UC Berkeley
UC Berkeley Electronic Theses and Dissertations

Title
Some Periodic Solutions of the Two-Dimensional Stokes-Oldroyd-B System with Stress Diffusion

Permalink
https://escholarship.org/uc/item/0bp939gf

Author
Isaacson, Erica Amy

Publication Date
2012

Peer reviewed|Thesis/dissertation

eScholarship.org Powered by the California Digital Library
University of California

Some Periodic Solutions of the Two-Dimensional Stokes-Oldroyd-B System with Stress Diffusion

by

Erica Isaacson

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Applied Mathematics

in the

GRADUATE DIVISION
of the
UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:
Professor Jon Wilkening, Chair
Professor James Sethian
Professor Ole Hald
Professor Ömer Savaş

Spring 2012

Some Periodic Solutions of the Two-Dimensional Stokes-Oldroyd-B System with Stress Diffusion

Copyright 2012
by
Erica Isaacson

Abstract

Some Periodic Solutions of the Two-Dimensional Stokes-Oldroyd-B System with Stress Diffusion

by

Erica Isaacson
Doctor of Philosophy in Applied Mathematics

University of California, Berkeley

Professor Jon Wilkening, Chair

We use a limited memory BFGS optimization method to seek time-periodic solutions of the Stokes-Oldroyd-B system of equations with a 4-roller forcing field and periodic boundary conditions. The gradient of the objective function for the optimization is found using a method which is based on the calculus of variations, and employs a pseudo-spectral implicit-explicit Runge-Kutta scheme. Once solutions are found, their asymptotic stability is calculated via an eigenvalue method. A variety of stationary and periodic solutions are found, plotted and systematized in a manner that suggests a global structure of periodic solutions.

Acknowledgements: I would like to thank my parents, Melanie and Eric Isaacson, and my thesis advisor, Jon Wilkening. I would also like to thank Ole Hald, James Sethian, Ömer Savaş, Becca Thomases, Alexandre Chorin, Grigory Isaakovich Barenblatt, Robert Saye, Matthias Morzfeld, Chris Rycroft, Barbara Salisbury, Valerie Heatlie, Jacob Rubinstein, Peter Sternberg, Scott Wissink, John Challifour, Robert Glassey, Linda McKinley, Alexander Dzierba, Katherine Isaacson, and Margaret Strong.

Contents

List of Figures

Acknowledgments

This work was supported in part by the National Science Foundation through Grant DMS-0955078 and by the Director, Office of Science, Computational and Technology Research, U.S. Department of Energy under Contract No.DE-AC02-05CH11231. Opinions, findings and conclusions are those of the author and do not reflect the views of the sponsors.

Chapter 1

Mathematical Models of Viscoelastic Fluids

1.1 A Short Discussion of Viscoelastic Fluids and their Properties

Viscoelastic fluids are a broad class of non-Newtonian fluids that exhibit properties of both viscous fluids and elastic materials. Most fluids have both viscous and elastic properties, but only those in which both properties play a significant role in the fluid's behavior are called viscoelastic. Molten metal, paint, bread dough, and biological fluids such as blood are all viscoelastic fluids. [19]

Viscoelastic fluids behave differently than Newtonian fluids because their microstructure is different. A Newtonian fluid's macroscopic behavior can be described mathematically by modeling it as a homogeneous medium. But a viscoelastic fluid has large molecules called polymers suspended in it, and these contribute significantly to its behavior. They do this chiefly in two ways. First, polymers tend to align with the direction of a flow, adding a stress in that direction which is called a normal stress. In addition, if the fluid is stretched, for example by pouring, the polymers are stretched by the viscosity of the surrounding flow and can exert a strong restoring force in response; this is called elongational viscosity. Once stretched, the polymers take some amount of time to recoil back to a rest state; as we will see, much of the fluid's behavior is determined by the ratio of the time scale on which this happens to the time scale of the fluid flow.

These effects lead to many interesting phenomena. If a jet of viscous fluid emerges from a pipe, the radius of the jet is a little smaller than that of the pipe; in a viscoelastic fluid, it is a little bigger. Think of pulling on a piece of rubber and letting it go again; the rubber gets shorter along the axis of pulling and fatter in the transverse dimension; the normal stresses of the viscoelastic fluid inside the pipe cause the same effect. A viscoelastic fluid also undergoes less jet breakup, because

the polymers in the fluid encourage it to cling to itself. In fact, polymers are often added to a fluid to stabilize its jet breakup in industrial applications. If you spin a rod in a Newtonian fluid, the centrifugal force from the rod pushes the fluid away and forms a dip around the rod. But a viscoelastic fluid will climb up the rod – think of cake batter climbing up the stem of an electric beater. Normal stresses are also responsible for this: the tension along the concentric stream lines pulls the fluid towards the center and pushes it up the rod. Finally, elongational viscosity in a viscoelastic fluid can cause it to exhibit what is called the tubeless siphon effect, in which the fluid pulls itself out of a beaker if a little bit goes over the side: an elastic solid would not be deformable enough to do this, and a purely viscous fluid would not adhere to itself enough [19]. You can see this effect by pouring egg whites out of a glass, as in Figure (1.1): the pouring is steady, and the bulge at the top of the fluid occurs because the egg whites are pulling themselves out of the glass.

Figure 1.1: An experiment you can do in your kitchen. Egg whites pull themselves out of a glass against gravity.

Viscoelastic fluids in the low Reynolds regime have been seen in experiments to exhibit complex behavior normally associated only with the high Reynolds regime of a Newtonian fluid. Low Reynolds numbers are often associated with simple fluid behavior, roughly because the high viscosity and low inertia prevent the fluid from forming eddies, vortices, and other complex and unstable formations commonly associated with high Reynolds number mixing. But experiments have shown that low Reynolds number polymer suspensions can do all of these things also, because the nonlinearities arising from the fluid's elastic properties replace (if not identically) those arising from the inertia term in the high Reynolds regime of the Navier Stokes equation. (see [23], [7], [9]). Our search for periodic solutions to a model of viscoelastic flow is partly motivated by a desire to reproduce some of these complex behaviors.

1.2 The Stokes-Oldroyd-B System

We will obtain the Stokes-Oldroyd-B model of viscoelastic flow by modifying a few simpler models. We are studying a two-dimensional flow on the domain $[0, 2\pi]^2$, and we assume that our solutions have periodic boundary conditions. As with all mathematical descriptions of a fluid, we begin with the continuity equation

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho u), \tag{1.1}$$

where ρ is the local density of the fluid and u is the fluid velocity. This is the mathematical statement of the local conservation of matter. If ρ remains constant in space and time, which we assume, it becomes simply

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

Next, we require the fluid to obey the law of balance of momentum:

$$\rho(\frac{\partial u}{\partial t} + (u \cdot \nabla)u) = f + \nabla \cdot S - \nabla p + \mu \triangle u. \tag{1.3}$$

The left side expresses the fluid's rate of change of momentum; f is an external force applied to the fluid, S is the polymer contribution to the stress tensor of the fluid, and the last two terms represent the contribution of pressure and Newtonian viscous forces to the fluid stress. This would be the Navier-Stokes equation if S were not there. It is a local equation, so the terms have units of force divided by volume. Since the fluid is incompressible, ρ has a constant value ρ_0. Let us nondimensionalize this equation by writing every variable as a dimensional constant times a dimensionless variable (the latter has the carat):

$$u = V\hat{u}, t = T\hat{t}, \nabla = \frac{1}{L}\hat{\nabla}, f = F\hat{f}, S = \sigma\hat{S}, p = P\hat{p}. \tag{1.4}$$

Notice that stress is an intrinsically local quantity, since it has units of force divided by area. Before completing the nondimensionalization, we take the intermediate step of dividing by μ, and declaring that LV is much smaller than the kinematic viscosity $\frac{\mu}{\rho_0}$, i.e., that the Reynolds number is small. Now we have

$$0 = \frac{V}{L^2}\hat{\triangle}\hat{u} + \frac{\sigma}{\mu L}\hat{\nabla} \cdot \hat{S} + \frac{F}{\mu}\hat{F} - \frac{P}{\mu L}\hat{\nabla}\hat{p}. \tag{1.5}$$

This is the Stokes equation. As was mentioned in the previous section, the disappearance of the nonlinear inertial terms gives rise to very different dynamics than those of the high Reynolds regime [17]. However, the nonlinearity of the constitutive equation governing the stress tensor S in a viscoelastic fluid can lead to effects similar to those normally associated with the high Reynolds regime [7]. Now we get rid of all the units by dividing by $\frac{V}{L^2}$:

$$0 = \hat{\triangle}\hat{u} + \frac{T\sigma}{\mu}\hat{\nabla}\cdot\hat{S} + \frac{L^2F}{V\mu}\hat{f} - \frac{L^2P}{V\mu L}\hat{\nabla}\hat{p} \tag{1.6}$$

Let us define β to be the coefficient in front of $\hat{\nabla}\cdot\hat{S}$. If we relate the flow time $T = \tau_{flow}$ to the other system parameters via

$$\tau_{flow} = \frac{\mu}{LF}, \tag{1.7}$$

we see that

$$\beta = \tau_{flow}\frac{\sigma}{\mu} = \frac{\mu}{LF}\frac{\sigma}{\mu} = \frac{\sigma}{LF}. \tag{1.8}$$

Thus β is a measure of the ratio of the polymer stress per unit length to the driving force.

Rewriting our dimensionless variables without carats for legibility, we have the nondimensional Stokes equation

$$0 = \triangle u + \beta\nabla\cdot S + f - \nabla p \tag{1.9}$$

We have defined the dimensionless variables f and p in such a way as to absorb the magnitude of the parameters that appeared in front of them in (1.6).

Notice that, since from (1.7) we have that

$$\tau_{flow} \propto \frac{1}{F}, \tag{1.10}$$

it is reasonable to guess that a similar relation exists between the polymer force $\frac{\sigma}{L}$ and the time scale on which the polymers relax:

$$\tau_{polymer} \propto \frac{\sigma}{L} \tag{1.11}$$

Then, since $\beta = \frac{\sigma}{LF}$, it is the case that

$$\beta \propto \frac{\tau_{flow}}{\tau_{polymer}} \tag{1.12}$$

or, defining the Weissenberg number

$$Wi = \frac{\tau_{polymer}}{\tau_{flow}} \tag{1.13}$$

we have

$$\beta \propto \frac{\tau_{flow}}{\tau_{polymer}} \tag{1.14}$$

where the constant of proportionality can be shown to be the product of $\tau_{polymer}$ with the ratio of the isotropic polymer stress due to microscopic drift to the solvent

viscosity [22]. Thus the value of Wi is dependent on the experiment – as can be seen from its being inversely proportional to the scale of the external force – but the product $\beta \times Wi$ is intrinsic to the fluid.

In our model, we use the value $\beta \times Wi = 0.5$, because this regime has been shown in [23] and [22] to produce mathematically interesting results.

We also choose f to be a 4-roller force defined by

$$f_1(x, y) = 2\sin(x)\cos(y) \tag{1.15}$$
$$f_2(x, y) = -2\cos(x)\sin(y). \tag{1.16}$$

To complete our system, we need an equation representing a constitutive law for the polymer stress tensor S. Most of the content of our model will reside in this equation. We will arrive at the equation by modifying some stress models of simpler fluids. This material is developed in Joseph [11] and Renardy [19]. Note that we do not include an energy equation, because we assume that the fluid's temperature is constant (see Chorin and Marsden, [2]).

The simplest fluid is a Newtonian fluid. The total stress experienced by a Newtonian fluid is given by

$$S_{total}(x, y, t) = \frac{\mu}{2}(\nabla u(x, y, t) + \nabla u(x, y, t)^T) - pI. \tag{1.17}$$

where p is the pressure function and u is the fluid velocity. (It makes sense that a fluid's internal stress should depend upon its velocity gradient, since if all particles of a fluid have the same velocity, the fluid will not interact with itself!) This model has the property that the stress depends only on the present value of the strain rate. This is not the case with a viscoelastic fluid because the polymers take time to recoil to a rest state even if the flow has stopped, giving the fluid a kind of memory of what has happened to it before.

A modification of Newton's model that gives the fluid a memory is the Maxwell model, in which we define the polymer contribution to the stress as the solution to the differential equation

$$\frac{\partial S}{\partial t} + \lambda S = \mu(\nabla u(x, y, t) + \nabla u(x, y, t)^T). \tag{1.18}$$

In this model, S is credited with a memory by making the Newtonian term a source term in a differential equation. It is not surprising that it has a Duhamel-type solution, namely the convolution of the solution to the corresponding homogeneous equation with the source term:

$$S(x, y, t) = \mu \int_{-\infty}^{t} e^{-\lambda(t-s)}(\nabla u(x, y, t) + \nabla u(x, y, t)^T)ds. \tag{1.19}$$

It is clear from this solution that λ, which has units of inverse time, represents the memory scale of the fluid: if λ is small, and therefore the time scale is large,

then $e^{-\lambda(t-s)}$ decays slowly, and values of the velocity gradient from a long time ago contribute significantly to the present value of the stress. This is an example of the general representation of a *linear* viscoelastic fluid's stress as a convolution

$$S(x,y,t) = \int_{-\infty}^{t} G(t-s)(\dot{\gamma}(x,y,s))ds \tag{1.20}$$

of what is called the relaxation modulus $G(t)$ with $\dot{\gamma}$, the rate of strain undergone by the fluid. $G(t)$ tells you how much influence the strain rate from t time units ago has on the present value of the stress.

Unfortunately, every linear model of a fluid stress tensor except Newton's model has the property that the stress tensor does not transform appropriately if the fluid is rotated (see Renardy [19]). We also note that Maxwell's model track the history of $\dot{\gamma}$ at a fixed point in space rather than following particle trajectories. To address these problems, we can modify the Maxwell model by replacing $\frac{\partial S}{\partial t}$ with what is variously called the upper convected time derivative or the Oldroyd derivative

$$\frac{\partial S}{\partial t} + (u \cdot \nabla)S - (S\nabla u + \nabla u^T S), \tag{1.21}$$

which measures a tensor quantity in a coordinate system that is rotating and stretching with the fluid (see Joseph [11] and Van and Matolcsi [14]). This gives us

$$\frac{\partial S}{\partial t} + (u \cdot \nabla)S - (S\nabla u + \nabla u^T S) + \lambda S = \mu(\nabla u + \nabla u^T). \tag{1.22}$$

Equation (1.22) is called the Oldroyd-B equation. As in the Maxwell model, λ has units of inverse time. Its value arises as a function of the size of the molecules in the polymers and from thermodynamical constants. Notice also that μ has different units than in Newton's model (this was the case in Maxwell's model also). It arises as a function of the drag force of the surrounding fluid on the polymers.

The Oldroyd-B equation has the weakness that it sometimes produces infinite stresses for finite strain rates [18]. This happens because, if the model is derived from microscopic principles, it represents the polymers as springs whose restoring force is a linear function of how far they are stretched, so the macroscopic flow is sometimes enough to stretch them infinitely. Some models, like the FENE-P (finitely extensible nonlinear elastic polymer) model, use a nonlinear spring law for the polymers that enforces a finite maximum length, but the mathematics involved are formidable [23]. Our solution is to add a stress diffusion term

$$\nu \triangle S \tag{1.23}$$

to the constitutive law. This effectively enforces a finite length for the polymers by transferring energy from highly stretched polymers to more relaxed ones [6]. Stress diffusion does occur in real polymers, but our coefficient ν is artificially large [22].

