

# Formation of Labyrinth Patterns in Langmuir Films

**George Tucker**

Andrew Bernoff, Advisor

Elizabeth Mann, Reader

May, 2008

**HARVEY MUDD**  
COLLEGE

Department of Mathematics

Copyright © 2008 George Tucker.

The author grants Harvey Mudd College the nonexclusive right to make this work available for noncommercial, educational purposes, provided that this copyright statement appears on the reproduced materials and notice is given that the copying is by permission of the author. To disseminate otherwise or to republish requires written permission from the author.

# Abstract

A Langmuir film is a molecularly thin fluid layer on the surface of a subfluid. When dipole–dipole forces are negligible, bounded films relax to energy-minimizing circular domains. We investigate numerically the case where dipole–dipole interactions are strong enough to deform the domain into highly distorted labyrinth-type patterns. Our numerical method is designed to achieve higher accuracy and better stability than previous work and exploits an analytic formulation that removes a singularity in the dipole–dipole forces without resorting to a small cut-off parameter. We calculate the relaxation rates for a linearly perturbed circular domain, and we verify them numerically. We are also able to numerically reproduce experimentally observed circle to dog-bone transitions with minimal area loss.



# Contents

<b>Abstract</b>	<b>iii</b>
<b>Acknowledgments</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Introduction . . . . .	1
1.2 Langmuir Films . . . . .	2
1.3 Line Tension . . . . .	2
1.4 Dipolar Repulsion . . . . .	3
<b>2 Numerical Simulation</b>	<b>7</b>
2.1 Formulation . . . . .	7
2.2 Stability . . . . .	10
2.3 Tangential Velocity . . . . .	13
2.4 Motion by Curvature . . . . .	14
2.5 Numerical Scheme . . . . .	15
<b>3 Implementation</b>	<b>17</b>
3.1 Differentiation and Integration . . . . .	17
3.2 Filtering the Spectrum . . . . .	17
3.3 Calculating $\Psi_s$ . . . . .	18
3.4 Equal Arc-length Redistribution . . . . .	18
3.5 Maximum Curvature . . . . .	19
3.6 Variable Time Step . . . . .	20
<b>4 Results</b>	<b>21</b>
4.1 Stability and Relaxation for a Perturbed Circular Domain . .	21
4.2 Error Order . . . . .	22
4.3 Transitions . . . . .	23

<b>5 Conclusion</b>	<b>27</b>
5.1 Future Work . . . . .	27
<b>Bibliography</b>	<b>29</b>

# List of Figures

1.1	Phases in a Langmuir film . . . . .	2
1.2	Circle to dog-bone transition . . . . .	5
4.1	Relaxation rates for a mode 2 perturbation . . . . .	22
4.2	Numerical simulation of a circle to dog-bone transition . . .	24
4.3	Numerical simulation of a four fold symmetry perturbation	25
4.4	Numerical simulation of a complex domain . . . . .	26



# Acknowledgments

I would like to thank my thesis advisor Professor Andrew Bernoff for his invaluable guidance on my thesis and throughout my Harvey Mudd career. I would also like to thank Claire Connelly for her tireless efforts to keep all of the mathematics servers running.



# Chapter 1

## Introduction

### 1.1 Introduction

A Langmuir film is a molecularly thin fluid layer on the surface of a subfluid. Such layers are model systems for the classic example of a molecularly thin film, the lipid bi-layer that forms the external membrane of a biological cell. Langmuir layers can have multiple phases, which can evolve a variety of morphologies. If the molecules in a phase have a strong vertical dipole moment (often a result of hydrophilic/hydrophobic chains orienting themselves on the subfluid), exotic morphologies such as dog-bone shapes and labyrinth patterns manifest themselves [4, 5]. Similar morphologies are observed in numerous other physical systems, including type-1 superconductors, chemical reaction-diffusion systems, and films of ferrofluids. The current hypothesis, which has been confirmed qualitatively, is that these patterns are caused by the competition between line tension forces and dipolar repulsion. The goal of my thesis project was to develop an accurate qualitative and quantitative understanding of the dipolar repulsion forces in Langmuir films. This entails extending a model for Langmuir films [1] to incorporate dipolar repulsion and observing changes in equilibrium states as dipole forces increase.

This report details the model and numerical methods that we used in developing the numerical simulation. In the first section, we describe the previous work conducted by Wintersmith on developing a manageable model for Langmuir films with line tension and the extended model with electrostatic forces. Then we describe how we formulate the equations of motions into a numerical method. To do this we exploit an analytic formulation that removes a singularity in the dipole-dipole forces. In the fol-

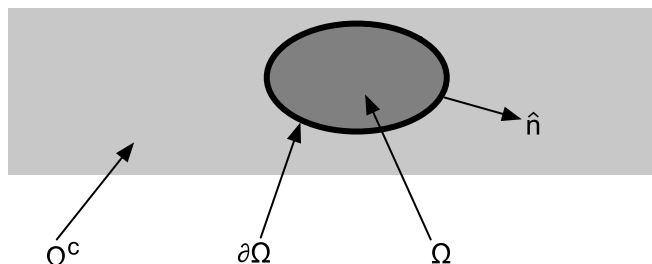


Figure 1.1: A top-down view of a Langmuir layer, defining the domain  $\Omega$ , the boundary  $\partial\Omega$ , the outer monolayer  $\Omega^c$  and the normal  $\hat{n}$ .

Following section, I describe several important implementation details. And finally, I describe the verification process and results from the numerical model.

## 1.2 Langmuir Films

In this section, we summarize the experimental framework that we work in. In our setup, we have a molecularly thin film on the surface of a quiescent subfluid. Typically, the subfluid is water and the gas above the film is air. The film phases can separate into surface liquids with two thicknesses. These phases form different domains within the Langmuir film. It is the formation and evolution of these domains as they are exposed to external forcing that we study. An illustration of the setup is shown in Fig. 1.2.

## 1.3 Line Tension

The dominant forces in the dynamics of the evolution are determined by the chemical properties of the Langmuir film. We assume that one phase is a localized domain  $\Omega$ , that its complement  $\Omega^c$  is a second phase that extends infinitely in the horizontal plane. The domain boundary  $\partial\Omega$  is described by a position vector  $\vec{\Gamma}(s, t)$  parameterized by arc length  $s$  and time  $t$ . We model these domains as incompressible inviscid two-dimensional Newtonian fluids. We also assume that the line tension  $\lambda$  is constant over  $\partial\Omega$ . Using dimensional analysis on the experimental data, Alexander et al.

[1] show at leading order, the subfluid is Stokesian and the Langmuir layer is inviscid. With these assumptions, Alexander et al. develop the *inviscid Langmuir layer Stokesian subfluid* (ILLSS) model. For their experiments, Alexander et al. [1] and Wintersmith [8] used cyano-biphenyl liquid-crystal (8CB), which has negligible dipolar forces. Basically, the evolution is governed by the line tension force, which is damped by the viscosity of the subfluid. Their dimensional analysis also suggests that at leading order the surface of the subfluid is flat and that the film has negligible thickness.

In the special case where the Langmuir domains have the same surface viscosity, a solution can be found with constant pressure and where the movement is in horizontal planes [7]. Alexander et al. developed their model develop a model with this property assuming the surface viscosity is negligible in both phases.

This model leads to the equation of motion for the interface

$$\vec{\Gamma}_t = (\Psi_s + \mathcal{U}_{ext}) \hat{n}, \quad (1.1)$$

where  $\vec{\Gamma}$  is the curve arc-length parameterized by  $s$ . Here

$$\begin{aligned} \Psi(s) &= \frac{-1}{2\pi} \oint_{\vec{\Gamma}} \kappa(s') \hat{t}(s') \cdot \hat{r}(s, s') ds', \\ \vec{r}(s, s') &= \vec{\Gamma}(s') - \vec{\Gamma}(s), \end{aligned} \quad (1.2)$$

where  $\vec{\Gamma}$  is the boundary curve parameterized by  $s$ ,  $\hat{t}$  is the unit tangent,  $\hat{n}$  is the unit outward point normal, and  $\vec{r}$  is a vector pointing from  $\vec{\Gamma}(s)$  to  $\vec{\Gamma}(s')$ . The full derivation can be found in [1].

