Study of Vortex Dynamics with Free Surface in a Shallow Water Regime

Christopher Curtis, Ph.D. Robert Insley Eunji Yoo
Hannah Barta Phoebe Coy Thomas Pecha Thomas Retzloff
Alec Todd

August 18, 2017

Contents

Abstract

We examine the deformation of varying shapes of collections of point vortices beneath a free surface of an inviscid fluid with a flat bottom boundary. Initial simulations investigate the deformation of vortex sheets in a manner which resembles the Kelvin-Helmholtz instability. With large numbers of point vortices, the simulations we run show that these sheets tend to deform into elliptical patches. Several other shapes of point vortex arrays are simulated, and almost all deform into an elliptical shape. Upon deciding to simulate an ellipse as a starting shape, we observe that an initial elliptical shape deforms the least with time. To increase the realism of these simulations, we introduce a mollified kernel, which slows the speeds of vortices that are close together and yields ellipses that retain their shape with time. We introduce a metric to measure the deformation of these ellipses before and after mollification.

1 Introduction

The study of point vortices under a free surface in shallow water is a relatively novel and unexplored field in hydrodynamics. While several works examine the impact of point vortices on free surface flows [1, 2, 3, 4, 5, 6, 7], none have thus far developed an approach which allows for arbitrary numbers of vortices beneath a free surface. Such an approach is necessary to develop a point-vortex based solver [8] for a free surface over an arbitrary distribution of vorticity.

Therefore, building off the methodology of [9], in [10] a method was developed for numerically simulating the motion of point vortices in two dimensions under a free surface in an incompressible fluid with an impenetrable flat bottom boundary. Using this method, the authors of [10] perform simulations consisting of two and four point vortices with varying negative and positive circulation strengths. Due to a singularity present in the kernel describing the motion of point vortices, the method in [10] causes vortices to reach velocities approaching infinity when within a sufficiently small distance of each other. Additionally, simulations with larger numbers of vortices have not yet been studied.

To extend the research of [10], we replace the existing singular kernel with a new "mollified" kernel that eliminates the singularity and thus decreases the velocity of vortices that are close together. Through computer simulation, we examine the evolution of vortex sheets as well as various other configurations of point vortices. In several of these simulations, we observe behavior resembling the Kelvin-Helmholtz Instability. Additionally, after numerous simulations of varying size, circulation, and geometry, we observe that an elliptical array of point vortices deforms the least with time; we then quantitatively measure the degree to which these ellipses deform from their initial positions.

The format of this paper is as follows: we first discuss the modeling of continuous patches of vorticity using collections of point vortices. We then explain the calculation and implementation of the mollified kernel, and describe the results of our simulations, particularly concerning the stability of ellipses. Finally, we present topics for further research.

2 The Free Surface Model with Vorticity

We now describe the basis of our research problem, introducing regions of vorticity and the discretization of continuous vortex patches. We introduce the Biot-Savart kernel and provide reasoning for our later work, specifically the implementation of a mollified kernel.

2.1 Continuous Regions of Constant Vorticity

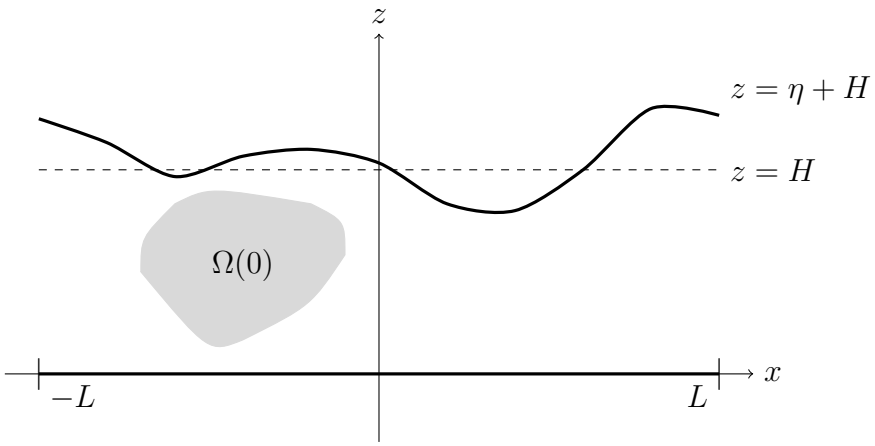


Figure 1: The free surface $z = \eta + H$ and flat bottom $z = 0$ at time $t = 0$, with vortex patch $\Omega(0)$ as shown. H and L represent the mean fluid depth from the bottom and wave length, respectively

Consider an inviscid, incompressible fluid with an impermeable bottom boundary $z = 0$ and free surface $z = \eta(x,t)+H$ in two dimensions. We assume the fluid domain is $2L$-periodic in x so that

$$\eta(x + 2L, t) = \eta(x, t).$$

At time $t = 0$, let $\Omega(0)$ denote a region of constant, nonzero vorticity with circulation strength Γ_0, and let all other points in the fluid have zero vorticity. Then, denoting by $\omega(x,z,t) = \nabla \times \boldsymbol{u}$ the vorticity profile of the fluid at time t, where $\boldsymbol{u}(x,z,t)$ is the fluid velocity profile, we have

$$\boldsymbol{\omega}(x,z,0) = \mathbb{1}_{\Omega(0)}\Gamma_0\hat{k},$$

where $\mathbb{1}_{\Omega(0)}(x,z)$ is the characteristic function on $\Omega(0)$, defined by

$$\mathbb{1}_{\Omega(0)}(x,z) = \begin{cases} 1 & \text{if } (x,z) \in \Omega(0) \\ 0 & \text{otherwise,} \end{cases}$$

and where $\hat{k}$ is the vector of unit length orthogonal to $\hat{x} = (1,0)$ and $\hat{z} = (0,1)$, according to $\hat{k} = \hat{x} \times \hat{z}$. Figure 1 illustrates this model.

Under the assumption of periodic boundary conditions, one may derive from the Biot-Savart Law the following equation for fluid velocity, following [11]:

$$\boldsymbol{u}(\boldsymbol{x},t) = \int_{\Omega(t)} \boldsymbol{K}(\boldsymbol{x},\tilde{\boldsymbol{x}})\,\omega(\tilde{\boldsymbol{x}},t)\,d\tilde{\boldsymbol{x}} + \nabla\phi, \tag{1}$$

where $\boldsymbol{x} = (x,z)$ is position, ω is the signed magnitude of vorticity, $\Omega(t)$ is the support of ω at time t, and $\boldsymbol{K}$ is a Biot-Savart kernel effectively describing the influence of points $\tilde{\boldsymbol{x}}$ with nonzero vorticity on fluid velocity. Additionally, ϕ is the velocity potential due to the bulk and surface of the fluid, subject to the constraints $\Delta\phi = 0$ and $\phi_z(x,0,t) = 0$.

Consider the flow map

$$\phi_t(x_0,z_0) = (x(t),z(t))$$

describing the time-evolution of an initial point $(x_0,z_0) = (x(0),z(0))$ in the fluid. Assuming that $\phi_t : \mathbb{R}^2 \to \mathbb{R}^2$ is differentiable, then it defines a diffeomorphism. Thus, if the initial region of vorticity $\Omega(0)$ is compact and connected, then

$$\Omega(t) = \phi_t(\Omega(0))$$

is compact and connected as well. As this holds for all $t \geq 0$, we see that $\Omega(t)$ is compact and connected at any given time $t \geq 0$.

2.2 Discretization Using Point Vortices

In order to describe the time-evolution of the initial vortex patch $\Omega(0)$, following the approach described in [8], we discretize the vorticity profile with N irrotational point vortices as follows:

$$\omega(\boldsymbol{x},t) = \frac{1}{2\pi}\sum_{\ell=1}^{N} \Gamma_\ell(t)\,\delta(\boldsymbol{x} - \boldsymbol{x}_\ell(t))\,d\boldsymbol{x}, \tag{2}$$

where $\boldsymbol{x}_\ell(t)$ is the position of the ℓ-th point vortex and $\Gamma_\ell(t)$ is its circulation strength. By requiring that the region $\Omega(t)$ follow streamlines of the fluid, we apply Kelvin's circulation theorem [12] and Stokes' theorem in turn to show that the sum of the point vortices' circulation strengths is constant with respect to time. In other words, there is a net circulation constant Γ_ω with

$$\Gamma_\omega = \sum_{\ell=1}^{N} \Gamma_\ell(t)$$

for all t. According to this equation, it is possible for point vortices to "exchange" vorticity as time changes, as long as their circulation strengths $\Gamma_\ell(t)$ sum to the constant Γ_ω. However, for our model we impose the restriction that $\Gamma_\ell(t) = \Gamma_\ell(0)$ for all t, so that each point vortex has constant circulation strength Γ_ℓ as time changes. Note that this restriction ensures constant net circulation Γ_ω according to the equation above.

We now turn our attention to the Biot-Savart kernel [11] $\boldsymbol{K}$ in equation (1), which may be expressed as

$$\boldsymbol{K}(\boldsymbol{x}, \tilde{\boldsymbol{x}}) = \tilde{\boldsymbol{K}}(x - \tilde{x}, z - \tilde{z}) - \tilde{\boldsymbol{K}}(x - \tilde{x}, z + \tilde{z}), \tag{3}$$

where

$$\tilde{\boldsymbol{K}}(x, z) = \frac{(-\sinh(\pi z/L), \sin(\pi x/L))}{4L(\cosh(\pi z/L) - \cos(\pi x/L)}. \tag{4}$$

To derive equations (3) and (4), observe that if we can find a streamfunction ψ satisfying $\boldsymbol{u} = (\psi_z, -\psi_x)$, then we are assured of our fluid being incompressible, as it follows that $\nabla \cdot \boldsymbol{u} = 0$. Additionally, from $\boldsymbol{u} = (\psi_z, -\psi_x)$, one may show that Poisson's equation $-\Delta\psi = \omega$ holds. Then, given a vorticity profile ω, a streamfunction ψ solving this equation is given by

$$\psi(x, z, t) = (G * \omega)(x, z, t) = \int_{\mathbb{R}^2} G(\boldsymbol{x} - \tilde{\boldsymbol{x}})\, \omega(\tilde{\boldsymbol{x}}, t)\, d\tilde{\boldsymbol{x}},$$

where G is Green's function

$$G(\boldsymbol{x}) = -\frac{1}{2\pi} \log(|\boldsymbol{x}|).$$

This streamfunction ψ then yields the fluid velocity

$$\boldsymbol{u}(x, z, t) = \int_{\mathbb{R}^2} \bar{\boldsymbol{K}}(\boldsymbol{x} - \tilde{\boldsymbol{x}})\, \omega(\tilde{\boldsymbol{x}}, t)\, d\tilde{\boldsymbol{x}},$$

where $\bar{\boldsymbol{K}}$ is the function

$$\bar{\boldsymbol{K}}(\boldsymbol{x}) = \frac{(-z, x)}{2\pi|\boldsymbol{x}|^2}.$$