Let us nondimensionalize this equation. Defining our nondimensional variables as in the discussion of the Stokes equation, and defining $\tau_{polymer}$ by $\lambda = \frac{1}{\tau_{polymer}}$, we get

$$\frac{\sigma}{\tau_{flow}}\frac{\partial \hat{S}}{\partial \hat{t}} + \frac{V\sigma}{L}[(\hat{u}\cdot\hat{\nabla})\hat{S} - (\hat{S}\hat{\nabla}\hat{u} + \hat{\nabla}\hat{u}^T\hat{S})] + \frac{\sigma}{\tau_{polymer}}\hat{S} = \frac{V}{L}(\hat{\nabla}\hat{u} + \hat{\nabla}\hat{u}^T) + \frac{\nu\sigma}{L^2}\hat{\triangle}\hat{S} \quad (1.24)$$

Dividing through by $\frac{\sigma}{\tau_{flow}}$ and once again rewriting the nondimensional variables without carats, we get

$$\frac{\partial S}{\partial t} + (u\cdot\nabla)S - (S\nabla u + \nabla u^T S) + \frac{1}{Wi}S = (\nabla u + \nabla u^T) + \nu \triangle S. \quad (1.25)$$

Here again is $Wi = \frac{\tau_{polymer}}{\tau_{flow}}$. We have adjusted the value of σ as necessary to make it equal to μ so that the parameter in front of $(\nabla u + \nabla u^T)$ disappears; this has the effect of rescaling the dimensionless variable S. We have also redefined ν to equal $\frac{\tau_{flow}}{L^2}\nu_{old}$.

Finally, if we define a new variable $\bar{S} = S + I$ and then rewrite $\bar{S}$ as just S again, we have

$$\frac{\partial S}{\partial t} + (u\cdot\nabla)S - (S\nabla u + \nabla u^T S) + \frac{1}{\lambda}(S - I) = \nu \triangle S. \quad (1.26)$$

This absorbs the term $(\nabla u + \nabla u^T)$ into the Oldroyd derivative and changes $\frac{1}{\lambda}(S)$ to $\frac{1}{\lambda}(S - I)$, which makes the equation more legible. Note that adding a constant matrix to S does not change its definition in the Stokes equation, since it appears there inside a divergence.

In our model, we define the value of the stress diffusion coefficient ν to be $\frac{0.01}{Wi}$. It is sensible for ν to be inversely proportional to Wi, since we expect the terms which together are equated with the Oldroyd derivative to depend on the value of Wi in the same way.

So, all together, the system we will use in our simulations is

$$\nabla \cdot u = 0$$

$$\beta\nabla \cdot S + \triangle u - \nabla p + f = 0$$

$$\frac{\partial S}{\partial t} = -(u\cdot\nabla)S + (\nabla u S + S\nabla u^T) - \frac{1}{Wi}(S - I) + \nu \triangle S. \quad (1.27)$$

The conservation of angular momentum requires that S is a symmetric matrix $\begin{pmatrix} S_{11} & S12 \\ S_{12} & S_{22} \end{pmatrix}$. S has three independent components; u has two components, and p is a scalar. Therefore, these are six equations in six unknowns.

Chapter 2

Some Theoretical Background

2.1 Overview

In the last chapter, we developed the system of equations that we will use in our simulations. In this one, we develop some methods to find periodic solutions of this system, and to study their stability properties. First, we define a functional G which measures a norm of the difference between the initial value of the stress tensor S and its value at some later time. G will depend upon the initial value S_0, which is represented as a sum of Fourier modes, and upon the final time T. We will minimize G using the BFGS algorithm, which will be introduced and discussed in section (3.6). To implement this algorithm, we will need both the value of G and its gradient when it is represented as a function of the discrete list of Fourier amplitudes of S_0. The calculation of this gradient is a lengthy one, to which most of this chapter is devoted. It requires the development of both the linearization of the Stokes-Oldroyd-B system and the adjoint of that linearization. Finally, we discuss how to calculate the stability of the solutions that we find.

Throughout this chapter, we treat the Oldroyd-B equation (1.26) as the main equation to solve, and S the main variable whose value we find. Since u and p obey static equations, we regard them as functions of the value of S at every time step.

2.2 Finding Periodic Solutions with an Optimization Method

Suppose that $S(x, y, t)$ is the 2×2 symmetric matrix solution, at time t, of our PDE, with initial condition

$$S(x, y, 0) = S_0(x, y). \tag{2.1}$$

Let us define a functional G to be the quantity

$$\frac{1}{2}\int_0^{2\pi}\int_0^{2\pi}\sum_{i,j=1}^{2}(S_{ij}(x,y,T)-S_{ij}(x,y,0))^2dxdy. \tag{2.2}$$

Notice that this is one half of the square of a norm of the distance between S and itself at some later time. Suppose that a solution of our equations has the initial value S_0 and is periodic with period T. Then we will have

$$G(S_0,T)=0. \tag{2.3}$$

Thus, we can seek a periodic solution by finding an initial condition S_0 and a final time T such that $G=0$. Since G can never have a negative value, this is the same as minimizing G.

To minimize G numerically, we need to represent it as a function of a finite list of variables. G is a functional that depends on the initial conditions $S_0(x,y)$ and T, so let us represent $S_0(x,y)$ as a discrete Fourier sum:

$$S_0=\sum_{m,n\in\Lambda}[a_{mn}e^{i(mx+ny)}+a_{(-m,-n)}e^{-i(mx+ny)}] \tag{2.4}$$

where

$$a_{-m,-n}=\overline{a}_{mn}, \tag{2.5}$$

since S is real-valued, and

$$\Lambda=\{(0,n):0\le n\le N\}\cup\{(m,n):1\le m\le N,-N\le n\le N\}, \tag{2.6}$$

i.e. Λ is a set of integer pairs in a region of the right half-plane. Now we can represent G as a function of a finite list of variables:

$$G=G(a_{mn},T). \tag{2.7}$$

The advantages of representing S in Fourier space instead of physical space will be discussed further in section (3.7).

We are going to use the BFGS optimization method to minimize G, and this method needs the gradient of G. We now find this gradient using a method developed in [1] which is based on the calculus of variations.

2.3 Calculating ∇G

Throughout this section we denote by $\langle:\rangle$ the following inner product for 2×2 matrix functions:

$$\langle A:B\rangle=\langle\left(\begin{smallmatrix}A_1&A_2\\A_3&A_4\end{smallmatrix}\right):\langle\left(\begin{smallmatrix}B_1&B_2\\B_3&B_4\end{smallmatrix}\right)\rangle:=\int_0^{2\pi}\int_0^{2\pi}(A_1B_1+A_2B_2+A_3B_3+A_4B_4)dxdy. \tag{2.8}$$

Using this notation, our definition of G becomes

$$G(S_0, t) = \frac{1}{2}\langle S(x, y, T) - S_0(x, y) : S(x, y, T) - S_0(x, y)\rangle, \tag{2.9}$$

or, suppressing the spatial arguments and writing the time values as subscripts,

$$G(S_0, t) = \frac{1}{2}\langle S_T - S_0 : S_T - S_0\rangle. \tag{2.10}$$

We must calculate the partial derivative of G with respect to each variable. These variables are the period T, and the amplitudes a_{mn} of the Fourier modes $e^{imx+iny}$.

First, we have that $\frac{\partial G}{\partial T}(S_0, T)$ is simply

$$\frac{\partial G}{\partial T}(S_0, T) = \langle (S_T - S_0) : \frac{\partial S_T}{\partial t}\rangle. \tag{2.11}$$

Next, for a given (m, n), the partial derivative of G with respect to a_{mn} is

$$\dot{G} = \frac{\partial}{\partial \epsilon}\Big|_{\epsilon=0} G(S_0 + \epsilon \dot{S}_0, T)$$

where $\dot{S}_0 = e^{imx+iny}$. (It will become clear in a moment why we introduce the notation $\dot{S}_0$.)

To find $\dot{G}$, we have to calculate the solution at time T to the Oldroyd-B equation with initial condition $S_0 + \epsilon \dot{S}_0$. Let $S(x, y, t, \epsilon)$ be this solution, and define

$$\dot{S}(x, y, t) = \frac{d}{d\epsilon}\Big|_{\epsilon=0} S(x, y, t, \epsilon). \tag{2.12}$$

$\dot{S}$ is a measure of the sensitivity of S_t to perturbations in its initial condition. Using this definition, we have

$$\frac{\partial G}{\partial \epsilon}(S_0 + \epsilon\sigma_0, T) = \lim_{\epsilon \to 0} \frac{G(S_0 + \epsilon\sigma_0, T) - G(S_0, T)}{\epsilon} = \langle S_T - S_0 : \dot{S}_T - \dot{S}_0\rangle. \tag{2.13}$$

If we define

$$\dot{u} = \frac{d}{d\epsilon}\Big|_{\epsilon=0} u(x, y, t, \epsilon), \tag{2.14}$$

and

$$\dot{p} = \frac{d}{d\epsilon}\Big|_{\epsilon=0} p(x, y, \epsilon), \tag{2.15}$$

then we can find the values of $\dot{S}$, $\dot{u}$, and $\dot{p}$ by linearizing the Stokes-Oldroyd-B system. Let us rewrite the system as

$$\nabla \cdot u = 0$$

$$\beta \nabla \cdot S + \triangle u - \nabla p + f = 0$$

$$\frac{\partial S}{\partial t} = UC(S,u) - \frac{1}{Wi}(S-I) + \nu \triangle S \tag{2.16}$$

where

$$UC(S,u) := -(u\cdot\nabla)S + (\nabla u S + S\nabla u^T) \tag{2.17}$$

is the spatial part of the upper convected time derivative. Because UC is a bilinear operator, we have

$$\frac{d}{d\epsilon}\Big|_{\epsilon=0} UC(S,u) = UC(\dot{S},u) + UC(S,\dot{u}). \tag{2.18}$$

Applying this to (2.16) shows that $\dot{S}$, $\dot{u}$, and $\dot{p}$ solve the equations

$$\nabla\cdot\dot{u} = 0$$

$$-\nabla\dot{p} + \triangle\dot{u} + \beta\nabla\cdot\dot{S} = 0$$

$$\frac{\partial\dot{S}}{\partial t} = UC(\dot{S},u) + UC(S,\dot{u}) - \frac{1}{Wi}\dot{S} + \nu\triangle\dot{S} \tag{2.19}$$

This system is called the linearization of the Stokes-Oldroyd-B system. In fact, if we consider the system (2.16) as being of the form

$$\frac{\partial S}{\partial t} = F(S), \tag{2.20}$$

then (2.19) is simply

$$\frac{\partial\dot{S}}{\partial t} = DF(S(t))\dot{S}, \tag{2.21}$$

where DF is the Frechet derivative of F with respect to S. Thus there is an analogy between the functional analysis we are doing and the simpler case of ODEs.

To calculate $\frac{\partial G}{\partial a_{mn}}$ using the above method, we must solve a different initial value problem to find $\dot{S}(x,y,t)$ for each choice of $\dot{S}_0(x,y)$! Fortunately, we can avoid this. After all, $\frac{\partial G}{\partial a_{mn}}$ is a directional derivative in what we can think of as the direction of $\dot{S}_0$, so it is reasonable to suspect that we can find a quantity $\frac{\delta G}{\delta S_0}$ that mimics a gradient in the sense that any directional derivative can be represented as an inner product of the direction with $\frac{\delta G}{\delta S_0}$:

$$\frac{\partial G}{\partial\epsilon}(S_0+\epsilon\dot{S}_0,T)\Big|_{\epsilon=0} = \dot{G} = \langle\frac{\delta G}{\delta S_0},\dot{S}_0\rangle \tag{2.22}$$

To find a formula for $\frac{\delta G}{\delta S_0}$, we use a trick. Suppose that there exists a function $\tilde{S}(x,y,s)$ with the property that

$$\langle\dot{S}(x,y,t) : \tilde{S}(x,y,T-t)\rangle \tag{2.23}$$

has a constant value for all $0 \leq t \leq T$. Then, suppressing spatial arguments and writing time arguments as subscripts as before, we have that

$$\langle \dot{S}_T : \tilde{S}_0 \rangle = \langle \dot{S}_0 : \tilde{S}_T \rangle. \tag{2.24}$$

Now let us define

$$\tilde{S}_0 = \tilde{S}(x, y, 0) = S(x, y, T) - S(x, y, 0). \tag{2.25}$$

Then

$$\dot{G} = \langle \tilde{S}_0 : \dot{S}_T - \dot{S}_0 \rangle. \tag{2.26}$$

We split up the terms to get

$$\langle \tilde{S}_0 : \dot{S}_T - \dot{S}_0 \rangle = \langle \tilde{S}_0 : \dot{S}_T \rangle - \langle \tilde{S}_0 : \dot{S}_0 \rangle, \tag{2.27}$$

and then by (2.23) we have

$$\langle \tilde{S}_0 : \dot{S}_T \rangle - \langle \tilde{S}_0 : \dot{S}_0 \rangle = \langle \tilde{S}_T : \dot{S}_0 \rangle - \langle \tilde{S}_0 : \dot{S}_0 \rangle = \langle \tilde{S}_T - \tilde{S}_0, \dot{S}_0 \rangle. \tag{2.28}$$

Comparing the above with (2.22) and (2.26), we see that

$$\tilde{S}_T - \tilde{S}_0 = \frac{\delta G}{\delta S_0}. \tag{2.29}$$

So let us find $\tilde{S}_T$. Taking the time derivative of both sides of (2.23), we obtain

$$\langle \frac{\partial}{\partial t} \dot{S}_t : \tilde{S}_{T-t} \rangle + \langle \dot{S}_t : \frac{\partial}{\partial t} \tilde{S}_{T-t} \rangle = 0 \tag{2.30}$$

or

$$\langle \frac{\partial \dot{S}_t}{\partial t} : \tilde{S}_s \rangle = \langle \dot{S}_t : \frac{\partial \tilde{S}_s}{\partial s} \rangle \tag{2.31}$$

where

$$s := T - t. \tag{2.32}$$

Substituting the right hand side of (2.19) for $\frac{\partial \dot{S}}{\partial t}$, we obtain

$$\langle\ UC(\dot{S}, u) + UC(S, \dot{u}) - \frac{1}{Wi} \dot{S} + \nu \triangle \dot{S} : \tilde{S}(x, y, s) \rangle = \langle \dot{S}(x, y, t) : \frac{\partial \tilde{S}(x, y, s)}{\partial s} \rangle. \tag{2.33}$$

Our goal is to integrate by parts the terms in the left hand side of the above equation to obtain $\langle \dot{S} : Q \rangle$, where Q is an expression involving $\tilde{S}$, S, and their derivatives. Then, we could equate this with the right hand side of (2.33) and conclude, because the inner product is non-degenerate, that in fact $\frac{\partial \tilde{S}}{\partial s} = Q$, and use this PDE to obtain the value of $\tilde{S}_T$.

