	0.5357	0.5751	5.02	3.0882
0.8	0.5169	0.4428	0.4938	3.99	3.0231

Table 3.1: Parameters and diagnostics for the various Norbury vortex rings under consideration where $\bar{V}_N$ and $\bar{T}_N$ are the non-dimensional translation and energy from (Norbury, 1973)).

From here onwards we drop the over-bar notation and all quantities presented will be rescaled to Norbury's non-dimensionalisation as described above.

$\alpha = 0.2$ Parameter Study

To demonstrate convergence of the method, a Norbury ring of radius $\alpha = 0.2$ was examined exploring various values of the renode parameter μ, the inter-filament distance (or filament resolution) d and grid refinement parameter γ' (in all of the Norbury cases γ_1 (3.23) is used).

An important property for attaining an accurate solution is to ensure the vortex core is well resolved. Vorticity profiles for various values of d and γ' are shown in figure 3.7. A clear improvement in the accuracy of the core profile is observed as d increases and γ' decreases. The majority of simulations are performed with $\mu = 2\pi/128$ which was found to give (qualitatively) the smoothest filaments and be computationally efficient (examples with $\mu = 2\pi/64$ and $\mu = 2\pi/256$ are given)

Table 3.2 includes details of the run parameters for each case, including filament number N_{fil}, total segment number N, total cells in the mesh N_{cells}, smallest cell width h_{min}. Table 3.3 contains the diagnostics of vorticity scale a, translation speed V and energy T, to compare to Norbury's results. All of these diagnostics are for $t = 0$, with the obvious exception of V which is computed after $T = 200$ time steps, $t \approx 6$ to 7.

The choices for the parameters are based on computational efficiency and the preservation of a physical solution. The values for γ' chosen are those for which a practical computation time is possible, while maintaining an accurate solution: Observing the results for V and T given in table 3.3 on decreasing γ (e.g. cases

case	d	μ	γ'	N_{fil}	N	N_{cells}	h_{min}
a	0.1	$2\pi/128$	0.7	28	3474	619005	$2\pi/256$
b	0.05	$2\pi/128$	fix	106	13208	864448	$2\pi/128$
c	0.05	$2\pi/128$	1.7	106	13208	294198	$2\pi/128$
d	0.05	$2\pi/128$	0.7	106	13208	797526	$2\pi/256$
d_1	0.05	$2\pi/64$	0.7	106	6832	425546	$2\pi/256$
d_2	0.05	$2\pi/256$	0.7	106	27074	406800	$2\pi/256$
e	0.025	$2\pi/128$	3.7	414	51577	267031	$2\pi/128$
f	0.025	$2\pi/128$	1.7	414	51577	802489	$2\pi/256$
g	0.0125	$2\pi/128$	7.7	1641	204647	255635	$2\pi/128$
h	0.0125	$2\pi/128$	3.7	1641	204647	805870	$2\pi/256$

Table 3.2: Mesh and filament parameters for runs with Norbury vortex ring initial condition with core radius $\alpha = 0.2$. Case (b) is the fixed grid case.

case	a	V	T	CPU time
a	7.19	0.8191	9.012	0.13
b	7.48	0.8347	9.272	0.21
c	7.48	0.8305	9.249	0.07
d	7.48	0.8351	9.289	0.22
d_1	7.45	0.8407	9.244	0.08
d_2	7.48	0.8322	9.301	0.15
e	7.67	0.8426	9.475	0.10
f	7.66	0.8478	9.523	0.36
g	7.84	0.8448	9.521	0.28
h	7.72	0.8511	9.624	1

Table 3.3: Diagnostics for Norbury vortex ring simulations with core radius $\alpha = 0.2$. Norbury's results are in table 3.1 and run parameters in table 3.2. CPU time is for $T = 200$ time steps and normalised to case (h) which took 2.04E6s on a single intel i7_X980 3.33GHz processor.
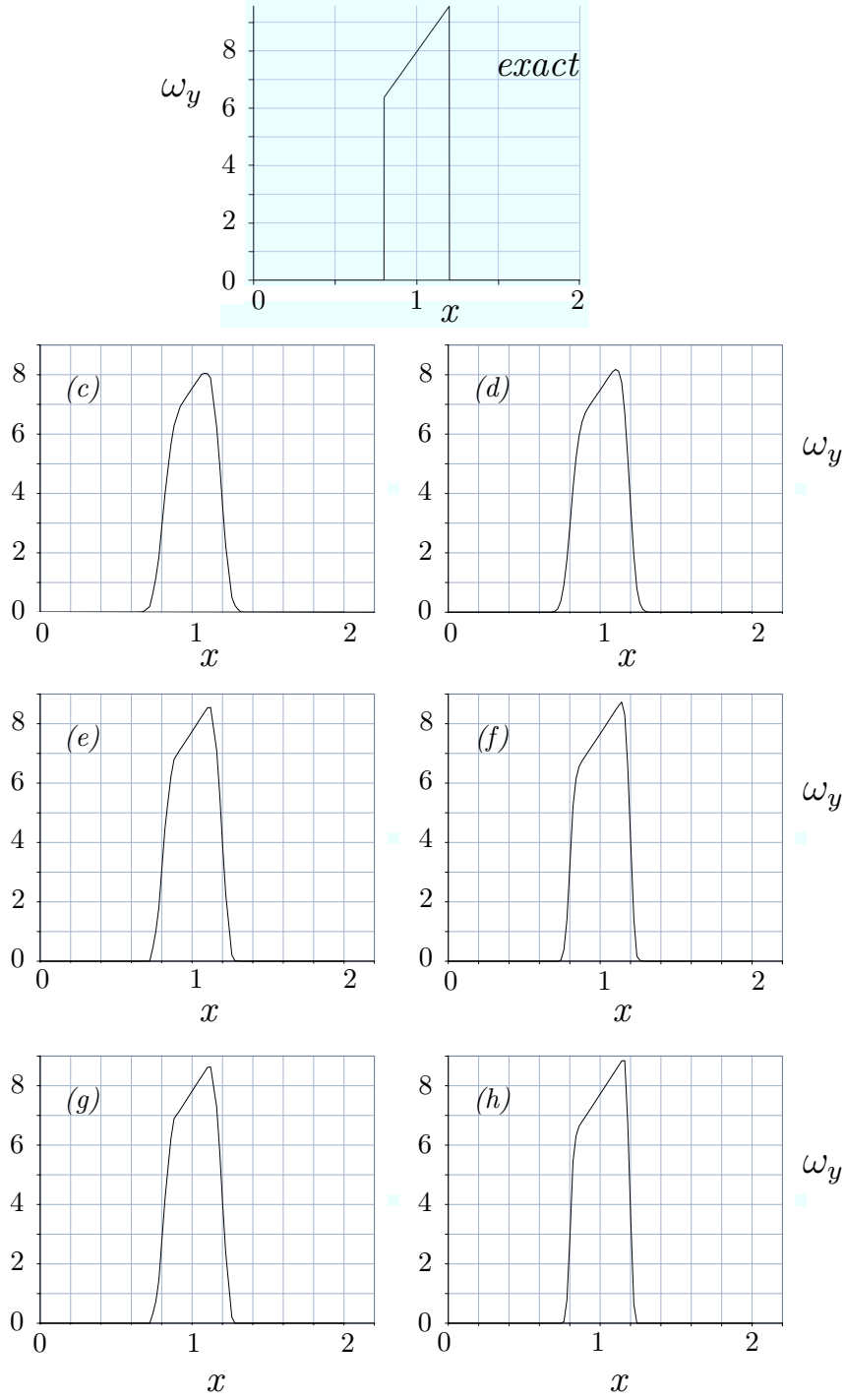
Figure 3.7: Vorticity profile for the Norbury vortex ring $\alpha = 0.2$. Plots shows ω_y against x, with $y = z = 0$, $\mu = 2\pi/128$ and (c) $d = 0.05$, $\gamma' = 1.7$, (d) $d = 0.05$, $\gamma' = 0.7$, (e) $d = 0.025$, $\gamma' = 3.7$, (f) $d = 0.025$, $\gamma' = 1.7$, (g) $d = 0.0125$, $\gamma' = 7.7$ and (h) $d = 0.0125$, $\gamma' = 3.7$ alongside the exact profile.
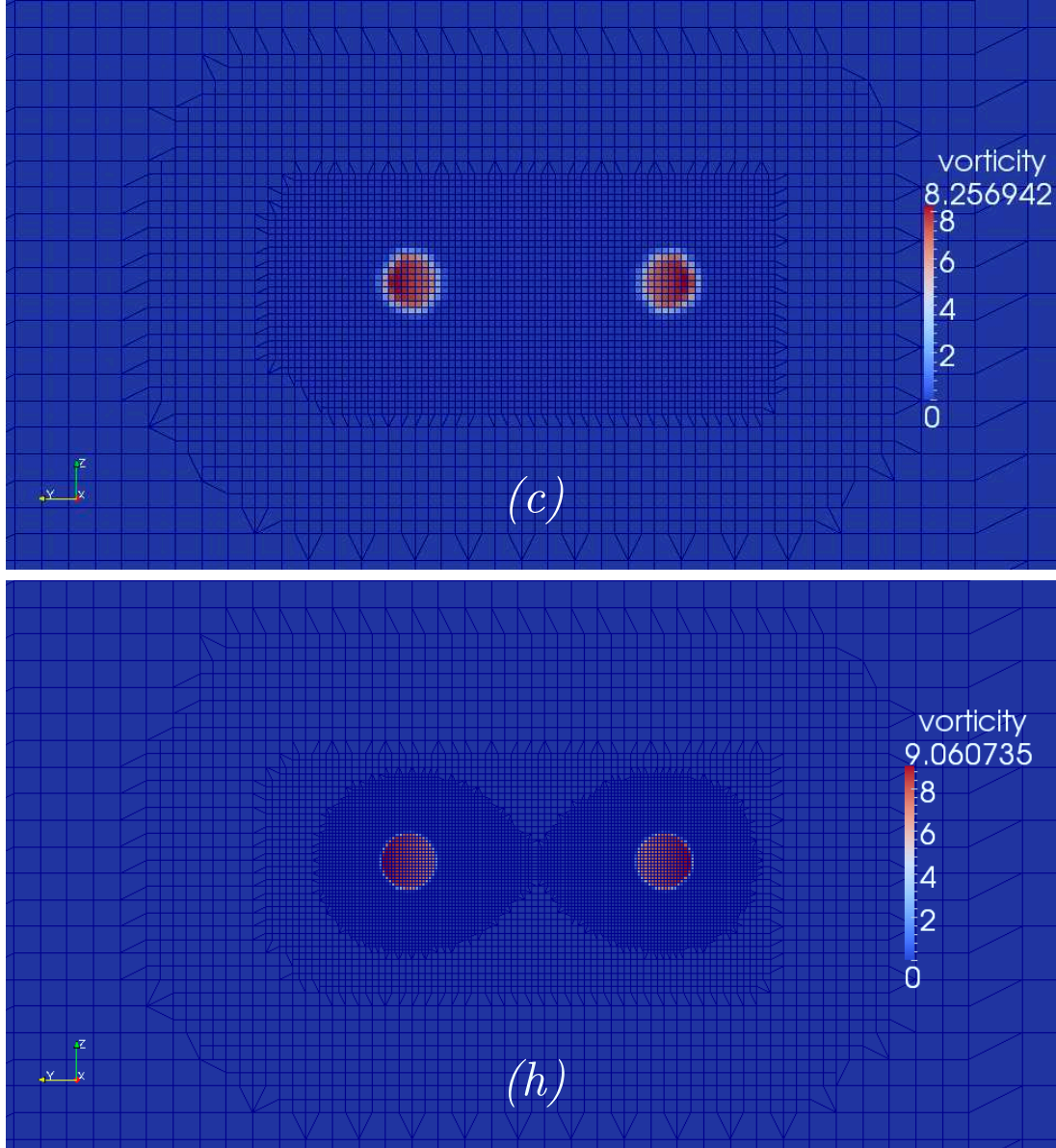
Figure 3.8: t=0 mesh configurations for case (c) (above) and (h) (below) for the $\alpha = 0.2$ Norbury ring. Colour represents vorticity magnitude $|\boldsymbol{\omega}|$.

Figure 3.9: Vortex filaments for the $\alpha = 0.2$ vortex ring, case (d), showing half of the filaments ($N = 6604$). Viewed from a latitude of 60°, and from a distance of 6 ring radii ($6R$). $T = 1$, 125, 250 time steps correspond to $t = 0.038$, 4.53, 8.60.

(c) vs (d), cases (e) vs (f)), we see good accuracy gains. We also see increased smoothness in the filaments themselves (figure 3.16) and better conservation of the invariants (Figures 3.10, 3.12, 3.13 show plots for energy, helicity, z-centroid and total mesh cells for the various cases outlined in table 3.2 for $\alpha = 0.2$).

It seems, however, that the parameter d, controlling the filament resolution and therefore profile accuracy seems to have the most influence on the accuracy of the solution. The results from table 3.3 indicate a significant improvement in both the accuracy of translation rate and the resolution of the vortex core. The results for $d = 0.0125$ may indicate that this trend is curtailed slightly where the filament resolution falls significantly below the grid scale. This is supported by the increase in variability in the plots for energy and helicity in figures 3.12 and 3.13. An argument can therefore be made to maintain higher ratios of d/h_i, which is consistent with retaining 'overlap' of filaments at the level of the grid (this is corroborated in the following section).

It is observed that the invariants are not well conserved at late times when instability is initiated. This is most likely due to mesh refinement not well resolving the filaments as they deform and generate smaller scales. This can be established from cases d_1 and d_2 where we see spurious growth in small scale noise and an associated loss in conservation of the invariants; see figures 3.14 and 3.15. As mentioned in section 3.2.3 various refinement criteria have been tested; in the following sections γ_2 is employed in an attempt to address this issue. In spite of this, for an inviscid calculation with *zero* numerical dissipation, the translation and conservation of these rings is well represented by our method. The introduction of numerical noise and its implications for gridded quantities is dealt with more thoroughly in section 3.3.2.

Variations in the filament descriptions themselves appear to have little influence upon the invariants and translation speed in these calculations. Figure 3.10 contains a plot of the total number of filament nodes N during the evolution.

103

Little correlation is observed between the value of N and the trends in T, H and $\bar{z}$. The influence of renoding is investigated in more detail in cases d_1 and d_2 which have $\mu = 2\pi/64$ and $\mu = 2\pi/256$ respectively. In addition these cases use γ_2 as the refinement criterion to address the issue raised above. Figure 3.14 shows energy, helicity and translation for these cases and figure 3.15 shows filaments at $T = 250$ ($t = 8.67$, 7.78 for d_1 and d_2 respectively). Much of what can be determined from these runs is broadly qualitative: the filaments in figure 3.15 are not as smooth as the corresponding plot in figure 3.16 for case d. The filaments in case d_1 are visibly more angular at the nodes and this lack of resolution results in them being unable to deform smoothly in response to the velocity gradients in the flow (any perceived efficiency gain over case d can be attributed to the mesh refinement). Case d_2 has the corresponding problem where the filament-wise resolution falls below the grid and the grid generated noise (see section 3.3.2) is magnified when the cubic interpolation in the renoding algorithm attempts to smooth the filaments. After much trial and error $\mu = 2\pi/128$ was found to be broadly optimal.

Convergence of the method is difficult to judge and harder to quantify. We see improvements in the method on increasing inter-filament and mesh resolution. However, our metrics for determining accuracy are dependent on our filaments representing some known vorticity field, which is itself a function of filament spacing *and* mesh resolution. This is before we consider time integration, velocity interpolation, renoding etc. For this reason it is only meaningful to comment that the method performs well (energy does not quickly diverge and translation rates match Norbury and Kelvin) and we observe accuracy gains with increased resolution. In section 3.3.3 we compare our method with OpenFOAM's in built incompressible solver and are able to judge better vortexFOAM's strengths.

Table 3.3 also contains the CPU time for the computations performed up to $T = 200$ time steps. This allows for some remarks to be made regarding the

efficiency of the method. The most striking characteristic of these results is that increases in mesh resolution most significantly effect computation time. Having argued above that the parameter d is most influential in gaining an accurate solution it is then tempting to dismiss mesh refinement and enforce a desired level of accuracy by increasing the number of filaments in the calculation. Clearly this is equally undesirable; velocity gradients will not be well resolved as small scales are generated. The rings in these simulations are broadly regular and do not challenge the method in the computation of strong velocity gradients. We are left with striking a balance between adequate mesh resolution, high ratios of d/h_i, and sufficient filaments to well resolve the vorticity profile. These facts inform the following results and will be discussed further in the following sections.

$\alpha = 0.4, 0.6$ **and** 0.8

Simulations were also carried out for vortex rings of varying size to verify that the method is robust at modelling different sizes of the vortex structure. Diagnostics and parameters for runs (i) through (viii), which have core radii $\alpha = 0.4, 0.6$ and 0.8 can be found in tables 3.4 and 3.5. Figure 3.17 shows typical filaments at various times for the three core sizes considered. The times allow one chosen to visualise the filaments away from transitions across the period boundary. Figures 3.18, 3.19 and 3.20 show similar diagnostics to those presented in the previous section, this time for the three larger core sizes considered.

The plots for $\bar{z}$ also now have to be computed in a slightly different fashion due to the frequent transitions with the periodic boundary. We compute $\bar{z}$ by incrementing an average over all filaments of the mean z filament displacement. Here

$$\bar{z}_{j,T} = \frac{\sum\limits_{p}^{N_j} z_p}{N_j}$$

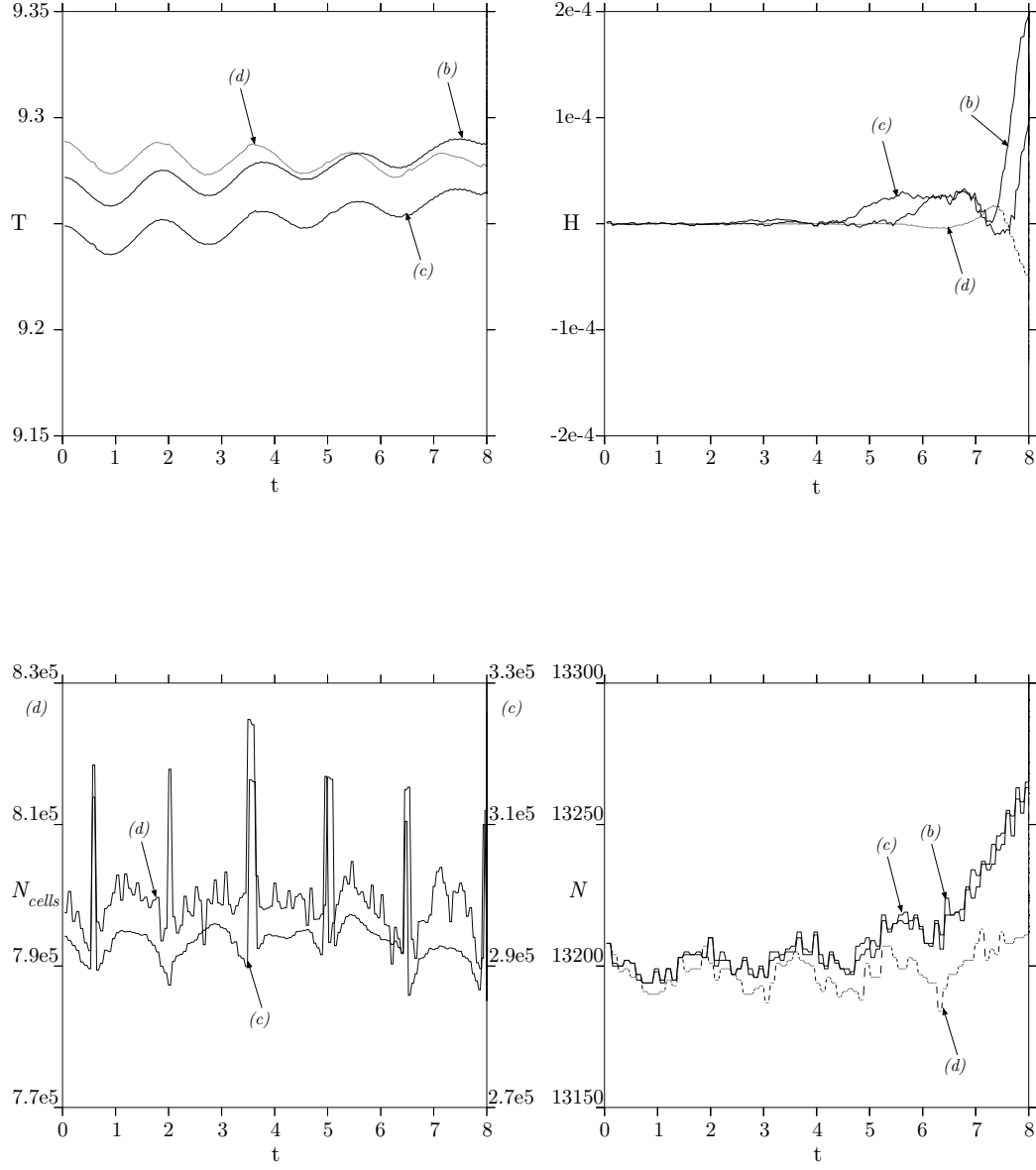is the mean z coordinate on filament j, given N_j nodes on that filament at time

Figure 3.10: Energy, helicity, number of grid cells (N_{cells}) and number of filament nodes (N) for the Norbury ring $\alpha = 0.2$ cases with $d = 0.05$, $\mu = 2\pi/128$, (b) fixed grid (no adaptation), (c) $\gamma' = 1.7$, (d) $\gamma' = 0.7$.

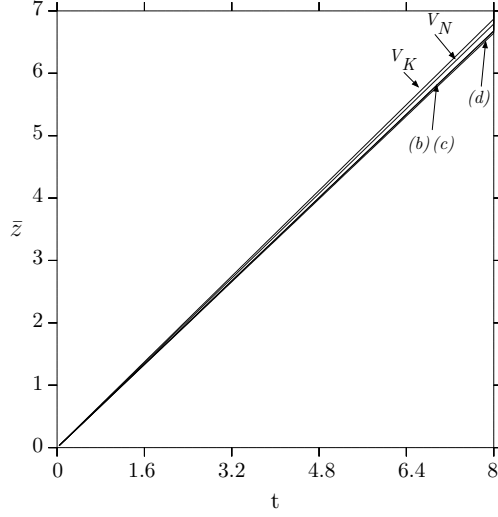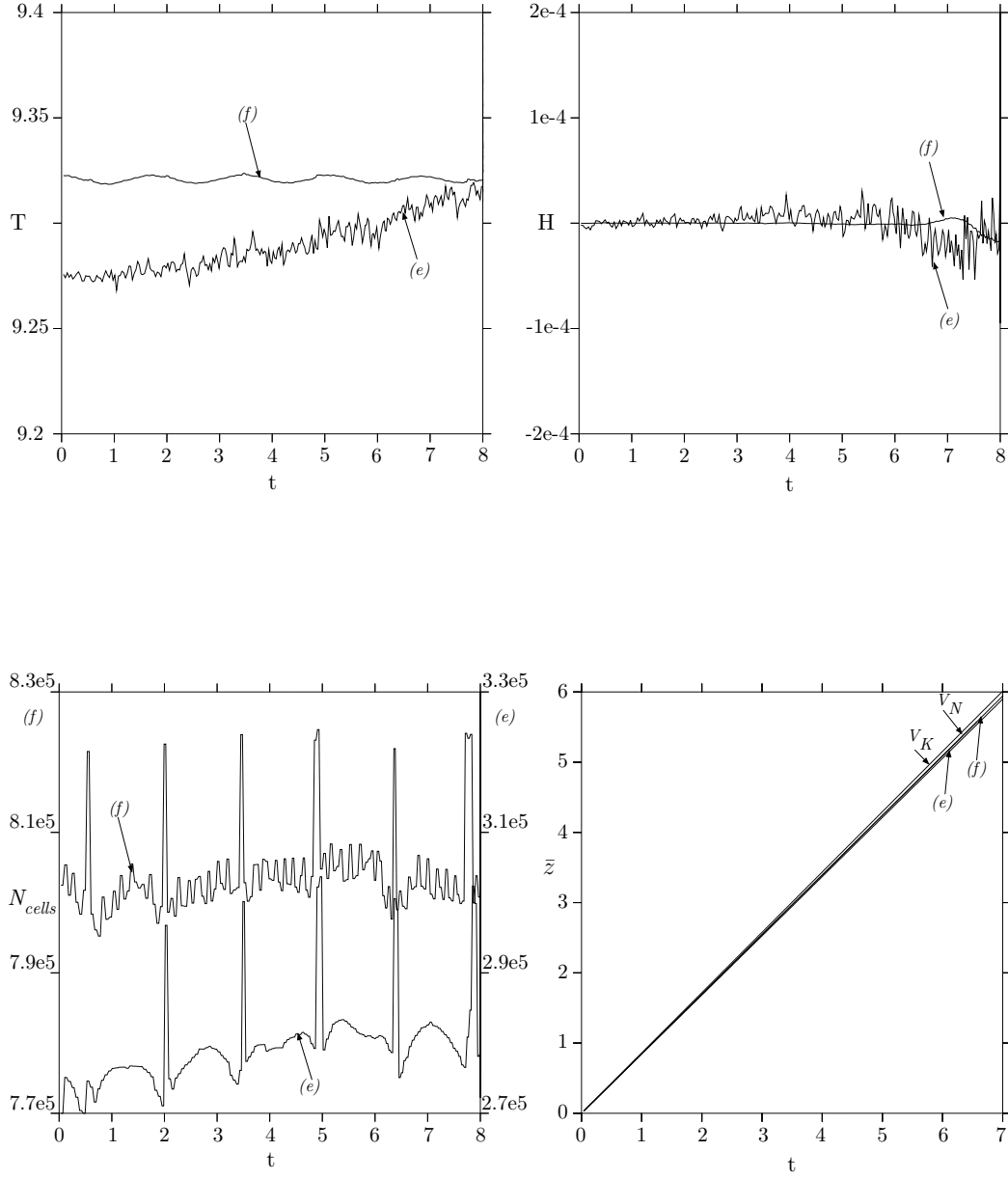Figure 3.11: z-centroid ($\bar{z}$) for the Norbury ring $\alpha = 0.2$ cases with $d = 0.05$, $\mu = 2\pi/128$, (b) fixed grid, (c) $\gamma' = 1.7$, (d) $\gamma' = 0.7$.

case	α	d	μ	γ'	N_{fil}	N	N_{cells}	h_{min}
i	0.4	0.1	$2\pi/128$	1.7	102	13310	365451	$2\pi/128$
ii	0.4	0.05	$2\pi/128$	1.7	405	51632	353761	$2\pi/128$
iii	0.6	0.154	$2\pi/128$	1.7	95	12561	445349	$2\pi/128$
iv	0.6	0.0755	$2\pi/128$	1.7	392	50976	445118	$2\pi/128$
v	0.6	0.0377	$2\pi/128$	2.5	1586	205812	447267	$2\pi/128$
vi	0.8	0.2	$2\pi/128$	1.7	101	13269	834822	$2\pi/128$
vii	0.8	0.1	$2\pi/128$	1.7	393	53082	833814	$2\pi/128$

Table 3.4: Mesh and filament parameters for runs with Norbury vortex ring initial conditions.