Furthermore, using energy analysis, Alexander et al. [1] show the that system dissipates energy by reducing the interface length. This suggests that any domain relaxes to a circular shape or possibly several disjoint circular shapes.

## 1.4 Dipolar Repulsion

In [5], Heinig et al. ran simulations with dipole–dipole forces, but they used an explicit integration method, which experienced significant area loss. We combine the model developed by Alexander et al. [1] with the formulation in Heinig [5] to achieve an order of magnitude increase in accuracy.

From the model in [1], we have

$$\mathcal{U} = \left( \frac{\partial \Psi}{\partial s} \right) \hat{n}, \quad (1.3)$$

where  $\Psi(s)$  is the stream function restricted to the boundary of the domain. The velocity stream function is computed as a boundary integral:

$$\Psi(s) = -\frac{1}{2\pi\eta'} \oint \mathcal{F}(s') [\hat{t}(s') \cdot \hat{r}(s, s')] ds', \quad (1.4)$$

where  $\eta'$  is the subfluid viscosity and  $\mathcal{F}$  is the amplitude of the normal force per unit length at the boundary,  $\hat{t}$  is the unit tangent vector, and  $\hat{r}(s, s')$  is a unit vector pointing from  $\vec{\Gamma}(s)$  to  $\vec{\Gamma}(s')$ . Because the subfluid viscosity only appears as a constant multiplicative constant of velocity which is independent of time, we can scale it out.

From Heinig et al. [5], the normal force per unit length at the boundary,  $\mathcal{F}$ , induced by a line tension and intermolecular forces can be computed as

$$\mathcal{F}(s) = \lambda\kappa(s) + \mu^2 \oint \hat{n}(s') \cdot \nabla\Phi(r(s, s')) ds', \quad (1.5)$$

where the first term is associated with the line tension,  $\lambda$ , and is proportional to the boundary's curvature,  $\kappa$ ; this was the sole term present in the numerical simulations done by Alexander et al. and Wintersmith [1, 8]. The new second term is induced by the dipole–dipole electrostatic interaction; here  $\mu$  is the dipole momentum and the potential  $\Phi$  is a radial function satisfying Laplace's equation,  $-\nabla^2\Phi = V(r)$  where  $\mu^2V(r)$  is the intermolecular potential. Classically, we can show for distances large compared to the size of the dipolar molecule that

$$V(r) \propto \frac{1}{r^3}. \quad (1.6)$$

However, the force is singular as  $r \rightarrow 0$ . From [4], one way to regularize the force is to let

$$V(r) = \frac{1}{(r^2 + \Delta^2)^{3/2}}, \quad (1.7)$$

where heuristically  $\Delta$  is the minimum separation of two adjacent molecules. Then, the force equation becomes

$$\mathcal{F}(s) = \lambda\kappa(s) - \mu^2 \oint \frac{\hat{n}(s') \cdot \hat{r}(s, s')}{r(s, s')} \left( \frac{1}{\sqrt{r(s, s')^2 + \Delta^2}} \right) ds'. \quad (1.8)$$



Figure 1.2: A transition from a circular state to a dog-bone. Pictures courtesy of Professor Elizabeth Mann at Kent State.



## Chapter 2

# Numerical Simulation

The evolution of the boundary curve in Langmuir layer is governed by the equation

$$\vec{\Gamma}_t = \mathcal{U}\hat{n} = (\Psi_s + \mathcal{U}_{ext})\hat{n}, \quad (2.1)$$

where  $\vec{\Gamma}$  is the curve arc-length parameterized by  $s$ . In this section, we present a numerical scheme that is second-order in time, essentially spectrally accurate in space, and semi-implicit which guarantees stability.

### 2.1 Formulation

The force term includes a regularization parameter  $\Delta$  that keeps the integral from diverging. Its physical meaning is that of the thickness of the phase boundaries in the case of Langmuir monolayers. Numerical calculation of the force term is complicated by  $\Delta$ . A particular numerical value of  $\Delta$  must be chosen so that the integral can be evaluated. Different values of  $\Delta$  give different numerical results. Moreover, typically  $|\Delta| \ll 1$  making numerical evaluation of the integrals challenging. Yet, the precise value of  $\Delta$  should only affect the short-range interactions (i.e. line tension) [4]. To this end, consider the long range force term:

$$\mathcal{F}_l(\alpha) = \mu^2 \int_0^{2\pi} \frac{\hat{n}(\alpha') \cdot \hat{r}(\alpha, \alpha')}{r} \left( \frac{1}{\sqrt{r^2 + \Delta^2}} \right) s_\alpha(\alpha') d(\alpha' - \alpha). \quad (2.2)$$

If  $\Delta = 0$ , then the integrand becomes singular when  $\alpha' = \alpha$ . However, near  $\alpha$ , we can expand the terms in the integrand as

$$\begin{aligned}
 \vec{r}(\alpha, \alpha') &= \vec{\Gamma}(\alpha') - \vec{\Gamma}(\alpha) = \vec{\Gamma}'(\alpha)[\alpha' - \alpha] + \frac{\Gamma''(\alpha)[\alpha' - \alpha]^2}{2} + \mathcal{O}([\alpha' - \alpha]^3) \\
 &= \hat{t}(\alpha)s_\alpha(\alpha)[\alpha' - \alpha] \\
 &\quad + \frac{1}{2}(\kappa(\alpha)\hat{n}(\alpha)s_\alpha(\alpha)^2 + s_{\alpha\alpha}(\alpha)\hat{t}(\alpha))[\alpha' - \alpha]^2 + \mathcal{O}([\alpha' - \alpha]^3)
 \end{aligned} \tag{2.3}$$

using Frenet–Serret identities. So that

$$\begin{aligned}
 r^2(\alpha, \alpha') &= s_\alpha(\alpha)^2[\alpha' - \alpha]^2 + \mathcal{O}([\alpha' - \alpha]^3), \\
 \hat{n}(\alpha') &= \hat{n}(\alpha) + \hat{n}'(\alpha)[\alpha' - \alpha] + \mathcal{O}([\alpha' - \alpha]^2) \\
 &= \hat{n}(\alpha) - \kappa(\alpha)\hat{t}(\alpha)[\alpha' - \alpha] + \mathcal{O}([\alpha' - \alpha]^2), \\
 \hat{n}(\alpha') \cdot \vec{r}(\alpha, \alpha') &= \frac{\kappa(\alpha)s_\alpha(\alpha)^2}{2}[\alpha' - \alpha]^2 + \mathcal{O}([\alpha' - \alpha]^3).
 \end{aligned} \tag{2.4}$$

Thus near  $\alpha' = \alpha$ , the integrand looks like

$$\frac{\hat{n}(\alpha') \cdot \hat{r}(\alpha, \alpha')}{r} \left( \frac{1}{\sqrt{r^2 + \Delta^2}} \right) s_\alpha(\alpha') \approx \frac{\kappa(\alpha)}{2} \left( \frac{1}{\sqrt{(s_\alpha(\alpha)[\alpha' - \alpha])^2 + \Delta^2}} \right) s_\alpha(\alpha'), \tag{2.5}$$

which can be integrated analytically.