Recalling our assumption that the fluid domain is horizontally $2L$-periodic, every point $(x, z) \in \Omega(t)$ with nonzero vorticity (where we take the region $\Omega(t)$ to be contained in the principal domain $[-L, L] \times \mathbb{R}$) corresponds to infinitely many points

$$\{\dots, (x - 2L, z), (x, z), (x + 2L, z), \dots\}$$

in the intervals

$$\dots, [-3L, -L], [-L, L], [L, 3L], \dots,$$

respectively. Then $\boldsymbol{u}$ may be rewritten as

$$\boldsymbol{u}(x, z, t) = \int_{\Omega(t)} \sum_{m=-\infty}^{\infty} \tau_m^h \left(\bar{\boldsymbol{K}}(\boldsymbol{x} - \tilde{\boldsymbol{x}})\right) \omega(\tilde{\boldsymbol{x}}, t)\, d\tilde{\boldsymbol{x}},$$

where $\tau_m^h \left(\bar{\boldsymbol{K}}(\boldsymbol{x} - \tilde{\boldsymbol{x}})\right) = \bar{\boldsymbol{K}}(x - \tilde{x} - 2mL, z - \tilde{z})$. Then, setting

$$\boldsymbol{K}(\boldsymbol{x}, \tilde{\boldsymbol{x}}) = \sum_{m=-\infty}^{\infty} \tau_m^h \left(\bar{\boldsymbol{K}}(\boldsymbol{x} - \tilde{\boldsymbol{x}})\right)$$

and evaluating the right-hand side yields equations (3) and (4) for the kernel $\boldsymbol{K}$.

Now, substituting the Biot-Savart kernel $\boldsymbol{K}$ and the discretized vorticity profile ω given in (2) into equation (1) for fluid velocity, we obtain

$$\boldsymbol{u}(\boldsymbol{x},t) = \sum_{\ell=1}^{N} \Gamma_\ell \, \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}_\ell(t)) + \nabla\phi. \tag{5}$$

Consider now the velocity of a given point vortex j, described by $\boldsymbol{u}(\boldsymbol{x}_j(t),t)$. In particular, consider a generic summand from (5):

$$\Gamma_\ell \, \boldsymbol{K}(\boldsymbol{x}_j, \boldsymbol{x}_\ell) = \Gamma_\ell \, \tilde{\boldsymbol{K}}(x_j - x_\ell, z_j - z_\ell) - \Gamma_\ell \, \tilde{\boldsymbol{K}}(x_j - x_\ell, z_j + z_\ell).$$

Suppose that point vortices j and ℓ are very close; more precisely, consider the limit $\boldsymbol{x}_j - \boldsymbol{x}_\ell \to \boldsymbol{0}$. Then $x_j - x_\ell \to 0$ and $z_j - z_\ell \to 0$, so that

$$\left|\tilde{\boldsymbol{K}}(x_j - x_\ell, z_j - z_\ell)\right| \to \infty$$

from (4). This singular behavior poses a central problem to computer simulations of interacting point vortices, as any two point vortices that become too close approach rapid speeds. To correct the singular behavior of $\tilde{\boldsymbol{K}}$ near the origin, we seek to replace $\tilde{\boldsymbol{K}}$ with a similar function $\tilde{\boldsymbol{K}}_\epsilon$ that removes the singularity of $\tilde{\boldsymbol{K}}$ near the origin while approximating $\tilde{\boldsymbol{K}}$ away from the origin. The calculation of such a function is the chief concern of the following section.

3 Mollification of the Kernel

Just as the Biot-Savart kernel $\boldsymbol{K}$ is expressed in terms of $\tilde{\boldsymbol{K}}$ according to (3), we define the mollified kernel $\boldsymbol{K}_\epsilon$ in terms of $\tilde{\boldsymbol{K}}_\epsilon$ as follows:

$$\boldsymbol{K}_\epsilon(\boldsymbol{x}, \tilde{\boldsymbol{x}}) = \tilde{\boldsymbol{K}}_\epsilon(x - \tilde{x}, z - \tilde{z}) - \tilde{\boldsymbol{K}}_\epsilon(x - \tilde{x}, z + \tilde{z}),$$

where $\tilde{\boldsymbol{K}}_\epsilon$ is to be determined. In particular, since we wish for $\tilde{\boldsymbol{K}}_\epsilon$ to approximate $\tilde{\boldsymbol{K}}$ while removing its singular behavior, we set

$$\tilde{\boldsymbol{K}}_\epsilon(\boldsymbol{x}) = (\tilde{\boldsymbol{K}} * \chi_\epsilon)(\boldsymbol{x}) = \int_{\mathbb{R}^2} \tilde{\boldsymbol{K}}(\boldsymbol{y}) \, \chi_\epsilon(\boldsymbol{y} - \boldsymbol{x}) \, d\boldsymbol{y}, \tag{6}$$

where χ_ϵ is a cutoff function with parameter ϵ, which we shall define shortly. Note that setting χ_ϵ equal to the Dirac delta function on $\mathbb{R}^2$ yields $\tilde{\boldsymbol{K}}_\epsilon(\boldsymbol{x}) = \tilde{\boldsymbol{K}}(\boldsymbol{x})$, which motivates a choice of cutoff function χ_ϵ that is "like" the delta function in some sense.

In this chapter, we discuss the derivation of a formula for $\tilde{\boldsymbol{K}}_\epsilon$ as an infinite series, and we then re-express it as a Fourier series. To distinguish between these two expressions for $\tilde{\boldsymbol{K}}_\epsilon$, we refer to the former infinite series as the direct summation expression for $\tilde{\boldsymbol{K}}_\epsilon$. We discuss why implementing $\tilde{\boldsymbol{K}}_\epsilon$ as a Fourier series is preferable with regard to computer simulation, and then discuss its effect on the existing code.

3.1 The Gaussian Cutoff Function

Consider a function $\chi : \mathbb{R} \to \mathbb{R}$, and define χ on $\mathbb{R}^2$ by setting $\chi(\boldsymbol{x}) = \chi(|\boldsymbol{x}|)$ for all $\boldsymbol{x} \in \mathbb{R}^2$. Formally, χ is said to be a cutoff function of order n (a positive integer) if it satisfies the following three conditions:

1. $\int_{\mathbb{R}^2} \chi(|\boldsymbol{x}|)\, d\boldsymbol{x} = 2\pi \int_0^\infty r\, \chi(r)\, dr = 1$.

2. For all positive integers n_1 and n_2 such that $0 < \sqrt{n_1^2 + n_2^2} \leq n - 1$, one has

$$\int_{\mathbb{R}^2} x^{n_1} z^{n_2}\, \chi(x, z)\, dx\, dz = 0.$$

3. For all positive integers n_1 and n_2 such that $\sqrt{n_1^2 + n_2^2} = n$, one has

$$\int_{\mathbb{R}^2} x^{n_1} z^{n_2}\, \chi(x, z)\, dx\, dz < \infty.$$

Now consider

$$\chi_\epsilon(r) = \frac{1}{\epsilon^2} \chi\left(\frac{r}{\epsilon}\right) \tag{7}$$

where

$$\chi(r) = \frac{1}{\pi} e^{-r^2} \tag{8}$$

is a Gaussian curve. Then χ is a cutoff function of order 2, which we now verify. In order to check condition 1, we compute

$$2\pi \int_0^\infty \frac{r e^{-r^2}}{\pi}\, dr = \int_0^\infty e^{-u}\, du = 1$$

using the change of variables $u = r^2$. For condition 2, we see that the only positive-integer pairs (n_1, n_2) satisfying $0 < \sqrt{n_1^2 + n_2^2} \leq 1$ are $(0, 1)$ and $(1, 0)$. In the case where $(n_1, n_2) = (0, 1)$, we notice that $z\, \chi(x, z)$ is odd in z, as

$$-z\, \chi(x, -z) = -z\, \chi(x, z),$$

so that

$$\int_{\mathbb{R}^2} z\, \chi(x, z)\, dx\, dz = 0.$$

The same reasoning applies to the case in which $n_1 = 1$ and $n_2 = 0$. Finally, by noting that the only positive integer pairs satisfying $\sqrt{n_1^2 + n_2^2} = 2$ are $(0, 2)$ and $(2, 0)$, we can confirm condition 3. If $(n_1, n_2) = (0, 2)$, then we change the integral $\int_{\mathbb{R}^2} z^2 \chi(x, z)\, dx\, dz$ to polar coordinates and take absolute value to obtain

$$\left| \int_0^{2\pi} \int_0^\infty r^3 \chi(r) \sin^2\theta\, dr\, d\theta \right| \leq \int_0^{2\pi} \int_0^\infty \left| r^3 \chi(r) \sin^2\theta \right| dr\, d\theta \leq 2\pi \int_0^\infty r^3 \chi(r)\, dr$$

One may then show that the rightmost integral converges. Specifically, write

$$2\pi \int_0^\infty r^3 \chi(r)\, dr = 2\int_0^1 r^3 e^{-r^2}\, dr + 2\int_1^\infty r^3 e^{-r^2}\, dr,$$

and notice that the integral from 0 to 1 on the right-hand side converges, since the integrand is continuous. Then, using the fact that $e^{-r^2} \leq e^{-r}$ for all $r \geq 1$, we note that

$$2\int_1^\infty r^3 e^{-r^2}\, dr \leq 2\int_1^\infty r^3 e^{-r}\, dr.$$

One may then show by induction that $\int_1^\infty r^n e^{-r}\, dr < \infty$ for all positive integers n, using integration by parts for the inductive step. Thus, we have shown that $\chi(r)$ as defined in (8) is a cutoff function of order 2.