How will we deal with terms like $(\dot{u} \cdot \nabla) S$, in which $\dot{S}$ does not appear? Observe that

$$\left\langle \frac{\partial \dot{S}}{\partial t} : \tilde{S} \right\rangle = \left\langle \begin{pmatrix} \frac{\partial \dot{S}}{\partial t} \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \tilde{S} \\ w \\ r \end{pmatrix} \right\rangle. \tag{2.34}$$

Now since

$$\begin{pmatrix} \frac{\partial \dot{S}}{\partial t} \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} UC(\dot{S}, u) + UC(S, \dot{u}) - \frac{1}{Wi}\dot{S} + \nu \triangle \dot{S} \\ -\nabla \dot{p} + \triangle \dot{u} + \beta \nabla \cdot \dot{S} \\ \nabla \cdot \tilde{u} \end{pmatrix}, \tag{2.35}$$

we have

$$\langle \frac{\partial \dot{S}}{\partial t} : \tilde{S} \rangle = \left\langle \begin{pmatrix} UC(\dot{S}, u) + UC(S, \dot{u}) - \frac{1}{Wi}\dot{S} + \nu \triangle \dot{S} \\ -\nabla \dot{p} + \triangle \dot{u} + \beta \nabla \cdot \dot{S} \\ \nabla \cdot \tilde{u} \end{pmatrix}, \begin{pmatrix} \tilde{S} \\ \tilde{u} \\ \tilde{p} \end{pmatrix} \right\rangle, \tag{2.36}$$

and now we can sort the terms into those containing $\dot{S}$, those containing $\dot{u}$, and those containing $\tilde{p}$:

$$\begin{aligned} \langle \frac{\partial \dot{S}}{\partial t} : \tilde{S} \rangle = \{ \langle UC(\dot{S}, u) : \tilde{S} \rangle - \langle \frac{1}{Wi}\dot{S} : \tilde{S} \rangle + \langle \nu \triangle \dot{S} : \tilde{S} \rangle + \langle \beta \nabla \cdot \dot{S} : \tilde{u} \rangle \} \\ + \{ \langle UC(S, \dot{u}) : \tilde{S} \rangle + \langle \triangle \dot{u} : \tilde{u} \rangle + \langle \nabla \cdot \dot{u} : \tilde{p} \rangle \} \\ - \{ \langle \nabla \dot{p} : \tilde{u} \rangle \} \end{aligned} \tag{2.37}$$

We isolate the dotted variables on the left side of each product using matrix identities and integration by parts. For example, the fact that for any matrices A, B, and C we have

$$\langle AB : C \rangle = \langle B : A^T C \rangle \tag{2.38}$$

gives us readily

$$\langle (\nabla u)\dot{S} : \tilde{S} \rangle = \langle \dot{S} : (\nabla u)^T \tilde{S} \rangle. \tag{2.39}$$

For the term

$$\langle \nabla \dot{u} S : \tilde{S} \rangle, \tag{2.40}$$

we use the identity

$$\langle AB : C \rangle = \langle A : CB^T \rangle \tag{2.41}$$

to write

$$\langle \nabla \dot{u} S : \tilde{S} \rangle = \langle \nabla \dot{u} : S\tilde{S} \rangle, \tag{2.42}$$

and use integration by parts to isolate $\dot{u}$:

$$\begin{aligned} \langle \nabla \dot{u} : S\tilde{S} \rangle &= \langle \left(\begin{smallmatrix} \partial_1 u_1 & \partial_2 u_1 \\ \partial_1 u_2 & \partial_2 u_2 \end{smallmatrix} \right) : S\tilde{S} \rangle \\ &= -\int_0^{2\pi} \int_0^{2\pi} u_1[\partial_1 (S\tilde{S})_{11} + \partial_2 (S\tilde{S})_{12}] + u_2[\partial_1 (S\tilde{S})_{21} + \partial_2 (S\tilde{S})_{22}] dxdy \\ &= \langle \left(\begin{smallmatrix} u_1 \\ u_2 \end{smallmatrix} \right), -\nabla \cdot (S\tilde{S}) \rangle. \end{aligned} \tag{2.43}$$

Notice that we have used the periodic boundary conditions to kill the boundary terms. The other terms are calculated similarly. When we put it all back together, we get

$$\left\langle \begin{pmatrix} \dot{S} \\ \dot{u} \\ \dot{p} \end{pmatrix}, \begin{pmatrix} \frac{\partial \tilde{S}}{\partial s} \\ 0 \\ 0 \end{pmatrix} \right\rangle = \left\langle \begin{pmatrix} \dot{S} \\ \dot{u} \\ \dot{p} \end{pmatrix}, \begin{pmatrix} Q \\ \triangle \tilde{u} - \nabla \tilde{p} - 2\nabla \cdot (S\tilde{S}) + \sum_{i,j=1}^{2} S_{ij} \nabla \tilde{S}_{ij} \\ \nabla \cdot \tilde{u} \end{pmatrix} \right\rangle, \tag{2.44}$$

where

$$Q = (u \cdot \nabla)\tilde{S} + (\nabla u^T \tilde{S} + \tilde{S} \nabla u) - \frac{1}{2}\beta(\nabla \tilde{u} + \nabla \tilde{u}^T) - \frac{1}{Wi}\tilde{S} + \nu \triangle \tilde{S}. \tag{2.45}$$

So the system that evolves ($\tilde{S}$,$\tilde{u}$,$\tilde{p}$) is

$$\nabla \cdot \tilde{u} = 0$$

$$\triangle \tilde{u} - \nabla \tilde{p} - 2\nabla \cdot (S\tilde{S}) + \sum_{i,j=1}^{2} S_{ij} \nabla \tilde{S}_{ij} = 0$$

$$\frac{\partial \tilde{S}}{\partial s} = (u \cdot \nabla)\tilde{S} + (\nabla u^T \tilde{S} + \tilde{S} \nabla u) - \frac{1}{2}\beta(\nabla \tilde{u} + \nabla \tilde{u}^T) - \frac{1}{Wi}\tilde{S} + \nu \triangle \tilde{S} \tag{2.46}$$

Since the variables of which G is a function are the amplitudes a_{mn} of the functions $e^{imx+iny}$, and since for a real-valued function we have $a_{mn} = \bar{a}_{-m,-n}$, we see that the components of ∇G are given by

$$\frac{\partial G}{\partial(\Re(a_{mn}))} = \langle \tilde{S}_T - \tilde{S}_0 : e^{imx+iny} + e^{-imx-iny} \rangle = 8\pi^2 \Re(\mathcal{F}(\tilde{S}_T - \tilde{S}_0)_{mn}) \tag{2.47}$$

and

$$\frac{\partial G}{\partial(\Im(a_{mn}))} = \langle \tilde{S}_T - \tilde{S}_0 : ie^{imx+iny} - ie^{-imx-iny} \rangle = 8\pi^2 \Im(\mathcal{F}(\tilde{S}_T - \tilde{S}_0)_{mn})), \tag{2.48}$$

where $\mathcal{F}$ denotes the Fourier transform. In other words, the components of ∇G are multiples of the components of the Fourier transform of $\tilde{S}_T - \tilde{S}_0$! This is convenient and elegant, but it is not special to the Fourier basis; it would have arisen with any orthogonal basis.

2.4 Stability of Periodic Solutions

Once we have found a periodic solution to the Stokes-Oldroyd-B system, we would like to compute its stability. Does the solution remain true over many periods, or do the small deviations introduced by roundoff error in the numerics, or by noise in the physical system, cause it to stray off course? To understand how to discover this, we discuss some elegant ODE theory from Coddington [3] and Verhulst [24].

We begin by revisiting the linearization of our system, this time in a different guise. Let us once again consider the Oldroyd-B equation as being of the form

$$\frac{dS}{dt} = F(S). \tag{2.49}$$

Suppose that $S(t)$ is a periodic solution to (2.49) and that ΔS is a small perturbation. Then ΔS satisfies

$$\frac{d\Delta S}{dt} = F(S + \Delta S) - F(S). \tag{2.50}$$

From the definition of the Frechet derivative, we have then that

$$\frac{d\Delta S}{dt} = DF(S) \cdot \Delta S + o(||\Delta S||). \tag{2.51}$$

It is nice that these calculus-like principles hold even for operators such as DF which send functions to functions. Note the similarity between this and the linearization that we obtained in the previous section. The difference is that, whereas ΔS equals the difference between two distinct solutions, $\dot{S}$ represents the *rate* at which a small perturbation in the initial condition grows over time, so the $o(|\Delta S|)$ term in its equation disappears when we take the limit. But the beautiful fact is that the presence of the $o(|\Delta S|)$ term in (2.51) will not stop the linearization obtained in the previous section - the same equation as above but without the $o(|\Delta S|)$ term - from governing the stability of the nonlinear Oldroyd-B equation. Now we will briefly describe why this is the case.

First, we can consider $DF(S)$ to be a matrix. It is a linear operator, and in discretizing the problem we have made it effectively one that acts on a finite dimensional space. As we will see, it is not necessary to compute the entries of this matrix, but knowing that there is one allows us to apply the following analysis to our situation.

Let us make our idea of stability more precise. A solution $x(t)$ to a differential equation is said to be asymptotically stable if any solution that starts out sufficiently

close to it eventually becomes arbitrarily close to it; that is, if there is a $\delta > 0$ such that if

$$||x(0) - y(0)|| < \delta \tag{2.52}$$

then

$$\lim_{t\to\infty} ||x(t) - y(t)|| = 0. \tag{2.53}$$

If the zero solution to (2.51) is asymptotically stable for a given periodic solution S, then if ΔS is initially small, $S + \Delta S$ will approach S over time.

We have already noted that the equation (2.51) can be regarded as a matrix ODE. By the same reasoning, the Oldroyd-B equation in the form (2.49) can also be regarded as a matrix ODE. The asymptotic stability of solutions to an equation of this form can understood by relating it to simpler cases. First, it is the case that if A is an $n \times n$ *constant* matrix, and x evolves according to

$$\frac{dx}{dt} = Ax, \tag{2.54}$$

then x is asymptotically stable if and only if A's eigenvalues all have negative real parts. This can be seen because the equation has the fundamental solution $x(t) = Ce^{At}$, where C is constant. If $A = P\Lambda P^t$ where Λ is A's Jordan form, then we have $x(t) = CPe^{\Lambda t}P^t$. If Λ is diagonal, $e^{\Lambda t}$ is readily seen to go to zero as t increases. If Λ is a proper Jordan matrix, then $e^{\Lambda t}$ is a direct sum of matrix exponentials of Jordan blocks, each of which is equal to $e^{\lambda_j}e^N$, where λ_j are the diagonal entries of Λ, and N is a nilpotent matrix. Since N is nilpotent, the series for e^N has a finite number of terms, so $e^{\Lambda t}$ approaches zero if and only if all the λ_j's have a negative real part.

So the eigenvalues of A tell us how x behaves. But in fact, they still tell us how x behaves if it evolves according to a nonlinear ODE whose nonlinear part is not too big. Suppose that

$$\frac{dx}{dt} = Ax + f(t, x) \tag{2.55}$$

where $|f(t, x)| = o(|x|)$ for all $t \geq 0$. Suppose that ϕ is a solution of (2.55). Then as long as $\phi(t)$ exists, it is equal to

$$e^{tA}\phi(0) + \int_0^t e^{(t-s)A} f(s, \phi(s))ds. \tag{2.56}$$

If A's real parts are negative, then there are positive constants K and α such that

$$|e^{At}| \leq Ke^{-\alpha t} \tag{2.57}$$

for $t \geq 0$. Picking x small enough that $|f(t, x)| \leq \frac{\epsilon|x|}{K}$ and applying the Gronwall inequality, we obtain the estimate

$$e^{\alpha t}|\phi(t)| \leq K|\phi(0)|e^{-(\alpha-\epsilon)t}, \tag{2.58}$$

so ϕ approaches zero as long as $\epsilon < \alpha$.

If x satisfies the same kind of equation but A is periodic in time, we can reduce back to the case in which A is constant. First, suppose that x satisfies

$$\frac{dx}{dt} = A(t)x \tag{2.59}$$

with $A(t+T) = A(t)$. Let Φ be a fundamental matrix solution of this system of equations, by which we mean that Φ is a real-valued $n \times n$ matrix whose columns comprise a basis for the solution space of the equation. We then have the matrix equation

$$\dot{\Phi}(t) = A(t)\Phi(t). \tag{2.60}$$

Since the above holds for any value of t, and A is periodic with period T, we have that

$$\dot{\Phi}(t+T) = A(t+T)\Phi(t+T) = A(t)\Phi(t+T), \tag{2.61}$$

i.e., the columns of $\Phi(t+T)$ are also solutions of the ODE. Since the columns of $\Phi(t)$ are a basis for the solution space, each column of $\Phi(t+T)$ is expressible as a linear combination of the columns of $\Phi(t)$ – that is, there is a real-valued constant matrix C such that

$$\Phi(t+T) = \Phi(t)C. \tag{2.62}$$

You can see immediately that, since

$$\Phi(t+mT) = \Phi(t)C^m \tag{2.63}$$

for any positive integer m, the eigenvalues of C will determine the behavior of $\Phi(t)$. However, we need a few more facts to see what will happen when we add the $o(|x|)$ term. Let B be a constant matrix such that TB is a logarithm of C, so $C = e^{BT}$. Let us define a matrix function $\Psi(t)$ by $\Psi(t) = \Phi(t)e^{-Bt}$. Then we can write Φ as

$$\Phi(t) = \Psi(t)e^{Bt}. \tag{2.64}$$

Now it is readily seen that

$$\Phi(t+T) = \Psi(t+T)e^{Bt}e^{BT} = \Phi(t)C = \Psi(t)e^{Bt}C = \Psi(t)e^{Bt}e^{BT}. \tag{2.65}$$

Therefore $\Psi(t+T) = \Psi(t)$. Thus, a fundamental solution to this system can always be written as a matrix product of a periodic matrix Ψ with a matrix e^{Bt}, where B

is a constant matrix. Furthermore, Ψ is nonsingular, since it is the product of the nonsingular matrix e^{-Bt} with the nonsingular matrix Φ (since Φ is a fundamental solution, its columns are linearly independent).

Finally, suppose that x satisfies

$$\frac{dx}{dt} = A(t)x + f(t,x) \tag{2.66}$$

where $f(t,x)$ is $o(|x|)$ for $t \geq 0$. Let $\Phi(t) = \Psi(t)e^{Bt}$ be a solution to the equation without the $o(|x|)$ term, where Ψ and B are defined as above. If we make the change of variables

$$w = \Psi^{-1}(t)x, \tag{2.67}$$

then w satisfies

$$\frac{dw}{dt} = Bx + \Psi^{-1}f(t, \Psi(t)w). \tag{2.68}$$

Since $\Psi(t)$ and $\Psi^{-1}(t)$ are periodic and nonsingular, and therefore harmless, it is the case that $\Psi^{-1}f(t, \Psi(t)w) = o(|w|)$ for $t \geq 0$. So w, and therefore also x, are asymptotically stable if and only if the real parts of B's eigenvalues are negative. Since, by defining C as above, we have $C = e^{BT}$, we see that our earlier intuition was correct, and a solution to our system will be asymptotically stable if the eigenvalues of C are less than one in magnitude. This is called the theorem of orbital stability [3].

Note that for an autonomous equation – that is, one that does not explicitly depend on time – there will always be one eigenvalue of magnitude exactly 1. This is because a phase shift in the solution produces another periodic solution, and one can make an arbitrarily small shift and produce two solutions that do not approach one another, but rather maintain a discreet (and discrete) distance. This is not an instability in the sense that solutions will wander away from one another; an appropriate phase shift will bring the two solutions arbitrarily close together.