Figure 3.12: Energy, helicity, number of grid cells (N_{cells}) and z-centroid ($\bar{z}$) for the Norbury ring $\alpha = 0.2$ cases with $d = 0.025$, $\mu = 2\pi/128$, (e) $\gamma' = 3.7$, (f) $\gamma' = 1.7$.
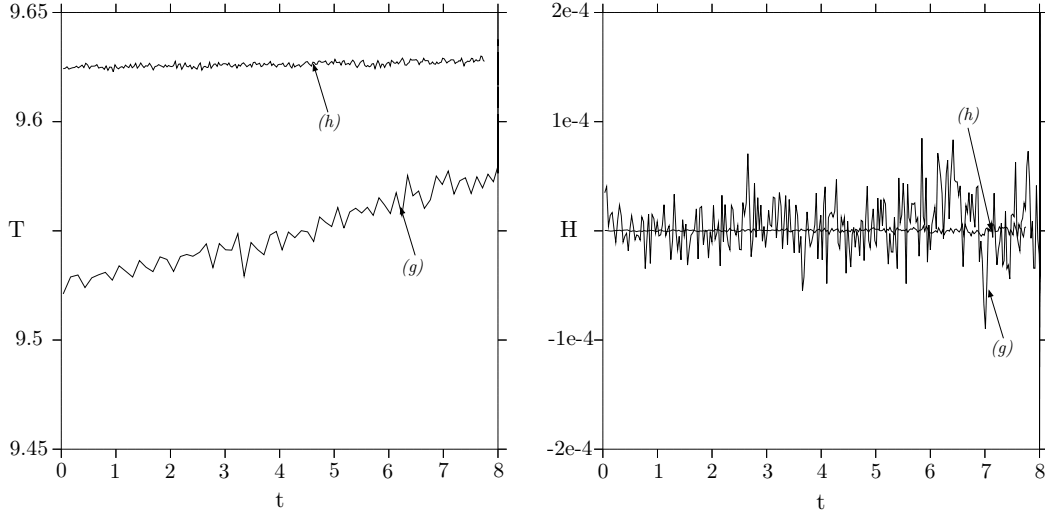
Figure 3.13: Energy and helicity for the Norbury ring $\alpha = 0.2$ cases with $d = 0.0125$, $\mu = 2\pi/128$, (g) $\gamma' = 3.7$, (h) $\gamma' = 7.7$.

case	α	a	V	T	CPU time
i	0.4	1.90	0.6276	6.189	0.09
ii	0.4	1.91	0.6416	6.368	0.13
iii	0.6	0.848	0.5011	4.518	0.15
iv	0.6	0.853	0.5143	4.678	0.20
v	0.6	0.845	0.5161	4.703	0.39
vi	0.8	0.474	0.4017	3.385	0.32
vii	0.8	0.476	0.4166	3.563	0.46

Table 3.5: Diagnostics for Norbury vortex ring simulations. Norbury's results are in table 3.1 and run parameters in table 3.4. CPU time is for $T = 200$ time steps and normalised to case (h) which took 2.04E6s on a single intel i7_X980 3.33GHz processor.
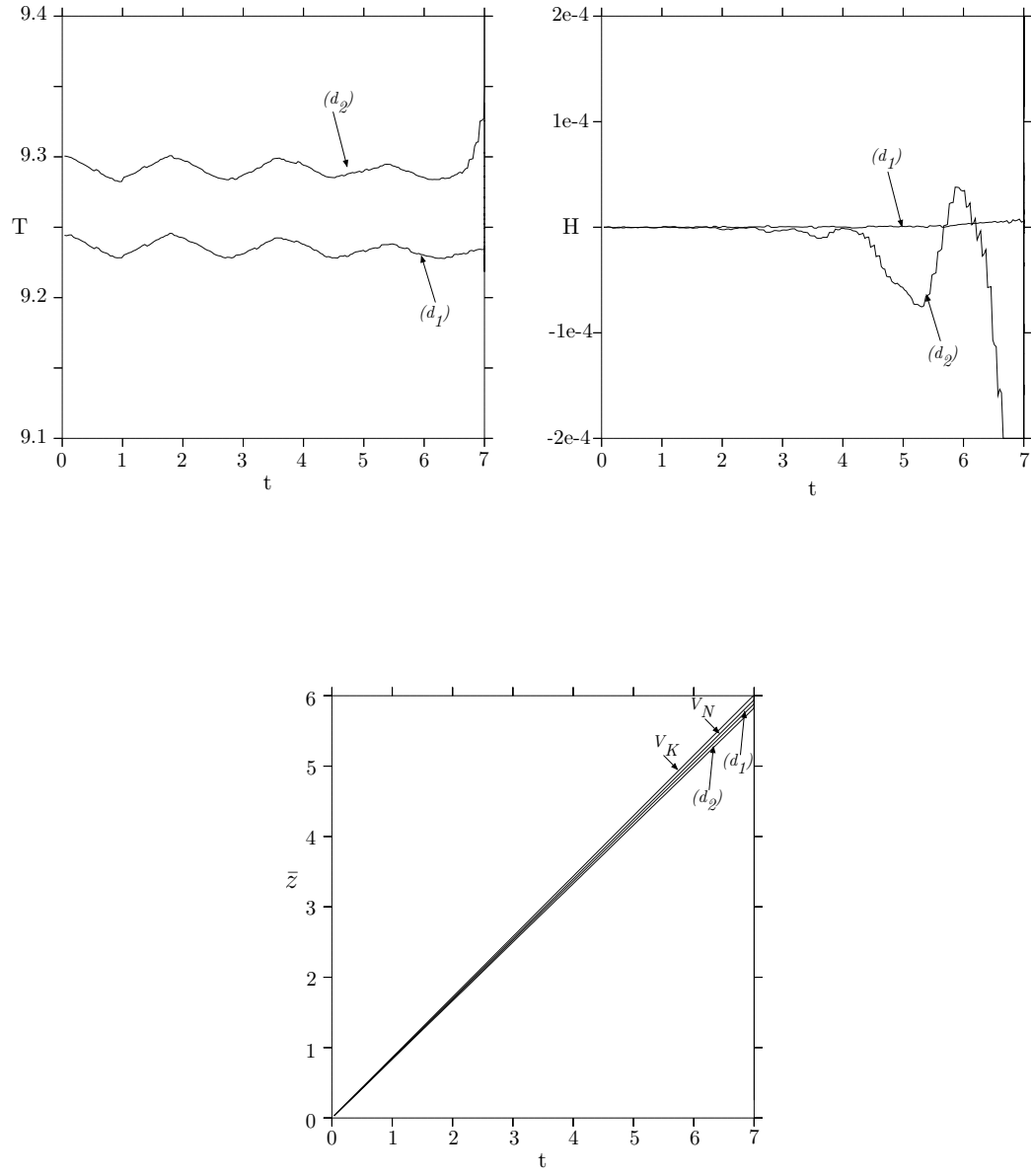
Figure 3.14: Energy, helicity and z-centroid ($\bar{z}$) for the Norbury ring $\alpha = 0.2$ cases with $\gamma' = 0.7$, $d = 0.05$ and (d$_1$) $\mu = 2\pi/64$, (d$_2$) $\mu = 2\pi/256$.
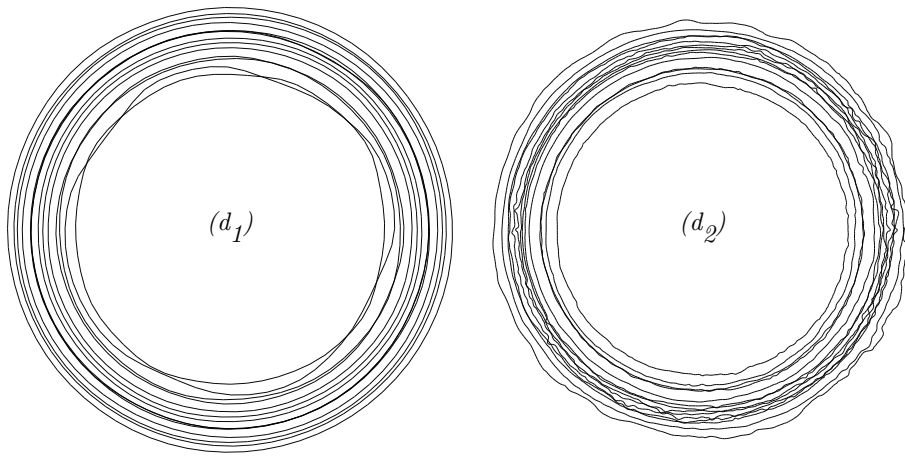
Figure 3.15: Vortex filaments for $T = 250$ ($t \approx 8.66$) of the Norbury ring $\alpha = 0.2$ cases with $\gamma' = 0.7$, $d = 0.05$ and (d$_1$) $\mu = 2\pi/64$, (d$_2$) $\mu = 2\pi/256$. Filaments are viewed from the z-axis and show an eighth of all filaments.
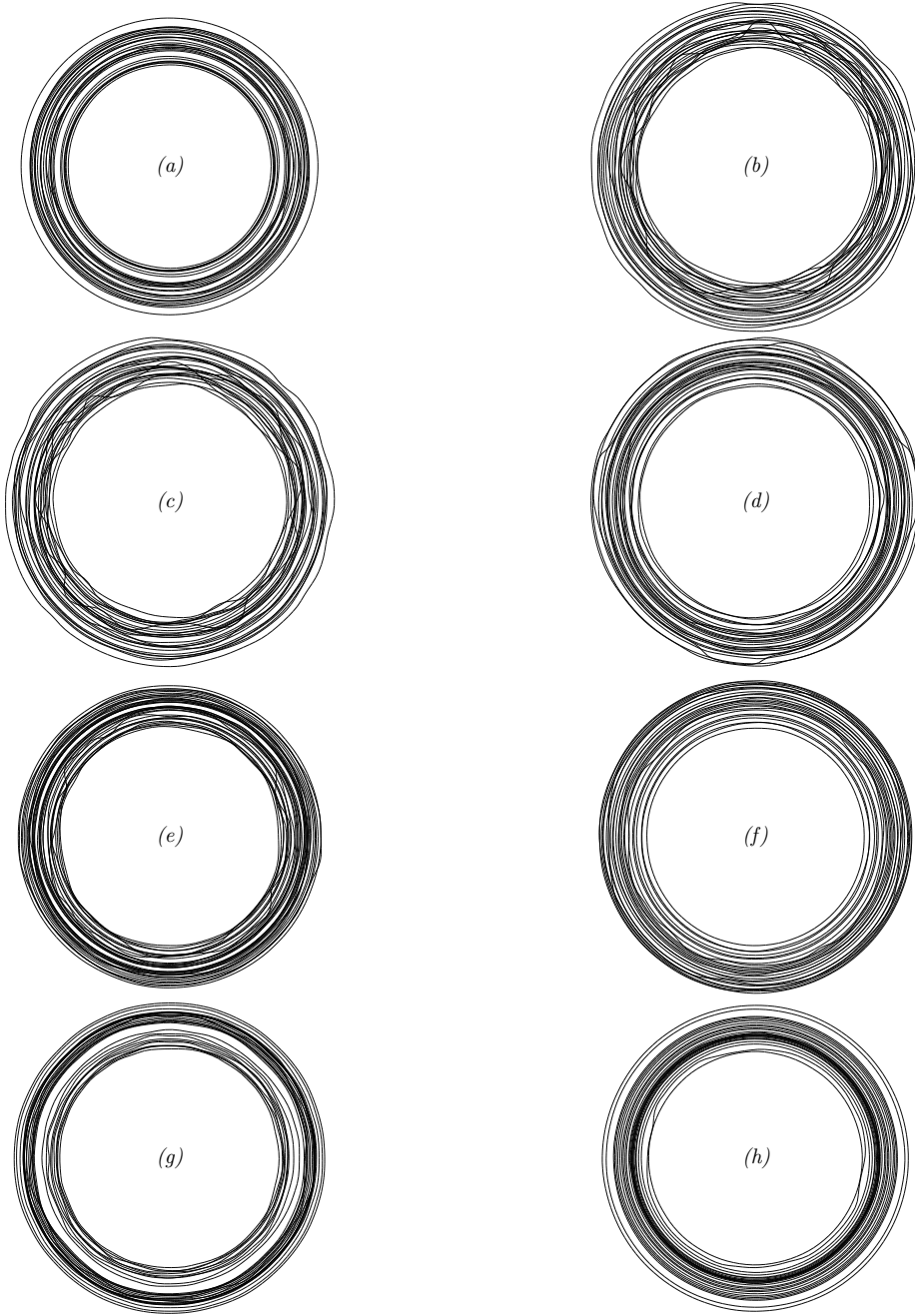
Figure 3.16: Vortex filaments for the $\alpha = 0.2$ vortex ring at $T = 250$ time steps ($t \approx 7 - 8$) for the cases detailed in table 3.2 (approximately 15-20 filaments are rendered for easy visualisation)

step T (we make use of the filaments in this post-processing due to the infrequency with which the mesh and associated fields are output. In future work a similar incremental computation will be made on the grid during runtime). Then the $\bar{z}$ *increment* is

$$\bar{z}' = \frac{1}{N_{fil}} \sum_j^{N_{fil}} \frac{1}{N_j} \sum_p^{N_j} (z_p - \bar{z}_{j,T})$$

and is subjected to a periodicity check to keep it within $(-\pi, \pi)$. This is then incremented from one time step to the next, $\bar{z}_{T+1} = \bar{z}_T + \bar{z}'$, to find the general $\bar{z}$ ignoring periodicity. We do see slight noise in the curves for $\bar{z}$, particularly when d is small (see figure 3.18) but this is preferable to $\bar{z}$ propagating across the boundary.

We find broadly similar results to those in section 3.3.1. We find the agreement with translation speed, V, decreases with core size, and, moreover, gains for increased mesh resolution are smaller. The reason from this is likely to stem from the fact that when the filament radius d is larger, the ratio of d/h is larger, and the vorticity profile, particularly at the edge of the ring, is poorly resolved (see figure 3.21). It remains clear from all of the plots for $\bar{z}$ that the translation is steady.

In energy conservation we see the expected improvements for higher mesh resolution, and the higher d/h ratios result in slightly improved helicity conservation.

3.3.2 Velocity divergence and smoothness

We expect the evolution of these rings to be broadly steady. However (Widnall et al., 1974) and subsequent studies (Moore and Saffman, 1974) (Widnall and Tsai, 1977) (Shariff et al., 1994) (Archer et al., 2008) have shown vortex rings to be unstable to azimuthal disturbances and to radial ones (Leung et al., 2007) . Numerical noise and grid anisotropy will introduce sufficient disturbances to
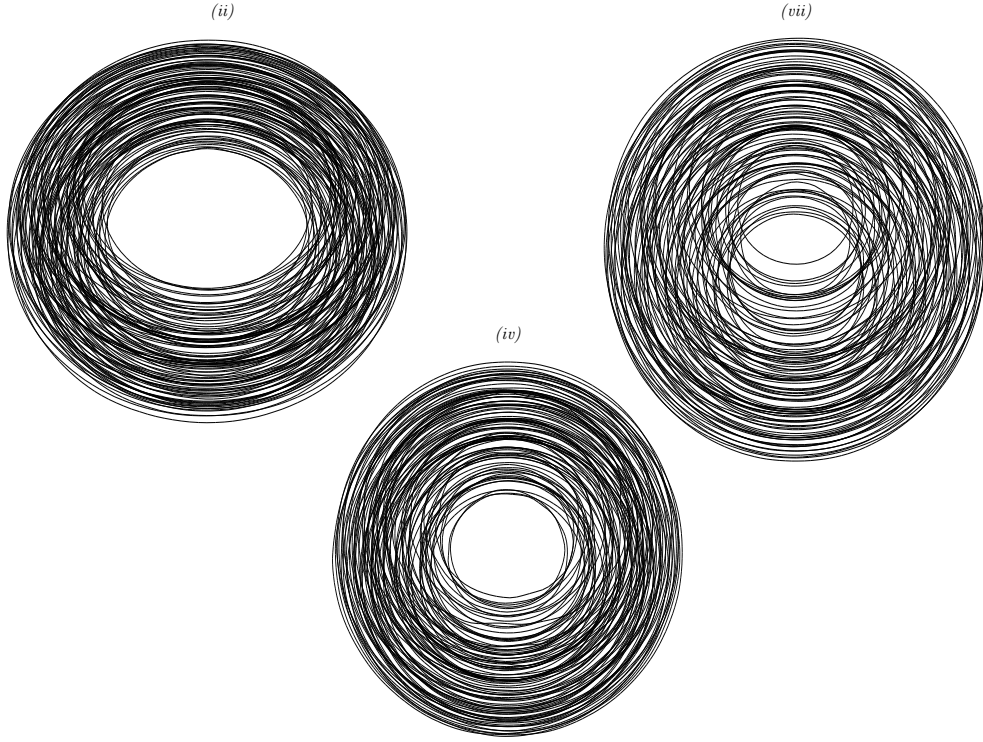
(ii)   *(vii)*

(iv)

Figure 3.17: Vortex filaments for (ii) $\alpha = 0.4$, (iv) $\alpha = 0.6$ and (vi) $\alpha = 0.8$ vortex rings at time steps $T = 150, 130$ and 170 ($t = 17.2$, 30.3, 69.1) respectively for the cases detailed in table 3.4 (one eighth of total filaments are rendered for easy visualisation and times chosen to avoid transitions with periodic boundary). Filaments are viewed from a latitude of $60°$ and a distance of $6R$.
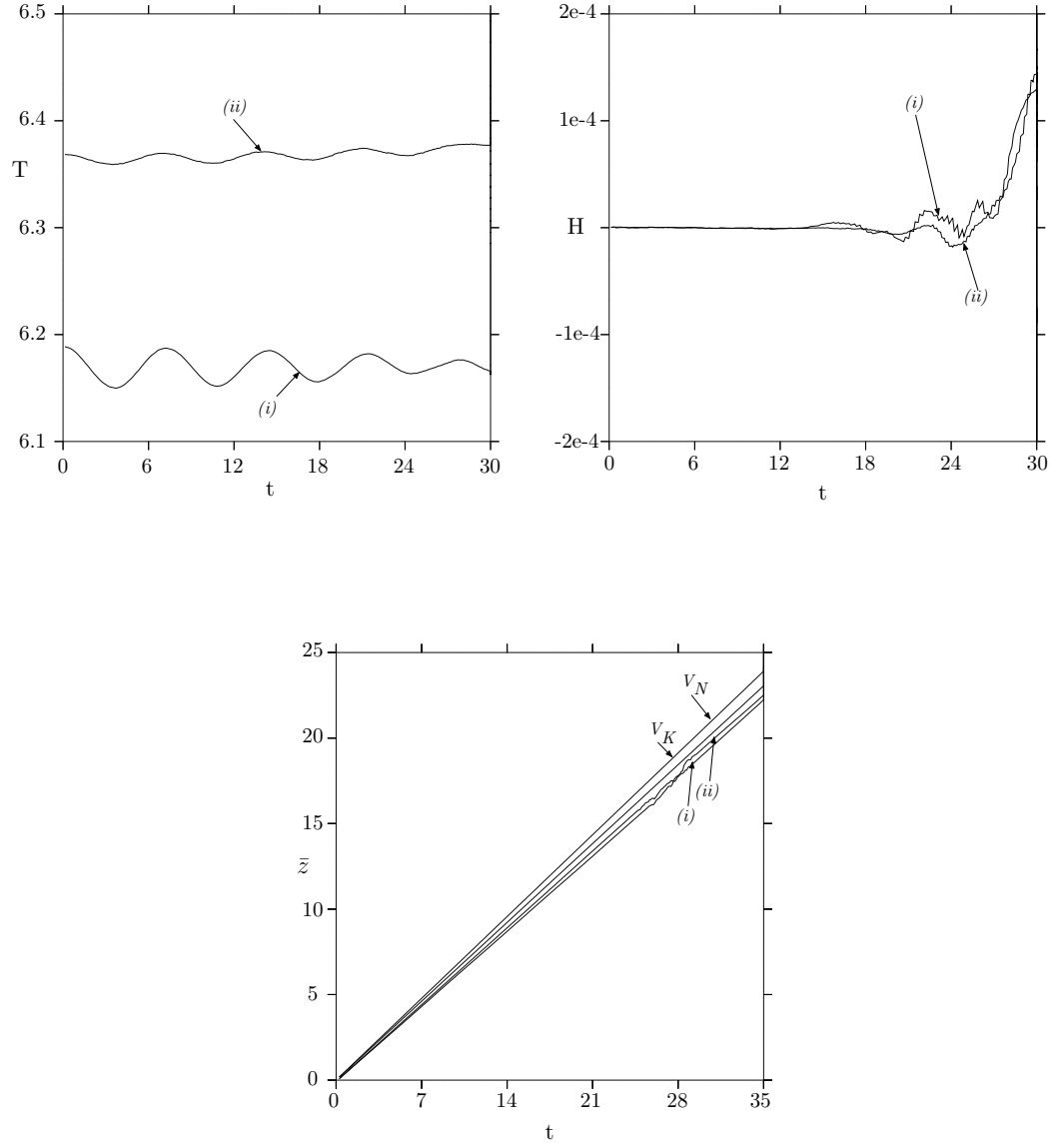
Figure 3.18: Energy, helicity and z-centroid ($\bar{z}$) for the Norbury ring $\alpha = 0.4$ cases with $\gamma' = 1.7$, $\mu = 2\pi/128$, (i) $d = 0.1$, (ii) $d = 0.05$.
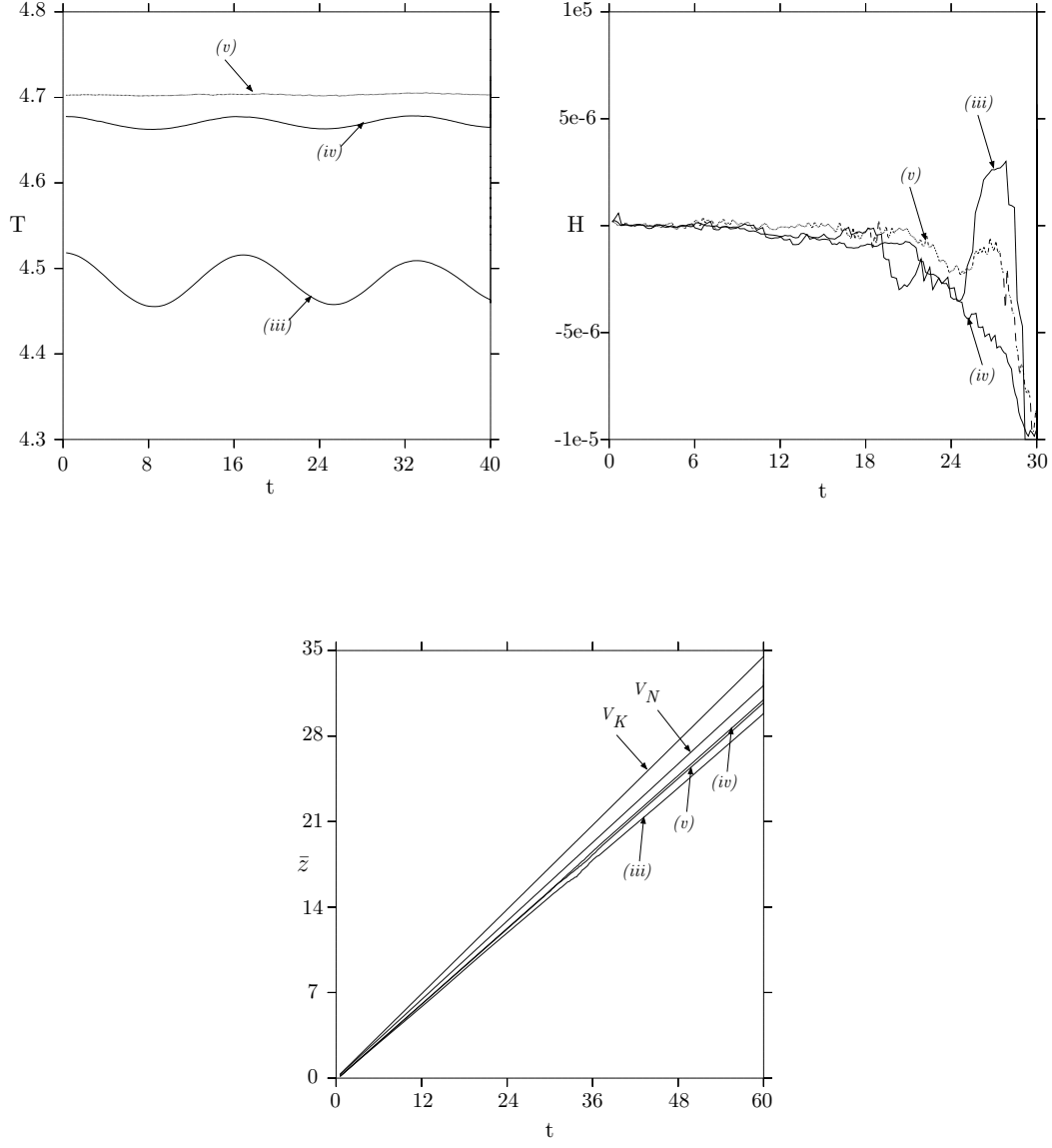
Figure 3.19: Energy, helicity and z-centroid ($\bar{z}$) for the Norbury ring $\alpha = 0.6$ cases with $\mu = 2\pi/128$, (iii) $d = 0.154$, (iv) $d = 0.075$, (v) $d = 0.0378$.