3.2 The Mollified Kernel $\boldsymbol{K}_\epsilon$

Substituting the Gaussian cutoff function χ_ϵ given in (7) into the convolution formula (6) for $\tilde{\boldsymbol{K}}_\epsilon$, we obtain

$$\tilde{\boldsymbol{K}}_\epsilon(\boldsymbol{x}) = \sum_{m=-\infty}^{\infty} \tau_m^h \left(\frac{(-z, x)}{|\boldsymbol{x}|^2} \int_0^{|\boldsymbol{x}|} r\, \chi_\epsilon(r)\, dr \right), \tag{9}$$

where τ_m^h denotes the horizontal translation operator $\tau_m^h(f(x,z)) = f(x - 2mL, z)$. One may then show that

$$\frac{1}{|\boldsymbol{x}|^2}\int_0^{|\boldsymbol{x}|} r\, \chi_\epsilon(r)\, dr = \frac{1}{|\boldsymbol{x}|^2}\int_0^{|\boldsymbol{x}|} \frac{re^{-r^2/\epsilon^2}}{\pi\epsilon^2}\, dr = \frac{1 - e^{-|\boldsymbol{x}|^2/\epsilon^2}}{2\pi|\boldsymbol{x}|^2}.$$

Note that this expression has a removable singularity at $|\boldsymbol{x}| = 0$, as

$$\lim_{|\boldsymbol{x}|\to 0} \frac{1 - e^{-|\boldsymbol{x}|^2/\epsilon^2}}{2\pi|\boldsymbol{x}|^2} = \frac{1}{2\pi\epsilon^2}.$$

Then (9) becomes

$$\tilde{\boldsymbol{K}}_\epsilon(x,z) = \sum_{m=-\infty}^{\infty} \frac{1 - e^{-[(x-2mL)^2+z^2]/\epsilon^2}}{2\pi[(x-2mL)^2+z^2]}(-z, x-2mL), \tag{10}$$

where the m-th term is defined to be 0 in the case where $(x-2mL)^2 + z^2 = 0$. We henceforth refer to (10) as the "direct summation" form of $\tilde{\boldsymbol{K}}_\epsilon$.

Notice that $\tilde{\boldsymbol{K}}_\epsilon$ is $2L$-periodic in x, as one may show that $\tilde{\boldsymbol{K}}_\epsilon(x+2L, z) = \tilde{\boldsymbol{K}}_\epsilon(x,z)$ by re-indexing the summation above. Thus, we may expand the components of $\tilde{\boldsymbol{K}}_\epsilon$ as Fourier series in x:

$$f(x,z) = \sum_{m=-\infty}^{\infty} \hat{f}_m(z) e^{i\pi m x/L}, \tag{11}$$

$$g(x,z) = \sum_{m=-\infty}^{\infty} \hat{g}_m(z) e^{i\pi m x/L}, \tag{12}$$

where $\tilde{\boldsymbol{K}}_\epsilon(x,z) = (f(x,z), g(x,z))$. We now claim that the Fourier coefficients for the series above are given by

$$\hat{f}_m(z) = \frac{1}{2L}\int_{-\infty}^{\infty} \frac{-z(1-e^{-[x^2+z^2]/\epsilon^2})}{2\pi[x^2+z^2]} e^{-i\pi mx/L}\,dx, \tag{13}$$

$$\hat{g}_m(z) = \frac{1}{2L}\int_{-\infty}^{\infty} \frac{x(1-e^{-[x^2+z^2]/\epsilon^2})}{2\pi[x^2+z^2]} e^{-i\pi mx/L}\,dx. \tag{14}$$

Using orthogonality relations, one may verify the following formula:

$$\hat{g}_m(z) = \frac{1}{2L}\int_{-L}^{L} g(x,z) e^{-i\pi mx/L}\,dx.$$

Then we have

$$\begin{aligned}
\hat{g}_m(z) &= \frac{1}{2L}\int_{-L}^{L} \sum_{k=-\infty}^{\infty} \frac{(x-2kL)(1-e^{-[(x-2kL)^2+z^2]/\epsilon^2})}{2\pi[(x-2kL)^2+z^2]} e^{-i\pi mx/L}\,dx \\
&= \frac{1}{2L}\sum_{k=-\infty}^{\infty}\int_{-L}^{L} \frac{(x-2kL)(1-e^{-[(x-2kL)^2+z^2]/\epsilon^2})}{2\pi[(x-2kL)^2+z^2]} e^{-i\pi mx/L}\,dx \\
&= \frac{1}{2L}\sum_{k=-\infty}^{\infty}\int_{-L-2kL}^{L-2kL} \frac{x(1-e^{-[x^2+z^2]/\epsilon^2})}{2\pi[x^2+z^2]} e^{-i\pi m(x+2kL)/L}\,dx \\
&= \frac{1}{2L}\int_{-\infty}^{\infty} \frac{x(1-e^{-[x^2+z^2]/\epsilon^2})}{2\pi[x^2+z^2]} e^{-i\pi mx/L}\,dx.
\end{aligned}$$

In the second line, we assume that the series for $g(x,z)$ may be integrated term-by-term. In the third line, we use a change of variables. In the fourth line, we note that

$$e^{-i\pi m(x+2kL)/L} = e^{-i\pi mx/L}$$

and that the intervals $[-L-2kL, L-2kL]$ cover the real line. Formula (13) follows the same reasoning.

We now evaluate the Fourier coefficients $\hat{f}_m(z)$ and $\hat{g}_m(z)$. First, we rewrite (13) as

$$\hat{f}_m(z) = \frac{1}{2L}\int_{-\infty}^{\infty} \frac{-ze^{-i\pi mx/L}}{2\pi[x^2+z^2]}\,dx + \frac{1}{2L}\int_{-\infty}^{\infty} \frac{ze^{-[x^2+z^2]/\epsilon^2}}{2\pi[x^2+z^2]} e^{-i\pi mx/L}\,dx. \tag{15}$$

We now evaluate each of these integrals separately. Define the Fourier transform of $f(x)$ to be

$$\hat{f}(\omega) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x}\,dx$$

and the inverse Fourier transform of $\hat{f}(\omega)$ to be

$$f(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}(\omega) e^{i\omega x}\,d\omega.$$

Recall the convolution theorem, which states that for functions $f(x)$ and $g(x)$, we have

$$\widehat{(fg)}(\omega) = \frac{1}{2\pi}(\hat{f} * \hat{g})(\omega),$$

where the right-hand side denotes the convolution of $\hat{f}(\omega)$ and $\hat{g}(\omega)$, defined as

$$(\hat{f} * \hat{g})(\omega) = \int_{-\infty}^{\infty} \hat{f}(\xi)\, \hat{g}(\omega - \xi)\, d\xi.$$

In other words, the Fourier transform of a product fg is the convolution of the Fourier transforms of f and g, up to a scalar multiple (in our case, $1/2\pi$, given our choice of convention for the Fourier transform). Now we claim that the Fourier transform of $h_1(x) = \frac{z}{x^2+z^2}$ is $\hat{h}_1(\omega) = \pi e^{-z|\omega|}$, which is readily obtained from the following formula:

$$\frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{2\alpha}{\alpha^2 + \omega^2} e^{i\omega x}\, d\omega = e^{-\alpha|x|}.$$

Then the first integral in (15) may be rewritten as

$$-\frac{1}{4\pi L} \hat{h}_1 \left(\frac{\pi m}{L}\right) = \frac{1}{4L} e^{-z|\pi m/L|}$$

Now let $h_2(x) = e^{-[x^2+z^2]/\epsilon^2}$; then the second integral in (15) may be rewritten as

$$\frac{1}{4\pi L} \widehat{(h_1 h_2)} \left(\frac{\pi m}{L}\right) = \frac{1}{8\pi^2 L} (\hat{h}_1 * \hat{h}_2) \left(\frac{\pi m}{L}\right), \tag{16}$$

using the convolution theorem. Now, to find $\hat{h}_1 * \hat{h}_2$, one may compute

$$\hat{h}_2(\omega) = \epsilon\sqrt{\pi} e^{-z^2/\epsilon^2} e^{-\epsilon^2\omega^2/4}$$

then write

$$\begin{aligned}
(\hat{h}_1 * \hat{h}_2)(\omega) &= \int_{-\infty}^{\infty} \hat{h}_1(\xi) \hat{h}_2(\omega - \xi)\, d\xi \\
&= \epsilon\pi^{3/2} e^{-z^2/\epsilon^2} \int_{-\infty}^{\infty} \exp\left[-z|\xi| - \frac{\epsilon^2}{4}(\omega - \xi)^2\right] d\xi \\
&= \epsilon\pi^{3/2} e^{-z^2/\epsilon^2} \int_{-\infty}^{\infty} \exp\left[-\frac{\epsilon^2}{4}\xi^2 + \frac{\epsilon^2\omega}{2}\xi - \frac{\epsilon^2\omega^2}{4} - z|\xi|\right] d\xi,
\end{aligned}$$

Observing that the integral in the last line is of the form

$$\int_{-\infty}^{\infty} e^{ax^2+bx+c+d|x|}\, dx,$$

with $a < 0$, we apply the integral formula (22) (see Appendix A) to evaluate this integral. Then we obtain

$$\hat{f}_m(z) = \frac{1}{4L}\left[e^{z\pi m/L}\frac{E_1(z)}{2} + e^{-z\pi m/L}\frac{E_2(z)}{2} - e^{-z|\pi m/L|}\right] \tag{17}$$

where

$$E_1(z) = \begin{cases} 1 - \operatorname{erf}\left|\frac{\epsilon\pi m}{2L} + \frac{z}{\epsilon}\right| & \text{if } z > -\frac{\epsilon^2\pi m}{2L} \\ 1 + \operatorname{erf}\left|\frac{\epsilon\pi m}{2L} + \frac{z}{\epsilon}\right| & \text{if } z \leq -\frac{\epsilon^2\pi m}{2L}, \end{cases}$$

$$E_2(z) = \begin{cases} 1 + \operatorname{erf}\left|\frac{\epsilon\pi m}{2L} - \frac{z}{\epsilon}\right| & \text{if } z < \frac{\epsilon^2\pi m}{2L} \\ 1 - \operatorname{erf}\left|\frac{\epsilon\pi m}{2L} - \frac{z}{\epsilon}\right| & \text{if } z \geq \frac{\epsilon^2\pi m}{2L}. \end{cases}$$

Similarly, the Fourier coefficients $\hat{g}_m(z)$ are given by the formula

$$\hat{g}_m(z) = \frac{i}{4L}\left[e^{z\pi m/L}\left(1 - \frac{E_1(z)}{2} - \theta(m)\right) + e^{-z\pi m/L}\left(\frac{E_2(z)}{2} - \theta(m)\right)\right] \tag{18}$$

where i is the imaginary unit,

$$\theta(x) = \begin{cases} 0 & \text{if } x < 0 \\ 1/2 & \text{if } x = 0 \\ 1 & \text{if } x > 0 \end{cases}$$

is the Heaviside step function, and where E_1 and E_2 are as before. To obtain this formula, we rewrite (14) as

$$\hat{g}_m(z) = \frac{1}{4\pi L}\hat{h}_3\left(\frac{\pi m}{L}\right) - \frac{1}{8\pi^2 L}(\hat{h}_2 * \hat{h}_3)\left(\frac{\pi m}{L}\right),$$

where

$$h_3(x) = \frac{x}{x^2 + z^2} = \frac{1}{2(x + iz)} + \frac{1}{2(x - iz)}.$$

Then, using the fact that

$$\frac{1}{2\pi}\int_{-\infty}^{\infty}\frac{1}{\alpha + ix}e^{i\omega x}\,dx = \theta(\omega)e^{-\alpha\omega}$$

for $\alpha \in \mathbb{R}$, one may compute $\hat{h}_3(\omega) = -i\pi\theta(\omega)(e^{z\omega} + e^{-z\omega})$. Then one may find the convolution $\hat{h}_2 * \hat{h}_3$ using similar reasoning to the derivation of Formula (22) in order to obtain the formula for the Fourier coefficient $\hat{g}_m(z)$.