To compute the eigenvalues of e^{BT}, we choose a basis such that $\Phi(0)$ is the identity matrix. Then $\Psi(0)$, and thus $\Psi(T)$, are also equal to the identity matrix. Therefore, $\Phi(T) = e^{BT}$. We choose an initial value of $\dot{S}_0$ and evolve it to $\dot{S}_T$ via the linearized equations. Since $\dot{S}_T$ is equal to $\Phi(T)$ applied to $\dot{S}_0$, one can use an iterative algorithm to update $\dot{S}_0$ at each iteration so that it converges to a basis of an invariant eigenspace of $\Phi(T)$. Thus, we can calculate the stability of a solution S without ever needing to find the matrix representing $DF(S)$: we only need to solve the linearized problem for an iterated sequence of initial conditions.

Chapter 3

Numerical Methods

3.1 Code Overview

Figure (3.1) shows the steps we used to find families of periodic solutions. To solve the forward equation (1.27), we first transform it into an ODE using the Fast Fourier Transform. Then we advance this ODE using an implicit-explicit Runge Kutta (IMEX-RK) scheme of fifth order, inverting the Fourier transform to perform the multiplications in physical space.

When the value of S is obtained at a final time T, the quantity $(S_T - S_0)$ is used for the initial condition of the adjoint equation as described in section (2.3). The adjoint equation is evolved using the same IMEX-RK scheme. If G and $||\nabla G||$ are small enough – about 10^{-16} and 10^{-10}, respectively – we declare the solution minimized, and compute its stability using ARPACK and the methods developed in section (2.4). If it is stable, we use it as the initial condition for a search for a periodic solution for a nearby value of Wi.

In this chapter, we develop and discuss the methods outlined above in more detail.

3.2 The First Periodic Solution

Many quasi-periodic solutions can be found by starting with S_0 equal to the identity matrix and letting it evolve for a long time. A typical such solution develops quasi-periodic behavior after one thousand to several thousand units of time. Figure (3.2) shows a plot of the first velocity component versus time of a solution that was found in this manner, with Wi=12. This was the first function that our algorithm refined into a periodic solution; its initial value of G was about 10^{-3}, and our algorithm brought it down to 10^{-16}. This solution was then used to start the process in Figure (3.1).

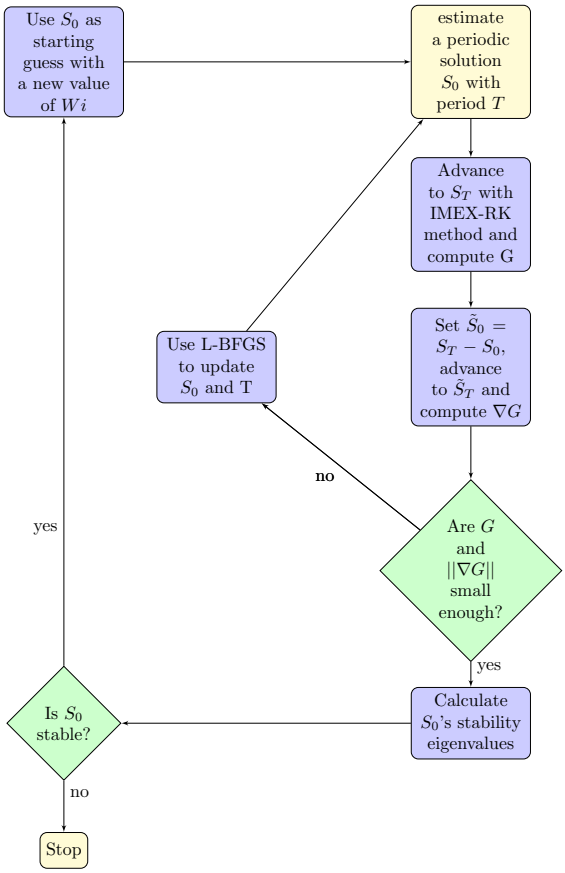

Figure 3.1: A process for finding families of periodic solutions using the BFGS algorithm, starting with the yellow box in the upper right corner.

3.3 Pseudospectral Methods

Our Fourier representation of S and its related functions will transform our system of PDE's into a system of ODE's by turning the spatial derivatives into various operators in Fourier space. Methods in which the data are computed in Fourier space are called spectral methods; our method is called pseudospectral because the derivatives are computed in Fourier space, but the multiplications are performed in physical space [16]. Let us examine what is the effect of such a calculation on accuracy and performance time. A representative term in our equations is

$$u_1(x,y)\partial_x S_{11}(x,y). \tag{3.1}$$

Both MATLAB and our C++ code use the fftw package to compute the Fourier transforms. This transform is accomplished in $O(n\log(n))$ time. Suppose our two-dimensional domain is discretized with n equally spaced points on a side in physical

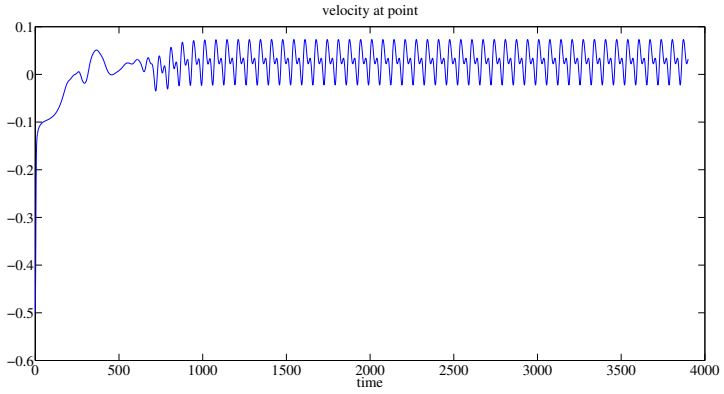

Figure 3.2: Evolution from the identity to develop quasi-periodic behavior for $Wi = 12$.

space, so that the quantities u_1 and S_{11} are each represented by n^2 values. A finite difference approximation of $\partial_x S_{11}(x, y)$ requires $O(n^2)$ calculations - say $2n^2$ for a central difference approximation - and multiplication by u_1 requires n^2 more calculations. If we use a central difference approximation, this calculation has an error of $O(h^2)$. On the other hand, if we compute the above term via

$$(\mathcal{F}^{-1}\hat{u}_1)(\mathcal{F}^{-1}(i\xi\hat{S}_{11})) \tag{3.2}$$

we still need n^2 multiplications for the physical multiplication, n^2 for the multiplication $(i\xi\hat{S}_{11})$, and $O(n\log(n))$ for each of the Fourier transforms. So more floating point operations are involved, but for the same $O(n^2)$ calculation we get an error of $O(e^{-Cn})$. This is called spectral accuracy. Note that it is made possible by the space periodicity (not the time periodicity) of our solutions, as well as by their smoothness, and facilitated by the simplicity of the domain on which our solutions are defined.

3.4 A Fourth or Fifth Order Implicit Explicit Runge Kutta Scheme

We face a dilemma in our choice of a scheme to advance S in time. On one hand, the Laplacian term has a maximum eigenvalue of $2 \times (\frac{N}{2})^2$ where N is the number of spatial grid points on one side, since the eigenvalues are sums of squares of Fourier frequencies. This means that the Laplacian term is stiff, and explicit numerical schemes work badly on stiff problems. On the other hand, the nonlinear terms in the PDE for S make an implicit scheme hard or impossible to implement. So we use an implicit-explicit or IMEX Runge Kutta scheme [13], which works by splitting the PDE into a linear and a nonlinear part, and treating them differently.

Let us write

$$\frac{\partial S}{\partial t} = f(S) + g(S) \tag{3.3}$$

where

$$f(S,u) = -u \cdot \nabla S + (\nabla u S + S \nabla u^T) - \frac{1}{Wi}(S - I) \tag{3.4}$$

and

$$g(S) = \nu \triangle S. \tag{3.5}$$

The implicit part of the scheme handles the stiff linear part of the PDE; the explicit part handles the non-linear part. (Note that $\frac{1}{Wi}(S-I)$ has been included in the nonlinear part of the operator; it is not stiff so we might as well keep it there.) The result is a scheme that is faster and easier to implement than a purely implicit method, and more robust than a purely explicit one. As in Chapter 2, the equations for u and p are treated as constraints at each time step.

The scheme is as follows:

$$\begin{aligned} l_1 &= g(S_n + h l_1 A_{11}) \\ k_1 &= f(S_n + h l_1 A_{11}) \end{aligned} \tag{3.6}$$

and for i from 2 to s

$$\begin{aligned} l_i &= g(S_n + h\sum_{j=1}^{i-1} A_{ij} l_j + h\sum_{j=1}^{i-1} \hat{A}_{ij} k_j + h A_{ii} l_i) \\ k_i &= f(S_n + h\sum_{j=1}^{i-1} A_{ij} l_j + h\sum_{j=1}^{i-1} \hat{A}_{ij} k_j + h A_{ii} l_i) \\ S_{n+1} &= S_n + h\vec{b} \cdot \vec{l} + h\vec{\hat{b}} \cdot \vec{k} \end{aligned} \tag{3.7}$$

$$S_{n+1} = S_n + h\vec{b} \cdot (\vec{l} + \vec{k}). \tag{3.8}$$

At each stage, we calculate l first, using the linearity of g, and calculate k using this latest l and all the previous stages. When all s $l's$ and $k's$ have been computed, a weighted average of them is used to update S. We use Kennedy and Carpenter's 4^{th} order scheme ARK4(3)6L[2]SA, which has $s = 6$ stages (see [13]), and a variant of their 5th order scheme ARK5(4)8L[2]SA, which was modified to obtain a dense output of order 4 (see [5], [26]).

Figure (3.4) shows a plot of the logarithm of the absolute error at ($x = 2\pi$, $y = 2\pi$), versus the logarithm of the step size in time, for the second component of S with $Wi = 14$, final time $t = 0.1$, and initial condition $S_0 = \left(\begin{smallmatrix} 1+0.1\sin x \sin y & 0 \\ 0 & 1+0.1\sin x \sin y \end{smallmatrix}\right)$.

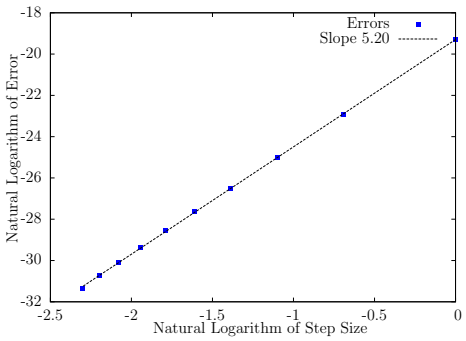

Figure 3.3: Fifth order convergence of the IMEX scheme with final time 0.1.

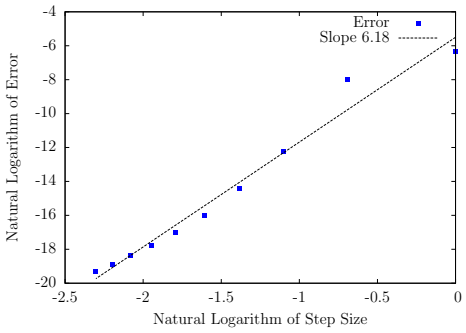

Figure 3.4: Slightly unruly convergence of the IMEX scheme due to instability for S with high frequency modes.

It is necessary to use a simple function like $1 + 0.1 \sin x \sin y$ to illustrate the scheme's convergence. Figure (3.4) shows why. The final time is still 0.1, but instead of the simple function, we use the periodic solution we found with $Wi = 14$. The high frequency modes in this solution make the convergence unruly. It bounces around for higher step sizes, and then suddenly appears to converge at a rate more rapid than fifth order. This is probably because there is significant amplification of error even for step size 0.05, and by the time this instability has been pacified, the truncation error has already almost disappeared compared to the roundoff error.

3.5 Computing the Adjoint Solution in Backwards Time

Advancing a solution via the adjoint equation is more challenging than via the original forward equation because the terms in the adjoint equation at time s involve the solution to the forward equation at time $t = T - s$. This complicates the code for two reasons. One is that the forward and adjoint computations cannot be done simultaneously, since forward values near $t = T$ are needed to compute adjoint values near $s = 0$. The forward data must be either stored, which takes memory space, or recomputed, which takes time. We handle this situation by defining the time grid for the forward equation as a coarse grid of N major time steps with d minor time steps between each major one. We store the forward data at the N+1 major grid values between 0 and T, and recompute the values on the minor grid at each step as they are needed to advance the adjoint solution. This procedure is illustrated in Figure (3.5).

The other problem is that, to advance the adjoint equation from time s to time $s + dt$ with a Runge-Kutta scheme, we need, not just the data from the forward equation at times $T - s$ and $T - s - dt$, but at several other times between those two values, for which we have not computed the values. In the fourth order method, we obtain approximations for these using a Hermite interpolation of third-order accuracy which is sufficient for fourth order accuracy in the IMEX-RK scheme, given by

$$\begin{aligned} f(t + \theta dt) \approx\ & (1 - \theta) f(t) + \theta f(t + dt) \\ & - \theta(1 - \theta)\Big[(1 - 2\theta)(f(t + dt) - f(t)) - (1 - \theta) dt \partial_t f(t) + \theta dt \partial_t f(t + dt)\Big] \end{aligned} \tag{3.9}$$

where $0 < \theta < 1$. In the fifth order method, we use the fourth order dense output formula presented by Wilkening in [26] to interpolate the solution at those intermediate times.

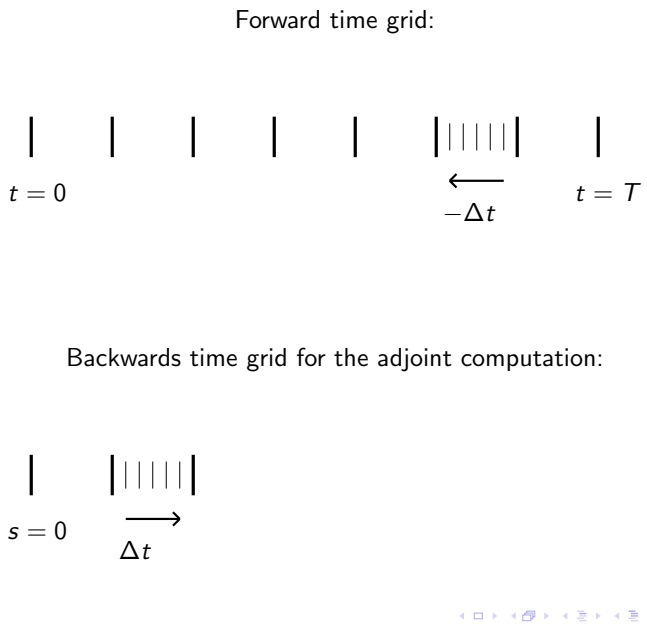

Figure 3.5: We compute S_T with N major grid steps and d minor ones, and store only the values at the major grid points. To compute the adjoint values in backwards time, we recompute the values of the forward equation between the major grid points.

3.6 BFGS minimizer

Now we will discuss the Broyden-Fletcher-Goldfarb-Shanno (BFGS) optimization algorithm, whose theory and implementation are described in Nocedal [15].

Consider a function $G(x)$ that we want to minimize. (In our case $x = [a_{mn}, T]$). As with all optimization methods, we make a starting guess x_0 and refine it via

$$x_{k+1} = x_k + p \tag{3.10}$$

until $G(x_k)$ is small enough to suit us. All of the content of the method resides in how p is chosen at each step.