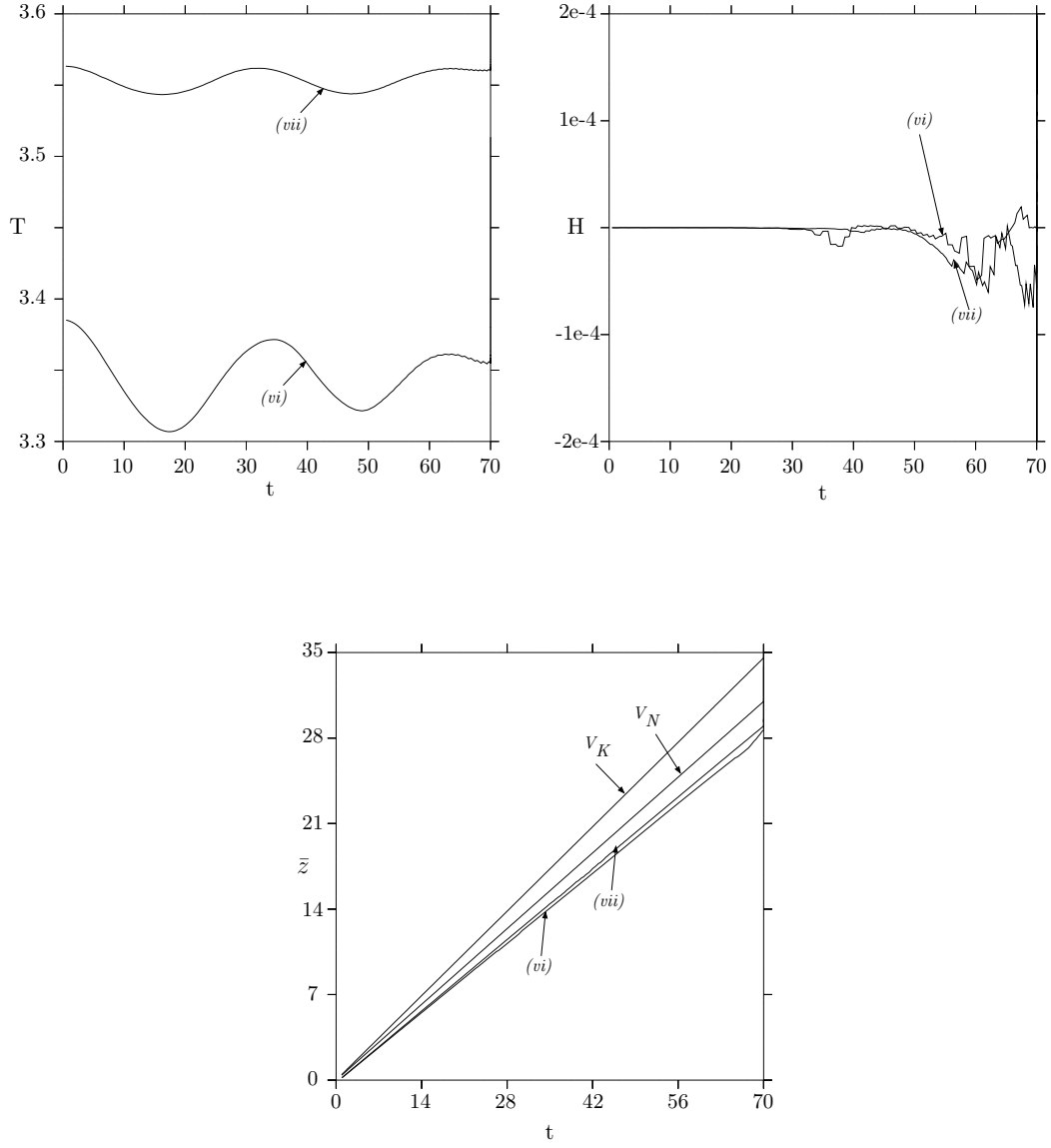Figure 3.20: Energy, helicity and z-centroid ($\bar{z}$) for the Norbury ring $\alpha = 0.8$ cases with $\mu = 2\pi/128$, (vi) $d = 0.2$, (vii) $d = 0.1$.
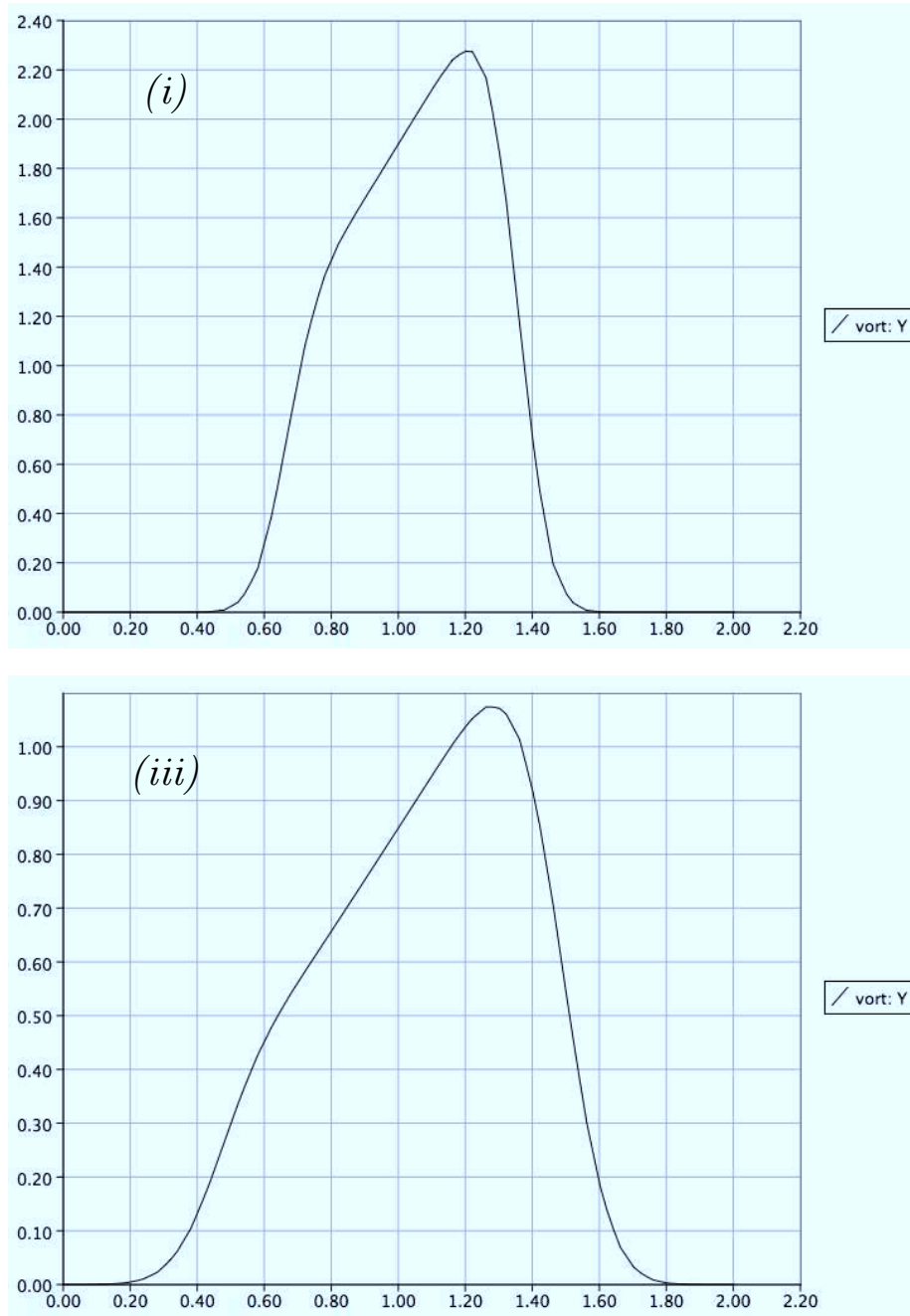
Figure 3.21: Vorticity profile (ω_y) for the Norbury vortex ring cases (i) and (iii) detailed in table 3.4.

instigate the instability of the ring. It is therefore critical that any observed instability be clearly physical in nature. Figures 3.9, 3.16 and 3.17 show filaments for the various Norbury ring cases considered thus far. The behaviour, while broadly similar displays signs of azimuthal inviscid instability (undulating vortex lines, production of halo vorticity and no viscous wake). The various cases show instability triggered at different times and with various degrees of smoothness. Understanding these differences is important in understanding how adaptivity effects the solutions.

The FVM discretisation employed by OpenFOAM allows for a degree of mesh non-orthogonality, where the vector joining cell centres of adjacent cells is not perpendicular to the cell face. This occurs in the locations where the cell sizes, or resolution changes; the refinement patterns. When calculating gradients at the faces between cells (which are used to calculate the laplacian) this introduces errors. For implicit gradients, i.e. inverting differential operators, this is corrected for by interpolations and a series of iterations.

In addition to the non-orthogonality of the mesh, the refinement patterns reduce the order of accuracy of explicit gradients (e.g. the curl of equation (3.5)). This stems from the non-uniformity and skewness of the cells in these regions which impinges upon the accuracy of the interpolation required to computed the cell centred gradients (see section A.1).

These two error sources create spurious gradients arising from the curling of vorticity to be transferred to the Lagrangian filaments near refinement patterns. In fact the velocity field is not exactly incompressible (despite being defined as a curl of a potential), therefore the advection of the filaments can violate Helmholtz's vortex laws. Some details of the numerical schemes involved and the corrections available in OpenFOAM are included in appendix A.

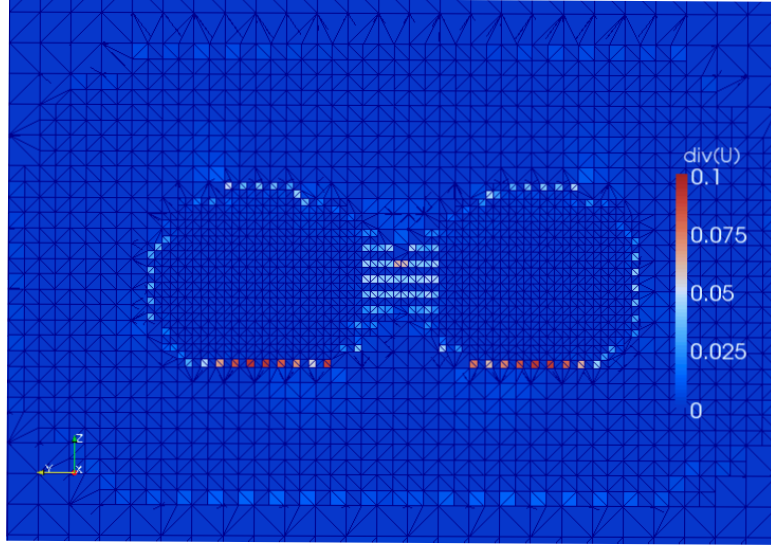This has motivated the refinement strategy outlined in section 3.2.3 so that

Figure 3.22: A typical mesh configuration for a simulation of a vortex ring, colours indicate $\nabla \cdot \boldsymbol{u}$.

we produce meshes with low non-orthogonality in locations near the vortex filaments. We must, however also be wary of such gradients still being introduced to the filaments via the velocity interpolation. Figure 3.23 demonstrates the consequence of a poor mesh refinement where the refinement pattern coincides with the filaments and spurious gradients are transferred to the filaments resulting in unphysical small scale fluctuations.

Figure 3.16 shows the filaments at $T = 250$ time steps ($t \approx 7$) for the $\alpha = 0.2$ ring. The accuracy discussed in section 3.3.1 and observation of the filaments indicate that a smooth representation of vorticity is a contributing factor to an accurate computation. For low filament resolution (large d, case (a)) the core profile for the Norbury ring is poorly represented. However due to smaller ratios of cell width, h_i to d, a smooth profile is obtained. In other words the individual filaments are represented smoothly and therefore are subjected to less noise from the grid (notice, however, these cases fail to give accurate propagation speeds etc. see case(a) in table 3.3 and figure 3.16). A similar argument can be made for small values of d, where the filaments are sufficiently dense to represent the
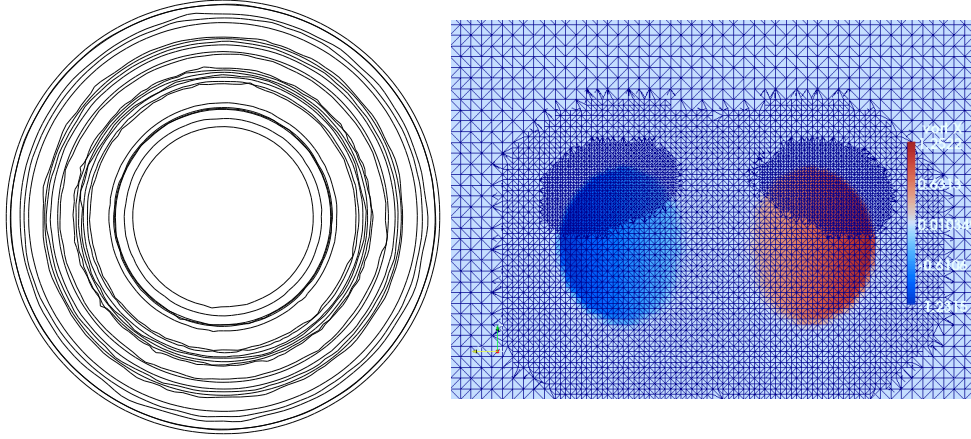
120

Figure 3.23: Filaments and mesh configuration where poor mesh refinement, combined with spurious velocity gradients conspire to introduce erroneous small scale oscillations to the filaments. Mesh on right has a refinement pattern across the core of the vortex ring (colour shows azimuthal vorticity).

discontinuous ring boundary smoothly. i.e. despite the individual filaments not being smoothly represented, their overlapping union is (see cases (g) and (h) in figure 3.16). It is therefore the intermediate values (here the $d = 0.05$ cases) where neither h_i/d is small, nor the filaments dense enough where the smoothness of vorticity beaks down. Cases (b), (c) and (d) show the most irregular filaments in figure 3.16.

Case (b) is included with no mesh refinement, i.e. a fixed uniform grid throughout the calculation. While the diagnostics in table 3.3 and figure 3.10 may indicate a slight advantage in this strategy, due to zero mesh variability, the filaments indicate the benefit of mesh refinement. Case (d) was computed with comparable CPU time, but has smoother filaments due the higher mesh resolution possible.

A breakdown in smoothness also has associated consequences for the calculation of the invariants upon the grid. In particular we see helicity grow that late times (albeit comparatively small relative to the characteristic magnitudes in the

flow, recall $\Gamma = 1$ and $R = 1$). It was discovered, however, that this growth is a further manifestation of mesh variability (in the calculation of H itself), rather than any consistent topological changes in the flow. This is demonstrated in the following section (3.3.4) where a filament approximation of helicity is computed and compared to the gridded quantity. We can also make some more general comments regarding the invariants in section 3.3.4 when the dynamics are more complex and renoding and mesh refinement are more acute.

This section serves to indicate a further complication in the development and calibration of this method. The various adaptive components introduce noise and variability in a dynamic fashion throughout the calculation. Most often this occurs with a degree of positive feedback; small scales generate more adaptation and increased variability. As such, pinpointing the precise cause of a particular feature within a calculation (desirable or not) is extremely difficult.

It should also, however, be remembered that the disturbances are observed after a significant time integration of an *inviscid* calculation. No numerical or physical diffusion is employed to maintain the steady translation rates for these rings. It is therefore inevitable that numerical noise will accumulate in the solution. We must then ensure then that the physical instability is dominant enough to render the numerical noise negligible.

In this effort it is possible to introduce some smoothing to relax the strict mesh properties. Employing Gaussian smoothing in the velocity interpolation (similar to the vorticity assignment) maintains a smoother solution, provided the smoothing length is greater than or equal to the mesh width (the natural scale at which the numerical noise is included). Following the format of section 3.2.1 we define interpolation weights as

$$\Phi_p(\boldsymbol{x}_i) = e^{-\frac{|\boldsymbol{x}_p - \boldsymbol{x}_i|^2}{\sigma_u h_i^2}} \tag{3.26}$$

122

such that velocity assignment is

$$\boldsymbol{u}_p = \boldsymbol{u}(\boldsymbol{x}_p) = \sum_i \left(\frac{\boldsymbol{u}_i \Phi_p(\boldsymbol{x}_i)}{\sum_i \Phi_p(\boldsymbol{x}_i)} \right) \tag{3.27}$$

where $\sigma_u \geq 1$. Figures 3.24 and 3.25 show diagnostics and filaments for the Norbury ring $\alpha = 0.2$ with $\mu = 2\pi/128$, $d = 0.05$ and Gaussian velocity interpolation with $\sigma_u = 1$. Cases have (s) $\gamma' = 0.7$ and (t) $\gamma' = 1.7$ for direct comparison to cases (c) and (d) from section 3.3.1. Case (s) finds $V = 0.8283$ and (t) $V = 0.7967$, compared to (c) $V = 0.8305$ and (d) $V = 0.8351$. We can therefore conclude that the smoothing is more dispersive; when the smoothing lengths are larger we may compute a smoother solution, but it is at the cost of accurate filament velocities.

There are also a number of options available to counter the divergence of velocity in locations of mesh non-orthogonality/uniformity. The easiest to implement but of limited scope is simply to adjust the interpolation scheme used in the computation of gradients (see appendix). Also relatively simple is to make a correction like that used for vorticity using a Helmholtz decomposition, i.e.

$$\boldsymbol{u}^* = \boldsymbol{u} + \nabla\chi \qquad \Rightarrow \qquad \nabla.\boldsymbol{u}^* = \Delta\chi$$

thus

$$\boldsymbol{u} = \boldsymbol{u}^* - \nabla\Delta^{-1}\nabla.\boldsymbol{u}^*.$$

Unfortunately this itself will suffer from inaccurate explicit gradients and requires the inversion of the Laplacian operator. Possibly the most promising line for future work is to devise an entirely new interpolation procedure for which the nodal (vortex) velocity is divergence free. This may be achievable by considering a non-collocated form for velocity, i.e. utilising the face normal components, weighted in some conservative fashion which preserves $\nabla \cdot \boldsymbol{u} = 0$.
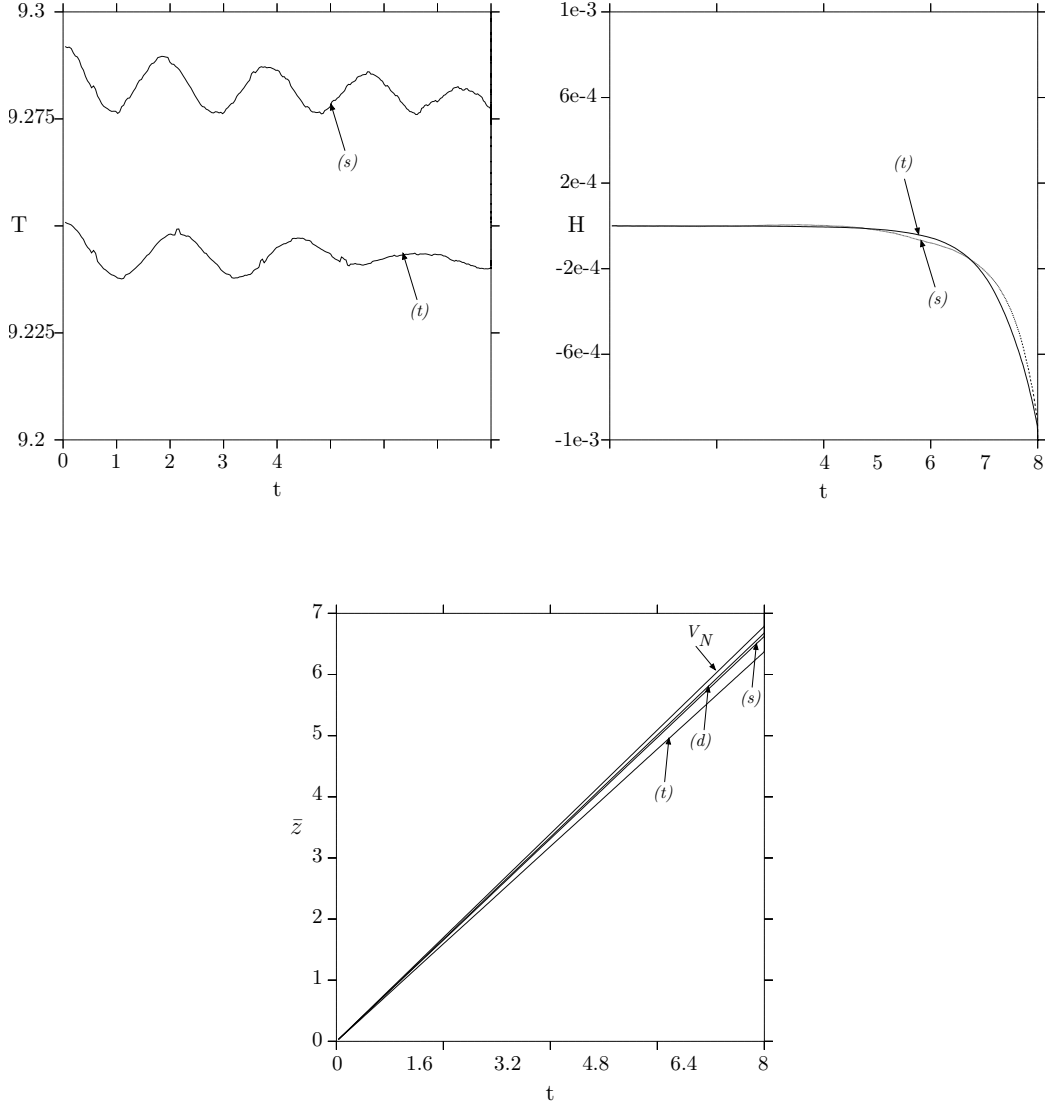
Figure 3.24: Energy, helicity, and z-centroid for the Norbury ring $\alpha = 0.2$ cases with $d = 0.05$, $\mu = 2\pi/128$, (s) $\gamma' = 0.7$, (t) $\gamma' = 1.7$ and Gaussian velocity smoothing.
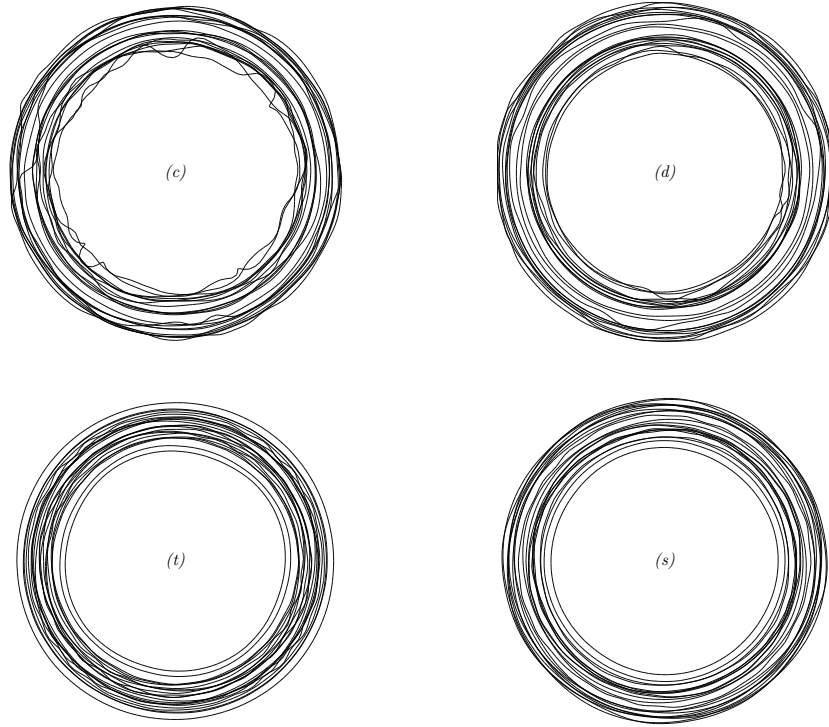
Figure 3.25: Vortex filaments for the $\alpha = 0.2$ vortex ring at $T = 300$ ($t \approx 10$) for the cases (c) and (d) detailed in table 3.2 and cases (s) and (t) with Gaussian velocity smoothing. One eighth of all filaments are shown.