3.3 Non-Dimensionalization of Vortex Velocities

We now seek formulas describing the motion of the point vortices in order to numerically simulate their motion. First, note that the velocity of a given vortex j at time t is given by $\boldsymbol{u}(x_j(t), z_j(t), t)$, where $\boldsymbol{x}_j(t) = (x_j(t), z_j(t))$ is the position of point vortex j at time t. In particular, using the discretized fluid velocity formula (5) and replacing the problematic kernel $\boldsymbol{K}$ with the mollified kernel $\boldsymbol{K}_\epsilon$, we have

$$\boldsymbol{u}(\boldsymbol{x}_j, t) = \sum_{\ell=1}^{N} \Gamma_\ell \, \boldsymbol{K}_\epsilon(\boldsymbol{x}_j, \boldsymbol{x}_\ell) + \nabla\phi.$$

In this section, we consider the partial velocity of vortex j,

$$(\dot{x}_j, \dot{z}_j) = \sum_{\ell=1}^{N} \Gamma_\ell \, \boldsymbol{K}_\epsilon(\boldsymbol{x}_j, \boldsymbol{x}_\ell),$$

dropping the bulk/surface potential gradient $\nabla\phi$. In particular, we seek to express the partial velocity components $\dot{x}_j$ and $\dot{z}_j$ in terms of dimensionless quantities, which allows us to characterize vortex behavior irrespective of length and time scales.

We introduce the rescaled variables

$$\tilde{x} = \frac{x}{L}, \qquad \tilde{z} = \frac{z}{H}, \qquad \tilde{t} = \frac{\sqrt{gH}}{L} t,$$

as well as the nondimensional parameters

$$\tilde{\Gamma}_j = \frac{\Gamma_j}{\Gamma}, \qquad \tilde{\epsilon} = \frac{\epsilon}{L}, \qquad \mu = \frac{d}{H}, \qquad \gamma = \frac{H}{L}, \qquad F = \frac{\Gamma}{\mu L \sqrt{gH}}.$$

Under this choice of dimensionless quantities, the non-dimensionalized horizontal component of velocity is

$$\frac{d\tilde{x}}{d\tilde{t}} = \frac{d\tilde{x}}{dx}\frac{dx}{dt}\frac{dt}{d\tilde{t}} = \frac{1}{\sqrt{gH}}\frac{dx}{dt}.$$

Then rescaling $\dot{x}_j$ by $1/\sqrt{gH}$ yields

$$\frac{1}{\sqrt{gH}}\dot{x}_j(t) = \frac{1}{\sqrt{gH}} \sum_{\ell=1}^{N} \Gamma_\ell [\tilde{K}_\epsilon^1(x_j - x_\ell, z_j - z_\ell) - \tilde{K}_\epsilon^1(x_j - x_\ell, z_j + z_\ell)], \tag{19}$$

where $\tilde{K}_\epsilon^1$ denotes the first component of $\tilde{\boldsymbol{K}}_\epsilon$. Similarly, one may compute

$$\frac{d\tilde{z}}{d\tilde{t}} = \frac{L/H}{\sqrt{gH}}\frac{dz}{dt}$$

and rescale $\dot{z}_j$ accordingly to obtain

$$\frac{L/H}{\sqrt{gH}}\dot{z}_j(t) = \frac{L/H}{\sqrt{gH}} \sum_{\ell=1}^{N} \Gamma_\ell [\tilde{K}_\epsilon^2(x_j - x_\ell, z_j - z_\ell) - \tilde{K}_\epsilon^2(x_j - x_\ell, z_j + z_\ell)], \tag{20}$$

3.3.1 Direct Summation Form of $\tilde{\boldsymbol{K}}_\epsilon$

We now derive non-dimensional formulas for the partial velocity components in Equations (19) and (20), using the direct sum form of $\tilde{\boldsymbol{K}}_\epsilon$ in (10). One may show that the non-dimensionalized horizontal component (19) becomes

$$-\frac{\mu\gamma F}{2\pi}\sum_{\ell=1}^{N}\tilde{\Gamma}_\ell\sum_{m=-\infty}^{\infty}\left[\frac{(\tilde{z}_j-\tilde{z}_\ell)(1-e^{-[(\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j-\tilde{z}_\ell)^2]/\tilde{\epsilon}^2})}{(\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j-\tilde{z}_\ell)^2}\right.$$
$$\left.-\frac{(\tilde{z}_j+\tilde{z}_\ell)(1-e^{-[(\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j+\tilde{z}_\ell)^2]/\tilde{\epsilon}^2})}{(\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j+\tilde{z}_\ell)^2}\right],$$

where, in the singular cases,

$$\begin{cases}\frac{(\tilde{z}_j-\tilde{z}_\ell)(1-e^{-[(\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j-\tilde{z}_\ell)^2]/\tilde{\epsilon}^2})}{(\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j-\tilde{z}_\ell)^2}=0 & \text{if } (\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j-\tilde{z}_\ell)^2=0\\ \frac{(\tilde{z}_j+\tilde{z}_\ell)(1-e^{-[(\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j+\tilde{z}_\ell)^2]/\tilde{\epsilon}^2})}{(\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j+\tilde{z}_\ell)^2}=0 & \text{if } (\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j+\tilde{z}_\ell)^2=0.\end{cases}$$

Likewise, the vertical component (20) becomes

$$\frac{\mu F}{2\pi\gamma}\sum_{\ell=1}^{N}\tilde{\Gamma}_\ell\sum_{m=-\infty}^{\infty}(\tilde{x}_j-\tilde{x}_\ell-2m)\left[\frac{1-e^{-[(\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j-\tilde{z}_\ell)^2]/\tilde{\epsilon}^2}}{(\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j-\tilde{z}_\ell)^2}\right.$$
$$\left.-\frac{1-e^{-[(\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j+\tilde{z}_\ell)^2]/\tilde{\epsilon}^2}}{(\tilde{x}_j-\tilde{x}_\ell-2m)^2+\gamma^2(\tilde{z}_j+\tilde{z}_\ell)^2}\right],$$

with the singular cases dealt with as above.

3.3.2 Fourier Series Expansion of $\tilde{\boldsymbol{K}}_\epsilon$

Expanding $K_\epsilon^1(x,z)=f(x,z)$ as a Fourier series, and using Formula (17) for the Fourier coefficients $\hat{f}_m(z)$, Equation (19) becomes

$$\mu F\sum_{\ell=1}^{N}\tilde{\Gamma}_\ell\sum_{m=-\infty}^{\infty}\left[\tilde{\hat{f}}_m(\tilde{z}_j-\tilde{z}_\ell)-\tilde{\hat{f}}_m(\tilde{z}_j+\tilde{z}_\ell)\right]e^{i\pi m(\tilde{x}_j-\tilde{x}_\ell)},$$

or equivalently,

$$\mu F\sum_{\ell=1}^{N}\tilde{\Gamma}_\ell\left[\tilde{\hat{f}}_0(\tilde{z}_j-\tilde{z}_\ell)-\tilde{\hat{f}}_0(\tilde{z}_j+\tilde{z}_\ell)+2\sum_{m=1}^{\infty}\left[\tilde{\hat{f}}_m(\tilde{z}_j-\tilde{z}_\ell)-\tilde{\hat{f}}_m(\tilde{z}_j+\tilde{z}_\ell)\right]\cos(\pi m(\tilde{x}_j-\tilde{x}_\ell))\right],$$

where

$$\tilde{\hat{f}}_m(\tilde{z})=\frac{1}{4}\left[e^{\gamma\tilde{z}\pi m}\frac{\tilde{E}_1(\tilde{z})}{2}+e^{-\gamma\tilde{z}\pi m}\frac{\tilde{E}_2(\tilde{z})}{2}-e^{-\gamma\tilde{z}\pi|m|}\right],$$

and where

$$\tilde{E}_1(\tilde{z}) = \begin{cases} 1 - \operatorname{erf}\left|\frac{\tilde{\epsilon}\pi m}{2} + \frac{\gamma\tilde{z}}{\tilde{\epsilon}}\right| & \text{if } \tilde{z} > -\frac{\tilde{\epsilon}^2\pi m}{2\gamma} \\ 1 + \operatorname{erf}\left|\frac{\tilde{\epsilon}\pi m}{2} + \frac{\gamma\tilde{z}}{\tilde{\epsilon}}\right| & \text{if } \tilde{z} \leq -\frac{\tilde{\epsilon}^2\pi m}{2\gamma}, \end{cases}$$

$$\tilde{E}_2(\tilde{z}) = \begin{cases} 1 + \operatorname{erf}\left|\frac{\tilde{\epsilon}\pi m}{2} - \frac{\gamma\tilde{z}}{\tilde{\epsilon}}\right| & \text{if } \tilde{z} < \frac{\tilde{\epsilon}^2\pi m}{2\gamma} \\ 1 - \operatorname{erf}\left|\frac{\tilde{\epsilon}\pi m}{2} - \frac{\gamma\tilde{z}}{\tilde{\epsilon}}\right| & \text{if } \tilde{z} \geq \frac{\tilde{\epsilon}^2\pi m}{2\gamma}. \end{cases}$$

Similarly, by expanding $K_\epsilon^2(x,z) = g(x,z)$ as a Fourier series, using Formula (18) for the coefficients $\hat{g}_m(z)$, Equation (20) becomes

$$\frac{\mu F}{\gamma} \sum_{\ell=1}^{N} \tilde{\Gamma}_\ell \sum_{m=-\infty}^{\infty} \left[\tilde{\hat{g}}_m(\tilde{z}_j - \tilde{z}_\ell) - \tilde{\hat{g}}_m(\tilde{z}_j + \tilde{z}_\ell)\right] e^{i\pi m(\tilde{x}_j - \tilde{x}_\ell)},$$

or equivalently,

$$\frac{2i\mu F}{\gamma} \sum_{\ell=1}^{N} \tilde{\Gamma}_\ell \sum_{m=1}^{\infty} \left[\tilde{\hat{g}}_m(\tilde{z}_j - \tilde{z}_\ell) - \tilde{\hat{g}}_m(\tilde{z}_j + \tilde{z}_\ell)\right] \sin(\pi m(\tilde{x}_j - \tilde{x}_\ell)),$$

where

$$\tilde{\hat{g}}_m(\tilde{z}) = \frac{i}{4}\left[e^{\gamma\tilde{z}\pi m}\left(1 - \frac{\tilde{E}_1(\tilde{z})}{2} - \theta(m)\right) + e^{-\gamma\tilde{z}\pi m}\left(\frac{\tilde{E}_2(\tilde{z})}{2} - \theta(m)\right)\right],$$

and where $\tilde{E}_1$ and $\tilde{E}_2$ are as above.