So we can reformulate the integral to remove the troublesome piece as

$$\begin{aligned}
 \mathcal{F}_l &= \mu^2 \int_0^{2\pi} \frac{\hat{n}(\alpha') \cdot \hat{r}(\alpha, \alpha')}{r} \left( \frac{1}{\sqrt{r^2 + \Delta^2}} \right) s_\alpha(\alpha') d(\alpha' - \alpha) \\
 &= \mu^2 \int_\beta^{2\pi-\beta} \frac{\hat{n}(\alpha) \cdot \hat{r}(\alpha, \alpha')}{r^2} s_\alpha(\alpha') d(\alpha' - \alpha) \\
 &\quad + \mu^2 \int_{-\beta}^\beta \frac{\hat{n}(\alpha') \cdot \hat{r}(\alpha, \alpha')}{r} \left( \frac{1}{\sqrt{r^2 + \Delta^2}} \right) s_\alpha(\alpha') d(\alpha' - \alpha) + \mathcal{O}(\Delta^2) \\
 &= \mu^2 \int_\beta^{2\pi-\beta} \frac{\hat{n}(\alpha') \cdot \hat{r}(\alpha, \alpha')}{r^2} s_\alpha(\alpha') d(\alpha' - \alpha) \\
 &\quad + \mu^2 \int_{-\beta}^\beta \frac{\hat{n}(\alpha') \cdot \hat{r}(\alpha, \alpha')}{r} \left( \frac{1}{\sqrt{r^2 + \Delta^2}} \right) s_\alpha(\alpha') d(\alpha' - \alpha) + \mathcal{O}(\Delta^2)
 \end{aligned}$$

$$\begin{aligned}
&= \mu^2 \int_{\beta}^{2\pi-\beta} \frac{\hat{n}(\alpha') \cdot \hat{r}(\alpha, \alpha')}{r^2} s_{\alpha}(\alpha') d(\alpha' - \alpha) \\
&\quad + \mu^2 \int_{-\beta}^{\beta} \left[ \frac{\hat{n}(\alpha') \cdot \hat{r}(\alpha, \alpha')}{r} \left( \frac{1}{\sqrt{r^2 + \Delta^2}} \right) - \frac{\kappa(\alpha)}{2} \frac{1}{\sqrt{(s_{\alpha}(\alpha)[\alpha' - \alpha])^2 + \Delta^2}} \right] s_{\alpha}(\alpha') \\
&\quad + \frac{\kappa(\alpha)}{2} \frac{1}{\sqrt{(s_{\alpha}(\alpha)[\alpha' - \alpha])^2 + \Delta^2}} s_{\alpha}(\alpha') d(\alpha' - \alpha) + \mathcal{O}(\Delta^2) \\
&= \mu^2 \int_{\beta}^{2\pi-\beta} \frac{\hat{n}(\alpha') \cdot \hat{r}(\alpha, \alpha')}{r^2} s_{\alpha}(\alpha') d(\alpha' - \alpha) + \mu^2 I_{\beta}(\Delta) + J_{\beta}(\Delta) + \mathcal{O}(\Delta^2),
\end{aligned} \tag{2.6}$$

where

$$\begin{aligned}
I_{\beta}(\Delta) = \int_{-\beta}^{\beta} \left[ \frac{\hat{n}(\alpha') \cdot \hat{r}(\alpha, \alpha')}{r} \left( \frac{1}{\sqrt{r^2 + \Delta^2}} \right) \right. \\
\left. - \frac{\kappa(\alpha)}{2} \frac{1}{\sqrt{(s_{\alpha}(\alpha)[\alpha' - \alpha])^2 + \Delta^2}} \right] s_{\alpha}(\alpha') d(\alpha' - \alpha)
\end{aligned} \tag{2.7}$$

and

$$J_{\beta}(\Delta) = \int_{-\beta}^{\beta} \frac{\kappa(\alpha)}{2} \frac{1}{\sqrt{(s_{\alpha}(\alpha)[\alpha' - \alpha])^2 + \Delta^2}} s_{\alpha}(\alpha') d(\alpha' - \alpha). \tag{2.8}$$

Now, the first piece of  $\mathcal{F}_l$  can be integrated numerically without problems. Further, the integrand of  $I_{\beta}$  remains finite as  $\Delta \rightarrow 0$ , in fact the limit at  $\Delta = 0$  is 0. We get that  $I_{\beta}$  depends weakly on  $\Delta$ , in that  $I_{\beta}(\Delta) = I_{\beta}(0) + \mathcal{O}(\Delta^2)$ . Finally,

$$\begin{aligned}
J_{\beta}(\Delta) &= \int_{-\beta}^{\beta} \frac{\kappa(\alpha)}{2} \frac{1}{\sqrt{(s_{\alpha}(\alpha)[\alpha' - \alpha])^2 + \Delta^2}} s_{\alpha}(\alpha') d(\alpha' - \alpha) \\
&= \frac{\kappa(\alpha)}{2} \int_{-\beta}^{\beta} \frac{1}{\sqrt{[\alpha' - \alpha]^2 + (\Delta/s_{\alpha})^2}} d(\alpha' - \alpha) \\
&= \frac{\kappa(\alpha)}{2} (2 \ln(2s_{\alpha}(\alpha)\beta) - 2 \ln(\Delta) + \mathcal{O}(\Delta^2)) \\
&= \kappa(\alpha) (\ln(2s_{\alpha}(\alpha)\beta) - \ln(\Delta) + \mathcal{O}(\Delta^2)),
\end{aligned} \tag{2.9}$$

because  $s_{\alpha}$  is not a function of  $\alpha$  in our parameterization.

Thus,

$$\mathcal{F}_l = \mu^2 I - \kappa(\alpha) \mu^2 \ln(\Delta) + \mathcal{O}(\Delta^2), \tag{2.10}$$

where

$$I = \int_{\beta}^{2\pi-\beta} \frac{\hat{n}(\alpha') \cdot \hat{r}(\alpha, \alpha')}{r^2} s_{\alpha}(\alpha') d(\alpha' - \alpha) + I_{\beta}(0) + \kappa(\alpha) \ln(2s_{\alpha}(s_{\alpha})\beta). \tag{2.11}$$

The  $\Delta$  dependence has been separated out of the integral, and as we initially expected, it contributes to the short range line tension force (i.e. it is proportional to curvature). It should also be noted that  $I$  is independent of  $\beta$ , and can be easily integrated by a Newton Cotes method or similar techniques. Combining the long range force term with the short range force term yields

$$\begin{aligned}\mathcal{F}(\alpha) &= \lambda\kappa(\alpha) - \mathcal{F}_l(\alpha) \\ &= \left(\lambda - \mu^2 \ln \frac{1}{\Delta}\right) \kappa(\alpha) - \mu^2 I + \mathcal{O}(\Delta^2) \\ &= \tilde{\lambda}\kappa(\alpha) - \mu^2 I + \mathcal{O}(\Delta^2),\end{aligned}\tag{2.12}$$

where  $\tilde{\lambda} = \lambda - \mu^2 \ln(1/\Delta)$  is the effective line tension. In our numerical simulations, we use Eq. 2.12 for the force term.

## 2.2 Stability

Numerically we find that explicit integration methods are susceptible to high-wavenumber instability. If an explicit time integration method is used, these systems impose strong stability constraints on the time step. The presence of such constraints are referred to as *stiffness* [3]. For this system, our problems arise due to the high-wavenumber terms. It can be shown that asymptotically, the high-wavenumber terms are governed by a simpler evolution law. Thus, if we can split the equations of motion into two pieces, we can solve one piece explicitly without stability constraints, and the high-wavenumber term implicitly. In this section, we find the growth rate for high-wavenumber terms.