3.3.3 Comparison to icoFOAM

In order to make some more quantitative statements regarding the accuracy of vortexFOAM and benchmark our solution against an established code, we perform some calculations using *icoFOAM*, OpenFOAM's incompressible Navier-Stokes solver. In order to avoid a lengthy exposition of the details of the icoFOAM solver, we refer the reader to the OpenFOAM online documentation at `http://www.openfoam.com` (2011) and `http://www.openfoamwiki.net` (2011). In broad terms icoFOAM uses the FVM method to solve the Navier-Stokes equations using the PISO (Pressure Implicit with Splitting of Operators) algorithm. Here we simply set the kinematic viscosity to zero and select the most conserving interpolation schemes (linear; see appendix for some details on interpolation schemes in the FVM and OpenFOAM). We select Crank-Nicholson time stepping with a time step $\delta t = 0.002$ to maintain a low Courant number. We use the same boundary conditions as vortexFOAM (3.7) and take a velocity field from the inversion of the vorticity field of the $\alpha = 0.2$ Norbury vortex ring (interpolated from the filaments in case (f)) as the initial condition. We employ fixed grids of two resolutions (the grids are fixed in time but are adapted such that there is uniform resolution $r < \pi$) and compare energy conservation properties and isosurfaces of enstrophy.

The energy conservation properties of the two methods under consideration are shown in figure 3.26. A clear decay in energy is observed for the icoFOAM cases (I) and (II) compared to the good conservation of vortexFOAM. Case (I) has a resolution $h_{min} = 2\pi/128$ and (II) $h_{min} = 2\pi/256$, equivalent to the vortexFOAM cases (b) and (f). In fact case (b) uses the same fixed grid as (I) for its inversion, i.e. with $h_{min} = 2\pi/128$ and the adaptation of (f) is such that $h_{min} = 2\pi/256$ for the duration of the simulation (see figure 3.12). The energy loss in the PISO algorithm stems from corrections introduced to alleviate the computational mode which results from using a collocated grid (all variables de-
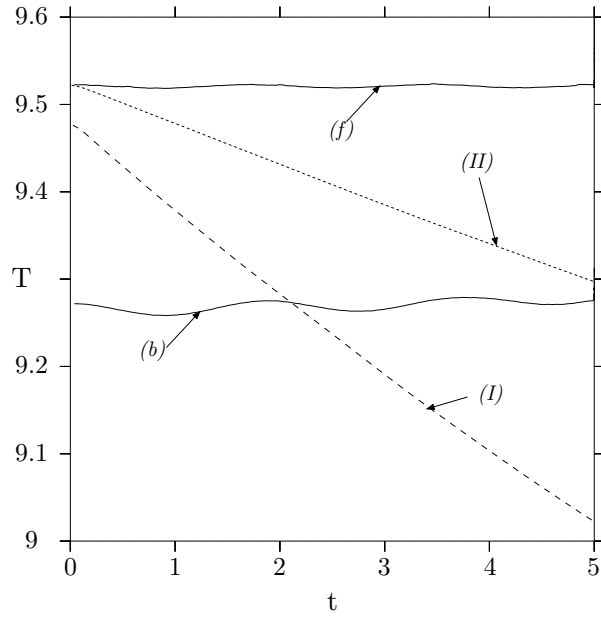
Figure 3.26: Curves for energy T for the $\alpha = 0.2$ Norbury vortex ring. (b) and (f) are the vortexFOAM cases detailed in section 3.3.1 and table 3.2. (I) and (II) are two icoFOAM runs on fixed grids with $h_{min} = 2\pi/128$ and $2\pi/256$ respectively to compare with the fixed grid of (b) with $h_{min} = 2\pi/128$ and the adaptive grid of (f) with lowest cell width $h_{min} = 2\pi/256$.

fined at cell centres), central differences and linear interpolation on orthogonal grids. This mode results in a decoupling of the cell centres from their face neighbours (so called 'checkerboard' effect) and hence spurious oscillations of pressure. The pressure correction (known as Rhie-Chow due to (Rhie and Chow, 1983)) is well known to dissipate energy (Shashank et al., 2010).

Certain caveats should be placed upon any conclusions regarding energy conservation; importantly icoFOAM is designed to be a robust viscous FVM solver, which is adaptable to many different fluid problems. Energy conservation in most of these contexts is insignificant compared to e.g. mass conservation, particularly in low Reynolds number cases where viscosity diffuses much of the energy in any case. A stronger test may be to compare to e.g. a pseudo-spectral inviscid code. In addition conclusions comparing the efficiency of the methods are difficult to draw. The icoFOAM cases shown here use fixed grids, however the functionality for an adaptive mesh is readily available in OpenFOAM and will therefore impact on the CPU time. Also vortexFOAM is still in a proof-of-concept stage of development and has yet to be optimised (for example velocity interpolation requires an unnecessary full grid sweep at each vortex node).

Since the grids under consideration will support the same range of scales we can observe how well those scales are managed by the different algorithms. To this end we present iso-surfaces of enstrophy ($\frac{1}{2}|\boldsymbol{\omega}|^2 = 0.1$) computed from the grids for the three cases (b), (I) and (II) in figure 3.27 along with the initial condition. This figure demonstrates the benefit of the treatment of vorticity used in vortexFOAM in maintaining accurate vorticity structure and highlights the diffusive nature of icoFOAM when used as an inviscid solver. We observe an improvement from (I) to (II) as resolution increases, however instability is still detected in (II) where none is apparent in the lower resolution vortexFOAM case (b) (case (f) is smoother still and is omitted for brevity). The instability observed is synonymous with the viscous transition where the core of the ring spreads and
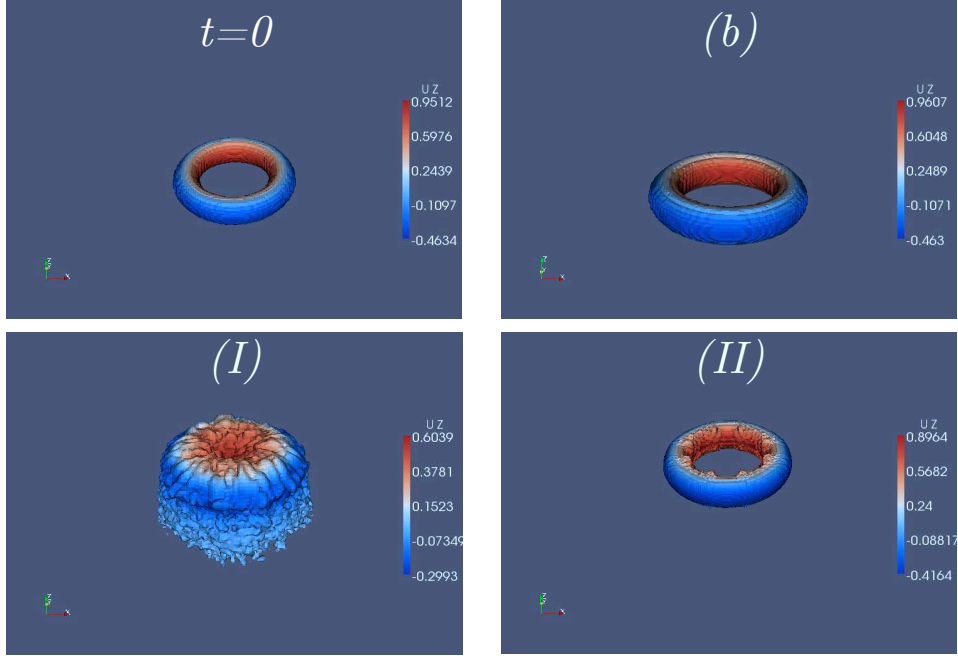
Figure 3.27: Iso-surfaces of enstrophy $\frac{1}{2}|\boldsymbol{\omega}|^2 = 0.1$ for the $\alpha = 0.2$ Norbury vortex ring. Shown are the initial condition $t = 0$, vortexFoam case (b) detailed in section 3.3.1 and table 3.2 and icoFoam cases (I) and (II) on fixed grids with $h_{min} = 2\pi/128$ and $2\pi/256$ respectively. (b), (I) and (II) iso-surfaces at $t = 5$ and colours indicate axial velocity .

a wake is ejected. In addition axial velocities (colours in figure 3.27) are shown to decay upon the iso-surfaces in the icoFOAM cases.

While we account for the strengths of the individual algorithms it seems clear that the formulation of vortexFOAM does hold some advantage when considering inviscid flows of this nature. This may give sufficient motivation to continue the development of the method and address some of the discovered short comings.

3.3.4 Widnall instability

The instability of vortex rings to azimuthal disturbances was first demonstrated by (Widnall et al., 1974) who considered bending modes on a thin core vortex ring of constant vorticity. Solutions are obtained asymptotically in $\varepsilon = \alpha/R$, the ratio of ring radius to core radius, with instability found to be an $O(\varepsilon^2)$ effect. Instability occurs when the magnitude of the strain upon the vortex due to its own curvature is larger than the rotation rate Ω of the bending mode disturbances of the axial vortex. The most amplified wavenumbers are thus those for which $\Omega = 0$, and the first such intersection of the dispersion relation occurs for the second radial mode at $\kappa = k\alpha = 2.51$ for a uniform vortex ring. The final criterion is simply that the ring can only support an integer number of waves around its circumference, i.e. $k = N_W/R$ where N_W is an integer. Thus for the first critical bending mode, given ε, the number of permitted waves can be expressed as

$$N_W = \frac{2.51}{\varepsilon} \qquad (3.28)$$

In order to validate an accurate computation of the azimuthal instability of the vortex ring, we simply choose a wavenumber N_W and, for a ring radius $R = 1$, compute the core size to which this wavenumber is most unstable. In general a vortex ring will be subject to a superposition of various wavenumber instabilities around N_W; therefore, to pre-empt the most unstable mode we must perturb the

ring sufficiently such that the growth rates of the neighbouring modes are negligible. A ring is thus initialised similarly to the Norbury vortex rings of section 3.3.1, only here a uniform radial grid is used, to yield a uniform vorticity profile, and the core is subject to a sinusoidal azimuthal perturbation of amplitude, A (inversely proportional to α; $A = 1/500\alpha$) and wavenumber N_W. Widnall's instability is for the thin core vortex ring regime, i.e. higher wavenumbers, which serves as an ideal test of the method; the smaller the scale of instability the more rigorous a test for our adaptive method. This motivates our choices for N_W below.

To damp grid generated small scales, it was found necessary to introduce the velocity smoothing outlined in section 3.3.2. Results are presented for the cases outlined in table 3.6 including an investigation into the influence of the velocity smoothing paramter σ_u. We also make use of γ_2 as the refinement criterion and this is found to perform more efficiently (for example, 200 time steps in the w_6 case detailed in table 3.6 takes approximately $4e5$ CPU seconds).

case	N_W	α	σ_u	d	γ'	N_{fil}	N	N_{cells}	h_{min}
w_1	13	0.193	1	0.0242	0.7	198	34752	140380	$2\pi/128$
w_2	13	0.193	2	0.0242	0.7	198	34752	140380	$2\pi/128$
w_3	15	0.167	1	0.0209	0.7	198	43070	138070	$2\pi/128$
w_4	15	0.167	2	0.0209	0.7	198	43070	138070	$2\pi/128$
w_5	17	0.147	1	0.0185	0.7	198	51497	138154	$2\pi/256$
w_6	17	0.147	2	0.0185	0.7	198	51497	138154	$2\pi/256$
w_7	19	0.132	1	0.0165	0.7	196	45178	188456	$2\pi/256$
w_8	19	0.132	2	0.0165	0.7	196	45178	188456	$2\pi/256$

Table 3.6: Parameters and diagnostics for various Widnall instability cases with $\mu = 2\pi/128$.

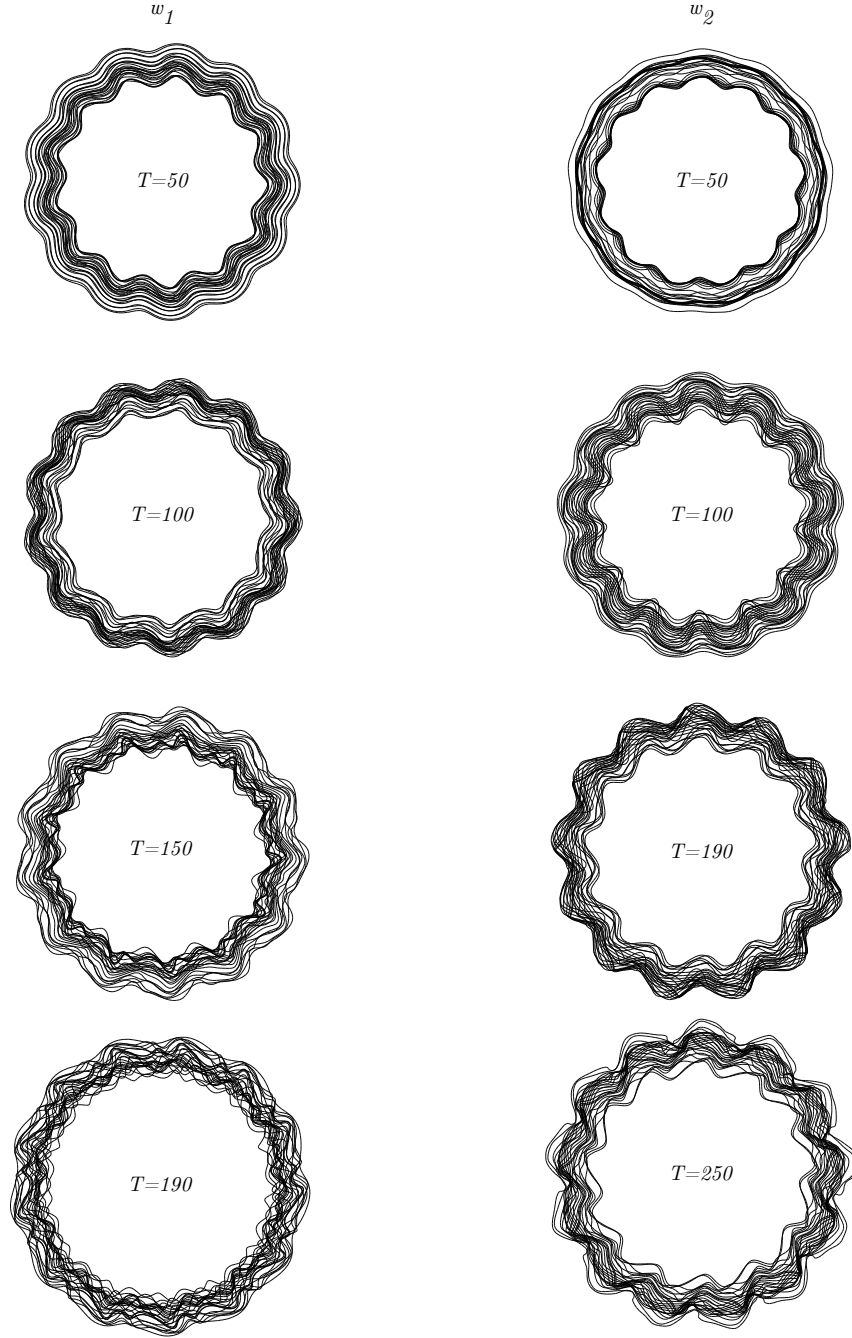In figure 3.28 we present filaments for the $N_W = 13$ cases w_1 and w_2 at

Figure 3.28: Vortex filaments for the $N_W = 13$ Widnall vortex ring instability for the cases w_1 and w_2 detailed in table 3.6. Filaments are viewed from the z-axis and one eighth of all filaments are shown ($T = 190$ corresponds to $t \approx 3.4$).
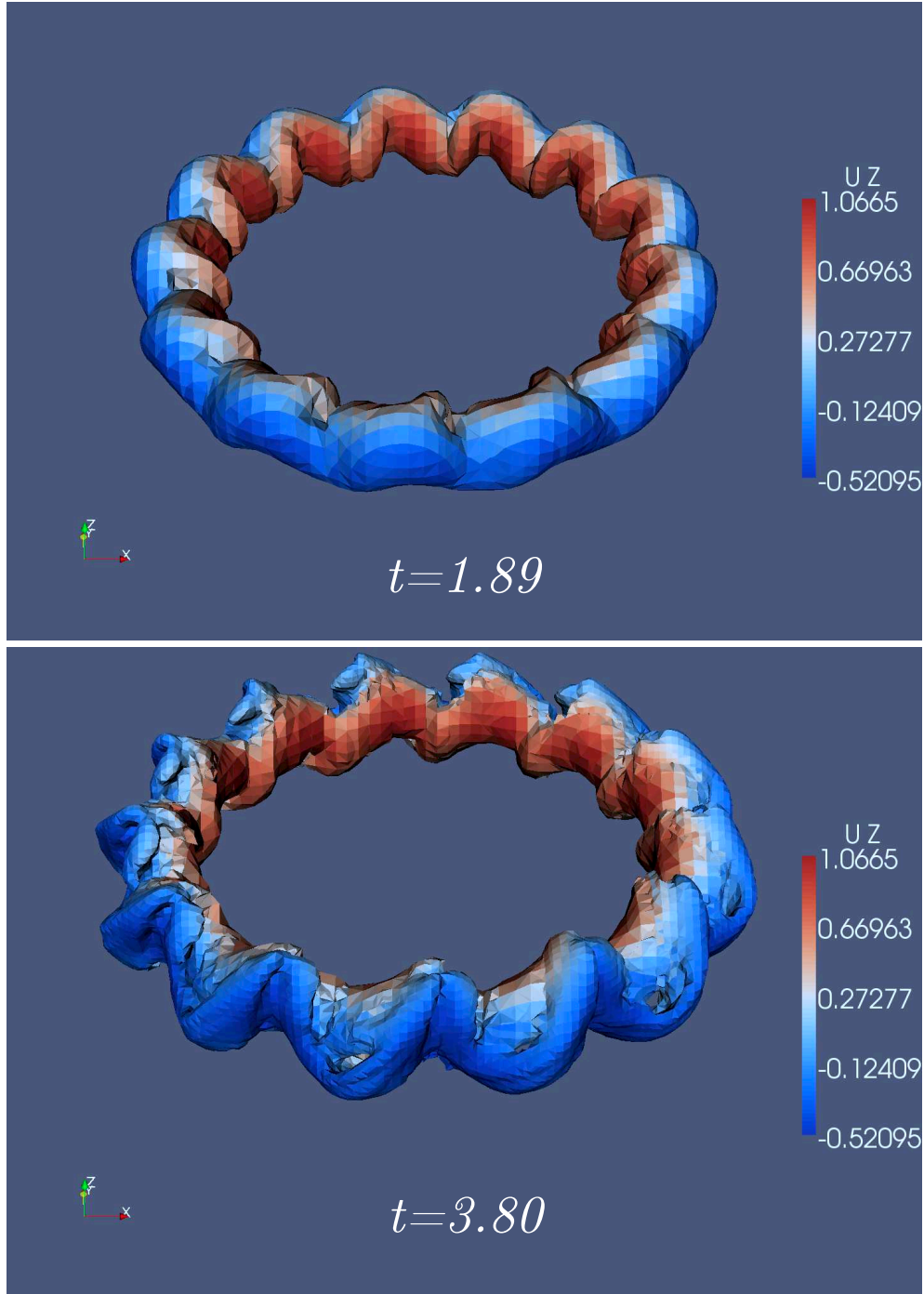
Figure 3.29: Enstrophy isosurfaces $\frac{1}{2}|\boldsymbol{\omega}|^2 = 25$ for the $N_W = 13$ case w_2 at $t = 1.89$ and $t = 3.80$. Colour shows vertical velocity.
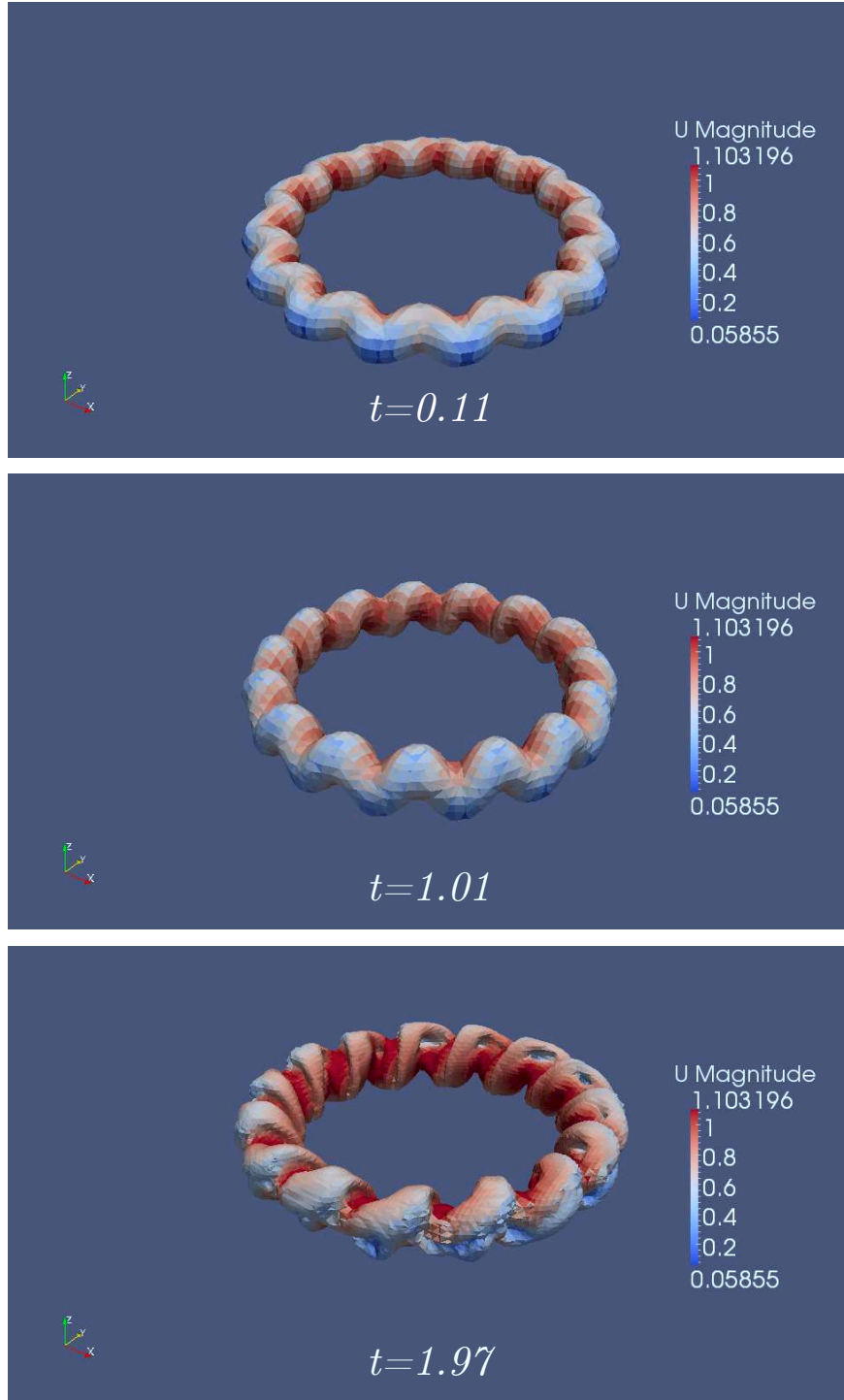
Figure 3.30: Enstrophy isosurfaces $\frac{1}{2}|\boldsymbol{\omega}|^2 = 30$ for the $N_W = 15$ case w_4 at $t = 0.11$, $t = 1.01$ and $t = 1.97$. Colour shows velocity magnitude.
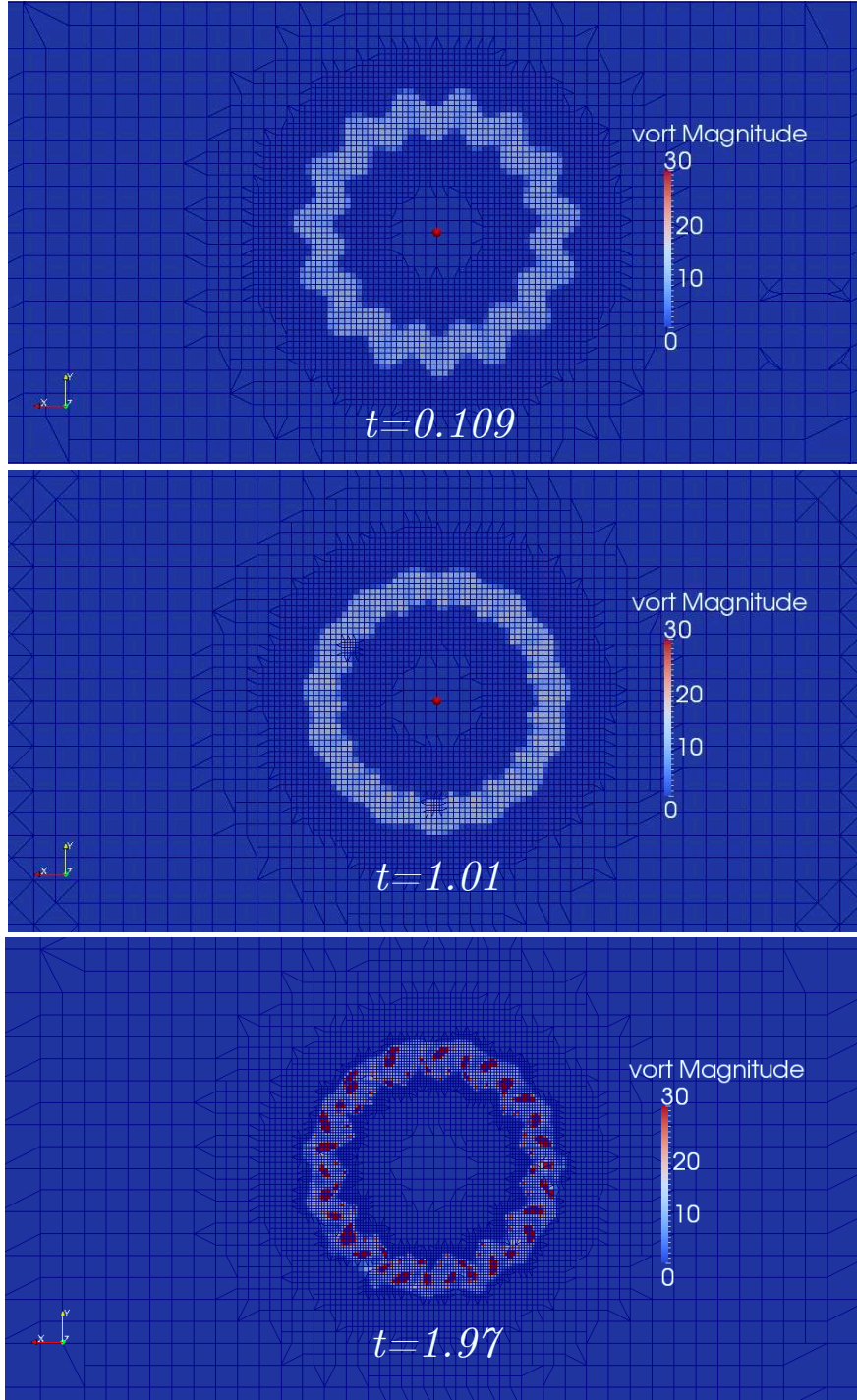
Figure 3.31: Mesh and vorticity magnitude for the $N_W = 15$ case in the xy-plane. Cross sections are at $z = 0$, $z = \pi/16$ and $z = \pi/8$ for times $t = 0.11$, $t = 1.01$ and $t = 1.97$ respectively.
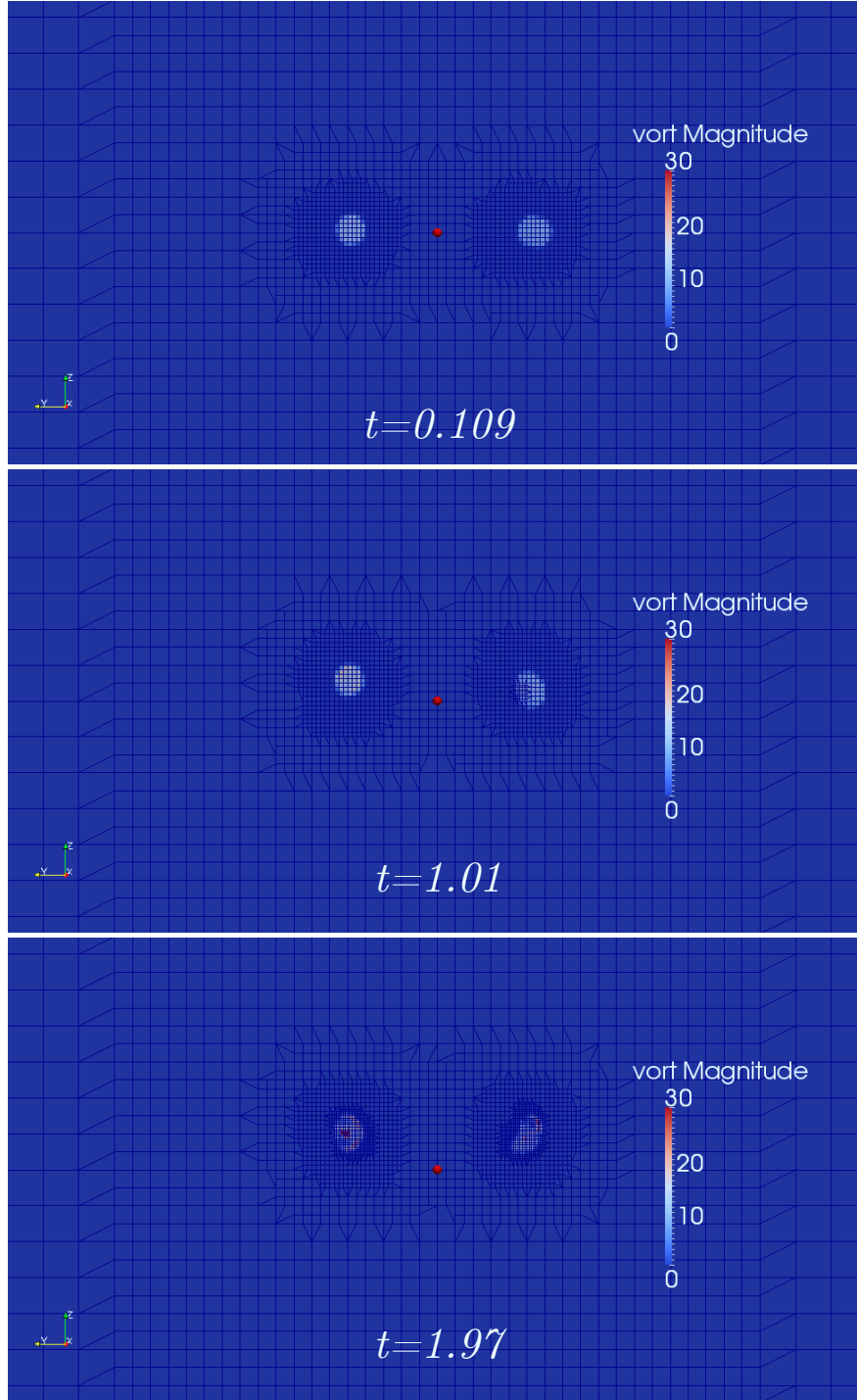
135