3.4 Selection of Kernel

After computing and implementing into the existing code both choices for the mollified kernel, we decide which is preferable for our simulations. Runtime is a consideration, but precision is more important. We want our mollified kernel to compute accurate simulations for low values of truncation. Therefore, we compute the difference between kernels, one truncated at ten terms higher than the other. Figure 2 shows these differences for increasing limits of truncation, with the blunt sum kernel in red and the Fourier series kernel in blue.

The Fourier series kernel converges very quickly, while the blunt sum kernel exhibits slow convergence. Therefore, we choose the Fourier series version of the mollified kernel for our final implementation of the code. The elementary example in figure 3 demonstrates that the mollified code causes vortices to behave in a more reasonable manner when close together, by not undergoing the unnaturally fast movement caused by the singularity in $\boldsymbol{K}$.

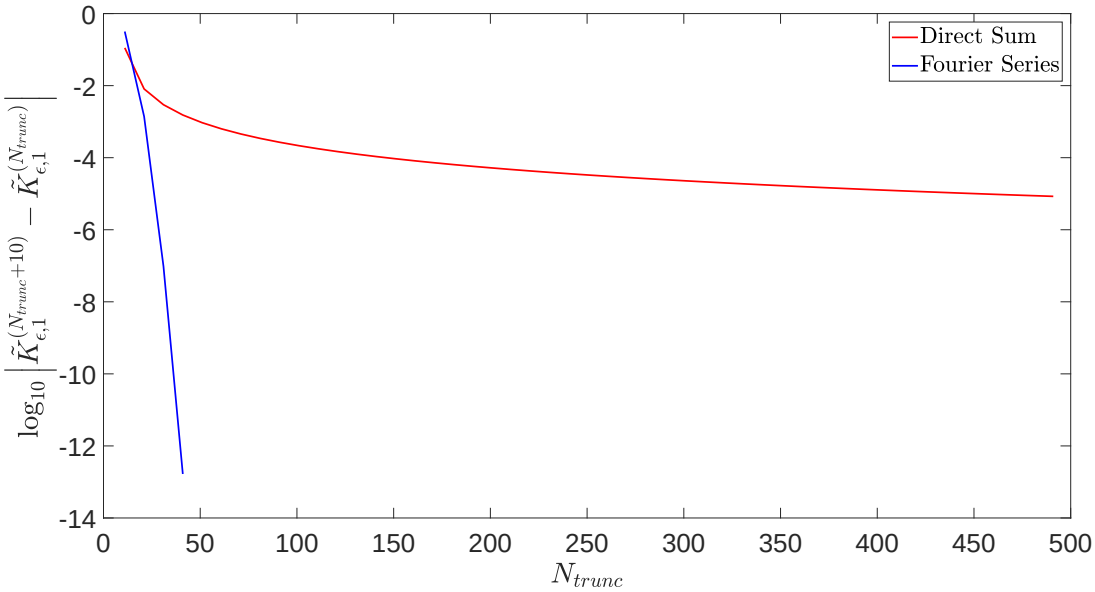
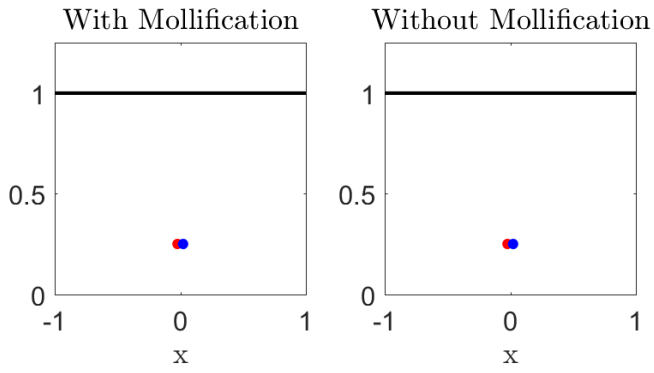
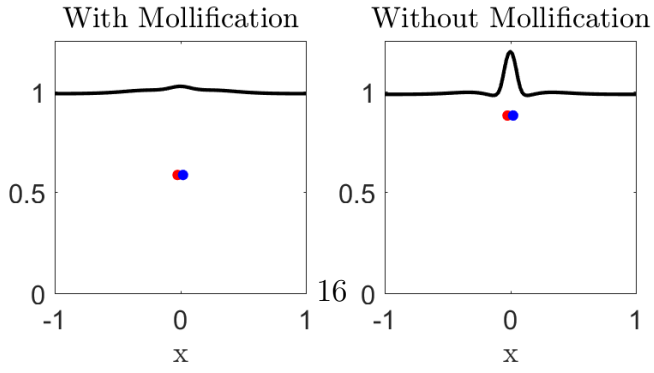


Figure 2: The x-axis is the limit of truncation of the respective series for selected x_j, x_l, z_j, z_l values, with $\epsilon = 0.05$. (Plots of other x and z values look very similar)





The increased realism and accuracy of the Fourier series kernel comes at a price, namely runtime. Despite our efforts to decrease runtime, such as implementing symmetries of the Fourier coefficients (namely $\hat{f}_m = \hat{f}_{-m}$ and $\hat{g}_m = -\hat{g}_{-m}$), and taking advantage of Matlab's efficient calculation of vectorized quantities, the code with the Fourier series kernel takes significantly longer to run than the method from [10].

As an example, using parameters rows=1, cols=8, $K = 128$, $\mu = 0.2$, $\gamma = \sqrt{0.2}$, $F = 0.2$, and $tf = 15$, the non-mollified code had a runtime of 25.6 seconds. The Fourier series mollified code, with additional parameters $\epsilon = .05$ and Ntrunc=250, had a runtime of 108.2 seconds.

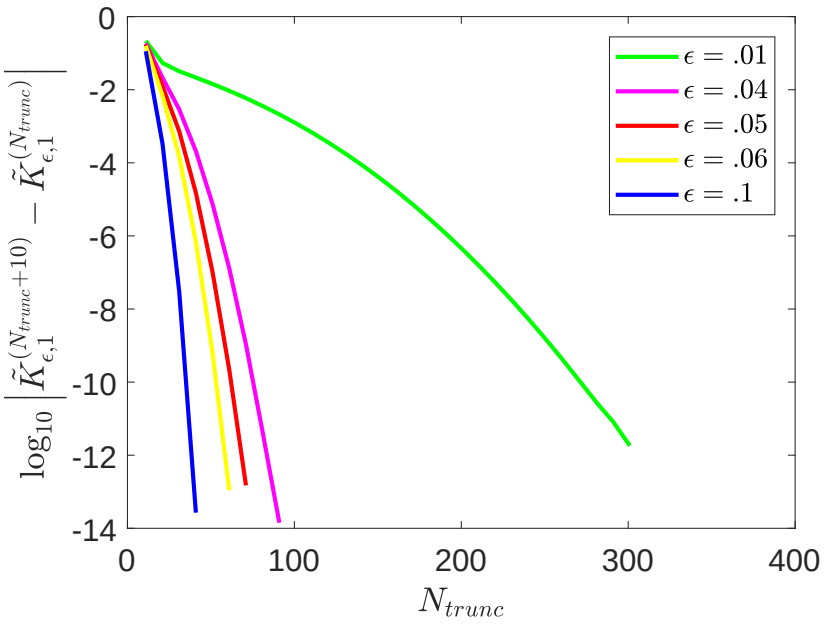


Figure 4: The x-axis is the limit of truncation of the respective Fourier series. Parameters used were $x_j = -0.7, x_l = 0.3, z_j = 0.4, z_l = 0.5$.

We also consider which value of ϵ is preferable to use for our mollifier. We aim to find a "Goldilocks" value, not too large and not too small. If the value is too small, we lose accuracy, and since the mollified kernel is very close to the original kernel, the speed of close-by vortices will still be too high. However, if ϵ is large, we may "over-mollify" and move farther from real vortex behavior. Figure 4 shows the same information as 2, but for different values of ϵ used in the Fourier version of the kernel. We see that $\epsilon = 0.05$ appears to be a good choice, as there is not a major increase in rate of convergence for lower values.

4 Numerical Results

We study the possibility that a patch of vortex points assembled in a specific shape preserves its structure over time. We begin with a description of the parameters used by our functions

in `Matlab`, and next examine a rectangular sheet of points. These rectangular sheets are composed of at least two rows of closely-spaced point vortices which stretch in a line several hundred point vortices long. Recall from section 3.3 that

$$\mu = \frac{d}{H}$$

and

$$\gamma = \frac{H}{L}.$$

When we take the product of these two parameters, we have $\mu\gamma$, or

$$\mu\gamma = \frac{d}{L}$$

which is a measurement for the distance between vortices. Changing one or both of these parameters then allows us to define what we mean by "closely-spaced."

We proceed to note that each vortex sheet exhibits the property of curling into itself on the edges, and we call this property "roll-up." The level of roll-up along the edges of the sheet is observed to be consistent with the Kelvin-Helmholtz Instability [13]. Additionally, it is discovered that ellipses preserve their shapes over a relatively long runtime. Furthermore, we analyze the mollification effects described in section 3 and compare simulations with and without mollification, and subsequently show that mollification improves the realism of vortex behavior by smoothing out the effects the vortices have on each other.

Finally, we create a metric which measures how much each ellipse has deformed relative to its initial configuration. This is expressed through a series of different tests, which compare the original shape to a reference frame, utilize the ellipse equation, and track clustering throughout the vortex patch.

4.1 Initial Conditions

In this section, we consider the initial conditions and the various parameters important to our experiments. In our rectangular vortex sheets, we use as many as two thousand point vortices, whereas in our simulations involving elliptical patches, we may use as few as thirty. In the majority of our simulations, we rely on the use of 128 pseudo-spectral modes, denoted in our `Matlab` code as `K`. In some of our simulations, we test using as low as 32 modes or 64 modes. However, we find that 128 modes provide greater frequency spectrum accuracy, since K accounts for the discretization of section 2, and thus for the accuracy of our approximations.