Suppose, we have an  $\epsilon$  single mode perturbation with Fourier mode  $k$ . From the form of the integral in the force term, we know that the force will die off quickly away from the basepoint, so we can perform the calculation on an infinite flat interface. In this case, our curve can be parameterized as

$$\vec{\Gamma}(\alpha) = \langle x(\alpha), y(\alpha) \rangle = \langle \alpha, \epsilon e^{ik\alpha} \rangle.\tag{2.13}$$

Then, without loss of generality, setting the basepoint at 0 yields

$$\begin{aligned}
\vec{r}(0, \alpha) &= \vec{\Gamma}(\alpha) - \vec{\Gamma}(0) = \langle \alpha, \epsilon (e^{ik\alpha} - 1) \rangle, \\
r(0, \alpha) &= \alpha + \mathcal{O}(\epsilon^2), \\
\vec{\Gamma}_\alpha &= \langle 1, ik\epsilon e^{ik\alpha} \rangle, \\
s_\alpha &= 1 + \mathcal{O}(\epsilon^2), \\
\hat{t} &= \langle 1, ik\epsilon e^{ik\alpha} \rangle, \\
\hat{n} &= \langle -ik\epsilon e^{ik\alpha}, 1 \rangle, \\
\kappa &= \frac{y_\alpha x_{\alpha\alpha} - x_\alpha y_{\alpha\alpha}}{s_\alpha^3} = \epsilon k^2 e^{ik\alpha}.
\end{aligned} \tag{2.14}$$

Alexander et al. [1] calculate the growth rate,  $\sigma_k$ , for the case when  $\mu = 0$ . To modify their calculation for  $\mu \neq 0$ , we calculate the force term with the long range interaction term. We have:

$$\begin{aligned}
\mathcal{F}(0) &= \lambda\kappa(0) - \mu^2 \int_{-\infty}^{\infty} \frac{\hat{n}(\alpha) \cdot \hat{r}(0, \alpha)}{r} \left( \frac{1}{\sqrt{r^2 + \Delta^2}} \right) s_\alpha(\alpha) d\alpha \\
&= \lambda k^2 \epsilon - \mu^2 \int_{-\infty}^{\infty} \frac{-\alpha ik\epsilon e^{ik\alpha} + \epsilon(e^{ik\alpha} - 1)}{\alpha^2 \sqrt{\alpha^2 + \Delta^2}} d\alpha \\
&= \lambda k^2 \epsilon - \mu^2 \epsilon \int_{-\infty}^{\infty} \frac{-\alpha ik(\cos(k\alpha) + i \sin(k\alpha)) + \cos(k\alpha) + i \sin(k\alpha) - 1}{\alpha^2 \sqrt{\alpha^2 + \Delta^2}} d\alpha \\
&= \lambda k^2 \epsilon - \mu^2 \epsilon \int_{-\infty}^{\infty} \frac{\alpha k \sin(k\alpha) + \cos(k\alpha) - 1}{\alpha^2 \sqrt{\alpha^2 + \Delta^2}} d\alpha \\
&= \lambda k^2 \epsilon - \mu^2 \epsilon k^2 \int_{-\infty}^{\infty} \frac{u \sin(u) + \cos(u) - 1}{u^2 \sqrt{u^2 + (k\Delta)^2}} du \\
&= \lambda k^2 \epsilon - 2\mu^2 \epsilon k^2 I(c),
\end{aligned} \tag{2.15}$$

where  $c = k\Delta$  and

$$I(c) = \int_0^\infty \frac{u \sin(u) + \cos(u) - 1}{u^2 \sqrt{u^2 + c^2}} du. \tag{2.16}$$

In our case,  $c \ll 1$  because  $\Delta \ll \frac{2\pi}{k}$ , a typical wavelength, so we can expand

$I$  in  $c$ :

$$\begin{aligned}
 I(c) &= \int_0^{\sqrt{c}} \frac{u \sin(u) + \cos(u) - 1}{u^2 \sqrt{u^2 + c^2}} du \\
 &\quad + \int_{\sqrt{c}}^{\infty} \frac{u \sin(u) + \cos(u) - 1}{u^2 \sqrt{u^2 + c^2}} du \\
 &= \frac{1}{2} \left( -\ln \frac{c}{2} + \frac{1}{2} - \gamma \right) + \mathcal{O}(c^2),
 \end{aligned} \tag{2.17}$$

where  $\gamma$  is Euler's constant and we have series expanded and subsequently evaluated the numerator and denominator of the two integral terms respectively. So,

$$\begin{aligned}
 \mathcal{F}(0) &= \lambda k^2 \epsilon - \epsilon \mu^2 k^2 \left( -\ln \frac{c}{2} + \frac{1}{2} - \gamma \right) + \mathcal{O}(c^2) \\
 &= \epsilon k^2 \left( \lambda - \mu^2 \ln(1/\Delta) + \mu^2 \left( \ln \frac{k}{2} + \gamma - \frac{1}{2} \right) \right) + \mathcal{O}(c^2) \\
 &= \epsilon k^2 \left( \tilde{\lambda} + \mu^2 \left( \ln \frac{k}{2} + \gamma - \frac{1}{2} \right) \right) + \mathcal{O}(c^2).
 \end{aligned} \tag{2.18}$$

From [1], for large  $k$  and a general interface, the growth rate is approximately

$$\begin{aligned}
 \lambda_k &= -\frac{k^2 \sigma_k}{\pi s_\alpha^2}, \\
 \sigma_k &= \tilde{\lambda} + \mu^2 \left( \ln \frac{k}{2} + \gamma - \frac{1}{2} \right).
 \end{aligned} \tag{2.19}$$

The additional term is short-wave stabilizing, but with large growth rates.

In splitting the high-wavenumber term, we do not have to exactly subtract it off, rather we need to subtract enough off so that it is stable for modest time steps. We can estimate the maximum wavenumber - suppose  $L$  is the length of the curve, and we have  $N$  points, then the maximum wavenumber is  $2\pi N/L$ . So the maximum decay rate is approximately

$$\sigma^* = -\frac{k^2 \left( \tilde{\lambda} + \mu^2 \left( \ln(\pi N/L) + \gamma - \frac{1}{2} \right) \right)}{\pi s_\alpha^2}.$$

Thus, closely following Hou et al. [6], we can split  $\vec{\Gamma}_t$  into two pieces:

$$\vec{\Gamma}_t = (\mathcal{U} - \sigma^* \kappa) \hat{n} + \sigma^* \kappa \hat{n}. \tag{2.20}$$

In the high-wavenumber limit, the low-wavenumber part vanishes and the interface curve is governed by normal motion proportional to its local curvature. To combine the two steps, we use a Strang splitting technique (accurate to second-order). In the following sections, we describe how our numerical scheme solves each piece, and the Strang splitting technique.

### 2.3 Tangential Velocity

In our calculations, we parameterize  $\vec{\Gamma}$  by  $\alpha$  from 0 to  $2\pi$ . We choose  $\alpha$  such that  $s_\alpha$  is constant initially, but we would also like to ensure that as the boundary evolves that  $s_\alpha$  is constant. This constraint makes the application of an implicit method for the curvature step trivial. The equation of motion is fully determined by the shape of the boundary, so we are free to add an arbitrary tangential velocity,  $W$ , so that the equation of motion is

$$\vec{\Gamma}_t = \mathcal{U}\hat{n} + W\hat{t}. \quad (2.21)$$

We choose  $W$  such that  $s_\alpha$  is constant with respect to  $\alpha$ , so that

$$L = \int_0^{2\pi} s_\alpha d\alpha = s_\alpha \int_0^{2\pi} d\alpha = 2\pi s_\alpha, \quad (2.22)$$

where  $L$  is the length of the curve. Then,

$$\begin{aligned} s_{\alpha t} &= \frac{2x_\alpha x_{\alpha t} + 2y_\alpha y_{\alpha t}}{2s_\alpha} = \hat{t} \cdot \frac{\partial}{\partial \alpha} \vec{\Gamma}_t \\ &= \hat{t} \cdot \frac{\partial}{\partial \alpha} (\mathcal{U}\hat{n} + W\hat{t}) \\ &= \hat{t} \cdot (\mathcal{U}_\alpha \hat{n} + \mathcal{U}\hat{n}_\alpha + W_\alpha \hat{t} + W\hat{t}_\alpha) \\ &= \hat{t} \cdot (\mathcal{U}_\alpha \hat{n} - \mathcal{U}\kappa s_\alpha \hat{t} + W_\alpha \hat{t} + W\kappa s_\alpha \hat{n}) \\ &= W_\alpha - \mathcal{U}\kappa s_\alpha, \end{aligned} \quad (2.23)$$

where we have used the Frenet–Serret identities:

$$\begin{aligned} \hat{n}_\alpha &= \hat{n}_s s_\alpha = -\kappa \hat{t} s_\alpha, \\ \hat{t}_\alpha &= \hat{t}_s s_\alpha = \kappa \hat{n} s_\alpha. \end{aligned} \quad (2.24)$$