Figure 3.32: Mesh and vorticity magnitude for the $N_W = 15$ case w_4 at $t = 0.11$, $t = 1.01$ and $t = 1.97$ in the zy-plane. Cross section is at $x = 0$.
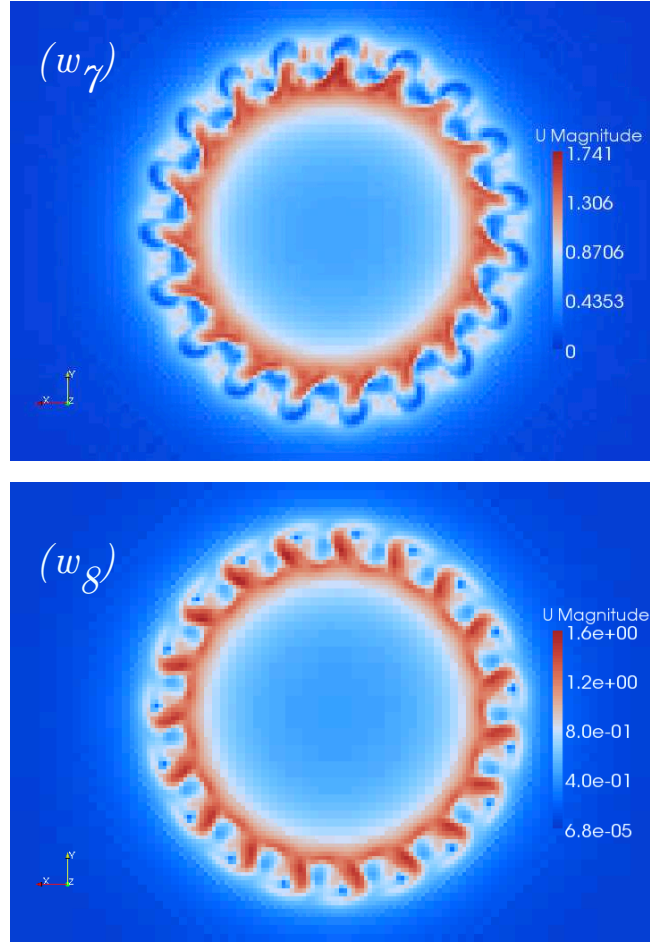
Figure 3.33: Velocity magnitude for the $N_W = 19$ cases w_7 and w_8 at $t = 1.97$ in the xy cross section at $z = \pi/8$.
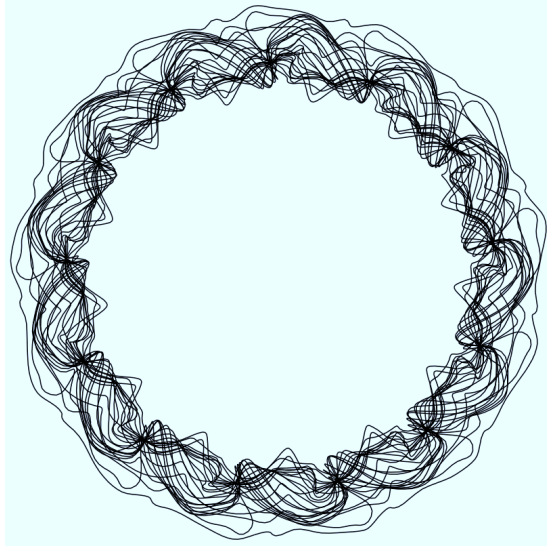
Figure 3.34: Vortex filaments for the w_2 case at the final time $t = 4.3$

several times throughout the simulation. At late times the smoothness of the w_2 case (which has double the velocity smoothing) is certainly improved over w_1 where small scales are observed developing. As we have already discussed, the introduction of noise into the simulation is inevitable. It is therefore an open question as to whether these small scales are due to mesh generated gradients or are in fact the physical secondary instabilities of the vortex filaments, which are being damped in the w_2 case. It is certainly true in all cases that we see the N_W perturbation grow, steepen and eventually break, causing the filaments to spiral around the core. Figure 3.34 shows the vortex filaments for the final time ($t = 4.3$) for the w_2 case and is included to demonstrate the compactness of the vortex lines at late times; there is no diffusion of the vorticity field, no viscous wake is produced and the vortex lines continue to spiral around the core generating secondary instability.

The filament representation will in general fall below the grid scale. Due to the relatively steady Norbury simulations considered thus far, we have had little cause to consider how well the grid tracks the vorticity representation. For this

reason we now examine the gridded vorticity field and the mesh structure for the Widnall instability cases. Figures 3.29 and 3.30 show enstrophy (defined as half the vorticity magnitude squared) isosurfaces at various times in the w_2 and w_4 cases. This enables us to determine how well the structure of the instabilities (or filaments) are represented upon the adapted grid. Again the steepening of the azimuthal instability is clear, as is the production of the so called "halo vorticity", where radial bands of vorticity sitting proud of the main core are observed. Colours show velocity and indicate the intensification of vorticity at the core.

Figures 3.31 and 3.32 show the mesh in xy and zy-planes respectively for the w_4 case. This example is particularly convenient as we observe the refinement tracking the vorticity profile, and also adapting to track small scales, and associated vorticity intensification. Figure 3.31 also shows how vortex stretching in the azimuthal bands intensifies vorticity as the vortex lines wrap around the core (see also figure 3.40 for a comparison with the filaments).

Figure 3.33 shows velocity magnitude for the cases w_7 and w_8 in an xy plane. This highlights the sharp gradients of velocity in the disturbances around the ring. The w_8 case is visibly more uniform as the instability progresses with more symmetry (see also figure 3.42 for a comparison with the filaments).

Having made some comment upon the gridded fields, we can turn our attention to the invariants, energy and helicity. Figures 3.35, 3.36, 3.38 and 3.39 show the invariants alongside diagnostics of mesh size (total cell number N_{cells}) and filament nodes (N). In contrast to the results of the section 3.3.1 we see significant growth in the magnitudes of energy and helicity. We also see significant growth in the number of mesh cells and filament nodes. Figures 3.28 to 3.33 showing the mesh and filaments make clear that this resolution increase is important in well resolving the instability. The question then remains whether we can correlate the loss of conservation to the adaptivity in the method and if this is justifiable.

Figure 3.35: Energy, helicity, number of grid cells (N_{cells}) and number of filament nodes (N) for the Widnall instability $N_W = 13$ cases with $d = 0.0242$, $\mu = 2\pi/128$, $\gamma' = 0.7$, w_1 has $\sigma_u = 1$ and w_2 has $\sigma_u = 2$.

Figure 3.36: Energy, helicity, number of grid cells (N_{cells}) and number of filament nodes (N) for the Widnall instability $N_W = 15$ cases with $d = 0.0209$, $\mu = 2\pi/128$, $\gamma' = 0.7$, w_3 has $\sigma_u = 1$ and w_4 has $\sigma_u = 2$.
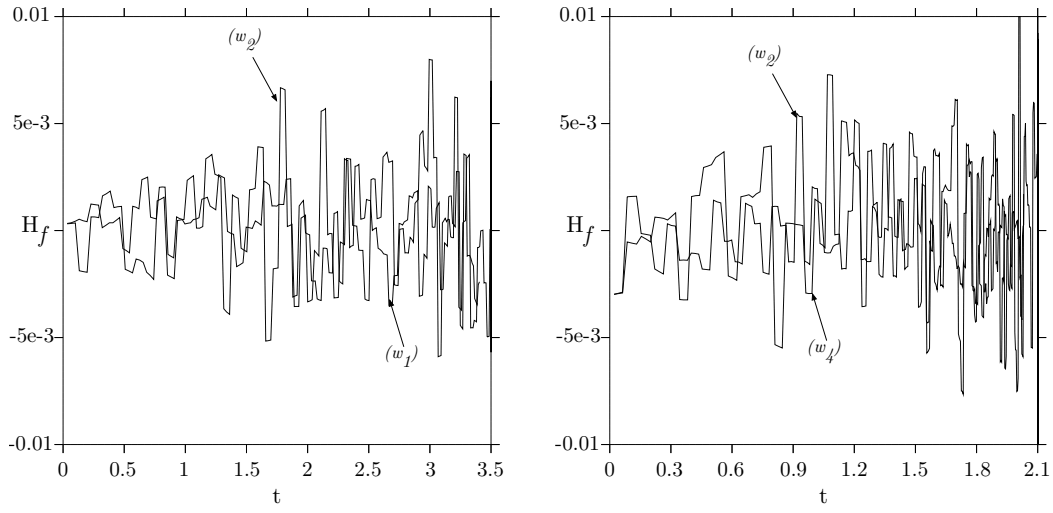
Figure 3.37: Filament helicity for the Widnall instability $N_W = 13$ (left) and $N_W = 15$ (right) cases with $d = 0.0209$, $\mu = 2\pi/128$, $\gamma' = 0.7$, w_1 and w_3 have $\sigma_u = 1$ and w_2 and w_4 have $\sigma_u = 2$.
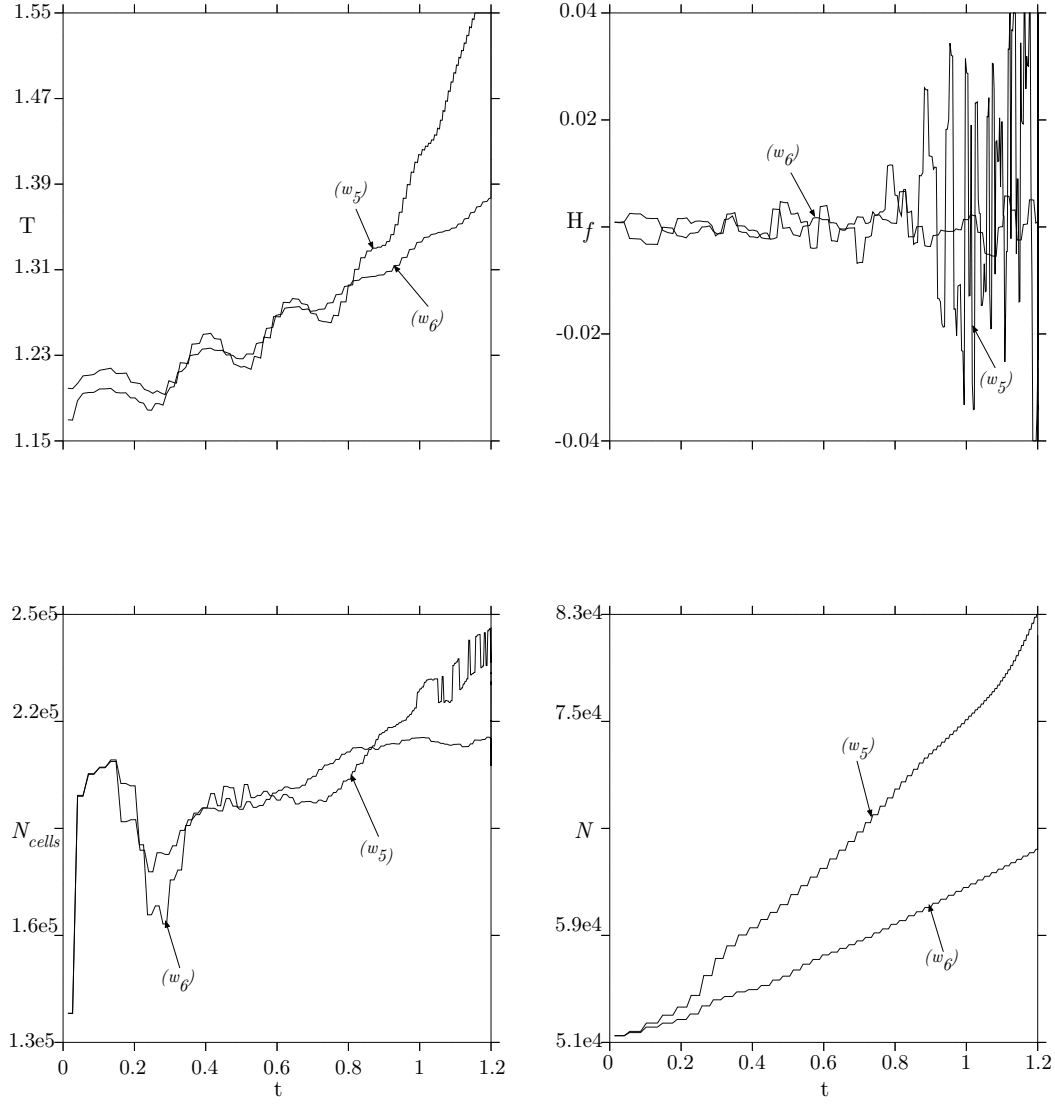
Figure 3.38: Energy, helicity, number of grid cells (N_{cells}) and number of filament nodes (N) for the Widnall instability $N_W = 17$ cases with $d = 0.0185$, $\mu = 2\pi/128$, $\gamma' = 0.7$, w_5 has $\sigma_u = 1$ and w_6 has $\sigma_u = 2$.
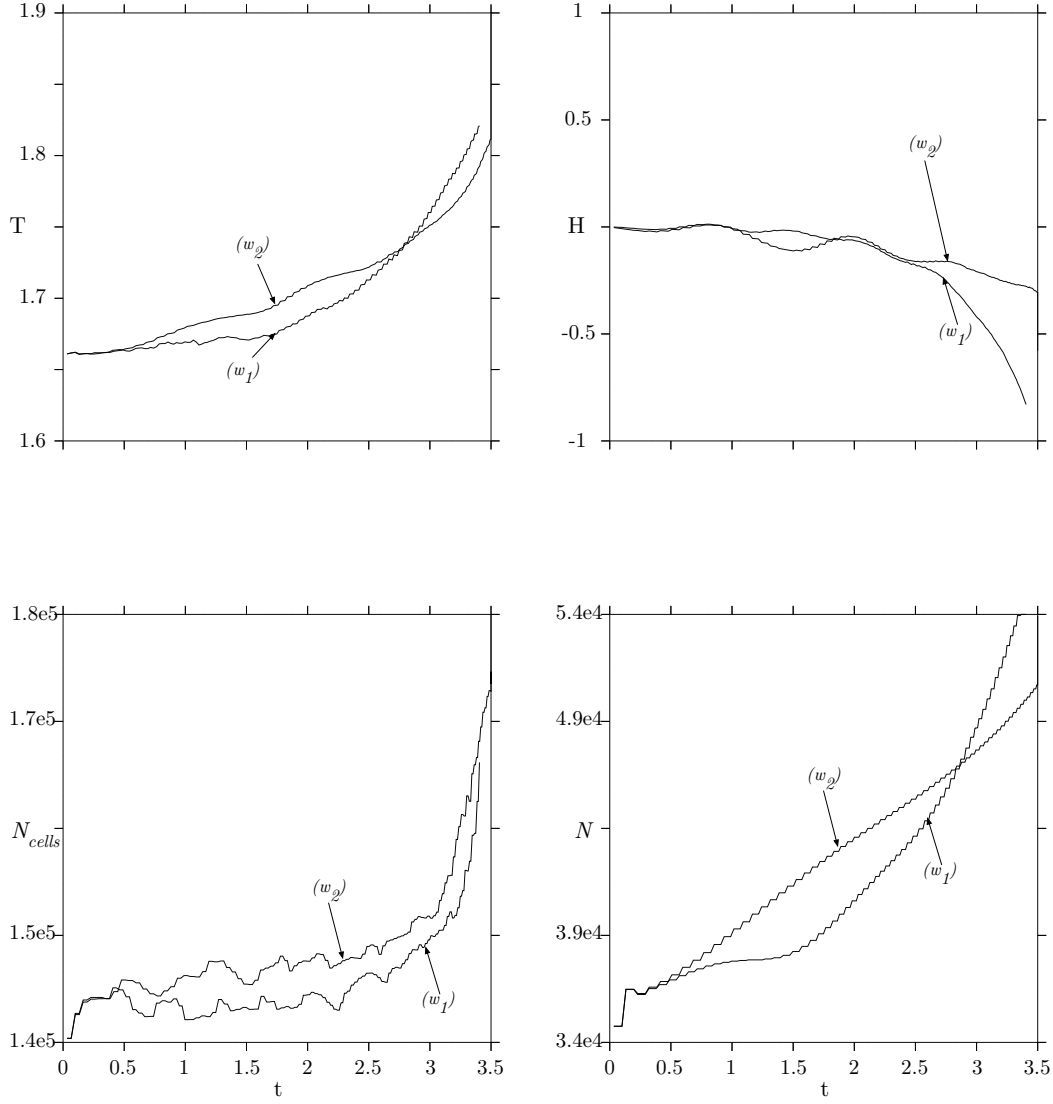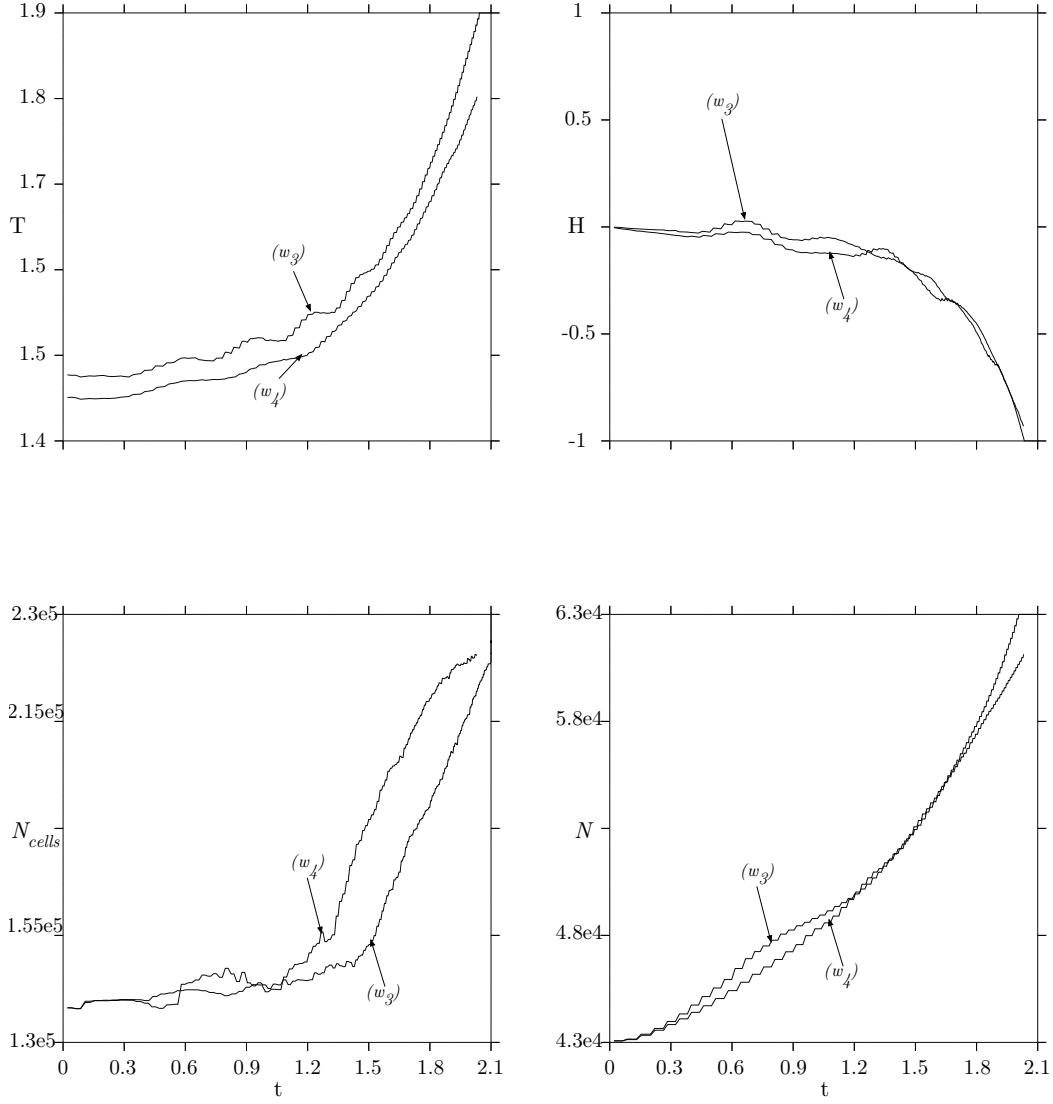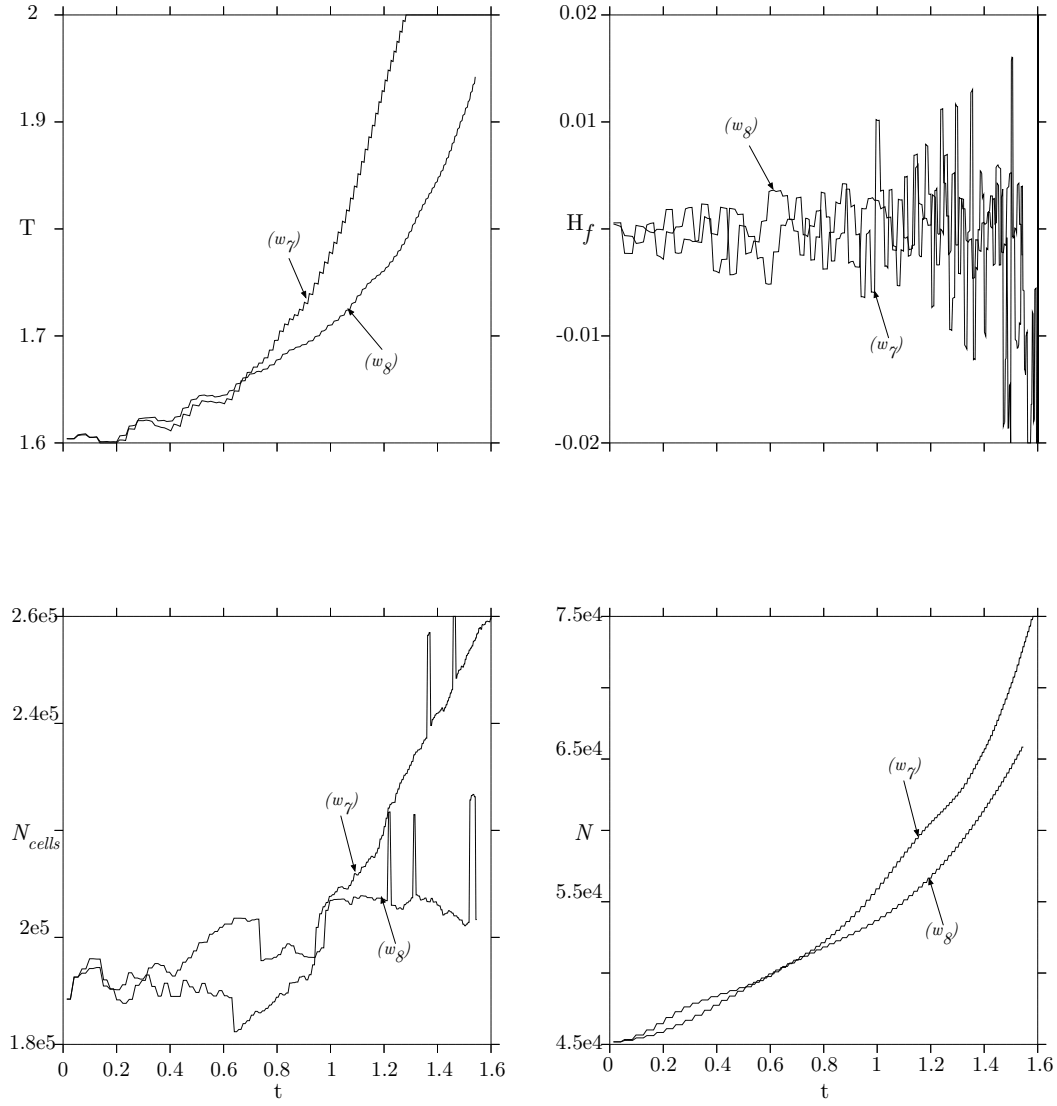
Figure 3.39: Energy, helicity, number of grid cells (N_{cells}) and number of filament nodes (N) for the Widnall instability $N_W = 19$ cases with $d = 0.0185$, $\mu = 2\pi/128$, $\gamma' = 0.7$, w_7 has $\sigma_u = 1$ and w_8 has $\sigma_u = 2$.