The BFGS algorithm is the most popular quasi-Newton line search algorithm. It is a line search algorithm because it computes the direction of p first and afterwards how far to walk along it. The simplest and most natural line search algorithm is to model $G(x)$ by a first order Taylor expansion

$$G(x_k + p) \approx m_k(p) := G(x_k) + \nabla G(x_k)^T p \tag{3.11}$$

which yields the search direction

$$p = -\nabla G(x_k) \tag{3.12}$$

at every step. A more accurate and more expensive refinement is to use the model

$$m_k(p) := G(x_k) + \nabla G(x_k)^T p + \frac{1}{2} p^T D^2 G_k p, \tag{3.13}$$

yielding the search direction

$$p = -(\nabla^2 G(x_k))^{-1} \nabla G(x_k) \tag{3.14}$$

at each step. But if we choose the model

$$m_k(p) := G(x_k + p) \approx G(x_k) + \nabla G(x_k)^T p + \frac{1}{2} p^T B_k p, \tag{3.15}$$

in which the Hessian matrix $D^2 G$ is replaced by some approximation B that is faster and easier to compute, we can devise a model that is more accurate than the first-order model, and less unwieldy than a second-order one involving the expensive computation of the Hessian. Thus, it is called a quasi-Newton method.

We have to find a suitable B_k at each step, and also a suitable step size α_k so that

$$x_{k+1} = x_k + \alpha_k p_k. \tag{3.16}$$

In this discussion we use the terms B and B_k interchangeably. First, we require B to be a symmetric, positive definite matrix, to mimic the behavior of the Hessian matrix of a smooth function at a local minimum, and because it simplifies calculations and guarantees the existence of a solution. Using the notation $G_k = G(x_k)$, the direction of steepest descent of the model m is

$$p_k = -B_k^{-1} \nabla G_k. \tag{3.17}$$

B's positive-definiteness guarantees both that its inverse exists and that this choice of p is a descent direction for the real function G (as is seen by taking the inner product with ∇G on both sides of the equation).

Instead of computing a completely new B_k at each step, we would like to update it iteratively by combining former values of B with the information gained at the latest step. A natural requirement is that B_{k+1} be chosen so that the gradient of the model m matches that of G at steps k and $k+1$. This will be the case if

$$B_{k+1} \alpha_k p_k = \nabla G_{k+1} - \nabla G_k. \tag{3.18}$$

This condition also requires the curvature of m_{k+1} to equal the average curvature of G over the line between x_k and x_{k+1}.

Since B is a symmetric, positive definite matrix, (3.18) will not have a solution unless

$$(\alpha_k p_k)^T (\nabla G_{k+1} - \nabla G_k) > 0. \tag{3.19}$$

This condition is satisfied, as can be seen with a little algebra, if we require α to satisfy the Wolfe conditions

$$G(x_k + \alpha_k p_k) \leq G(x_k) + c_1 \alpha_k \nabla G_k^T p \tag{3.20}$$

$$\nabla G(x_k + \alpha_k p_k)^T p_k \geq c_2 \nabla G_k^T p_k \tag{3.21}$$

where $0 < c_1 < c_2 < 1$. Equations (3.20) and (3.21) are called, respectively, the condition of sufficient decrease and the curvature condition. They ensure that G is reduced enough at each step, and also that it is not decreasing too much at the new point, which would indicate that we could do better to move even farther in that direction. Notice that, if G is convex, c_1 needs to be less than 1, or else (3.20) cannot be satisfied.

The $n \times n$ symmetric matrix B has $\frac{n(n+1)}{2}$ degrees of freedom, but only n constraints are imposed upon it by condition (3.18). The BFGS algorithm determines B_{k+1} by defining its inverse H_{k+1}, and requiring that H_{k+1} satisfies

$$H_{k+1} = min||H - H_k|| \tag{3.22}$$

over all symmetric positive definite matrices H satisfying the underdetermined system (3.18). Here $||\cdot||$ is the weighted Frobenius norm (see Nocedal, [15]). The unique H_{k+1} that meets these requirements is given by the relation

$$H_{k+1} = (I - \rho_k y_k s_k^T) H_k (I - \rho_k y_k s_k^T) + \rho_k s_k s_k^T \tag{3.23}$$

where $s_k = \alpha_k p_k$, $y_k = \nabla G_{k+1} - \nabla G_k$, and $\rho_k = s_k^T y_k$. By rewriting this as

$$H_{k+1} = H_k - \rho_k H_k y_k s_k^T - \rho_k s_k y_k^T H_k + \rho_k^2 s_k y_k^T H_k y_k s_k^T \tag{3.24}$$

one can see readily that it is a rank 2 update: the second term sends any input vector to a multiple of $H - k y_k$; the last two send it to a multiple of s_k.

The matrices H_k are generally dense, and therefore expensive to compute and store. We can improve the algorithm by exploiting the fact that it is built up by outer products of vectors. One can think of the BFGS update (3.23) as the construction of a matrix H_{k+1} from all previous values of s_j and y_j: in the *limited memory* BFGS (L-BFGS) algorithm, we construct it from *some* of the previous s_j and y_j's

$$s_{k-1}, y_{k-1}, ...s_{k-m}, y_{k-m} \tag{3.25}$$

for some value m; earlier pairs are discarded as new ones are added. So instead of storing the entire dense matrix H_k, we choose an initial H_k^0 - often as a multiple of the identity matrix - and update it by inner products of the last m steps and gradient differences. This update costs $O(n)$ calculations, where n is the size of the problem. In our program, we used $m = 200$ pairs of vectors.

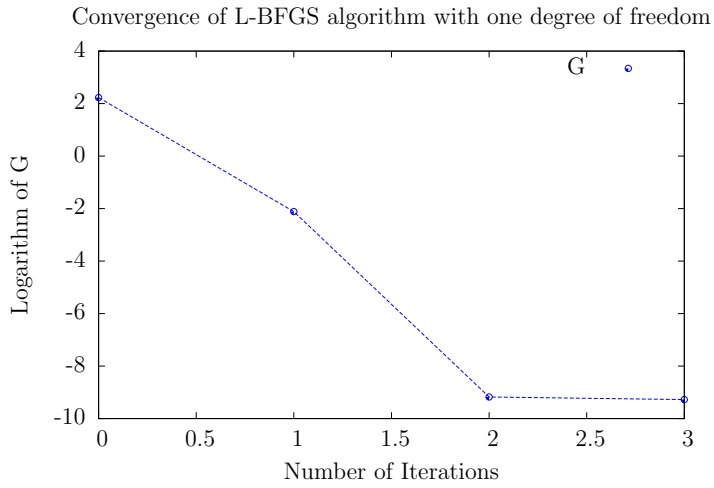

Figure 3.6: The convergence is superlinear when only T is allowed to vary.

Figure (3.6) shows a typical convergence of the L-BFGS algorithm when only the period T is allowed to vary. The convergence is superlinear but usually the final value will only be around 10^{-6} or 10^{-10}. So our procedure is to vary T until this value is achieved and use the period obtained, along with S_0, as the starting guess for the minimization using the full gradient of G as developed in section (2.3).

Figure (3.7) shows the decrease of the logarithm of G over 45 iterations of the L-BFGS algorithm when S_0 and T are both allowed to vary. The shape of the graph is very typical of all our use of the algorithm: the first approximation of the Hessian is a poor one, so the first update of G is several orders of magnitude larger than the starting value. The rest of the decrease is characterized by long plateaus of practically no decrease, interspersed with rapid plunges of one or several orders of magnitude, especially at the very end. What is happening is that the quasi-quadratic model becomes more and more accurate as the updates to H_k accumulate; eventually the approximate Hessian is so accurate that it finds a very good search direction, resulting in a plunge; then it is in a new region of $[S_0, T]$ space for which a new Hessian approximation is needed, so it plateaus for awhile as the new Hessian accumulates, and so on. Sometimes as many as 300 iterations are needed.

3.7 Fourier Representations

There is a naturalness in using wave amplitudes as the variables to represent the motion of a fluid, as this is, in some sense, the natural language of the fluid. More importantly, it is practical, because many calculations are simpler and more accurate when performed in Fourier space, like finding the components of the adjoint function

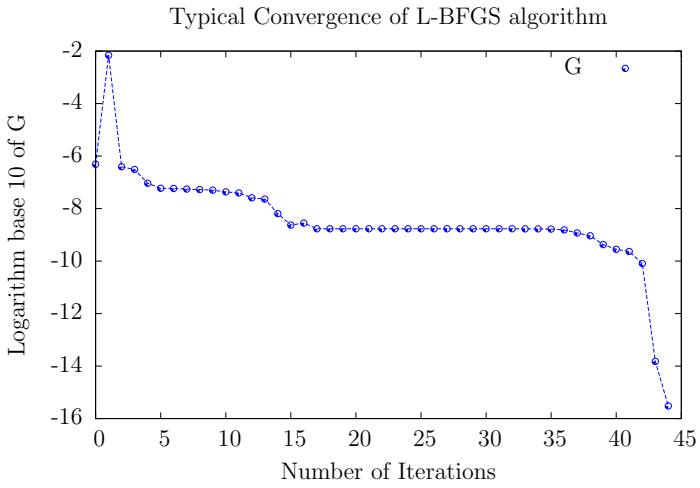

Figure 3.7: The convergence is slower when the full gradient for G is used, with many plateaus during which the algorithm accumulates its approximation of the Hessian matrix.

in section (2.3). We also take advantage of the Fourier representation when doing integrals: if a function $f(x, y)$ has the Fourier representation

$$f(x,y) = \sum_{(m,n)\in\Lambda} \hat{f}_{mn} e^{imx} e^{iny}, \tag{3.26}$$

then

$$\int_0^{2\pi}\int_0^{2\pi} f(x,y) = 4\pi^2 \sum_{(m,n)\in\Lambda} \hat{f}_{mn} \tag{3.27}$$

We can thus perform all the integrations in calculating G and ∇G with spectral accuracy, as opposed to the polynomial accuracy of integration by finite volume approximation, and less computation time than Gaussian quadrature and other approximation methods in physical space, by simply adding up their Fourier coefficients. Again, this is a feature of any orthogonal basis representation whose basis functions are easy to integrate.

It is slightly awkward to store the variables a_{mn} from equation (2.7). To illustrate why, consider the discrete Fourier transform of a 1-dimensional array with n entries, where n is even. Such a transform represents the physical data as

$$\sum_{k=0}^{\frac{n}{2}} a_k e^{ikx} + \sum_{k=-\frac{n}{2}+1}^{-1} a_k e^{ikx}, \tag{3.28}$$

and stores it as

$$[a_0, a_1, ...a_{n/2}, a_{-\frac{n}{2}+1}, a_{-\frac{n}{2}+2}, ...a_{-1}]. \tag{3.29}$$

This array is easy to store and manipulate; in particular, we can take advantage of the fact that, in a real-valued function, we have

$$a_n = \bar{a}_{-n} \tag{3.30}$$

by storing only the left half of the vector. ($\frac{n}{2}$ is called the Nyquist frequency, and unlike the other nonconstant modes it has no conjugate partner, making it a nuisance to work with, since it is not clear whether to define it as belonging to $e^{\frac{inx}{2}}$ or $e^{\frac{-inx}{2}}$. In our code, as is standard in this situation, we simply zero it out.) For a two-dimensional Fourier representation, we similarly have the Fourier coefficients

$$a_{mn} = \bar{a}_{(-m,-n)}, \tag{3.31}$$

but there is no longer an obvious way to order them. One sensible approach, which we adopt, is to store the modes that lie in the right half-plane minus $\{(0, -\mathbb{N})\}$, spiraling out clockwise from $(0, 0)$: see Figure (3.8). In addition to being simple to remember and understand, this ordering has the advantage of making it easy to change the mesh size because new modes get added onto the end instead of being interspersed with old ones, as they would be if we counted them row by row or column by column.

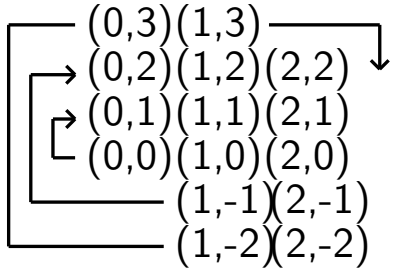

Figure 3.8: A spiral ordering of two-dimensional Fourier modes.

A drawback of spectral and pseudospectral methods is that errors develop in the higher part of the spectrum, since high frequency oscillations have large derivatives that can amplify roundoff error. This is a problem because many of a function's delicate features, for instance its singularities, are characterized by the higher part of its Fourier spectrum. A common method of suppressing this error is called the $\frac{2}{3}$ method: one simply removes the highest one third of the frequency spectrum after each time step. This is simple to implement, but errors are spread out over the solution in space instead of concentrating near singularities, and the abrupt cutoff

in the spectrum can also cause the Gibbs phenomenon [10]. A better method is the so-called 36 rule, in which we multiply the k^{th} mode by

$$\rho(k/N) = e^{-36(\frac{|k|}{N})^{36}} \tag{3.32}$$

where N is the highest mode number (one half the grid size). The value of ρ is very close to 1 until $\frac{|k|}{N}$ is about 78 percent, whereupon it plunges rapidly but smoothly towards zero. Thus we preserve a few more modes than in the $\frac{2}{3}$ method, and the smoothness of the cutoff discourages bad behavior near the boundaries.

3.8 Software

Most of the code was written in the C++ computer language and run on the computer called apps, a 12-core intel Xeon 5680 westmere rackmount server with 24 GB of RAM and a clock speed of 3.33 GHz, running ubuntu linux version 11.10 and using the intel composer XE compiler version 12.1 and the math kernel library; this computer is located in Earl Warren Hall on the corner of Hearst Avenue and Oxford Street in Berkeley, California. Some backup code and much of the data processing was written in MATLAB. The eigenvalues and eigenvectors of the linearized equations were computed using ARPACK, a collection of Fortran77 subroutines for solving large scale eigenvalue problems using the Implicitly Restarted Arnoldi Method (IRAM) [4]. The Fourier transforms were computed using the fftw package. We parallelized the Fourier transform by computing different rows of the same array on different processors, and transforming multiple functions simultaneously when possible.

Chapter 4

Results

4.1 Overview of Results

We have found two families of periodic solutions and two families of stationary solutions, with Weissenberg values ranging between $Wi = 5.0$ and $Wi = 18.0$. The first family of periodic solutions connects to one of the stationary families at the value $Wi \approx 7.5$. All of the stationary solutions are determined to be unstable using the method described in section (2.4), so there are potentially many other families of periodic solutions emerging from them.

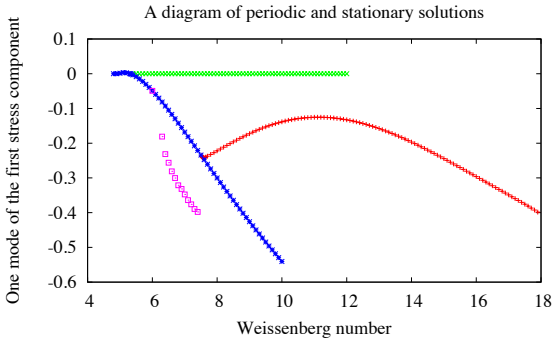

Figure 4.1: A global plot of some periodic and stationary solutions. The red and magenta dots represent periodic solutions, and the green and blue dots represent stationary states.

Figure (4.1) shows a global map of the stationary and periodic solutions we discovered. The horizontal axis shows the value of Wi, and the vertical axis shows the value of the real part of the (1,0) mode of S_{11} at the moment when the trajectory of the (1,1) mode crosses its axis of symmetry. (It does this twice over one period; we made an arbitrary conventional choice of which one to pick.)