Then we have

$$L_t = \int_0^{2\pi} s_{\alpha t} d\alpha = \int_0^{2\pi} W_\alpha - \kappa \mathcal{U} s_\alpha d\alpha = -s_\alpha \int_0^{2\pi} \kappa \mathcal{U} d\alpha \quad (2.25)$$

because  $W$  is  $2\pi$ -periodic. But also differentiating Eq. 2.22 with respect to  $t$  and using Eq. 2.23 yields

$$L_t = 2\pi s_{\alpha t} = 2\pi \left( \frac{W_{\alpha}}{s_{\alpha}} - \kappa \mathcal{U} \right) s_{\alpha} = \left( \frac{W_{\alpha}}{s_{\alpha}} - \kappa \mathcal{U} \right) L. \quad (2.26)$$

So that

$$W_{\alpha} = \kappa \mathcal{U} s_{\alpha} - \frac{s_{\alpha}^2}{L} \int_0^{2\pi} \kappa \mathcal{U} d\alpha = \kappa \mathcal{U} s_{\alpha} - \frac{1}{2\pi} \int_0^{2\pi} \kappa \mathcal{U} s_{\alpha} d\alpha \quad (2.27)$$

and by integrating

$$W(\alpha) = W(0) + \int_0^{\alpha} \kappa \mathcal{U} s_{\alpha} d\alpha - \frac{\alpha}{2\pi} \int_0^{2\pi} \kappa \mathcal{U} s_{\alpha} d\alpha \quad (2.28)$$

as the tangential velocity. We can choose  $W(0)$  arbitrarily because our condition is on  $W_{\alpha}$ , so we choose  $W(0) = 0$ . Note that subtracting  $\frac{1}{2\pi} \int_0^{2\pi} \kappa \mathcal{U} s_{\alpha} d\alpha$  in the equation for  $W_{\alpha}$  effectively zeros out the constant Fourier mode. This is numerically convenient because the constant mode is problematic for pseudo-spectral differentiation because it involves a division by zero. So instead of subtracting the term, we can zero out the constant mode and pseudo-spectrally integrate  $\kappa \mathcal{U} s_{\alpha}$  to get  $W$ .

## 2.4 Motion by Curvature

In the high-wavenumber limit, the motion is governed by normal motion proportional to its local curvature:

$$\vec{\Gamma}_t = \sigma^* \kappa \hat{n}. \quad (2.29)$$

To solve the equation implicitly, we first reformulate the equation using the Frenet–Serret identities:

$$\begin{aligned} \vec{\Gamma}_t &= \sigma^* \kappa \hat{n} = \sigma^* \frac{\partial \hat{t}}{\partial s} \\ &= \sigma^* \frac{\partial}{\partial s} \left( \frac{\vec{\Gamma}_{\alpha}}{s_{\alpha}} \right) = \frac{\sigma^*}{s_{\alpha}} \frac{\partial}{\partial \alpha} \left( \frac{\vec{\Gamma}_{\alpha}}{s_{\alpha}} \right) \\ &= \frac{\sigma^*}{s_{\alpha}} \left( \frac{\vec{\Gamma}_{\alpha\alpha} s_{\alpha} - \vec{\Gamma}_{\alpha} s_{\alpha\alpha}}{s_{\alpha}^2} \right) \\ &= \frac{\sigma^*}{s_{\alpha}^2} \vec{\Gamma}_{\alpha\alpha} \end{aligned} \quad (2.30)$$

because we have chosen  $\alpha$  such that  $s_\alpha$  is constant with respect to  $\alpha$  (although it varies in time). The system is linear in  $\alpha$ , so we can take the Fourier transform with respect to  $\alpha$  to get an ODE in  $t$ :

$$\begin{aligned}
 \hat{\Gamma}_t(k, t) &= \int_{-\infty}^{\infty} \vec{\Gamma}_t(\alpha, t) e^{-ik\alpha} d\alpha \\
 &= \mathcal{F}\{\vec{\Gamma}_t(\alpha, t)\} \\
 &= \mathcal{F}\left\{\frac{\sigma^*}{s_\alpha^2} \vec{\Gamma}_{\alpha\alpha}(\alpha, t)\right\} \\
 &= -\frac{\sigma^* k^2}{s_\alpha^2} \hat{\Gamma}(k, t),
 \end{aligned} \tag{2.31}$$

which admits an exact solution

$$\hat{\Gamma}(t + \Delta t) = \hat{\Gamma}(t) \exp\left(-\sigma^* k^2 \int_t^{t+\Delta t} \frac{1}{s_\alpha(t')^2} dt'\right). \tag{2.32}$$

We can Taylor expand the integrand to find a second-order in time approximation as

$$\begin{aligned}
 \frac{1}{[s_\alpha(t + \Delta t)]^2} &= \frac{1}{[s_\alpha(t)]^2} - \frac{2}{[s_\alpha(t)]^3} s_{\alpha t} \Delta t + \mathcal{O}[(\Delta t)^2] \\
 &= \frac{1}{[s_\alpha(t)]^2} + \frac{2}{2\pi} \frac{s_\alpha(t)}{[s_\alpha(t)]^3} \int_0^{2\pi} \kappa \mathcal{U} d\alpha \Delta t + \mathcal{O}[(\Delta t)^2] \\
 &= \frac{1}{[s_\alpha(t)]^2} + \frac{1}{\pi [s_\alpha(t)]^2} \int_0^{2\pi} \kappa \mathcal{U} d\alpha \Delta t + \mathcal{O}[(\Delta t)^2]
 \end{aligned} \tag{2.33}$$

using Eqs. 2.25 and 2.26 where  $\mathcal{U} = \sigma^* \kappa$  in this case. So for the Fourier coefficients, we get

$$\hat{\Gamma}(t + \Delta t) = \hat{\Gamma}(t) \exp\left(-\sigma^* k^2 \left[\frac{1}{s_\alpha^2} \Delta t + \left[\frac{1}{2\pi s_\alpha^2} \int_0^{2\pi} \kappa \mathcal{U} d\alpha\right] (\Delta t)^2\right] + \mathcal{O}[(\Delta t)^3]\right) \tag{2.34}$$

as the update equation. Then, we apply the inverse Fourier transform to find  $\vec{\Gamma}$  at the next time step.

## 2.5 Numerical Scheme

To summarize, we can write our equation of motion as

$$\vec{\Gamma}_t = [(\mathcal{U} - \sigma^* \kappa) \hat{n} + W \hat{f}] + [\sigma^* \kappa \hat{n}]. \tag{2.35}$$

Let  $A^\tau$  and  $B^\tau$  be solutions to the differential equations  $\vec{\Gamma}_t = (\mathcal{U} - \sigma^* \kappa) \hat{n} + W\hat{t}$  and  $\vec{\Gamma}_t = \sigma^* \kappa \hat{n}$  for a time  $\tau$ , respectively. As we have shown in the previous section,  $B^\tau$  can be solved by Fourier series; numerically it is solved by a pseudo-spectral method, where the Fourier series is replaced by the discrete Fourier transform. Numerically, we can approximate  $A^\tau$  by a Modified Euler method. Closely following Bernoff [2], a Strang split-step scheme is used where to evolve in time  $\Delta t$ , we apply  $B^{(\Delta t)/2} A^{\Delta t} B^{(\Delta t)/2}$ . This scheme has an error  $\mathcal{O}(T(\Delta t)^2)$  when integrated for time  $T$  (hence why we use a second-order in time approximation for  $A^\tau$  and  $B^\tau$ ).

So in summary our numerical scheme to advance  $\Delta t$  is:

1. Compute a half step of curvature motion implicitly.
2. Compute a full step according to the equation  $\vec{\Gamma}_t = (\mathcal{U} - \sigma^* \kappa) \hat{n} + W\hat{t}$  explicitly using Modified Euler.
3. Compute a half step of curvature motion implicitly.