Making a direct comparison between the plots for energy and total filament nodes (N), the small jumps in energy are with a frequency comparable to the renode frequency. It therefore appears that renoding does not well conserve energy, despite every effort having been made for renoding to be as conservative as possible. It may be the case, however, that the introduction of the small scales made by renoding are not well resolved, and still yet more work should be done to maintain a smooth solution or increase mesh resolution to compensate. Evidence for this is perhaps available figure 3.38 where the curves for the w_6 case show shallower growth. The implication being that the smoother the solution less rapid adaptivity is required and the invariants show less growth. The variability on the grid is potentially compromising an accurate estimate of the energy and helicity encountered by the filaments. In addition, the conservation of these invariants relies on strict boundary conditions, which may not be met (additional boundary resolution may be required).

To investigate this possibility an estimate of *filament* helicity was found via

$$H_f = \sum_p^N v_p \boldsymbol{\omega}_p \cdot \boldsymbol{u}_p$$

where $\boldsymbol{u}_p$ is the nodal velocity given by the smoothed interpolation formula (3.27). This is a justifiable estimate as the filaments will represent precisely locations where vorticity is non-zero, therefore contributions to H away from the filaments will be negligible and potentially prone to boundary errors. Figure 3.37 shows H_f for the cases w_1, w_2, w_3 and w_4. These plots indicate improved conservation, backing up the conclusions regarding spurious errors upon the mesh.

It should also be mentioned that as the cells in the mesh and therefore resolution increases, we should also expect to see a change in the estimate of energy and helicity since the volume integrations are performed with greater accuracy and velocity calculations improve. This is a further instance of the inability of rigorous conclusions to be made due to the multi-adaptability of the method.
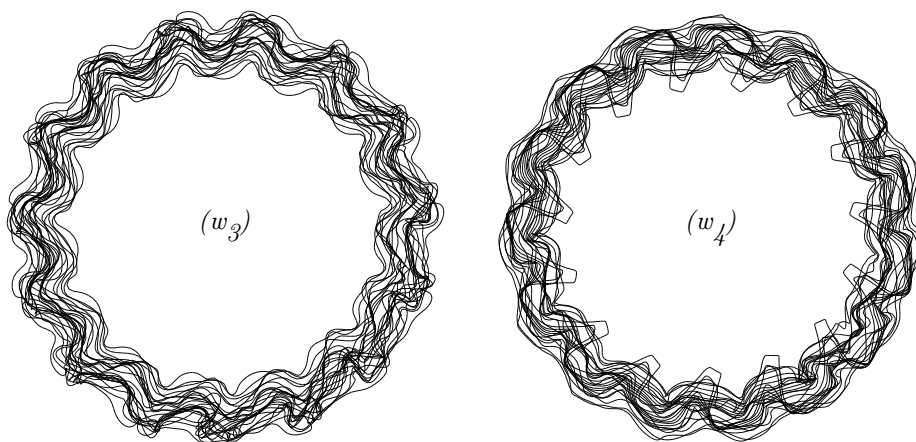
Figure 3.40: Vortex filaments at $T = 200$ ($t \approx 3.56$) for the Widnall instability $N_W = 15$ cases with $d = 0.0209$, $\mu = 2\pi/128$, $\gamma' = 0.7$, w_3 has $\sigma_u = 1$ and w_4 has $\sigma_u = 2$.
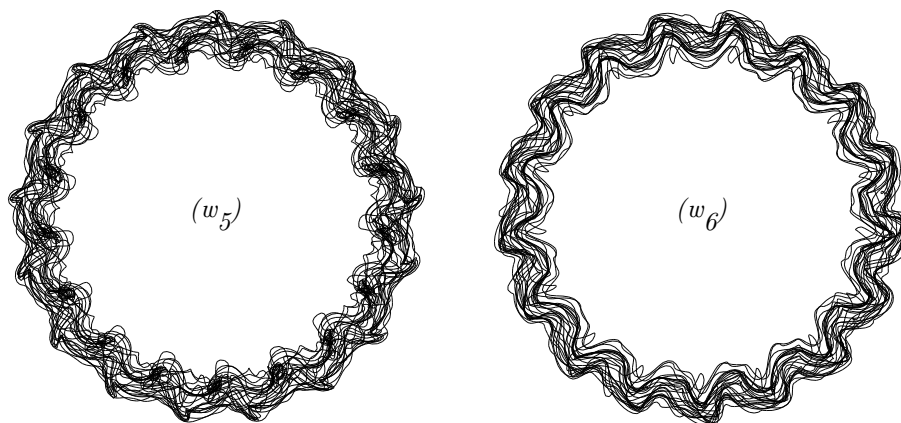

Figure 3.41: Vortex filaments at $T = 200$ ($t \approx 1.19$ for w_5 and $t \approx 1.49$ for w_6) for the Widnall instability $N_W = 17$ cases with $d = 0.0185$, $\mu = 2\pi/128$, $\gamma' = 0.7$, w_5 has $\sigma_u = 1$ and w_6 has $\sigma_u = 2$.
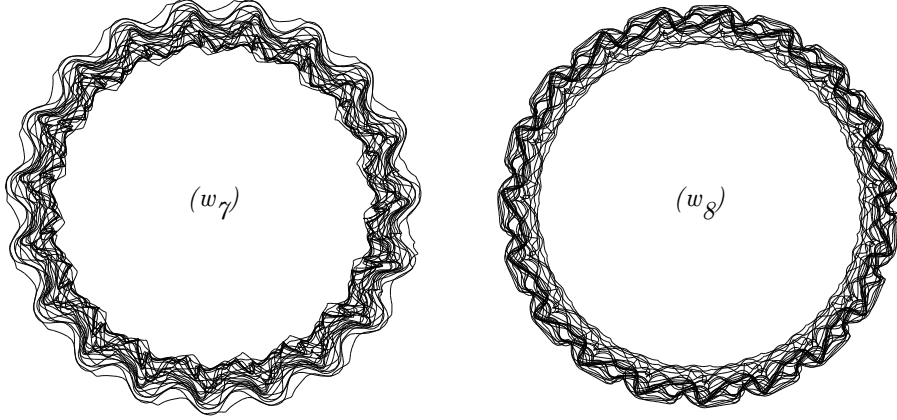
Figure 3.42: Vortex filaments at $T = 200$ ($t \approx 1.37$ for w_7 and$t \approx 1.48$ for w_8) for the Widnall instability $N_W = 19$ cases with $d = 0.0165$, $\mu = 2\pi/128$, $\gamma' = 0.7$, w_7 has $\sigma_u = 1$ and w_8 has $\sigma_u = 2$.

3.3.5   Helical equilibria

As a further validation of the method we present a number of calculations using the vortex equilibrium states found in (Lucas and Dritschel, 2009). These states are initialised by laying the starting nodes of each filament upon a planar Cartesian mesh, where the points lie within the contour computed in chapter 2. The filaments are then projected vertically with helical symmetry and unit pitch. Here α remains the core size ($\bar{R}$ in chapter 2) and R is the centroid radius (d in chapter 2). We retain V as the translation rate; however, this can equally be considered the rotation rate Ω, due to helical symmetry (see chapter 2). We employ linear velocity interpolation in these calculations with no velocity smoothing.

In order to verify the translation, or rotation rate we are able to make a direct comparison with that calculated for the equilibria in chapter 2. We estimate the rotation of the helix by computing the inverse tangent of the ratio of y-centroid

to x-centroid of axial vorticity in a constant z plane, i.e.

$$\bar{\theta} = \tan^{-1}\left(\frac{\bar{y}}{\bar{x}}\right)$$

where

$$\bar{x} = \frac{\iint_A \omega_z x \, \mathrm{d}A}{\iint_V \omega_z \, \mathrm{d}A}$$

and similarly

$$\bar{y} = \frac{\iint_A \omega_z y \, \mathrm{d}A}{\iint_A \omega_z \, \mathrm{d}A}.$$

where A is the plane $z = 0$. These integrals have to be computed on a uniform grid, with values interpolated from the adapted inversion grid, since the adaptation offsets cells of different sizes, therefore a plane is not easily found. To avoid periodicity considerations, we compute the sum of $\bar{\theta}$ increments from one time-step to the next and plot this alongside Ωt after non-dimensionalisation with Γ. Similarly with the Norbury ring calculations, non-dimensionalisation follows from the ratio of circulations:

$$\bar{V} = \frac{\bar{\Gamma} V}{\Gamma} = \frac{\pi \alpha^2 V}{\Gamma}$$

where $\bar{\Gamma} = \pi \alpha^2$ is the circulation of the equilibiria.

α	R	d	γ'	N_{fil}	N	N_{cells}	h_{min}	$\bar{V}$	Ω
0.51	0.9	0.0625	0.5	233	36916	355721	$2\pi/128$	0.0906	0.0974
2.01	2.5	0.098	0.5	574	88970	854569	$2\pi/128$	1.074	1.112
1	1	0.094	0.7	392	70625	597858	$2\pi/128$	0.2634	0.2828

Table 3.7: Parameters and diagnostics for various helical vortex equilibria with $\mu = 2\pi/128$.

Table 3.7 lists various parameters, and the non-dimensional translation speed $\bar{V}$ computed by our method and rotation rate Ω found in chapter 2, for three cases with widely varying d and α. We see fair agreement between the rotation estimates. It should be recalled, however, that the process of computing $\bar{\theta}$ is

148

subject to more error than that for the Nobury z-centroid, due to the interpolation and the lower resolution, uniform grid on which they are computed. In addition, time constraints did not permit these simulations to be carried out at particularly high resolution. An alternative for any future study would be to compute an estimate for angular velocity directly from the nodes of the filaments which would be independent of the grid.

Figures 3.43, 3.45 and 3.46 show the invariants and translation for the equilibria and figures 3.44, 3.47 and 3.48 show filaments at various times. Figure 3.49 shows the adapted mesh and vorticity magnitude for the equilibria $\alpha = 0.51$, $R = 0.1$. The method captures well the steady translation rate of these rings over a significant time integration. Invariants are similarly conserved as in the cases explored in section 3.3.1, however the simulations have not proceeded to the point where any instability has been triggered. An interesting piece of future work would be to explore the full three-dimensional stability of these equilibria.

3.4 Conclusion

This chapter has described the formulation of a new hybrid vortex method, utilising techniques from vortex-in-cell methods, vortex filament and blob methods, mesh adaptivity and space curve renoding, for modellling vortex stretching across a broad range of scales. We began by defining the vorticity field in terms of line element segments of Lagrangian space curves which define the centre lines of vortex filaments. The OpenFOAM CFD library provides finite volume method solvers to carry out the inversion problem on grids which can be dynamically refined during the calculation. In section 3.2.1 we presented details of the ellipsoidal Gaussian smoothing used in the vorticity interpolation. This procedure is analogous to providing the filaments with a Gaussian core profile and we use adaptive smoothing lengths based on the dimensions (volumes) of the segments
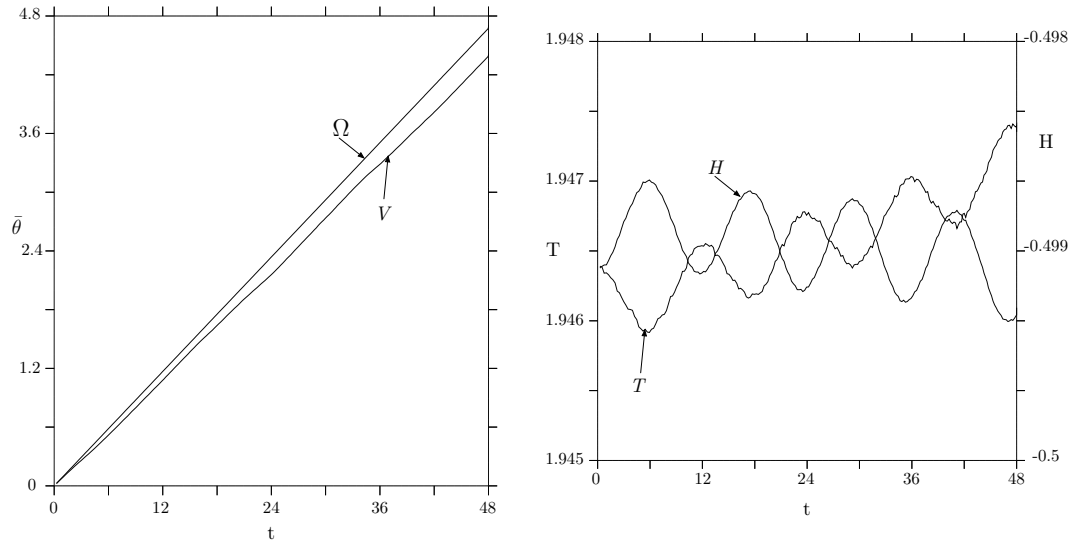
Figure 3.43: Plots showing $\bar{\theta}$ for the computed translation (V) and equilibria rotation (Ω), and Energy T and helicity H for the helical equilibria $\alpha = 0.51$, $R = 0.1$ case.
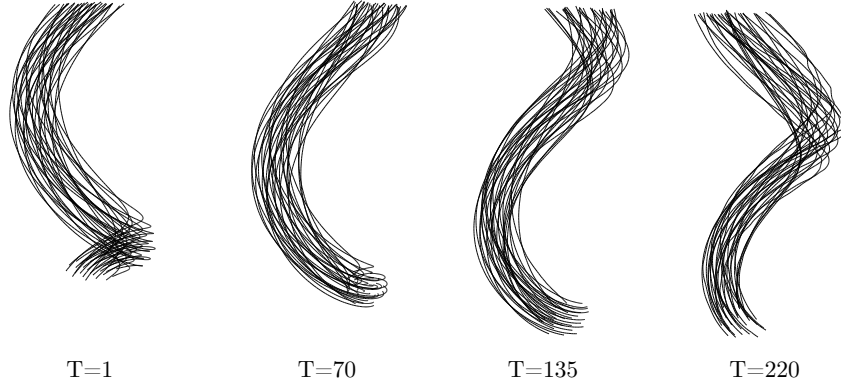
Figure 3.44: Plots showing vortex filaments for the helical equilibria $\alpha = 2.01$, $R = 2.5$ case. One eighth of filaments are rendered from a 30° latitude and a radial distance of 7. Times correspond to $t \approx 0.25, 16.4, 30.4, 46.4$.
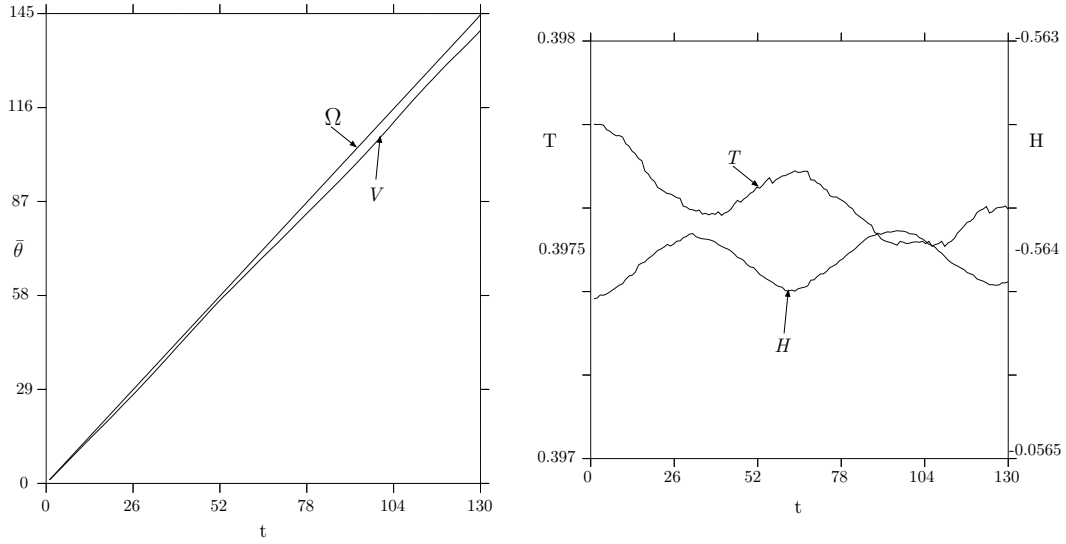
Figure 3.45: Plots showing translation V and rotation Ω, and Energy T and helicity H for the helical equilibria $\alpha = 2.01$, $R = 2.5$ case.
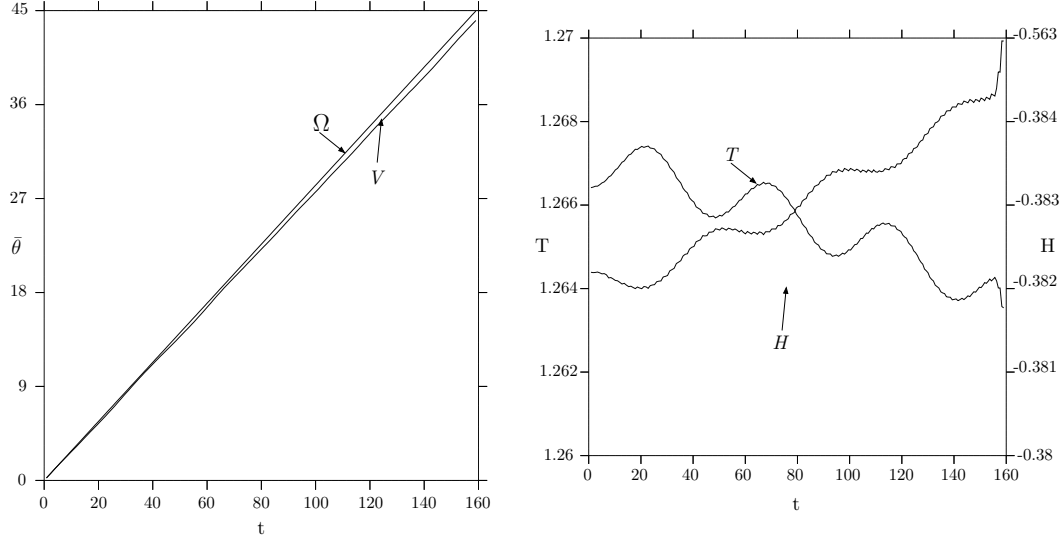
Figure 3.46: Plots showing translation V and rotation Ω, and Energy T and helicity H for the helical equilibria $\alpha = 11$, $R = 1$ case.
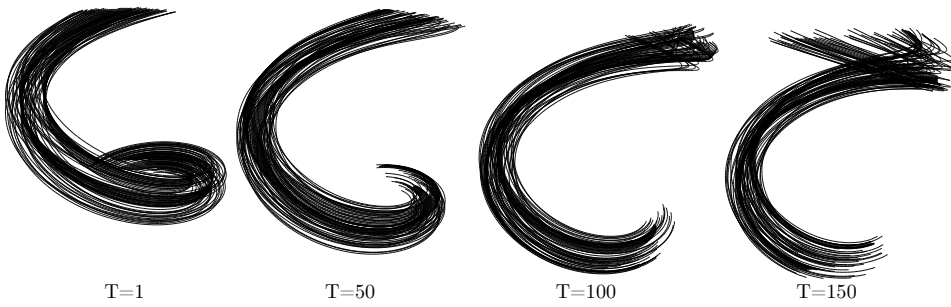
Figure 3.47: Plots showing vortex filaments for the helical equilibria $\alpha = 2.01$, $R = 2.5$ case. One eighth of filaments are rendered from a $30°$ latitude and a radial distance of 8. Times correspond to $t \approx 0.99$, 49.7, 99.3, 149.0.