In addition to our parameters μ and γ, we have our Froude number F, which is first defined as

$$F = \frac{\Gamma}{\mu L\sqrt{gH}},$$

where Γ represents our vortices' circulation strengths, and where g is our gravity constant by [10]. We maintain our Froude number at $F = 0.2$ for our simulations. We fix our time step,

denoted by δt computationally and by `dt` in our `Matlab` functions, as $\delta t = 1 \times 10^{-2}$. With our usage of the fourth-order Runge-Kutta approximation method, our error from the time step is an estimated 10^{-8} [10]. We have ϵ, a measurement of the amount of mollification present in the model and which is kept at a value of either .05 or .1 for the purposes of our simulations. Finally, we have N_{trunc}, the number at which the kernel sum is truncated. Our simulations with mollification maintain N_{trunc} at either 100 or 150. Finally, in the descriptions of our simulations that follow, we have t_f as the time value for which the simulation runs, and each simulation also produces a value called the RK4 stability parameter, which is a balance between K, the number of modes we choose, and our time step δt.

4.2 Vortex Sheets

Our first experiments concern rectangular sheets of point vortices, which we call vortex sheets, and which provide evidence of the realism of our simulations. Upon simulating a long, thin sheet of point vortices, the edges of the sheet exhibit roll-up or a curling along the edges of the sheet. This roll-up is seen in the Kelvin-Helmholtz Instability, a phenomenon of fluid sheets where, for example, two fluids with different velocities meet and interact according to difference in velocity and then change due to differences in air speeds and pressures [13]. The fact that our sheets exhibit this characteristic is a strong indicator that our simulations are realistically modeling fluid dynamics. Of further note is the fact that our vortex sheets curl in on themselves, and tend to deform into an ellipse. This tendency provides the basis for many of our experiments that regard stable ellipses (see section 4.3) and a deformation metric for said ellipses (see section 4.4).

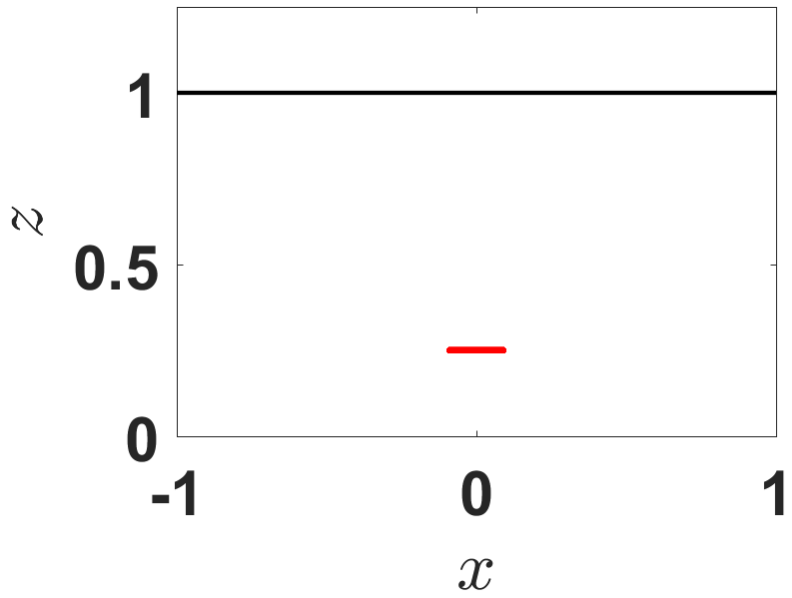


Figure 5: For this simulation, we had 2000 vortices, $K = 128$, $\mu = .1$, $\gamma = \sqrt{.2}$, $F = 0.2$, and $t_f = 10$. Our stability parameter is RK4 = .2987, and runtime was 2322.017857 seconds.

The above figure showcases the rectangular sheet of point vortices in its initial state, before any "roll-up" occurs. Other aspects of the code we changed for this simulation included `z-range = .004`, the use of Cartesian coordinates, and turning on the `sqr-grid` parameter. We describe `z-range` as the vertical range or diameter of the vortex formation, while `sqr-grid` is characterized by allowing the grid spacings in both the z and x directions to be the same. These grid spacings are denoted in Matlab as `dx` and `dz`.

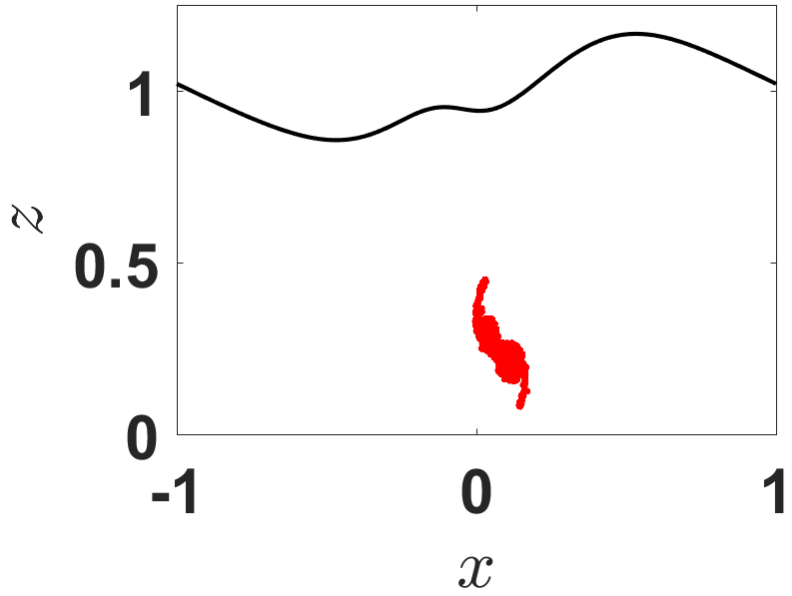


Figure 6: The tenth frame in this animation shows some deformation and roll-up of the vortex sheet shown previously.

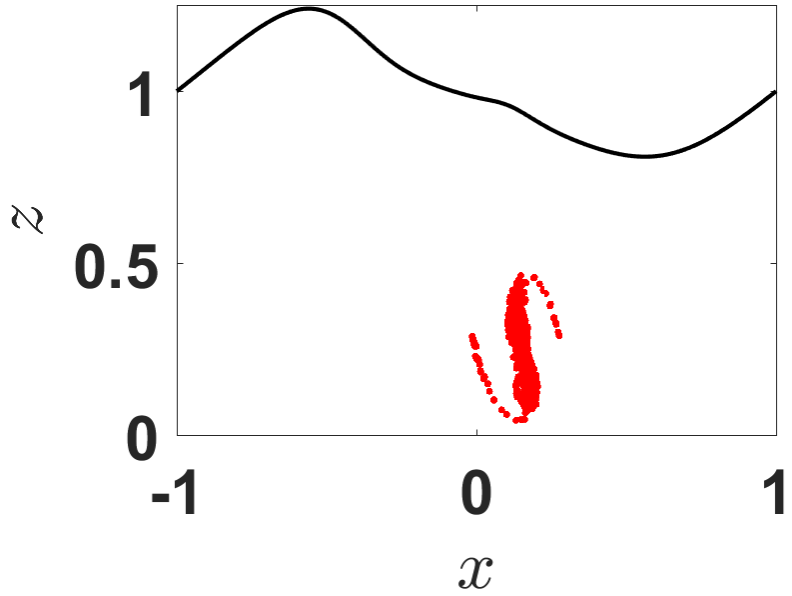


Figure 7: The vortex sheet begins to roll into an "S" configuration.

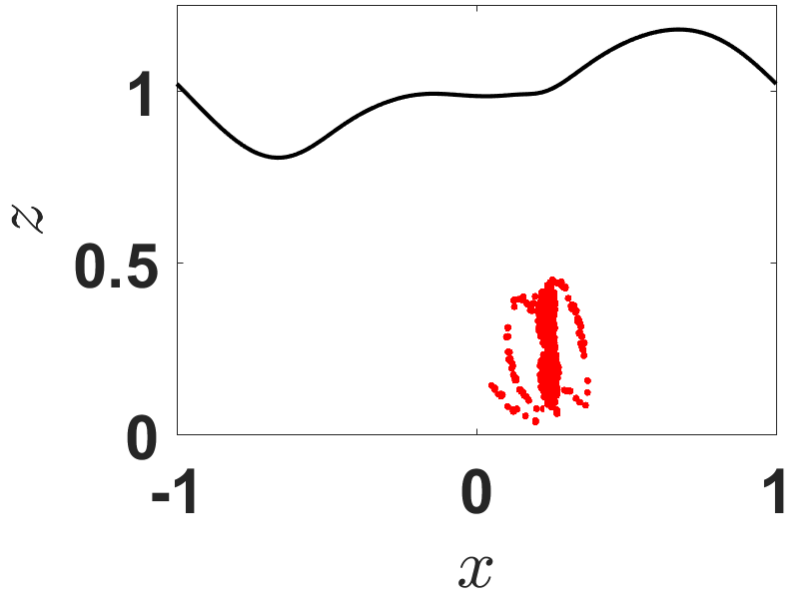


Figure 8: At frame 30 of 55, the vortex patch has effectively rolled up into itself, exhibiting the mixing property of the Kelvin-Helmholtz Instability [13].

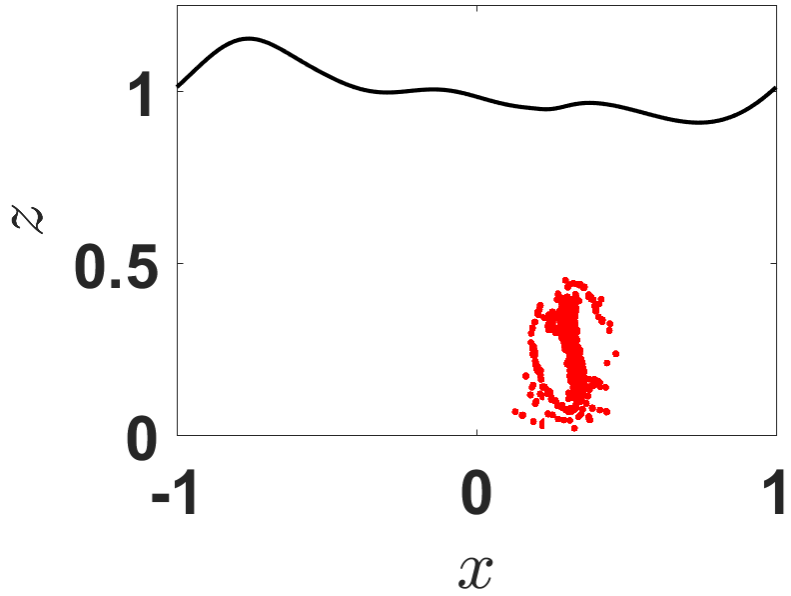


Figure 9: The vortex sheet continues to deform.