## Chapter 3

# Implementation

Our first step is to discretize the continuous curve  $\vec{\Gamma}$ . We choose an equal arc length discretization because it allows us to use spectral differentiation and integration techniques. So, we only know the boundary curve at  $n$  equally spaced points along the curve  $\alpha_j, j = 0, \dots, n - 1$ . In this section, we address some of the implementation issues that arose.

### 3.1 Differentiation and Integration

To compute the derivative of a function that we know only at  $n$  equally spaced points, we turn to spectral differentiation. Closely following Trefethen [9], we can express a  $2\pi$ -periodic function  $f$  in Fourier space using the Discrete Fourier Transform (DFT). Then it is simple to integrate or differentiate by either dividing or multiplying each Fourier coefficient by  $ik$  to integrate or differentiate respectively.

### 3.2 Filtering the Spectrum

As suggested by [6], we filter out the high-wavenumbers in the spectrum. At specific points in the time step, we apply the filter

$$\tilde{a}_k \mapsto \tilde{a}_k \exp \left[ - \left( \frac{k}{(1-f)n/2} \right)^{16} \right], \quad (3.1)$$

where  $f$  is the fraction of the wavenumbers that we wish to filter out. For our application, we chose  $f = 0.4$ . The exact shape of the filtering function

is unimportant, but it is important that it is smooth. Using a smooth filter function ensures that we do not introduce unnecessary distortions (ie. Gibb's phenomenon) in the physical space.

High-wavenumbers grow more rapidly than low-wavenumbers, so by filtering the spectrum, we ensure that small errors do not accumulate and grow unbounded.

### 3.3 Calculating $\Psi_s$

Recall that the velocity term requires us to calculate

$$\begin{aligned}\Psi_s(\alpha) &= -\frac{1}{2\pi\eta's_\alpha(\alpha)}\frac{\partial}{\partial\alpha}\int_0^{2\pi}\mathcal{F}(\alpha')[\hat{t}(\alpha')\cdot\hat{r}(\alpha,\alpha')]d(\alpha'-\alpha), \\ \mathcal{F}(\alpha) &= \tilde{\lambda}\kappa(\alpha) - \mu^2I, \\ I &= \int_\beta^{2\pi-\beta}\frac{\hat{n}(\alpha')\cdot\hat{r}(\alpha,\alpha')}{r^2}s_\alpha(\alpha')d(\alpha'-\alpha) \\ &\quad + I_\beta(0) + \kappa(\alpha)\ln(2s_\alpha(s_\alpha)\beta), \\ I_\beta(0) &= \int_{-\beta}^\beta\left[\frac{\hat{n}(\alpha')\cdot\hat{r}(\alpha,\alpha')}{r^2} - \frac{\kappa(\alpha)}{2s_\alpha(\alpha)|\alpha|}\right]s_\alpha(\alpha')d(\alpha'-\alpha).\end{aligned}\tag{3.2}$$

To compute  $\Psi_s$  numerically, we first compute  $\Psi(\alpha_j)$  for  $j = 0, \dots, n-1$ , then differentiate spectrally to find  $\Psi_s$ . We approximate the integrals with a composite 9-point closed Newton-Cotes integration rule [3]. Newton-Cotes methods provide a high degree of accuracy given that the function is sufficiently smooth. The  $I_\beta$  integral has a removable singularity at  $\alpha' = \alpha$  and the limit at the point is 0. We ensure that the singularity point is the endpoint of a panel, so  $\beta$  is a panel width. The  $\Psi$  integral also has a discontinuity at  $\alpha' = \alpha$ ; the left hand limit is opposite the right hand limit. We ensure that it is an endpoint in our composite rule, and we can safely ignore the value at the discontinuity because right and left sides cancel.

### 3.4 Equal Arc-length Redistribution

Both the spectral differentiation and integration and the Newton-Cotes method require that our discretization be equally spaced in the  $\alpha$  parameter. Additionally, we have stipulated that our parameterization is such that  $s_\alpha$  is constant. This implies that an equally spaced discretization of  $\alpha$  is also equally

spaced in arc length. Previously, we described how we choose the tangential velocity to ensure that the points stayed equally spaced. However, we also redistribute the points at the end of every time step to improve accuracy and correct any drift. So at the end of a time step, we reparameterize the curve at points  $\alpha'_j, j = 0, \dots, n-1$ , where  $\alpha'_0 = \alpha_0$  and the  $\alpha'_j$  are equally spaced in arc length. That is

$$\int_{\alpha'_0}^{\alpha'_j} s_\alpha d\alpha = j \frac{L}{n}, \quad (3.3)$$

where  $L$  is the length of the curve. To find each  $\alpha'_j$ , we solve for the root of the function

$$F(x) = \int_{\alpha'_0}^x s_\alpha d\alpha - j \frac{L}{n}. \quad (3.4)$$

This can be done using any number of root finders, but we find that Newton's method converges quickly and accurately (we compute the derivative pseudo-spectrally). Once we have the  $\alpha'_j$ , we use Fourier interpolation to evaluate  $\vec{\Gamma}$  at those points.

### 3.5 Maximum Curvature

Using a finite number of points means that we cannot resolve curves with high curvature without using many points. We can estimate the number of points necessary to resolve a curve by looking at the point with highest curvature. Suppose we require  $N_c$  points to resolve a circle and the curve has maximum curvature  $\kappa^*$ . Consider a circle tangent to the curve at the point of maximum curvature and agreeing in curvature. The curvature of a circle is  $-1/R$  where  $R$  is the radius of the circle. Let  $l$  be the length of the arc that the circle and curve agree at. Then

$$\frac{l}{L} = \frac{N_c}{N_{\vec{\Gamma}}} \frac{l}{2\pi R} = \frac{N_c \kappa^*}{N_{\vec{\Gamma}} 2\pi}, \quad (3.5)$$

where  $N_{\vec{\Gamma}}$  is the number of points needed to resolve the curve. Then

$$N_{\vec{\Gamma}} = \frac{\kappa^* L}{2\pi} N_c. \quad (3.6)$$

This gives a rough estimate of the number of points needed based on the maximum curvature of the curve.

### 3.6 Variable Time Step

To control amount of area lost or gained during the evolution of the curve, we use a variable time step. At the end of each time step, we calculate the area by

$$Area = \frac{1}{2} \int_0^{2\pi} xy_\alpha - yx_\alpha d\alpha \quad (3.7)$$

using pseudo-spectral derivatives and the trapezoid rule. We compare this value to the previously computed area, and if the difference is above a threshold (typically 0.1 percent), we recompute the step using a smaller time step. If the difference is lower than a threshold (typically 0.001 percent), then we increase the time step for the next step. This allows us to constrain the area loss or gain at every step, and to take larger steps when we can compute the area accurately. When  $\mu = 0$ , this is effective, but when  $\mu \neq 0$ , the small area losses can still lead to instability, so area loss is not an effective measure of stability.