152

T=1 T=50 T=100 T=150
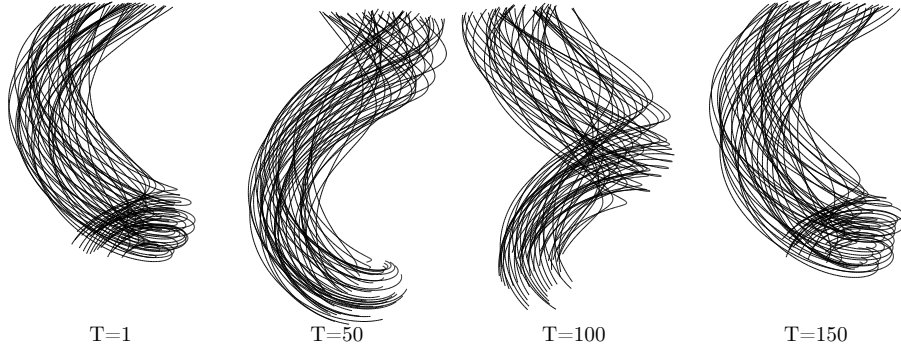
Figure 3.48: Plots showing vortex filaments for the helical equilibria $\alpha = 1$, $R = 1$ case. Times are correspond to $t \approx 0.84$, 43.2, 86.8, 130.5. One eighth of filaments are rendered from a 30° latitude and a radial distance of 7.
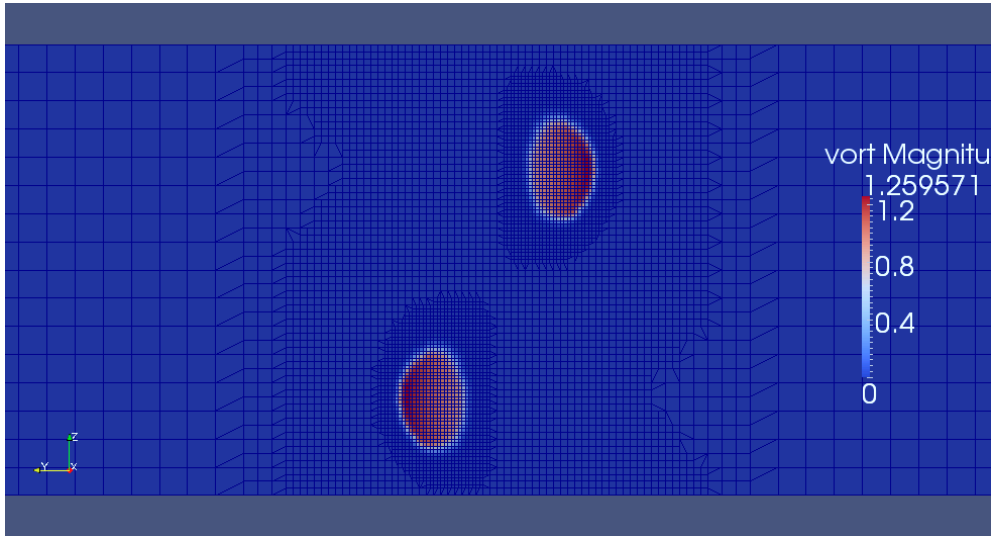
Figure 3.49: Plots showing mesh and vorticity magnitude for the helical equilibria $\alpha = 0.51$, $R = 0.1$ at $t = 2.24$.

of the space curve. To enable accurate computation of the velocity gradients generated by the stretching and folding of the vortex filaments, we adapt the grid via a refinement criteria based around the smoothing lengths which represent the typical scales of the filament segments. In section 3.2.2 a novel renoding procedure is presented with the aim of maintaining smooth adaptive filaments where a node density function is defined in terms of curvature averages and cubic interpolation defines the filament from the piecewise linear representation. This ensures filaments are well resolved when subjected to stretching and folding, and updates the scales represented in the vorticity description in a self-similar fashion. Put together, self-similarity is at the heart of this method, each component adapting to track the generation of small scales.

Validation of the method has been carried out using Norbury vortex ring equilibria, helically symmetric equilibria and Widnall's instability of the vortex ring.

Section 3.3.1 contained a thorough exploration of resolution parameters, and accuracy gains were observed when increasing filament and mesh resolution. A compromise is sought between mesh resolution, high ratios of filament to cell widths, d/h, and high filament resolution to correctly resolve the core. Accuracy in comparisons to Norbury's results were found to be sensitive to correctly resolving the vorticity core profile. A possible avenue for further study is to consider an optimisation method for the placement of filaments matching a known vorticity distribution. This could follow the method in (Dritschel and Reinaud, 2004) where the configurations of an optimal number of point vortices match the spatial moments of an ellipsoidal quasi-geostrophic vortex.

The stability and smoothness of these simulations were discussed in section 3.3.2 and a velocity smoothing was introduced in an attempt to damp mesh generated noise which is an inevitable characteristic of these calculations. Using this smoothing, section 3.3.4 investigated the instability of vortex rings to

154

azimuthal disturbances and we observed strong Widnall instability growth. We found in these cases the invariants are strongly dependent on the adaptation in the method and various explanations were given. Finally in section 3.3.5 a brief survey is made of the helical equilibria computed in chapter 2. We find good agreement with the rotation rate of the equilibria, calculated in chapter 2, and this rotation is broadly steady for the integration period we examined.

In summary the method captures the expected dynamics well. Any numerical method contains compromises and it is argued that the promising results presented display sufficient scope for developing the method further. As we have alighted upon in previous sections, the high variability in the method brought about by the various adaptive components make it unusually difficult for definitive statements to be made. However we have presented a careful study of the parameters in the method and discussed possible short-comings and solutions in detail. Finally it should be reinforced that the calculations presented display accurate results over a significant time integration for an inviscid flow with absolutely no dissipation mechanism.

Chapter 4

Future work

The incompressible Euler equations have a deceptively innocent simplicity about them; indeed their siren song has tempted many young scientists, somewhat like Ulysses, towards the twin rocks called Frustration and Despair.

After a career spent in puzzlement, the sadder but wiser researcher is forced to admit how subtle and difficult they are.

(Gibbon, 2008)

Having designed our inviscid method to adaptively track vortical structures and model vortex stretching across a wide range of scales, in a self-similar and automatic fashion, it remains to consider the open problems which vortexFOAM is suited to tackle.

4.1 Finite-time blow-up.

An important unresolved question in the field of fluid mechanics is whether or not an ideal fluid can develop a singularity in the vorticity field. It is generally agreed that indeed, given long enough, the absence of viscous diffusion will lead to a rapid build up of vorticity at small scales. However a more important question concerns the presence of singularities at some finite time, i.e. the finite-time blow-up problem.

This problem may be roughly stated as follows: if, beginning from a smooth initial condition, can, through the strongly nonlinear evolution of the flow, the solutions to the Euler equations become singular within a finite time? Such a loss of regularity is manifest at the smallest scales, where the complexity of the flow presents a formidable challenge to mathematical models. This is one of the reasons that the finite-time blow-up problem has remained open for so long.

The analogous problem for the Navier-Stokes equations is regarded as such an important challenge to the modern age of mathematics that the Clay Institute offers one million dollars for a proof for the existence of solutions (Fefferman, 2000). There is a strong argument that the question of regularity of the Euler equations is in fact the more challenging and interesting problem. As mentioned in section 1.4 the Euler equations lack even unique weak solutions (Shnirelman, 1996). A commonly held yet unproven belief is that even a small amount of viscosity will be enough to prevent blow-up in Navier-Stokes flows. However if there is evidence to suggest that inviscid flows in fact remain regular, then the problem is likely to be solved for viscous flows as well. Unfortunately there are no such proofs and the million dollar prize remains unclaimed.

In section 1.4 we noted that the cascade to small scales is principally driven by the vortex stretching term, $\boldsymbol{\omega}.\nabla\boldsymbol{u}$, from (1.4), which causes vortex lines to be stretched in the direction of maximum positive strain, thereby intensifying

vorticity into generally thin coherent structures such as vortex filaments and sheets. In the process, vortex stretching is responsible for the generation of vorticity; kinetic energy is transferred from the mean flow into the turbulent vortex. From (Beale et al., 1984) (see section 1.4) we know that the blow-up of the Euler equations is entirely controlled by the maximum of vorticity, i.e. we need not concern ourselves with growth in e.g. strain or enstrophy. It is therefore unquestionable that vortex stretching is of utmost importance in dealing with such flows and in tackling the finite-time blow-up problem. Given that our only available method of experimentation with an ideal fluid lies with numerical simulation, any method must be sure to accurately model this strong stretching of vortex filaments.

There have been a vast number of numerical studies from a variety of authors searching for convincing evidence for finite-time blow-up, particularly in the past two or three decades as computers have become increasingly capable in tackling these complex flows. These studies have been far from conclusive and have sparked fierce debate; in a recent list compiled in a review article (Gibbon, 2008) the current 'score' is 9-7 in favour of a finite-time singularity.

The efforts made to find evidence of finite time blow up in 3D Euler flows have been many and varied, with authors utilising many different numerical techniques, from the seminal calculations of (Kerr, 1993) who employed a Chebyshev polynomial method, taking advantage of symmetry in his anti-parallel vortex tube initial condition, to the high resolution pseudospectral methods employed by (Hou and Li, 2006) on the same initial condition. Studies by these two sets of authors arguably contain the more hotly contested science. They are the only studies to tackle the same initial condition, granted by two differing techniques, and to come to entirely different conclusions! While agreeing for the majority of the calculation, for late times Kerr argues evidence for singularity is seen. Hou and Li argue that in fact, resolving vortex stretching more fully, that a process

of nonlinear depletion halts the growth of vorticity. This more recent study, revisiting Kerr's work from the early 90s has been able to take advantage of the increases in computing power and some recent theoretical work.

(Frisch et al., 2003) points out that whether or not we conclude that singularities are stalled by nonlinear depletion, there is a fundamental inconsistency with simple phenomenology. We expect solutions of the Euler equations to behave as solutions to

$$\frac{Ds}{Dt} = s^2$$

which we know will blow up in time $t = 1/s(0)$ for $s(0) > 0$. Clearly the debate would not be so acute were the problem this simple; some mechanism must be stunting vorticity growth.

A crucial result by (Beale et al., 1984) furnishes us with an elegant characterisation for the regularity of the Euler equations. The Beale-Kato-Madja (BKM) theorem, in simple terms states that regularity is controlled by the infinity (maximum) norm of vorticity: solutions will exist provided that for any $T > 0$

$$\int_0^T \|\boldsymbol{\omega}(t)\|_\infty \mathrm{d}t < \infty.$$

A consequence of this remarkable theorem is of great importance in numerical experiments. Its states that blow up must behave as

$$\|\boldsymbol{\omega}(t)\|_\infty \sim (t_s - t)^p$$

with $p \geq 1$. If the data shows a power law relationship with $p < 1$ the blow-up is entirely a numerical artefact. This serves as the primary indicator for blow-up in the numerical evidence found in the literature. However a few recent conditional regularity results have found constraints on the geometry of the vorticity field for blow-up.

159

The focus of non-blow up arguments concerns the process of dynamic depletion of vortex stretching. Vortex flattening is observed at late times; vortex lines become geometrically more regular and the flow locally tends to a 2D regime, at which point vortex stretching slows. What results is a cancellation of the nonlinear advection and nonlinear stretching terms in the Euler equations and singularity formation is stopped. The inference is therefore that in locations of vorticity extrema there must be a degree of curvature in the vortex lines to enable further vortex stretching. This is the motivation for a variety of geometric regularity results (Constantin and Fefferman, 1996) (Cordoba, 2001) (Deng et al., 2005) (Deng et al., 2006).

(Constantin and Fefferman, 1996) showed that if the vorticity vector $\boldsymbol{\xi} = \boldsymbol{\omega}/|\boldsymbol{\omega}|$ remains smooth and the velocity, $\boldsymbol{u}$ bounded, no blow-up can occur. (Cordoba, 2001) demonstrate that (again with the assumption of bounded velocity) in order to achieve the vortex tube collapse associated with blow-up, the tube must undergo considerable bending and twisting such that its definition as a coherent vortex tube is under question (there exists rather a tangle of vortex filaments). Both of these results are proved for some $O(1)$ region, more localised results are obtained by (Deng et al., 2005) (Deng et al., 2006) which are potentially more amenable to use in numerical simulations where it is observed that the regions of extreme vorticity tend to shrink in time.

In their first study Deng et al. (Deng et al., 2005) presented two new results. Firstly they show that if the divergence of $\boldsymbol{\xi}$ along the vortex line segment $[s_1, s_2]$ which contains the maximum vorticity is integrable then no singularity is possible i.e.

$$\left| \int_{s_1}^{s_2} \nabla \cdot \boldsymbol{\xi} \mathrm{d}s \right| \leq C(T) \qquad 0 \leq t \leq T$$

where s is arc length and T the proposed singularity time. The implication of this theorem is that if there is blow-up at one point in this segment then vorticity

160

must diverge along the whole segment at the same rate.

The second result of (Deng et al., 2005) deals with a family of vortex lines. Suppose there exists some segment, denonted L_t, along which the maximum vorticity is comparable to the global maximum. Then if $\ell(t)$ is the arc length of the line segment, $\ell(t)||\kappa||_{\infty(\ell(t))}$ (where κ is the curvature of L_t) remains bounded, and the maximum tangential and normal velocities along the vortex line are integrable, then the length of the vortex line segment $\ell(t)$ can shrink to zero as $t \to T$ and no blow-up is possible. (See (Deng et al., 2005) for a more formal description and proof of these theorems).

These two theorems give geometric conditions on the vortex lines at near singular locations and make certain strict descriptions of vortex stretching while alluding to the possible nonlinear depletion mechanism. We focus on these results due to their striking suitability for use in the numerical method described in chapter 3. We readily have at our disposal a numerical approximation of the vortex lines of our flow and every effort has been made to ensure they are appropriately resolved; vortex stretching and folding being automatically followed.

A strategy for future research, which will exploit the novelties of the new method, will be to attempt to recreate near singular conditions, e.g. the anitparallel vortex tubes considered in (Kerr, 1993) and (Hou and Li, 2006). We can also develop diagnostics based on the conditional regularity results discussed above, directly from the computational vortex lines. In this way new insight may be afforded us via the high adaptability and hence broad range of scales represented in the method.

Furthermore a study could be performed to investigate the prevalence of near singular conditions in a general 'turbulent' inviscid fluid. The studies discussed thus far are concerned with actively searching for singularity, using very esoteric initial data. An important question might well be, how likely are we to see

161

the development of such conditions in a generic Euler flow, given that we might expect an unbounded turbulent cascade to small scales?

4.2 Method generality

Until now, the motivation and application for the method has focussed on the inviscid context and the problems surrounding the regularity of the equations. However certain aspects of the method may allow for it to be extended to tackle a variety of other less idealised problems.

Clearly a potentially rich avenue for future research already touched upon is into vortex structures, particularly at small scales. The modelling of fairly general vortex structures allowed by this method could allow for various vortex stability problems to be reinvestigated with greater precision and with an increased understanding of the dynamics of the vortex lines in the flow, as we have them readily available for diagnostics and visualisation. One example of such a study would be to investigate the full nonlinear 3D stability of the helical vortex equilibria computed in chapter 3 (Lucas and Dritschel, 2009).

Another obvious candidate for development is to include a vortex reconnection mechanism by extending the contour surgery algorithm of (Dritschel, 1988b). A version of such an algorithm and motivation for such a scheme has been presented in (Chorin, 1993). Combined with a viscosity scheme (see chapter 5 of (Cottet and Koumoutsakos, 2000)) this may open the method up to the study of various high Re Navier-Stokes problems with greater efficiency in resolving the inertial range. This may make the method attractive to workers with limited computing facilities, i.e. those without access to, for example, massively parallel clusters.

The renoding subroutine in the method could be adapted for use in a number of regimes where space curves can model field lines. One example of such an

application could be superfluids where quantised vortex lines of fixed strength and infinitesimal core structure exist. At broad scales superfluids have a striking similarity to conventional vortex dynamics, and the turbulent tangle of super-fluid vortex lines is therefore of great interest to quantum physicists and fluid dynamicists alike (Barenghi et al., 2001). There is sufficient precedence of using vortex filaments in modelling turbulent superfluids to expect a robust and efficient renoding and surgery routine to benefit numerical simulations (Barenghi et al., 1997).

4.2.1 Non-conservative forcing

Many problems where the modelling of a complex small-scale vorticity field is necessary require extensions of the method to include non-conservative effects. For example, vortex methods have a rich history in the study of wake problems where flow over an obstacle generates vorticity in it's lee (Koumoutsakos, 1995) (Ploumhans et al., 2002), (Liu, 2001), (Winckelmans et al., 2005). The generality of the OpenFOAM framework allows for arbitrary boundaries to be included; moreover the treatment of the emission of a near-boundary vorticity flux is well established (Koumoutsakos, 1995). This would mean the method could then be utilised for any problem involving flow over a solid body since the inclusion of vorticity generated by the no-slip condition would be theoretically possible.

Another field where additional forces impinge on the vorticity evolution is in magnetohydrodynamics (MHD). Here the Lorentz force arising in an electrically-conducting fluid creates or destroys vorticity. This force resists vortex motion, due to the tension in magnetic field lines. In the context of modelling the solar corona, for example, it is conjectured that the process of magnetic reconnection is responsible for heating the solar atmosphere to temperatures much higher than the solar surface (see (Zirker, 1993) and references therein). In addition, the

ejection of plasma from the solar atmosphere in the form of solar flares and coronal mass ejections (Zhang, 2005) can travel to the Earth and disturb the Earth's own magnetic field, disrupting communication satellites, electricity networks and even endangering astronauts in orbit. The generality of OpenFOAM means that the additional MHD equations are readily solvable, provided there is an appropriate scheme for the treatment of vorticity.

To achieve this additional generality, and enable the inclusion of such effects, a scheme has partially been designed to enable vorticity increments to be assigned to the Lagrangian line elements. This does, however, require a relaxation in the connectivity of the vortex line segments. Instead, vorticity will be defined by individual, unconnected, line elements which can be thought of in a particle context as overlapping blobs of vorticity. An algorithm has already been constructed which will apply adjustments to these vectors and iterate to match the Lagrangian and Eulerian vorticity descriptions. A consequence of the lack of connectivity in the vorticity field means that the topological constraint, $\nabla \cdot \boldsymbol{\omega} = 0$ is not automatically satisfied. However, the scheme described can be employed to perform corrections to maintain zero divergence.

In this way any additional sources or sinks of vorticity arising from boundary conditions or general forcing may be readily included.

4.3   Algorithm Development

The possibility for efficiency improvements are extensive. Now that the basic method has been validated, certain benchmarking can be performed to investigate the efficiency of the inversion portion of the method. It may be advantageous to invest some time considering alternatives to the current Poisson solver to establish if efficiency and accuracy may be improved.

The problem of maintaining incompressibility in vortexFOAM requires to be rigorously addressed before the code can be stress tested with blow-up type initial data. As has been discussed in section 3.3.2 a number of options have been explored, none of which have seen great improvements made. Some time should be dedicated to developing an efficient velocity interpolation which ensures the nodal velocities are divergence free. In addition the velocity interpolation is open to some computational saving by making some use of the particle tracking classes in openFOAM to avoid lengthy mesh searches.

Perhaps the most obvious efficiency boost, and the one to which priority should be given, is to parallelise the code. As the method stands, the mesh and solver have domain decomposition options built in via OpenFOAM. However the Lagrangian component requires this functionality to be added and dovetailed into the OpenFOAM parallelisation classes. In principle this should be straightforward as we simply need a processor designation for each node on the filament corresponding to the section of the domain the node lies within. This can then be tracked and nodes which translate between processor domains updated. The main difficulty will lie in efficiently distributing the computational burden across the processors, with problems involving little symmetry being particularly challenging.

4.4   Conclusions

In this thesis we have focussed on vortex structures in three-dimensions, sought their equilibria and developed a novel numerical method with which to compute their evolution.

In chapter 1 we provided the motivation for this work, placing emphasis on robust numerical approximations in making advances in our understanding of complex flows. Turbulence and the regularity of the equations are two rich areas

of research in this regard. The ubiquitous nature of vorticity in turbulence and the regularity of the inviscid equations for bounded vorticity are intrinsically linked via the process of vortex stretching, whereby vorticity is intensified by the stretching of vortex tubes. The self-similar modelling of this process, as the scales of motion cascade to increasingly smaller scales, is the crux of the numerical approach presented in chapter 3.

This hybrid vortex method models the inviscid (Euler) equations via a Lagrangian description of vorticity in terms of space curves representing the core of smooth vortex filaments which are material lines due to Kelvins circulation theorem. An underlying grid is used to perform an inversion to determine velocities, and is subjected to adaptive refinement (using the OpenFOAM CFD library) to concentrate numerical calculations in regions of intense stretching. In addition the filaments are constantly adjusted via a novel renoding algorithm, whereby additional nodes are included along the filaments in locations of high curvature. The algorithm has been designed such that this distribution of numerical effort is completely automatic, the scales of motion being represented in a completely self-similar way and limited purely by the capability of the machine.

The method has been validated using well known vortex structures. First vortex rings were considered. Known equilibria exist in the form of Norbury's rings (Norbury, 1973) and the method was validated using a number of these rings. We find the method well represents the steady translation of these rings and conservation of the invariants, energy and helicity was found to be good. These results showed the expected improvement on increasing the Lagrangian and Eulerian resolution. In addition vortex ring instability was studied via the Widnall instability, whereby vortex rings are perturbed sinusoidally in with azimuthal wavenumber inversely proportional to core radius. These waves were observed to grow as expected and secondary instability was observed, particularly in cases with low velocity smoothing. Finally vortex equilibria possessing

helical symmetry, presented in chapter 2, are used to verify the method. Good agreement with translation rates and invariant conservation is again found.

These helical equilibria presented in chapter 2 are computed via a numerical method, where a restriction on the streamfunction on contours bounding uniform axial vorticity is sufficient to define the equilibria due to the conservation of axial vorticity when vortex lines are helical. States are parameterised by mean radius (or area) and centroid location, and the method is extended to include multiple vortices evenly azimuthally spaced about the z-axis. The equilibria are computed in a rotating frame, the rotation of the states being given by their own self-induced rotation. This rotation rate is computed as part of the problem and presents itself as an ideal validation tool for new three-dimensional, inviscid numerical methods.

By way of summary, we have found that the accumulation of small scale, grid generated noise is inevitable in the VortexFOAM method. It can be mitigated with smoothing in the velocity interpolation and there is potential in future research to address the problem with some correction procedure. The method highlights the advantages of a Lagrangian description of vorticity and an adaptive mesh in resolving vortex stretching in 3D vortex dynamics in a consistent and self-similar way. The level of accuracy gained on a single processor for a three-dimensional, inviscid calculation is remarkably high given the CPU times, making this method attractive for future development and application.

Appendix A: OpenFOAM

This appendix aims to give a brief outline of the OpenFOAM (FOAM standing for "Field Operation And Manipulation") suite of codes, with an emphasis on their use in the vortexFOAM method. The OpenFOAM library and documentation is distributed free by OpenCFD under the GNU General Public License, and as such much of what follows is reproduced from the user guide and online content at `http://www.openfoam.com` (2011). A further intention of this section is to allow enough of a working knowledge of the code to allow a user to perform their own calculations using vortexFOAM.

OpenFOAM is an object oriented c++ library, focussed on the simulation of continuum mechanics problems. Although not restricted to this, the primary application is to fluid mechanics and solvers are available to study a wide variety of fluid problems. We first describe the discretisation and solvers employed before describing the practicalities of using the codes.

A.1   OpenFOAM discretisation

OpenFOAM makes use of the finite volume method (FVM) to discretise the solution space. This has a variety of advantages for CFD applications, not least that it allows great generality in mesh structures. We will first give a very brief synopsis of the FVM.

Scalars, vectors and even tensors are handled consistently through the object oriented structure used in OpenFOAM and the standard operations (multiplications, additions, vector and scalar products etc.) are all available as appropriate.

The FVM considers a domain discretised as a union of small non-overlapping control volumes with 'nodes' at cell centres. The PDE is then transformed by integrating over control volumes and converting divergence terms into surface

fluxes and gradient terms (including curl) into surface integrals using the divergence theorem. The matching of fluxes across the surfaces of adjacent control volumes makes the method conservative. For example given the poisson equation

$$\Delta \boldsymbol{\psi} = -\omega$$

then

$$\int_{cv} \Delta \boldsymbol{\psi} \mathrm{d}V = -\int_{cv} \boldsymbol{\omega} \mathrm{d}V \Rightarrow \int_{sf} \nabla \boldsymbol{\psi} . \boldsymbol{n}_{sf} \mathrm{d}S = -\int_{cv} \boldsymbol{\omega} \mathrm{d}V$$

Surface integrals can thus be computed as a summation and gradient terms are approximated via finite differences across the face. This can be complicated by the non-orthogonality of the mesh, i.e. where surface normals, $\boldsymbol{n}_{sf}$, and cell centres are not aligned and additional interpolation is required. These corrections can, however, be iterated over within the solve to reduce the error.