Figure (4.2) shows the trajectories of one of the Fourier modes of the first stress component S_{11} of the periodic solutions in the red family as Wi increases from 7 to 16.

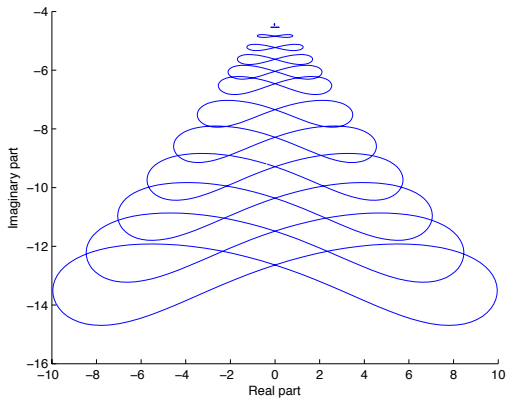

Figure 4.2: The amplitude of e^{iy} in S_{11} for Wi from 16.0 down to 7.0.

The shapes are similar and the amplitude shrinks as Wi decreases. This illustrates the manner in which this family of periodic solutions connects with the family of stationary ones: the oscillation shrinks in amplitude as we approach the bifurcation until, at the bifurcation, the oscillation disappears.

4.2 Two Families of Stationary Solutions

The two families of stationary solutions in Figure (4.1) appear to connect to one another, but there is great sensitivity near the bifurcation: if the value of Wi changes by as little as 0.05, one obtains stationary solutions that do not appear to belong to either family.

The two families have different characters. The solutions in the family represented by the horizontal line of green dots in Figure (4.1) appear to be symmetrical; the ones in the family represented by the blue dots have stronger vorticity in the upper right quadrant.

For Wi=5.5 near the putative bifurcation, the flows are similar, although the asymmetry is starting to be apparent in the second one; see Figure (4.3).

Far from the bifurcation, at $Wi = 10$, the asymmetry in the solution from the second family is more apparent; see Figure (4.4).

The asymmetry of the second family leads to very different mixing properties than for the symmetrical flows. Note that, in this context, to be a stationary state does not mean that there is no flow: it means that the values of the stress tensor and the velocity field of the flow remain constant in time. Figures (4.5) and (4.6) show a

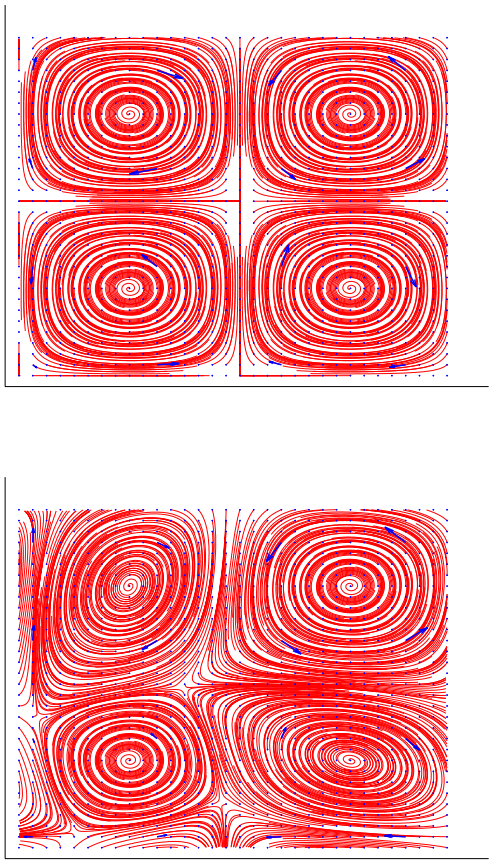

Figure 4.3: Streamline plots of two stationary states with $Wi = 5.5$ on $[0, 2\pi]^2$.

particle tracker simulation for, respectively, the symmetric and asymmetric velocity fields pictured in Figure (4.4). At time $t = 0$, the particles are evenly spaced as in the left-hand figure. As time passes, one sees how this initial configuration is mixed by the flow. This was achieved by requiring each particle's position (x, y) to evolve according to $(x(t), y(t)) = u(x(t), y(t))$, where u is the velocity field. The particle positions are updated using the same IMEX-RK method as was used to update S and u, with the velocity between grid points obtained by a fourth-order spatial interpolation. The symmetric flow shows no mixing between quadrants, even after 300 units of time. The asymmetric flow preserves most of the matter in the area of strong vorticity, with smaller areas from the off-diagonal quadrants remaining intact but sheared and distorted by the flow.

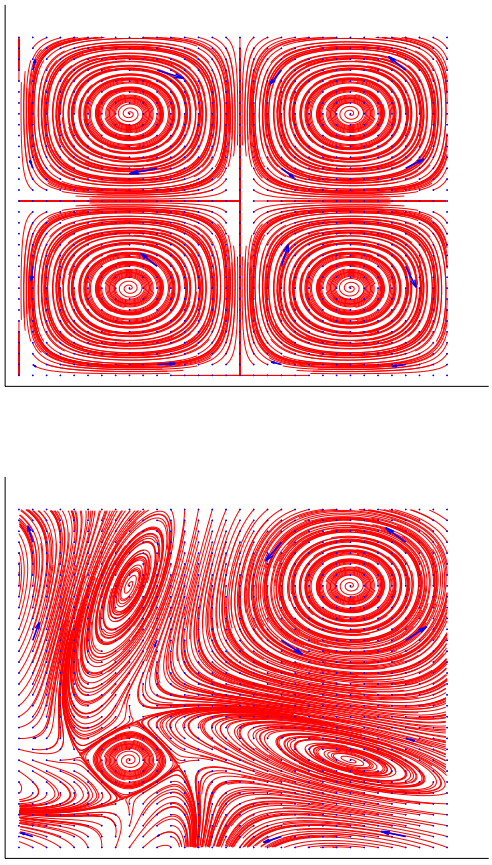

Figure 4.4: Streamline plots of two stationary states with $Wi = 10$ on $[0, 2\pi]^2$..

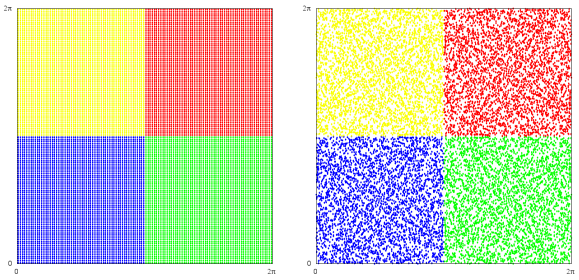

Figure 4.5: Particle mixture for a symmetrical stationary flow on $[0, 2\pi]^2$ with $Wi = 10$, at times t=0 and t=300.

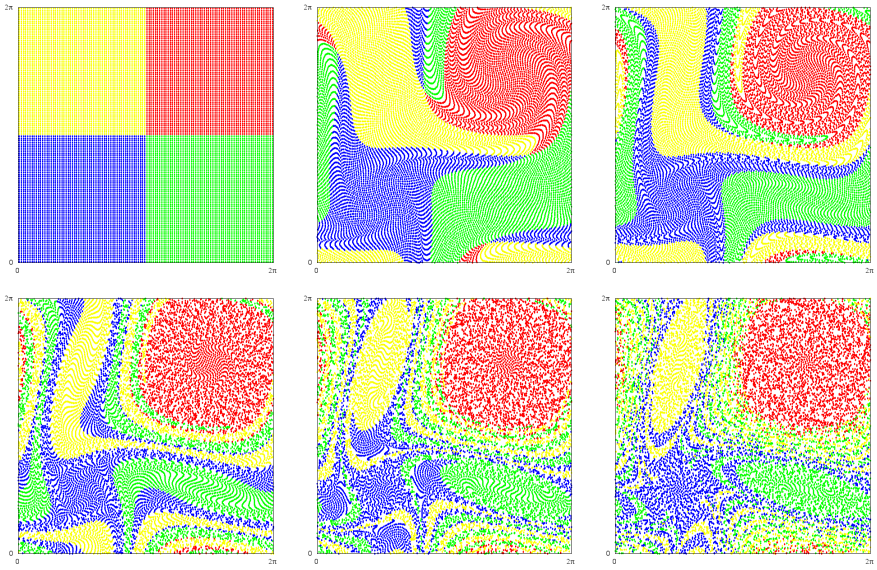

Figure 4.6: Particle mixture for an asymmetrical stationary flow on $[0, 2\pi]^2$ with Wi = 10 at times 0, 6, 12, 25, 50, and 100.

The asymmetrical stationary states all have a hidden symmetry: one can check numerically that $S(x, y, t) = S(\pi - x, \pi - y, t)$ for all of the ones that we found.

Since the original problem is symmetrical, it is likely that there exist other families of asymmetrical solutions with strong vorticity in the other three quadrants. Note that asymmetry is a known phenomenon in viscoelastic flows, thought to be caused by the polymers in the fluid aligning with the flow.

4.3 Periodic Solutions with Higher Weissenberg values

The first family of periodic solutions are those in which the value of Wi ranges between 7.5 and 18.0. It is represented by the red points in Figure (4.1).

In Figure (4.7), the periods of some of this family of solutions are plotted versus their Weissenberg numbers. The periods of the solutions get higher as the value of Wi increases, ranging from 42 units of time for $Wi = 7$ to 106 units of time for $Wi = 17$ (see Figure 4.7). As Wi increases, the period increases in a smooth and monotonic manner.

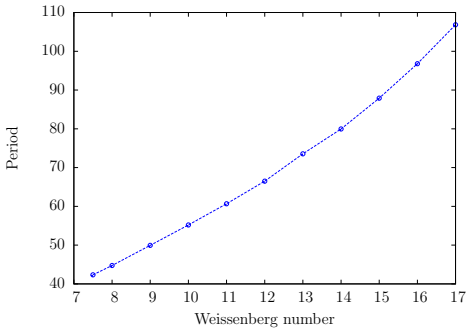

Figure 4.7: A plot of the period T versus the Weissenberg number in a family of periodic solutions with high Weissenberg values.

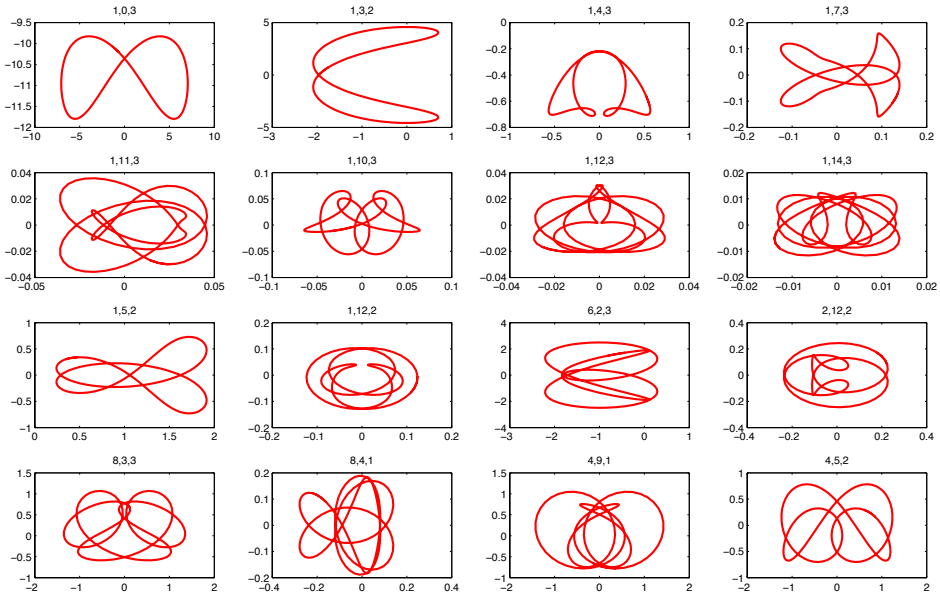

Figure 4.8: Trajectories in the complex plane of some of the Fourier modes of S for Wi=14. The label (m,n,k) indicates the $(m,n)^{th}$ mode of the k^{th} component of S.

Shown in Figure (4.8) are some of the trajectories of the lower Fourier coefficients of a stationary solution with $Wi = 14$. The trajectory of the coefficient of the $(m,n)^{th}$ Fourier mode is symmetric about the real axis if $(m+n)$ is even, and symmetric about the imaginary axis if $(m+n)$ is odd. In fact, for the solutions with Wi equal to 14 or higher, it is the case that if $a_{mn}(t)$ is the coefficient of $e^{imx+iny}$, then

$$a_{mn}(t+\frac{T}{2}) = (-1)^{m+n}\bar{a}_{mn}(t) \tag{4.1}$$

This actually reveals a simple spatial symmetry: plugging the above into our Fourier representation for S, we see that

$$\begin{aligned} S(x,y,t+\frac{T}{2}) &= \sum_{m,n\in\Lambda} [\overline{a}_{mn}(t)(-1)^{m+n}e^{i(mx+ny)} + a_{mn}(t)(-1)^{m+n}e^{-i(mx+ny)}] \\ &= \sum_{m,n\in\Lambda} [\overline{a}_{mn}(t)e^{-i\pi(m+n)}e^{i(mx+ny)} + a_{mn}(t)e^{i\pi(m+n)}e^{-i(mx+ny)}] \\ &= \sum_{m,n\in\Lambda} [\overline{a}_{mn}(t)e^{-i(m(\pi-x)+n(\pi-y))} + a_{mn}(t)e^{i\pi(m+n)}e^{i(m(\pi-x)+n(\pi-y))}] \\ &= S(\pi-x,\pi-y,t) \end{aligned} \tag{4.2}$$

It is easy to show that, if $[S(x,y,t), u(x,y,t), p(x,y,t)]$ solves the Stokes Oldroyd-B system with our choice of f, then $[S(\pi-x,\pi-y,t), -u(\pi-x,\pi-y,t), p(\pi-x,\pi-y,t)]$ solves the same system with f replaced by -f. However, this does not predict that this transformation will be achieved every half-period, as is the case with these solutions!

Between Weissenberg values of 13 and 14, this symmetry breaks down. The trajectories of the Fourier coefficients in the complex plane still have bilateral symmetry, but the moment at which the coefficient achieves its own mirror image no longer happens after one half period, but varies in time from mode to mode.

A helpful way to picture the periodic motion of these solutions is to look at the vector field of the velocity's deviation from an average value over time. In Figure (4.9), we plot the average velocity of the periodic solution with Weissenberg value 14.0, computed over one period. It is not surprising that it exhibits the asymmetry characteristic of the second family of stationary solutions, since it is from that family that this family of periodic solutions emerges.

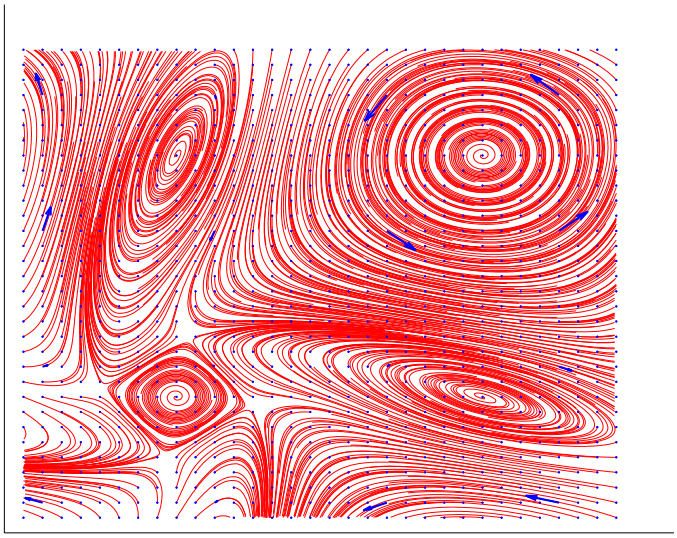

Figure 4.9: Average velocity over one time period for $Wi = 14.0$ on $[0, 2\pi]^2$.