# Chapter 4

## Results

### 4.1 Stability and Relaxation for a Perturbed Circular Domain

In this section, we consider the relaxation of a linear perturbation to the boundary of a circular domain with radius  $R$ . Following Alexander et al., we modify their calculations for the case where  $\mu \neq 0$ . The rates of relaxation of the various modes are characteristic and give one way to verify that our numerical code is correct. We describe the boundary of the domain in polar coordinates as

$$\vec{\Gamma}(\theta, t) = R + \epsilon\beta(\theta, t), \quad (4.1)$$

where  $\beta(\theta, 0)$  is an initial perturbation. Rotational symmetry guarantees that angular Fourier modes will decouple in the linear stability problem. Consequently, we expand  $\beta$  as

$$\beta(\theta, t) = \sum_{n=1}^{\infty} [a_n \cos(n\theta) + b_n \sin(n\theta)] e^{\lambda_n t} \quad (4.2)$$

and compute to first order in  $\epsilon$ . We essentially replace the force term in Alexander et al.'s calculation with our force term and carry through the changes. Unlike in the  $\mu = 0$  case, we were not able to obtain a closed form for the general decay rate, but we developed a Maple worksheet able to calculate the decay rates for specific modes. In particular, for the  $n = 2$  mode, the decay rate is

$$\lambda_2 = \frac{4(-3\lambda + (-7 + 9\ln(2) + 3\ln(R))\mu^2)}{(4 - 1/4)\pi R^2}, \quad (4.3)$$

$\mu$	Expected	Observed	Relative error
0	-0.254635	-0.254598	0.000147
0.25	-0.247644	-0.247607	0.000151
0.5	-0.226671	-0.226635	0.000161
1.0	-0.142779	-0.142751	0.000193
2.0	0.192790	0.192468	0.001669

Figure 4.1: Relaxation rates for a mode 2 perturbation. The parameters for the simulation were:  $R = 2, \lambda = 1, \epsilon = 0.02, N = 64, \Delta t = 0.01, T = 1$ .

which reduces to the relaxation rate in [1] when  $\mu = 0$ . Additionally, we can solve for the stability critical radius, which for the  $n = 2$  mode gives

$$R_2 = \frac{1}{8} e^{\frac{\lambda}{\mu^2} + \frac{7}{3}}. \quad (4.4)$$

The stability critical radius agrees with the literature [4, 5]. For  $R > R_2$ , the domain is unstable and for  $R < R_2$  the domain is stable.

To confirm our numerical code, we choose an initial condition of a circle plus a single mode perturbation,  $\vec{\Gamma}(\theta, 0) = R + \epsilon \cos(n\theta)$ , for  $n$  positive. We allow the curve to relax back to a circle, and at each time step we record the log of the coefficient corresponding to the  $n$ th mode. We expect that the values should decay linearly with slope  $\lambda_n$  and that it is accurate up to second-order in time. The results of such a study are shown in Fig. 4.1. The error is relatively small and less than the order of the perturbation, but as we increase  $\mu$ , the error appears to increase. This may be indicative of an error in the code.

We can also determine the critical radius numerically within 0.2 percent of the actual critical radius, which is another confirmation of our numerical code's correctness.

## 4.2 Error Order

We expect that the numerics are accurate to second-order. We test this by measuring area loss in refinement studies, where we run the simulations to the same time, but increase the number of time steps in the time period. In the case of an unperturbed circle and for small amplitude perturbations (i.e.  $\epsilon = R/10$ ), the numerics exhibit second-order error for various values of  $\mu$  and  $\lambda$ , that is an increase in the number of time steps by two, results in

a decrease in error by four. This is further evidence that the numerics are correct.

### **4.3 Transitions**

Transitions from a perturbed circle to a dog-bone are observed experimentally and have been reproduced in [5]. We are able to produce qualitatively similar transitions from a perturbed circle to a dog-bone. We can also introduce higher wavenumber perturbations, and qualitatively, our results agree with the literature.

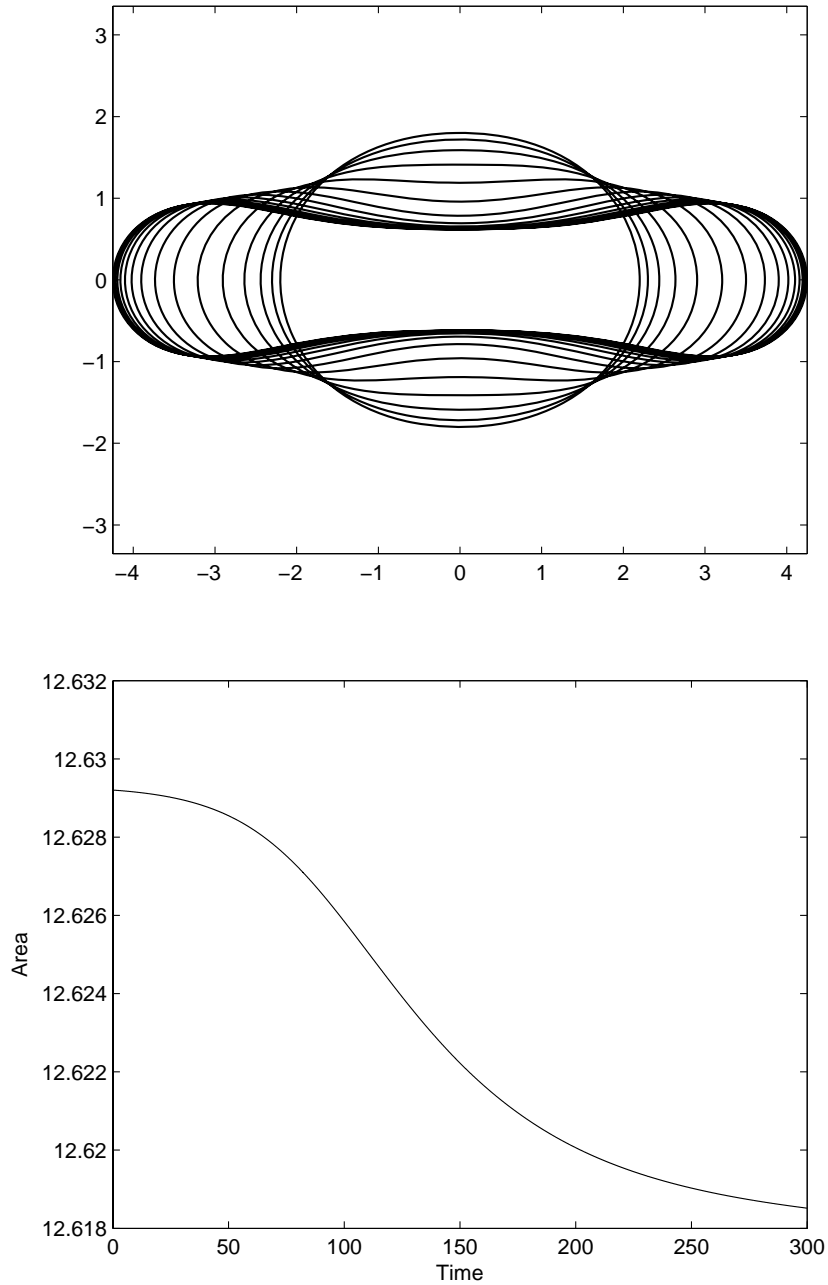


Figure 4.2: A transition from a mode 2 perturbation to a dog-bone. The radius is greater than critical radius, so we observe the ellipse evolving towards a dog-bone morphology. From the lower plot, we observe that the area loss is minimal and we see that the domain is approaching stability as a dog-bone. The parameters used for the simulation were:  $R = 2, \epsilon = 0.2, \lambda = 1, \mu = 2, N = 128, \Delta t = 0.01, T = 300$ .