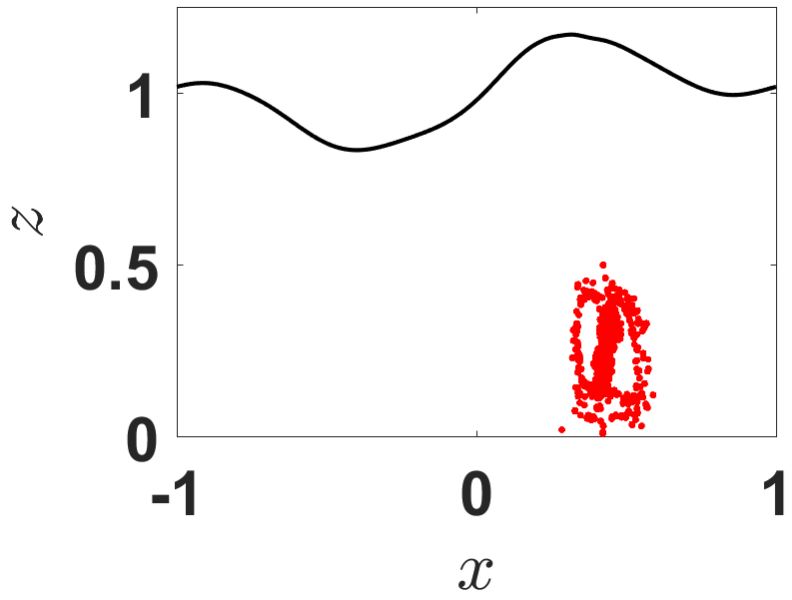


Figure 10: In the final frame of this animation, we have the vortex sheet evidently deforming into an elliptical pattern.

4.3 Experiments in Modeling a Stable Structure

We now search for a vortex patch which retains its shape over a period of time. We test shapes such as square grids, equilateral triangles, and circles, but notice that, as in the case with the vortex sheet, these have the tendency to deform into an ellipse, and we subsequently find that ellipses maintain a stable form over a period of time. We first examine these structures without mollification, and in section 4.5, we study the effects mollification has on our simulations.

The formation of our ellipses include changing several parameters in the `Matlab` code. In particular, we use polar coordinates, and we let `p-shape = 'circle'`, where for an array of x positions, we have `xpos(ij) = xpos1+(apos1+(ii-1)*da)*costh`, with `ij` denoting the `i`th row and `j`th column position of the vortices in the x direction. Similarly, we let `zpos(ij) = zpos1+(bpos1+(ii-1)*db)*sinth`. In the preceding equations, `apos1` represents the length of the semimajor axis of the ellipse in the x direction, while `bpos1` represents the length of the semiminor axis of the ellipse, which, for our purposes, is in the z direction. We define `apos1` to equal `da`, a grid spacing equal to the vertical grid spacing `dz/2` and we define `bpos1` to equal `db`, a grid spacing equal to `dz/4`.

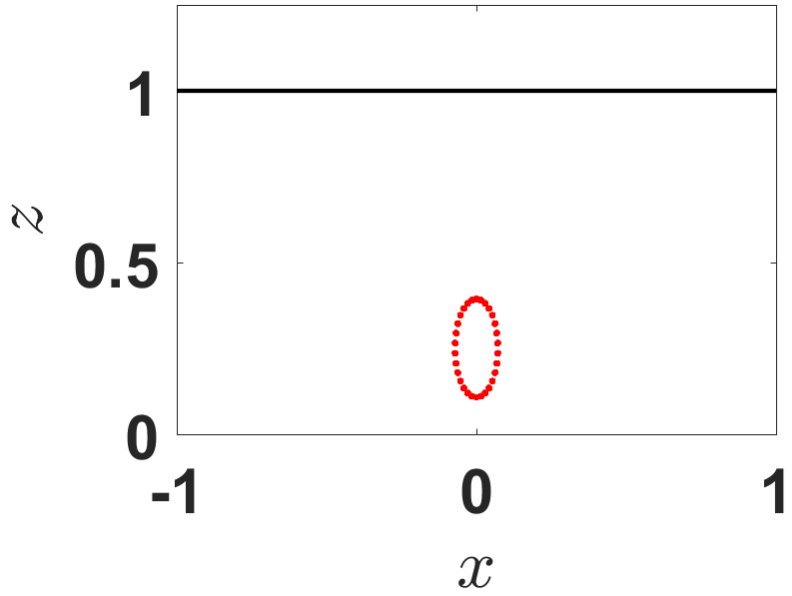


Figure 11: For this simulation, we have rows = 1, cols = 30, $K = 128$, $\mu = .08$, $\gamma = \sqrt{.2}$, $F = .2$, and $t_f = 30$. Our stability parameter is .2987, and runtime was 289.900563 seconds.

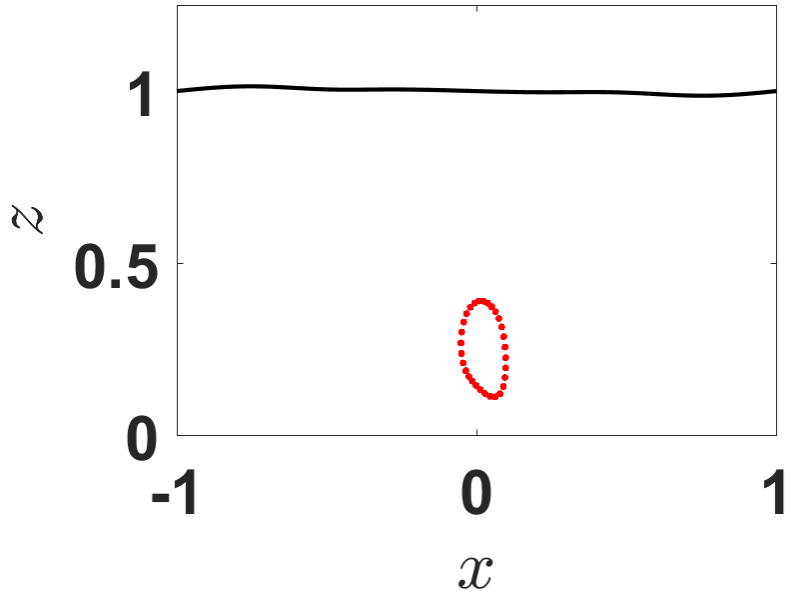


Figure 12: At frame 25 of 301, the ellipse still shows a maintenance of its shape, but we see that the lower half of the ellipse shows some caving in. In section 4.5, we will examine the effects of mollification on this ellipse.

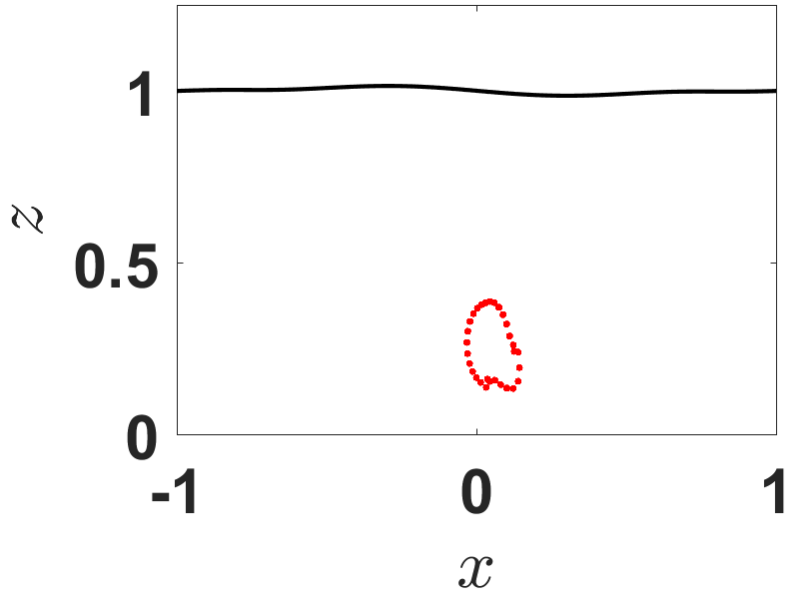


Figure 13: At frame 50 of 301, we notice that the ellipse has begin a deformation away from its original shape.

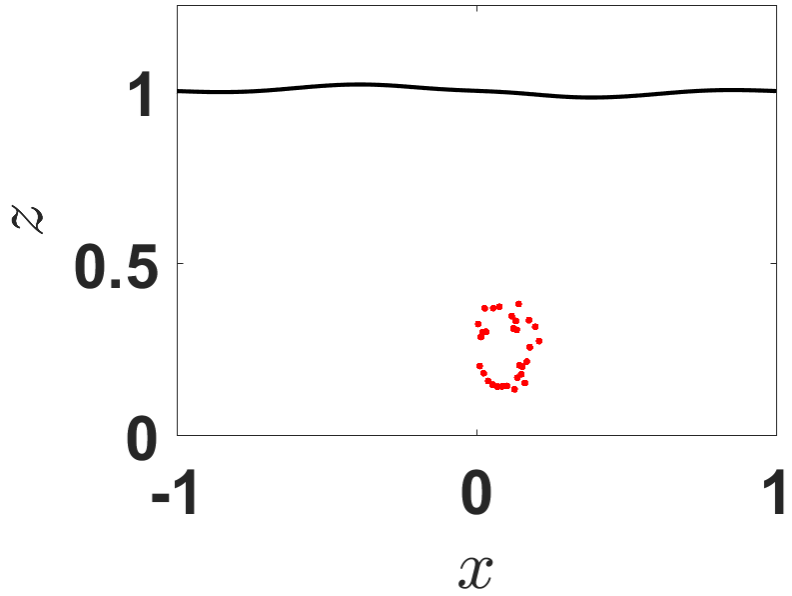


Figure 14: By frame 100, we notice that the ellipse has deformed from its original shape.

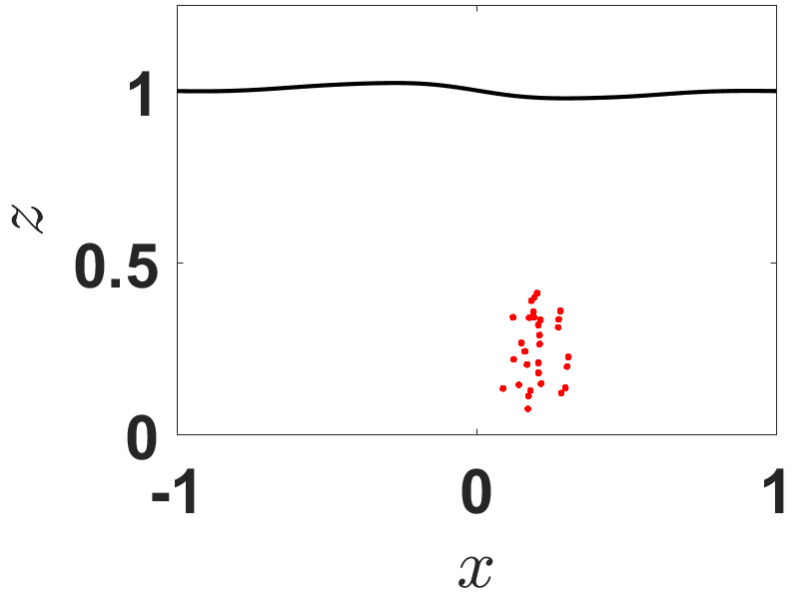


Figure 15: The ellipse begins clustering.