In Figures (4.10) and (4.11), the deviation from the average flow value is shown at four equal time intervals over one period for the solution with $Wi = 14$; the progress is along the rows. The maximum deviation is about 10 percent of the average value. Small vortices appear and disappear. It appears that the flow at a point in time is close to the opposite flow half a period later; however, a visual inspection reveals that this symmetry is not exact. It is possible that it would become exact for a driving force of smaller amplitude.

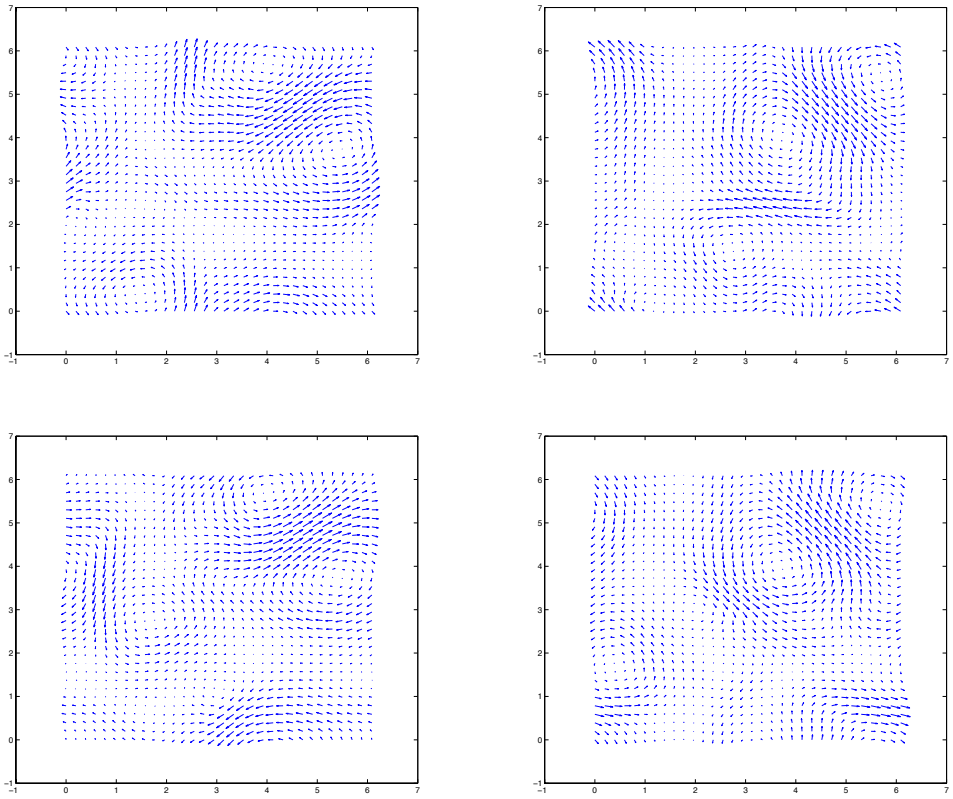

Figure 4.10: A vector field plot of the deviation from average velocity of a solution with $Wi = 14.0$ on $[0, 2\pi]^2$ at equal time intervals over one period. The progress is read along the rows.

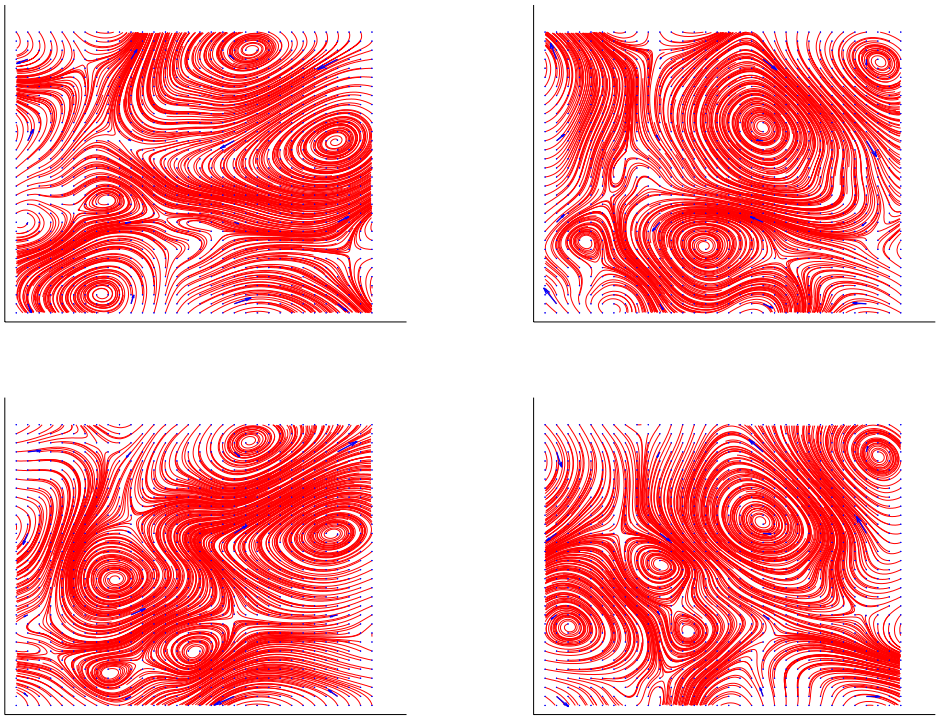

Figure 4.11: Deviation from average velocity for $Wi = 14.0$ on $[0, 2\pi]^2$ at equal time intervals over one period, pictured as streamlines.

4.4 Periodic Solutions with Lower Weissenberg values

The second family of periodic solutions is represented by the magenta points in Figure (4.1). Note that one magenta point overlaps with the asymmetrical family of stationary solutions. It seems likely that the periodic solutions in this family connect with this stationary solution as the value of Wi decreases from 6.3 to 6.0. However, the solutions in this regime are so sensitive that finding periodic behavior is difficult. If one decreases the value of Wi from 6.3 by as little as 0.025, then the new solution experiences very light damping as it evolves, decreasing in amplitude over tens of thousands of time units without achieving quasiperiodic behavior or becoming stationary.

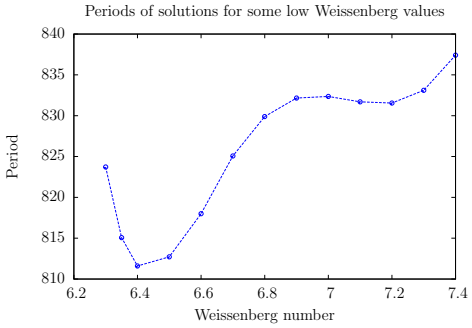

Figure 4.12: A plot of the period T versus the Weissenberg number in a family of periodic solutions with low Weissenberg value.

The solutions in the second family have periods of more than 800 units of time, and their variation versus the Weissenberg number is more complex than those of the first periodic family. See Figure (4.12).

In these solutions, we see a special case of the symmetry exhibited by the higher family of solutions, in which the Fourier coefficient a_{mn} is purely real if the sum $(m+n)$ is even, and purely imaginary if it is odd. Thus we cannot make plots in the complex plane as in Figure (4.8).

Figure (4.13) shows a plot of the variation, over one period, of the constant modes of the first and third stress components versus one another. It appears that the value of one is the value of the other shifted by half a period, which seems to suggest a symmetry between S_{11} and S_{22}. However, this relation does not hold for other Fourier modes of these two stress components.

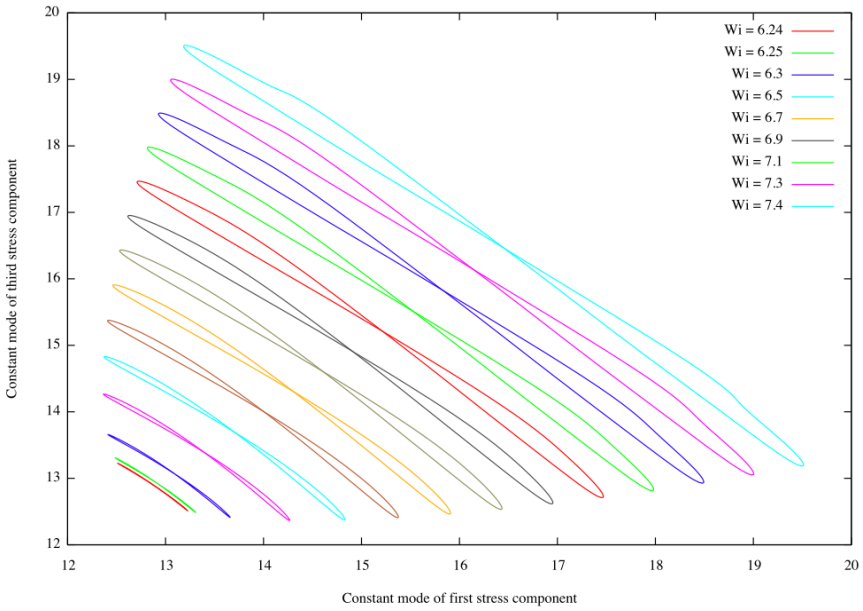

Figure 4.13: Symmetry in the constant mode of the first versus the third stress components for some low Weissenberg values.

Figure (4.14) shows the average velocity, computed over one period, of the solution with $Wi = 6.3$.

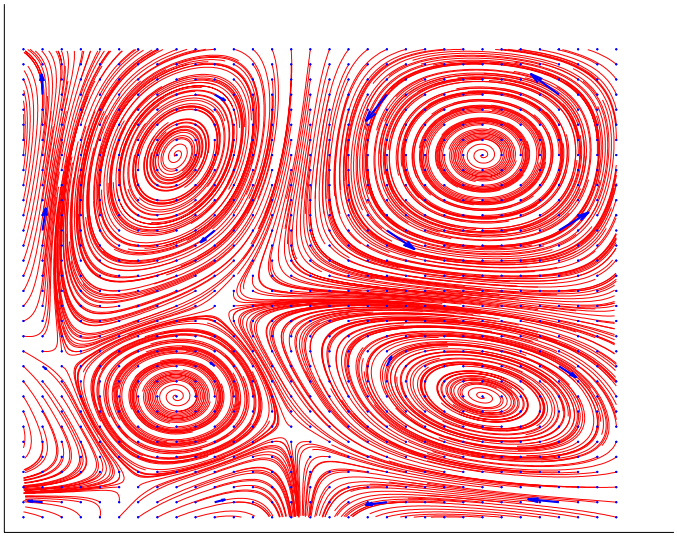

Figure 4.14: Average velocity over one time period for $Wi = 6.3$ on $[0, 2\pi]^2$, pictured as streamlines.

Figures (4.15) and (4.16) show the deviation from the average velocity of the solution with $Wi = 6.3$, once again read along the rows over equal times intervals of one quarter period. The motion is noticeably different than in the first family of solutions. As mentioned above, it happens over a longer time period, and rather than having several small vortices that appear to travel around over time, there are only two at fixed points. The general sense is more of a back and forth motion, as compared with the roiling motion suggested in Figure (4.10).

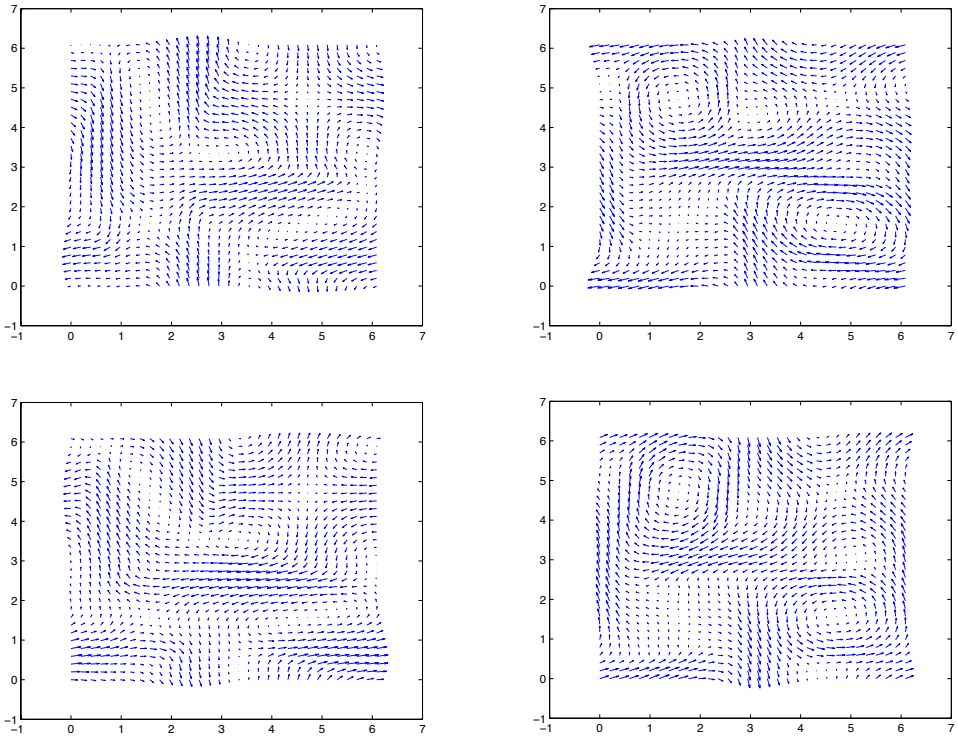

Figure 4.15: A vector field plot of the deviation from average velocity for $Wi = 6.3$ on $[0, 2\pi]^2$ at equal time intervals over one period. The progress is read along the rows.

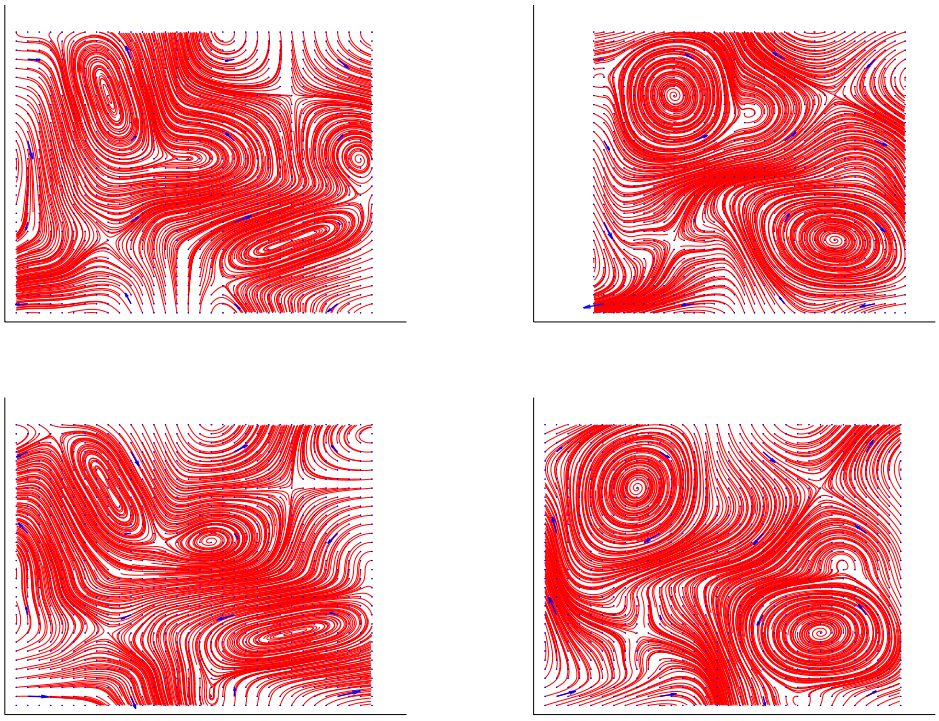

Figure 4.16: Deviation from average velocity for $Wi = 6.3$ on $[0, 2\pi]^2$ at equal time intervals over one period, pictured as streamlines.