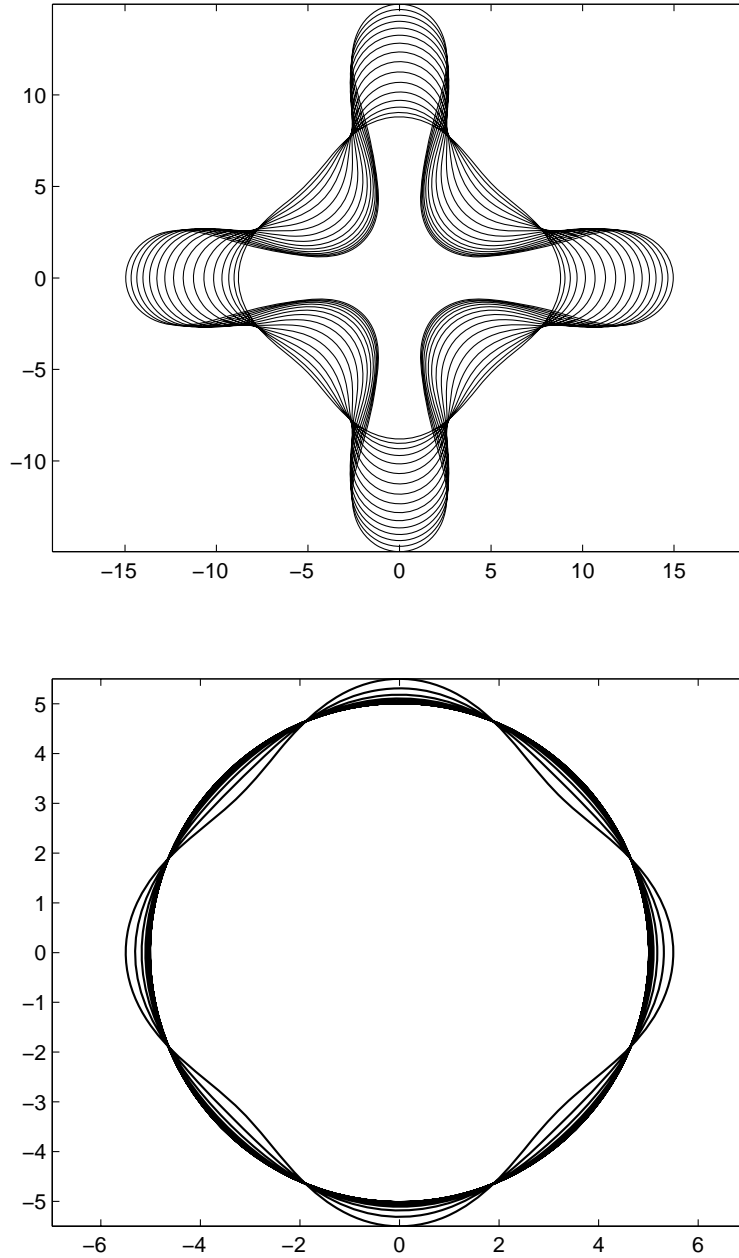


Figure 4.3: In the top figure, a transition from a mode 4 perturbation to a clover morphology. The radius is greater than critical radius, so as we expect, the perturbed circle is not stable. In the bottom figure, the radius is less than critical radius so the perturbed circle is stable and relaxes back to a circle. The parameters used for the simulation were:  $R = \{8, 5\}$ ,  $\epsilon = 1$ ,  $\lambda = 1$ ,  $\mu = 1$ ,  $N = 128$ ,  $\Delta t = 0.01$ ,  $T = 200$ .

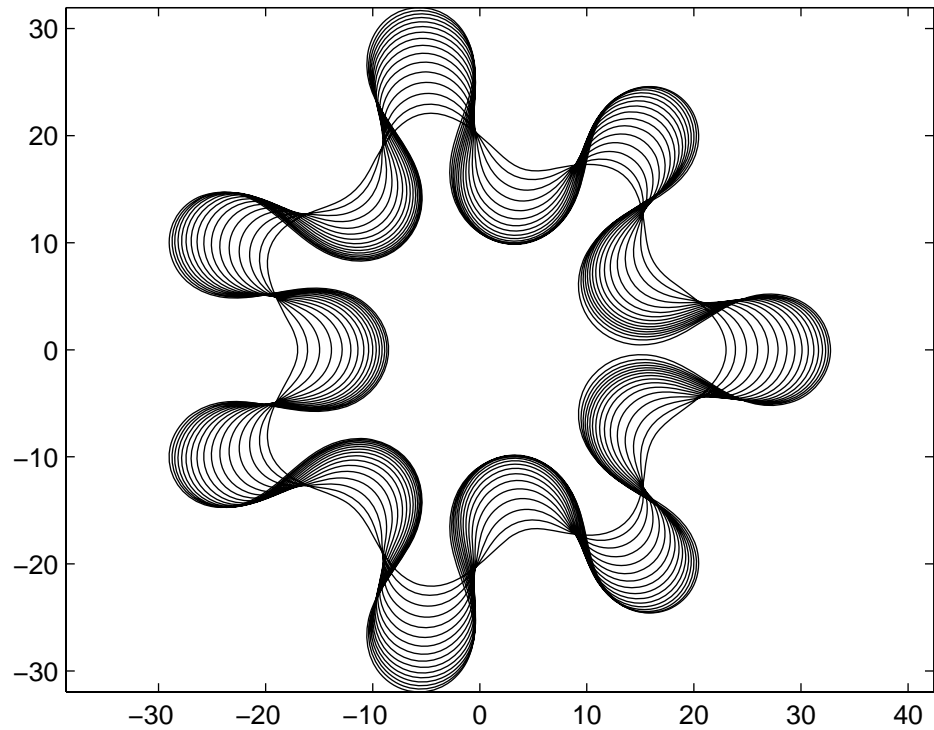


Figure 4.4: An evolution of unstable perturbed circle. We added in a mode 7 perturbation with amplitude 2 and a mode 3 perturbation with amplitude 1. We experience an 0.8 percent loss in area over the simulation. The behavior qualitatively mimics results from experiments and literature. Unfortunately, running the simulation longer results in a pinching off, which our code is not able to handle. The parameters used for the simulation were:  $R = 20$ ,  $\lambda = 1$ ,  $\mu = 1$ ,  $N = 128$ ,  $\Delta t = 0.01$ ,  $T = 280$ .

# Chapter 5

## Conclusion

We set out to develop a numerical model that would combine the high accuracy methods developed in [1, 6, 7] with a model of Langmuir films that included dipolar forces [5, 4]. Previous simulations done by Heinig et al. [5] had qualitatively reproduced experimental results, but they suffered from numerical inaccuracy due to using explicit methods and resorting to a small cut-off parameter. We aimed to use an implicit method to get around some of the instabilities, and we also reformulated the equations of motion to remove a singularity without resorting to a small cut-off parameter. As the project comes to an end, I believe we have achieved our goals. We were able to reproduce many of the results in the literature to a much higher accuracy than previously obtained. However, we have been unable to reproduce labyrinth patterns yet and we are just observing some interesting dynamics, such as pinch-off.

### 5.1 Future Work

Future work would involve further exploring dynamics through simulations that cannot be calculated analytically. This might involve producing phase plots, indicating regions of stability and instability for different parameter values. We would also like to improve the numerics so that we can generate labyrinth patterns. Recent simulations have shown pinch-off. Future work might involve investigating whether pinch-off is energetically favorable as well for what parameters it occurs. New methods would need to be investigated to simulate pinch-off because our current method is unable to deal with separating domains.



# Bibliography

- [1] J. C. ALEXANDER, A. J. BERNOFF, E. K. MANN, J. A. MANN JR., AND J. R. WINTERSMITH, *Domain relaxation in langmuir films*, *J. Fluid Mech.*, 571 (2007), pp. 191–219.
- [2] A. J. BERNOFF, *Slowly varying fully nonlinear wavetrains in the ginzburg-landau equation.*, *Physica D*, 30 (1988), pp. 363–381.
- [3] RICHARD L. BURDEN AND J. DOUGLAS FAIRES, *Numerical Analysis*, Thomson Brooks/Cole, 8 ed., 2005.
- [4] R. DEKOKER AND H. M. MCCONNELL, *Circle to dogbone—shapes and shape transitions of lipid monolayer domains*, *J. Phys. Chem.*, 97 (1993), pp. 13419–13424.
- [5] P. HEINIG, L. E. HELSETH, AND TH. M. FISCHER, *Relaxation of patterns in 2d modulated phases*, *New Journal of Physics*, 6 (2004), p. 189.
- [6] T.Y. HOU, J.S. LOWENGRUB, AND M.J. SHELLEY, *Removing the stiffness from interfacial flow with surface tension*, *Journal of Computational Physics*, 114 (1994), pp. 312–338.
- [7] D. K. LUBENSKY AND R. E. GOLDSTEIN, *Hydrodynamics of monolayer domains at the air-water interface*, *Phys. Fluids*, 8 (1996), pp. 843–854.
- [8] J. M. PUGH, *Numerical simulation of domain relaxation in Langmuir films*. Senior Thesis, Department of Physics, Harvey Mudd College, 2006.
- [9] LLOYD N. TREFETHEN, *Spectral methods in MATLAB*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000.