Let us now consider the approximation of $\boldsymbol{u} = \nabla \times \boldsymbol{\psi}$. The standard method in OpenFOAM is to again use Gauss integration to compute $\boldsymbol{u}$, i.e.

$$\int_{cv} \boldsymbol{u} \mathrm{d}V = \int_{cv} \nabla \times \boldsymbol{\psi} \mathrm{d}V = \int_{sf} \boldsymbol{n}_{sf} \times \boldsymbol{\psi} \mathrm{d}S$$

so for a given cell, i,

$$\boldsymbol{u}_i = \frac{1}{V_i} \sum_{faces} \boldsymbol{n}_{sf} \times \boldsymbol{\psi}_f.$$

Therefore it is necessary to transfer values of $\boldsymbol{\psi}$ from the cell centre locations to the cell face centres. The accuracy of this interpolation is related to the skewness and relative sizes of the adjacent cell centres (skewness is the degree to which the cell centre vector is from intersecting the cell face centre), which in turn impacts upon the overall accuracy of the velocity $\boldsymbol{u}$. This is the reason why we find divergence in refinement patterns, as described in section 3.3.2. Using linear interpolation, as in the simulations presented, results in a reduction of the solution to first order accurate near the refinement patterns. Choosing either a least squares scheme (instead of Gauss integration) or *skewCorrected*

interpolation increases the accuracy (*skewCorrected* includes a correction which is a projection of the surface normal gradient in the skewed direction between $\boldsymbol{n}_{sf}$ and the cell centre vector). As has already been discussed this inaccuracy is an outstanding difficulty requiring additional future work.

The application of the boundary conditions is remarkably straight forward, given that our equation has been transformed into an expression involving surface quantities. When such a face of a cell is a boundary face the prescribed boundary condition can be applied, and OpenFOAM allows a variety of conditions to be specified. For the purposes of our algorithm, we desire free-slip conditions upon the x and y boundaries and periodic, or *cyclic*, condition on the z boundary. In terms of the discretisation this simply requires the vector potential $\boldsymbol{\psi} = 0$.

The result is a system of equations which, as with many other numerical methods, can be expressed in terms of a matrix of coefficients along with vectors of solution variables and source terms i.e.

$$A\boldsymbol{u} = \boldsymbol{b}$$

where the source term $\boldsymbol{b}$ contains the right hand side of the poisson equation plus boundary data. Iterative schemes for the solution of this system will be discussed in section A.3.

A.2 The mesh and refinement

The physical partitioning of the domain into control volumes is performed by a mesh generator. There are several ways to achieve this, including many third party programs, however openFOAM comes with its own mesh generation routine *blockMesh*. The FVM allows for a great generality in its mesh, which is one of the reasons for its wide use in the field of CFD. Although our domain is very simple, OpenFOAM can handle complex geometries and a wide variety of

cell shapes through its 'arbitrary unstructured' mesh paradigm.

blockMesh begins with a user defined decomposition of the domain with 1 or more hexahedral blocks and the mesh is then generated with a specified number of cells per block. Edges are not limited to being lines but can be curves, defined either piecewise or as splines. For our case of a cuboidal domain, we specify a single block, with the eight vertices on each corner of the domain. Boundaries must also be specified to *blockMesh* via 'patches', again given as a vertex list, with the periodic boundary as a 'cyclic' of two patches. An initial mesh $48 \times 48 \times 16$ is generated (this was found to be optimum to allow the boundary conditions to be accurately discretised) before a preprocessing utility *setVfMesh* is called which performs an initial refinement of the mesh, given the vortex filaments for the simulation considered.

The mesh is defined as a *dynamicFvMesh* type, which allows for changes to the mesh during runtime. The mesh is adjusted according to a scalar field defined on the mesh, `gamma`, as discussed in section 3.2.3.

Due to the interpolations performed by the OpenFOAM classes it was discovered necessary to flag cells satisfying the refinement criteria ($\gamma_i > \gamma'$, where γ' is a specified constant) with a large value to force refinement. The scalar field `gamma`=10 is set for cells where $\gamma_i > \gamma'$. If $\gamma_i < \gamma'/10$ the mesh is allowed to unrefine and `gamma`=0. All other cells have `gamma`=0.5 and the thresholds are detailed in section A.3.

The preprocessing performed by *setVfMesh* allows sufficient resolution locally to the vortex filaments such that the runtime mesh refinement laid out in section 3.2.3 can perform satisfactorily. *setVfMesh* uses *dynamicFvMesh* updating similar to during runtime but instead refines a cylindrical zone $r < \pi$.

171

A.3   Solving

Most of the code for what is described above is several layers of inherited class structures below the user. The necessary classes and thus functions are included via header files in the main source code. While this makes for a neat and tidy code, which is very efficient to adapt to specific uses, it can be a challenge for the uninitiated to follow the inner workings of the library. Understanding the nature of the method and the available options and schemes is essential in utilising this powerful code to best advantage.

The equation and variable discretisation classes make the solve line in the code very straight forward:

```
solve (fvm::laplacian(psi)+vort);
U = fvc::curl(vort)
```

where `fvm` are the implicit terms in the discretisation and `fvc` the explicit. The velocity and vorticity fields (`U` and `vort`) are defined as `volVectorField` type, implying a volume centred vector field, as required to form the discrete equations.

The control of this solve is not set until runtime, therefore applications can be compiled with a certain generality in this regard. For any given run of an OpenFOAM application a directory is created containing certain essential subdirectories and 'dictionary' files.

The first directory, `constant`, contains the primary mesh data and initial conditions for the run. Contained within is the `polyMesh/` directory holding the mesh data generated by *blockMesh* via the `blockMeshDict` dictionary. This file is common for all runs and simply outlines the 36864 cell uniform global mesh before it is updated by *setVfMesh*. The `constant` directory contains the `dynamicMeshDict` dictionary file which controls the mesh refinement parameters for the run. This dictionary is necessary for both the preprocessing of

172

setVfMesh and for the vortexFOAM run itself. The dictionary contains various parameters including the thresholds on `gamma` for refinement/unrefinement (`lowerRefineLevel=1`, `upperRefineLevel=10000`, `unrefineLevel= 0.1`), the maximum cell limit (50000000 cells), the maximum number of refinement levels (20) and the frequency of mesh updating (every 3 steps).

`constant` must also contain initial conditions for any global variables defined on the mesh which require outputting, e.g. `U` and `vort` are always of this type. The second directory recognises the zero time initial condition of the mesh and solver variables, denoted by `0/`, containing a copy of `polyMesh/` and `U` and `vort`.

The final directory is `system/` and contains a number of dictionary files which are read by the application and prescribe the options for solvers, interpolations etc. The main control parameters are set in the `controlDict` file. These include time parameters (time step, run length, time precision) and output parameters (write format, frequency, precision) for writing meshed variables to file. The output is contained within a series of subdirectories named by the current time and containing the same data format as `0/`.

The other two dictionary files in the `system/` directory are `fvSolution` and `fvSchemes` which prescribe the linear solver options and discretisation options respectively. `fvSolution` must contain within it the appropriate solver choice for every variable the application solves for. For example in the case of vortex-FOAM this includes `U` and also `gamma`, which is used to correct for the solenoidal constraint before writing vorticity.

There are a substantial number of options for linear solvers in openFOAM, the main three being preconditioned conjugate gradient, smoothing solvers and a generalised geometric-algebraic multigrid. Each of these has its own set of further options associated with it; conjugate gradients require a preconditioner, smooth solvers (e.g. Gauss-Seidel) must be set, and the multi-grid method requires var-

ious parameters dictating coarsening, cycles etc. Details of these options can be found in section 4.5 of the user guide. In addition to choosing the solver type, tolerances are also required to control convergence of the iterative scheme. `tolerance` denotes a general threshold upon the value of the residual for halting the solver, and `relTol` is a relative tolerance from one iteration to the next.

For vortexFOAM we generally employ the default settings. No substantial gains in accuracy or speed were observed with changing the solver preferences. We employ preconditioned conjugate gradient iteration with a diagonal incomplete-Cholesky preconditioner. `tolerance` is set at 10^{-10}, and `relTol` to 0.01.

The options for the numerical schemes employed in the finite volume method are contained in the `fvSchemes` file. There are various options for the treatment of the different terms in the equations of interest and in handling the necessary interpolations the method requires. The default for derivative terms is to employ standard Gaussian finite volume integrations which requires surface information on the control volumes. An associated interpolation is needed to transfer data from the cell centres. The default for the interpolations throughout is linear, although the options in OpenFOAM are extensive (see section 4.4.1 in the user guide for details). For the purposes of the Poisson solver in vortexFOAM it was found that anything more elaborate was not efficient enough to justify an improvement in the algorithm. The choices were explored in regard to alleviating the mesh non-orthogonality/uniformity issue described in 3.3.2; however, other choices were found to produce negligible differences.

The final pieces of data required for a vortexFOAM run are the Lagrangian vortex elements and non-OpenFOAM parameters. Various initialisation routines are available to construct filaments. All will output an `elements.dat` file which includes a list of 3D points which represent the space curve along with an integer label denoting the subsequent node along the filament. The `params.dat` file includes parameters for time stepping, i.e. the RK4 time step, output frequency

174

and total run duration, and parameters relating to the filaments, i.e. vorticity scaling c_p (initially constant over all segments/filaments), transverse smoothing length σ_p (filament width), renode parameter μ, refinement parameter γ' and the total number of filaments N_{fil}.

In summary, to perform a run using vortexFOAM a `<case>` directory is required containing the following file structure:

- params.dat
- elements.dat
- constant/
    - polyMesh/
    - dynamicMeshDict
    - vort
    - U
- system/
    - controlDict
    - fvSchemes
    - fvSolution
- 0/
    - polyMesh/
    - vort
    - U

Once inside this directory, with OpenFOAM installed and vortexFOAM compiled the job can be started by simply calling the program:

```
> vortexFOAM
```

Data will be output into a number of files. The file `vect.dat` contains the Lagrangian vortex elements, output at the interval determined in `params.dat` with the same format as `elements.dat`. This data can be visualised with the gplot programs *vecprint* and *vecdisp*. Diagnostics are output into the files `output_time.dat` and `diagnostic.dat`, the first containing the number of cells in the mesh, the total number of Lagrangian nodes, a diagnostic for the vorticity interpolation accuracy, and the magnitudes of the divergences in velocity and vorticity. `diagnostic.dat` contains energy, helicity, and axial centroid location for vortex ring propagation.

Eulerian data is output in the OpenFOAM format into directories named by time with the same format as `0/`. The preferred method to visualise this data is to use the *ParaView* visualisation application which comes packaged with OpenFOAM along with various plugin/interface script options to enable users to visualise their data.

In due course all the necessary codes and files will be packaged and made available with the necessary documentation to allow any OpenFOAM user to perform calculations with vortexFOAM.

Bibliography

AFP/Getty Images. Severe weather warning as hurricane katia tail approaches. `http://www.guardian.co.uk/uk/2011/sep/11/severe-weather-warning-hurricane-katia`, September 2011.

Jürg Alean. Etna steam ring. `http://www.swisseduc.ch/stromboli/etna/etna00/etna0002photovideo-en.html?id=2`, September 2011.

SV Alekseenko, PA Kuibin, VL Okulov, and SI Shtork. Helical vortices in swirl flow. *Journal of Fluid Mechanics*, 1999.

AS Almgren, T Buttke, and P Colella. A fast adaptive vortex method in three dimensions. *Journal of Computational Physics*, 1994.

Wilde Analysis. Polyhedral mesh and pressure distribution on an f1 car post-processed using ansys cfd-post software. `http://wildeanalysis.co.uk/cfd/software/ansys/fluid-dynamics/fluent`, September 2011.

CR Anderson. A method of local corrections for computing the velocity field due to a distribution of vortex blobs. *Journal of Computational Physics*, 1986.

PJ Archer, TG Thomas, and GN Coleman. Direct numerical simulation of vortex ring evolution from the laminar to the early turbulent regime. *Journal of Fluid Mechanics*, 2008.

LA Barba, A Leonard, and CB Allen. Advances in viscous vortex methods-meshless spatial adaption based on radial basis function interpolation. *International Journal for Numerical Methods in Fluids*, 2005.

C F Barenghi, R J Donnelly, and W F Vinen. *Quantized Vortex Dynamics and Superfluid Turbulence*. Springer, 2001.

CF Barenghi, DC Samuels, and GH Bauer. Superfluid vortex lines in a model of turbulent flow. *Physics of Fluids*, 1997.

George Keith Batchelor. *An introduction to fluid dynamics*. Cambridge University Press, 1967.

J Beale, T Kato, and A Majda. Remarks on the Breakdown of Smooth Solutions for the 3-D Euler Equations. *Commun. Math. Phys*, 1984.

JT Beale and A Majda. Vortex methods. I: Convergence in three dimensions. *Mathematics of Computation*, 1982.

William L Briggs, Van Emden Henson, and Stephen Fahrney McCormick. *A multigrid tutorial*. Society for Industrial Mathematics, 2000.

AJ Chorin. Scaling laws in the vortex lattice model of turbulence. *Communications in Mathematical Physics*, 1988.

Alexandre Chorin. Hairpin removal in vortex interactions II. *Journal of Computational Physics*, 1993.

Alexandre Joel Chorin. *Vorticity and turbulence*. Springer, 1994.

I Christiansen. Numerical simulation of hydrodynamics by the method of point vortices. *Journal of Computational Physics*, 1973.

P Constantin and C Fefferman. Geometric constraints on potentially singular solutions for the 3-D Euler equations. *Communications in Partial Differential Equations*, 1996.

D Cordoba. On the Collapse of Tubes Carried by 3D Incompressible Flows. *Communications in Mathematical Physics*, 2001.

G-H Cottet and P D Koumoutsakos. *Vortex Methods: Theory and Practice.* Cambridge University Press, 2000.

GH Cottet, MLO Salihi, and M El Hamraoui. Multi-purpose regridding in vortex methods. *ESAIM Proceedings*, 1999.

GH Cottet, P Koumoutsakos, and MLO Salihi. Vortex methods with spatially varying cores. *Journal of Computational Physics*, 2000.

Peter Alan Davidson. *Turbulence.* Oxford University Press, USA, 2004.

Jian Deng, Thomas Y Hou, and Xinwei Yu. Geometric properties and nonblowup of 3D incompressible Euler flow. *Communications in Partial Differential Equations*, 2005.

Jian Deng, Thomas Hou, and Xinwei Yu. Improved Geometric Conditions for Non-Blowup of the 3D Incompressible Euler Equation. *Communications in Partial Differential Equations*, 2006.

J Douglas Jr, RB Kellogg, and RS Varga. Alternating direction iteration methods for N space variables. *Mathematics of Computation*, 1963.

DG Dritschel. The stability and energetics of corotating uniform vortices. *Journal of Fluid Mechanics*, 1985.

DG. Dritschel. Contour dynamics/surgery on the sphere. *J. Comput. Phys.*, 79: 477–483, December 1988a.

DG Dritschel. Contour surgery: a topological reconnection scheme for extended integrations using contour dynamics. *Journal of Computational Physics*, 1988b.

DG Dritschel. Generalized helical Beltrami flows in hydrodynamics and magnetohydrodynamics. *Journal of Fluid Mechanics*, 1991.

DG Dritschel. A general theory for two-dimensional vortex interactions. *Journal of Fluid Mechanics*, 1995.

DG Dritschel and MHP Ambaum. A contour-advective semi-Lagrangian numerical algorithm for simulating fine-scale conservative dynamical fields. *Quarterly Journal of the Royal Meteorological Society*, 1997.

DG Dritschel and JN Reinaud. The quasi-geostrophic ellipsoidal vortex model. *Journal of Fluid Mechanics*, 2004.

DG Dritschel, LM Polvani, and AR Mohebalhojeh. The contour-advective semi-Lagrangian algorithm for the shallow water equations. *Monthly Weather Review*, 1999.

CL Fefferman. Existence and smoothness of the Navier-Stokes equation. *The millennium prize problems*, 2000.

J Fontane and DG Dritschel. The HyperCASL algorithm: A new approach to the numerical simulation of geophysical flows. *Journal of Computational Physics*, 2009.

U Frisch. *Turbulence*. Cambridge University Press, 1995.

U Frisch, T Matsumoto, and J Bec. Singularities of Euler flow? Not out of the blue! *Journal of Statistical Physics*, 2003.

Y Fukumoto and V L Okulov. The velocity field induced by a helical vortex tube. *Physics of Fluids*, 2005.

J Gibbon. The three-dimensional Euler equations: Where do we stand? *Physica D: Nonlinear Phenomena*, 2008.

C Greengard. Convergence of the vortex filament method. *Mathematics of Computation*, 1986.

H. Helmholtz. Über Integrale der hydrodynamischen Gleichungen, welche den Wirbelbewegungen entsprechen. *Journal für die reine und angewandte Mathematik (Crelles Journal)*, 1858.

RD Henderson. Unstructured spectral element methods for simulation of turbulent flows. *Journal of Computational Physics*, 1995.

MJM Hill. On a Spherical Vortex. *Proceedings of the Royal Society of London*, 1894.

T Hou and R Li. Dynamic depletion of vortex stretching and non-blowup of the 3-D incompressible Euler Equations. *Journal of Nonlinear Science*, 2006.

TY Hou. Convergence of a variable blob vortex method for the Euler and Navier-Stokes equations. *SIAM Journal on Numerical Analysis*, 1990.

MY Hussaini. Spectral methods in fluid dynamics. *Annual Review of Fluid Mechanics*, 1987.

Lord Kelvin. Vibrations of a Columnar Vortex. *Phil. Mag.*, 1880.

RM Kerr. Evidence for a singularity of the threedimensional, incompressible Euler equations. *Physics of Fluids A Fluid Dynamics*, 1993.

G Kirchhoff. *Vorlesungen über mathematische Physik.* 1876.

A N Kolmogorov. The local structure of turbulence in incompressible viscous fluid for very large Reynold's numbers. *C. R. Acad. Sci. USSR*, 1941.

P Koumoutsakos. High-resolution simulations of the flow around an impulsively started cylinder using vortex methods. *Journal of Fluid Mechanics*, 1995.

PA Kuibin and VL Okulov. Self-induced motion and asymptotic expansion of the velocity field in the vicinity of a helical vortex filament. *Physics of Fluids*, 1998.

Horace Lamb Sir. *Hydrodynamics*. Cambridge Univ Press, 1993.

MJ Landman. On the generation of helical waves in circular pipe flow. *Physics of Fluids A Fluid Dynamics*, 1990.

A Leonard. Vortex methods for flow simulation. *Journal of Computational Physics*, 1980.

A Leonard. Computing three-dimensional incompressible flows with vortex elements. *Annual Review of Fluid Mechanics*, 1985.

R Leung, S Idelsohn, and E Ońate. A Study Of Inviscid Vortex Ring Core Instability Using Contour Dynamics. *Computational Mechanics: New Trends and Applications*, 2007.

CH Liu. A three-dimensional vortex particle-in-cell method for vortex motions in the vicinity of a wall. *International Journal for Numerical Methods in Fluids*, 2001.

T Loiseleux, JM Chomaz, and P Huerre. The effect of swirl on jets and wakes: Linear instability of the Rankine vortex with axial flow. *Physics of Fluids*, 1998.

D Lucas and D Dritschel. A family of helically symmetric vortex equilibria. *Journal of Fluid Mechanics*, 2009.

C Macaskill and W Padden. The CASL algorithm for quasi-geostrophic flow in a cylinder. *Journal of Computational Physics*, 2003.

Andrew Majda and Andrea L Bertozzi. *Vorticity and incompressible flow.* Cambridge University Press, 2001.

HK Moffat and S Kida. Stretched vortices-the sinews of turbulence; large-Reynolds-number asymptotics. *Journal of Fluid Mechanics*, 1994.

HK Moffatt. Some developments in the theory of turbulence. *Journal of Fluid Mechanics*, 1981.

JJ Monaghan. Extrapolating B splines for interpolation. *Journal of Computational Physics*, 1985.

DW Moore and PG Saffman. The motion of a vortex filament with axial flow. *Philosophical Transactions for the Royal Society of London. Series A, Mathematical and Physical Sciences*, 1972.

DW Moore and PG Saffman. A note on the stability of a vortex ring of small cross-section. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 1974.

J Norbury. A family of steady vortex rings. *Journal of Fluid Mechanics*, 1973.

JJ O'Connor and EF Robertson. Claude Louis Marie Henri Navier. `http://www-history.mcs.st-andrews.ac.uk/Biographies/Navier.html`, September 2011a.

JJ O'Connor and EF Robertson. Lewis Fry Richardson. `http://www-history.mcs.st-andrews.ac.uk/Biographies/Richardson.html`, September 2011b.

V Okulov and J Sørensen. Optimum operating regimes for the ideal wind turbine. *Journal of Physics: Conference Series*, 2007a.

V L Okulov. On the stability of multiple helical vortices. *Journal of Fluid Mechanics*, 2004.

V L Okulov and J N Sørensen. Stability of helical tip vortices in a rotor far wake. *Journal of Fluid Mechanics*, 2007b.

SA Orszag. Numerical simulation of incompressible flows within simple boundaries: accuracy. *Journal of Fluid Mechanics*, 1971.

SA Orszag. Spectral methods for problems in complex geometries. *Journal of Computational Physics*, 1980.

P Ploumhans, G S Winckelmans, J K Salmon, A Leonard, and M S Warren. Vortex Methods for Direct Numerical Simulation of Three-Dimensional Bluff Body Flows: Application to the Sphere at Re=300, 500, and 1000. *Journal of Computational Physics*, 2002.

D Pullin and P Saffman. Vortex dynamics in turbulence. *Annual Review of Fluid Mechanics*, 1998.

O Reynolds. *An experimental investigation of the circumstances which determine whether the motion of water shall be direct or sinuous and of the law of resistance in parallel channels.* Philosophical Translations of the Royal Society, 1883.

CM Rhie and WL Chow. Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA journal*, 1983.

RL Ricca. The effect of torsion on the motion of a helical vortex filament. *Journal of Fluid Mechanics Digital Archive*, 2006.

Lewis Fry Richardson. Weather prediction by numerical process. 1922.

LF Richardson. Atmospheric diffusion shown on a distance-neighbour graph. In *Proceedings of the Royal Society of London Series A*, 1926.

L Rosenhead. The formation of vortices from a surface of discontinuity. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 1931.

P G Saffman. *Vortex dynamics*. Cambridge University Press, 1995.

PG Saffman. The number of waves on unstable vortex rings. *Journal of Fluid Mechanics*, 1978.

Jörg Schumacher, Maik Boltes, Herwig Zilken, Marc-André Hermanns, Bruno Eckhardt, and Charles R Doering. Enstrophy amplification events in three-dimensional turbulence. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 2008.

K Shariff and A Leonard. Vortex rings. *Annual Review of Fluid Mechanics*, 1992.

K Shariff, R Verzicco, and P Orlandi. A numerical study of three-dimensional vortex ring instabilities: viscous corrections and early nonlinear stage. *Journal of Fluid Mechanics*, 1994.

Shashank, Johan Larsson, and Gianluca Iaccarino. A co-located incompressible Navier–Stokes solver with exact mass, momentum and kinetic energy conservation in the inviscid limit. *Journal of Computational Physics*, 2010.

A Shnirelman. On the nonuniqueness of weak solution of the Euler equation. *Journées Équations aux dérivées partielles*, 1996.

J Smagorinsky. General circulation experiments with the primitive equations. *Monthly Weather Review*, 1963.

A Staniforth. Semi-Lagrangian integration schemes for atmospheric models- A review. *Monthly Weather Review*, 1991.

`http://www.openfoam.com`. The open source computational fluid dynamics (cfd) toolbox, silicon graphics international corp. webpage, December 2011.

`http://www.openfoamwiki.net`. Unofficial openFOAM wiki, Media Wiki. webpage, December 2011.

Henk Kaarle Versteeg and Weeratunge Malalasekera. *An introduction to computational fluid dynamics.* Prentice Hall, 2007.

A Vincent. The dynamics of vorticity tubes in homogeneous turbulence. *Journal of Fluid Mechanics*, 1994.

J H Walther, M Guénot, E Machefaux, J T Rasmussen, P Chatelain, V L Okulov, J N Sørensen, M Bergdorf, and P Koumoutsakos. A numerical study of the stabilitiy of helical vortices using vortex methods. *Journal of Physics: Conference Series*, 2007.

S Widnall. The stability of a helical vortex filament. *Journal of Fluid Mechanics*, 1972.

S Widnall and C Tsai. The instability of the thin vortex ring of constant vorticity. *Philosophical Transactions of the Royal Society of London. Series A. Mathematical and Physical Sciences.*, 1977.

S Widnall, D Bliss, and CY Tsai. The instability of short waves on a vortex ring. *Journal of Fluid Mechanics*, 1974.

G Winckelmans, R Cocle, and L Dufresne. Vortex methods and their application to trailing wake vortex simulations. *C. R. Physique*, 2005.

M Zhang. The hydromagnetic nature of solar coronal mass ejections. *Annu Rev Astron Astrophys*, 2005.

U Ziegler. An ADI-based adaptive mesh Poisson solver for the MHD code NIR-VANA. *Computer Physics Communications*, 2004.

JB Zirker. Coronal heating. *Solar physics*, 1993.