4.5 Solution Stability

Figures (4.17) and (4.18) show respectively some eigenvalues in the complex plane of some of the stable and periodic solutions we computed using the methods discussed in section (2.4). For each solution, the 11 eigenvalues of highest magnitude are plotted.

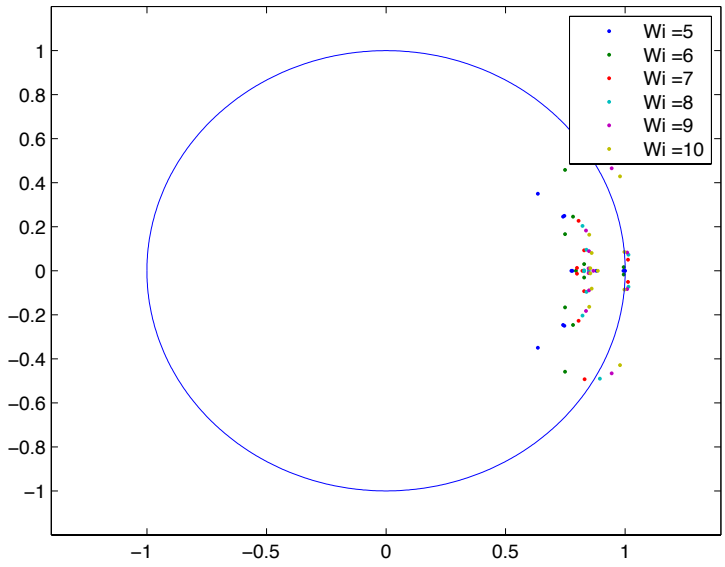

Figure 4.17: Largest eigenvalues of some stationary solutions. The unit circle is shown for reference.

Figure (4.18) indicates that the periodic solutions with Weissenberg values of 9 or lower have an eigenvalue outside the unit circle and are therefore unstable; those with Weissenberg values of 10 or higher have no eigenvalues outside the unit circle and are therefore stable. As discussed in section (2.4), all the periodic solutions have one eigenvalue equal to 1.

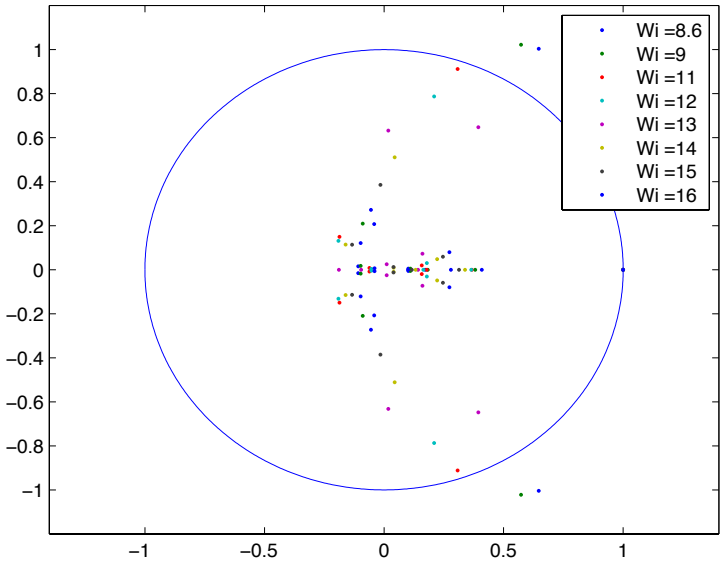

Figure 4.18: Largest eigenvalues of some periodic solutions. The unit circle is shown for reference. Those solutions with an eigenvalue outside the unit circle are unstable.

We can check that the magnitude of these eigenvalues reflects the stability of their associated solutions by evolving the solutions for a long time to see what happens to them. Figure (4.19) shows a plot of one mode of S for $Wi = 14$ over 200 periods. The largest nonunit eigenvalue of this solution has a magnitude of 0.5, and after 200 periods, the value of G is still 5×10^{-16}.

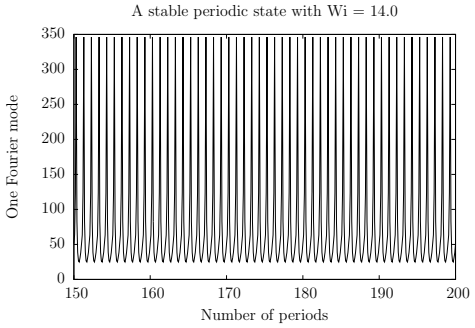

Figure 4.19: The evolution of a stable periodic state with $Wi = 14.0$ over 200 periods.

On the other hand, Figure (4.20) shows the instability that eventually develops in the stationary solution with $Wi = 16.2$, whose largest eigenvalue has magnitude 1.05. Figure (4.21) shows the emergent quasiperiodic behavior in more detail.

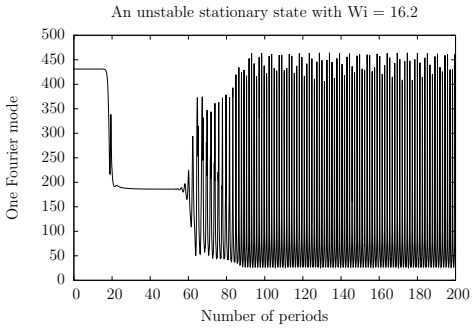

Figure 4.20: The evolution of an unstable stationary state with $Wi = 16.2$ over 200 periods.

Meanwhile, the periodic solution with $Wi = 8.6$ has two eigenvalues with magnitude 1.19. After about 90 periods, it develops a wobble, and then a striking new quasi-periodic motion develops, as pictured in Figure (4.22). This new behavior appears to have a period of about 420 units of time; the BFGS minimizer is able to decrease the value of G to about 6×10^{-5} after 300 iterations; it is possible that it represents a new kind of periodic behavior.

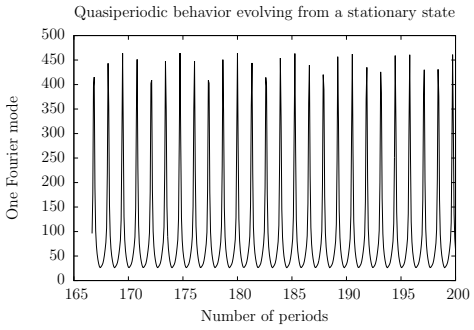

Figure 4.21: Quasi-periodic behavior evolving from the stationary state with $Wi = 16.2$.

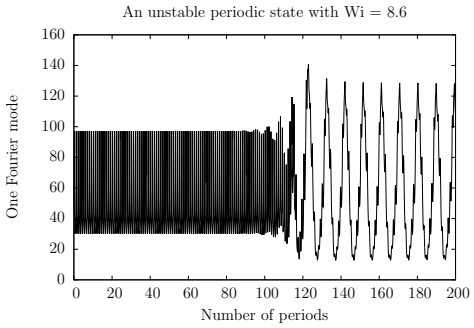

Figure 4.22: Development of new quasi-periodic behavior from an unstable periodic state with $Wi = 8.6$.

4.6 Computation Times

Because the periods of the solutions we found were all so long, the computations in finding these solutions were lengthy. A typical step size in time was approximately 0.06. For example, for a solution with period 70 time units, characteristic of the family described in section (4.3), we used 1200 time steps to advance the solution by one period. On the computer called apps mentioned in section (3.8), this computation takes about one minute.

Typically, a solution was first approximated by evolving an initial approximation (such as a periodic solution with a nearby Weissenberg value) for about 20 periods to allow it to stabilize, taking about 20 minutes.

Next, the value of G for the resulting function was minimized by varying the period only. Typically the size of $\frac{\partial G}{\partial T}$ was brought down to within 10^{-10} in about four iterations. No significant additional time is needed to compute $\frac{\partial G}{\partial T}$ at each step, nor the next guess for T, so this step takes about four minutes and brings the value of G down to between 10^{-6} and 10^{-10}.

The resulting function is now fed into the L-BFGS minimizer with the full value of ∇G computed via the adjoint equation as discussed in section (2.3). Depending on whether the forward S values can be stored at every step, or must be recomputed at some steps as discussed in section (3.5), the adjoint computation takes between 1.5 and 2 times as long as the forward computation; extra time is needed even when all the forward values of S are stored because the forward values of u must still be recomputed, and because the adjoint system has more terms than the forward one. Since this implementation of L-BFGS takes between 45 and 300 iterations to achieve a G value of 10^{-16}, a typical such minimization takes on the order of 8 hours to complete.

In computing the stability of the solution, the forward and linearized equations are calculated simultaneously. To do this over one period takes about 3 minutes. ARPACK typically takes about 60 iterations to find 11 eigenvalues of the linearized system, so this calculation takes about 3 hours. So, all together, to find a periodic solution in this first family and compute its stability took around 12 hours.

The computation of stationary solutions was much faster. We evolved stationary solutions to a final time of $t = 3.5$, so a stationary solution could be approximated and minimized and have its stability calculated in about 6 hours.

On the other hand, the solutions with low Weissenberg values, described in section (4.4), have periods of approximately 10 times the periods of those with high Weissenberg values, so the computations take 10 times as long. The computations for the solutions in this family took almost one week apiece.

4.7 Summary and Conclusion

The periodic solutions we have found to the Stokes Oldroyd-B system exhibit complex behavior not associated with the low Reynolds regime for Newtonian fluid flow. These solutions vary qualitatively for high versus low values of the Weissenberg number. When $Wi > 10$, small traveling vortices develop over periods in the range of 70 to 100 units of time. When Wi is between 6 and 7, low frequency oscillations arise with periods of more than 800 units of time. For Newtonian fluids, periodic orbits are thought to play an important role in the transition to turbulence in Couette flow (see [8] [12], [25]) and pipe flow (see [21], [20]). We may be observing similar transitions to elastic turbulence (see [7], [23]) in the low Reynolds number regime. In addition, many of the flows we discovered have asymmetrical vorticity which leads to interesting mixing properties.

The bifurcation plot in Figure (4.1) suggests a global structure of solutions whose details may emerge in future work. It is likely that other families of solutions exist with strong vorticity in different quadrants, and possible that these solutions emerge from the family of symmetrical stationary solutions that seems to connect to the asymmetrical states that we found, perhaps even at the same value of the Weissenberg number. The quasi-symmetry observed in some solutions, in which the periodic flow's deviation from its average value appears to almost reverse itself every half period in time, may become an exact symmetry for different amplitudes of the 4-roller forcing field. It is also likely that more complex periodic behavior will emerge if the fluid model is subjected to a 16-roller forcing field.

Bibliography

[1] David M. Ambrose and Jon Wilkening. Computation of time-periodic solutions of the Benjamin–Ono equation. *J. Nonlinear Sci.*, pages 277–308, 2010.

[2] Alexandre J. Chorin and Jerrold E. Marsden. *Mathematical Introduction to Fluid Mechanics*. Springer, 1993.

[3] Earl A. Coddington and Norman Levinson. *Theory of Ordinary Differential Equations*. McGraw Hill, 1955.

[4] James Demmel. *Applied Numerical Linear Algebra*, volume 1. Society for Industrial and Applied Mathematics, 3600 University City Science Center, Philadelphia, PA, 1997.

[5] E.Hairer, S.P. Norsett, and G. Wanner. *Solving Ordinary Differential Equations: Nonstiff Problems*, volume 1. Springer, 1987.

[6] Ardith W. El-Kareh and L. Gary Leal. Existence of solutions for all Deborah numbers for a non-Newtonian model modified to include diffusion. *Journal of Non-Newtonian Fluid Mechanics*, 33:257 – 287, 1989.

[7] A. Groisman and V. Steinberg. Elastic turbulence in a polymer solution flow. *Nature*, 405:53–55, 2000.

[8] J. M. Hamilton, J. Kim, and F. Waleffe. Regeneration mechanisms of near-wall turbulence structures. *J. Fluid Mech.*, 287:317–348, 1995.

[9] O. G. Harlen, J.M. Rallison, and M.D. Chilcott. High-Deborah-number flows of dilute polymer solutions. *Journal of Non-Newtonian Fluid Mechanics*, 34:319–349, 1989.

[10] T. Y. Hou and R. Li. Computing nearly singular solutions using pseudo-spectral methods. *J. Comput. Phys.*, 226:379–397, 2007.

[11] Daniel D. Joseph. *Fluid Dynamics of Viscoelastic Liquids*. Springer, 2007.

[12] G. Kawahara and S. Kida. Periodic motion embedded in plane Couette turbulence: regeneration cycle and burst. *J. Fluid Mech.*, 449:291–300, 2001.

[13] C.A. Kennedy and M.H. Carpenter. Additive Runge–Kutta schemes for convection-diffusion-reaction equations. *Applied Numerical Mathematics*, 44:139–181, 2003.

[14] T. Matolcsi and P. Van. Absolute time derivatives. *Journal of Mathematical Physics*, 48, 2007.

[15] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*, volume 1. Springer Science+Business Media, LLC, 233 Spring Street, New York, NY 10013, 2006.

[16] Roger Peyret. *Spectral Methods for Incompressible Viscous Flow*, volume 148. Springer-Verlag New York, 2002.

[17] E.M. Purcell. Life at low Reynolds numbers. *American Journal of Physics*, 45:3–11, 1977.

[18] J.M. Rallison and E.J. Hinch. Do we understand the physics in the constitutive equation? *Journal of Non-Newtonian Fluid Mechanics*, 29:37–55, 1988.

[19] Michael Renardy. *Mathematical Analysis of Viscoelastic Flows*, volume 1. Cambridge University Press, Cambridge, United Kingdom, 2000.

[20] T. M. Schneider, B. Eckhardt, and J. Vollmer. Statistical analysis of coherent structures in transitional pipe flow. *Phys. Rev. E*, 75(2)(6):066313, 2007.

[21] T. M. Schneider, B. Eckhardt, and J. A. Yorke. Turbulence transition and the edge of chaos in pipe flow. *Phys. Rev. Letters*, 99(3):034502, 2007.

[22] Becca Thomases and Michael Shelley. Emergence of singular structures in Oldroyd-B fluids. *American Institute of Physics*, 2007.

[23] Becca Thomases, Michael Shelley, and J.L. Thiffeault. A Stokesian viscoelastic flow: Transition to mixing and oscillations. *Physica D 240*, 20:1602–1614, 2011.

[24] Ferdinand Verhulst. *Nonlinear Differential Equations and Dynamical Systems*. Springer-Verlag Berlin Heidelberg, 1990.

[25] D. Viswanath. Recurrent motions within plane Couette turbulence. *J. Fluid Mech.*, 580:339–358, 2007.

[26] Jon Wilkening. A fifth order additive Runge–Kutta method with a fourth order dense output. *Technical Report, Lawrence Berkeley National Lab*, 2012.