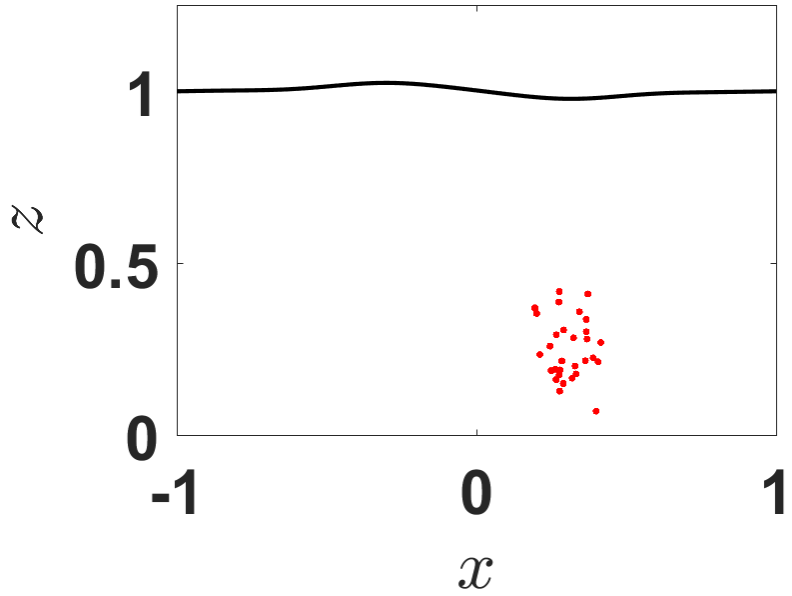


Figure 16: By the final frame, we see that the ellipse has completely deformed.

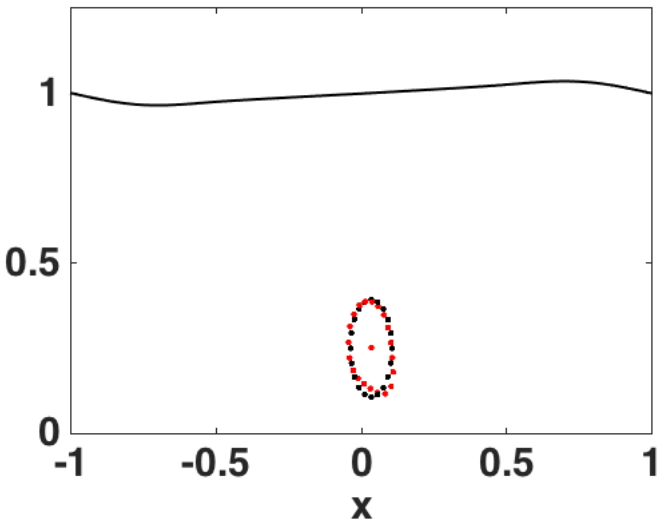
Since the ellipse deforms over time, an interesting question is how we can model said deformation in the Matlab code or through equations. Because of how the ellipse deforms,

forming an equation for how the ellipse changes over time might be difficult or nearly impossible. Thus, our focus is on modifying the Matlab code to create a a metric for determining the amount of deformation the elliptical patch exhibits over time.

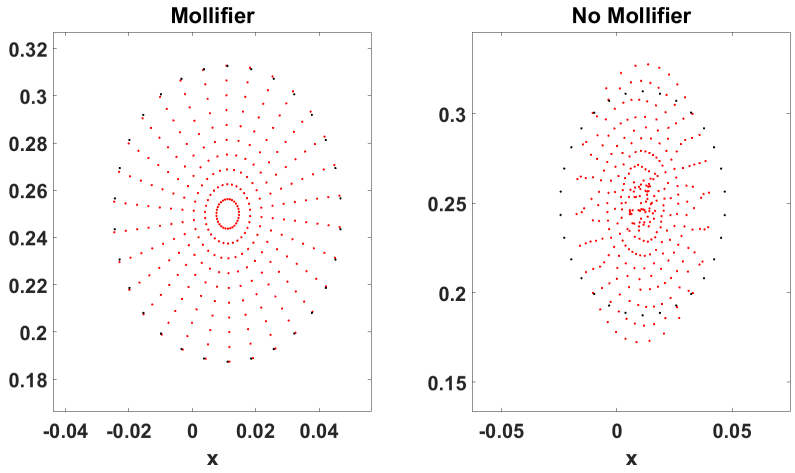
4.4 Deformation Metric

We studied the evolution of vortex patch shapes over time. One shape of interest became the ellipse, as many times patches that were not originally elliptic in shape evolved to eventually become elliptic. As a starting case, we chose to track the shape of an ellipse of vortex points, both as a hollow ring and a filled in patch.

The first direction in tracking the evolution of the shape of an elliptic vortex ring/patch was to simply visualize the change in shape of a vortex patch. In order to so, we first took the boundary of the patch when the program started and then plotted the points as points with nonzero vorticity. This frame would not move on the vertical axis, but rather the horizontal axis at a rate equal to the average change in position of the vortex points between time steps. The following figure gives a visual of what the frame is supposed to look like after a few time steps.



This method was also used to compare the shape between the system with and without mollification. Using the frame showed that the point vortices from mollified code followed closer to the reference frame while the points from the unmollified code did not adhere to the shape. This behavior is shown in the following figure, where the reader can see that the vortex points from the code using mollification stayed in an elliptic shape while the code without mollification ended up in a chaotic pattern.



With the reference frame sufficiently complete, the second path we chose to quantitatively track the change in shape of an elliptic hollow ring. This method includes plugging in vortex point positions into into the ellipse equation

$$\frac{(x-x_0)^2}{a^2}+\frac{(y-y_0)^2}{b^2}=1.$$

The center point (x_0, z_0) was taken from the center of the elliptic frame, as this could be easily taken from the Matlab program that updates the reference frame positions. This path posed the goal: to find how close could we get points to equal 1 when plugged into the ellipse equation. Furthermore, the program would count the number of points that when plugged into the ellipse equation would give a value within ϵ of 1. We found that the code for this metric confirmed the less quantitative results above from the reference frame. Using a ring with twenty points, we plot together how many points are within ϵ of 1. Several figures as well as their respective ϵ values are given below, note that the plots to the right comes from the mollified system, while the plots to the left uses the unmollified system.

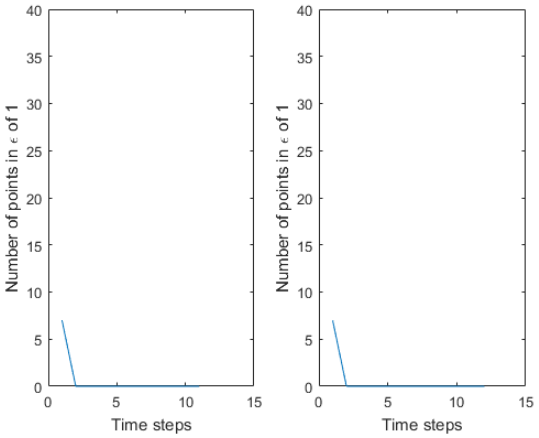


Figure 17: $\epsilon = 0$

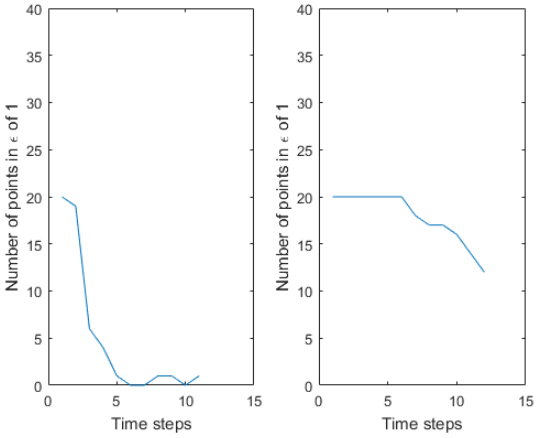


Figure 18: $\epsilon = 0.01$

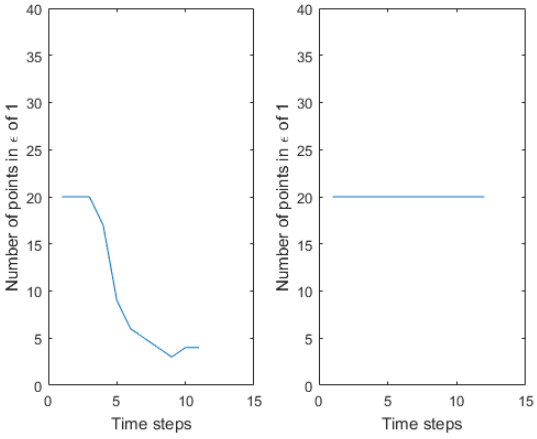


Figure 19: $\epsilon = 0.05$

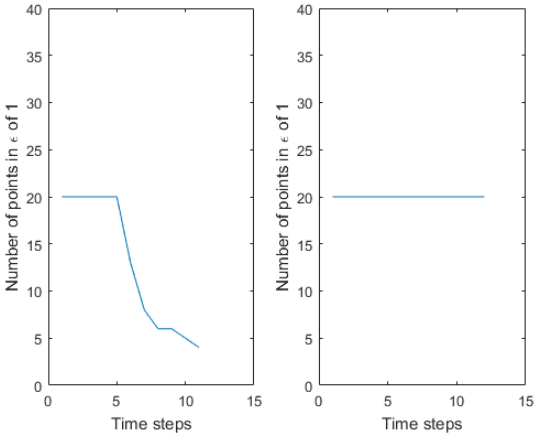


Figure 20: $\epsilon = 0.1$

From the figures above that when $\epsilon \approx .01$, we have that the mollified system stops satisfying the ellipse equation. Furthermore, we see that even for $\epsilon = .1$, the unmollified system doesn't satisfy the ellipse equation for long.

Although the metric we just described works well, one notices that it does not account for vortex point clustering. This means that we could hypothetically have a few vortices clustered around only a few points on the reference frame, thus tricking the program into thinking that we have an ellipse.

In studying point clustering, we decided to place circular bubbles, or rather δ-balls at reference points that would move at the same rate as the reference frame. These bubbles would count the number of points that they contain at each point in time. The following figure serves as a visual for how the δ-balls would look.

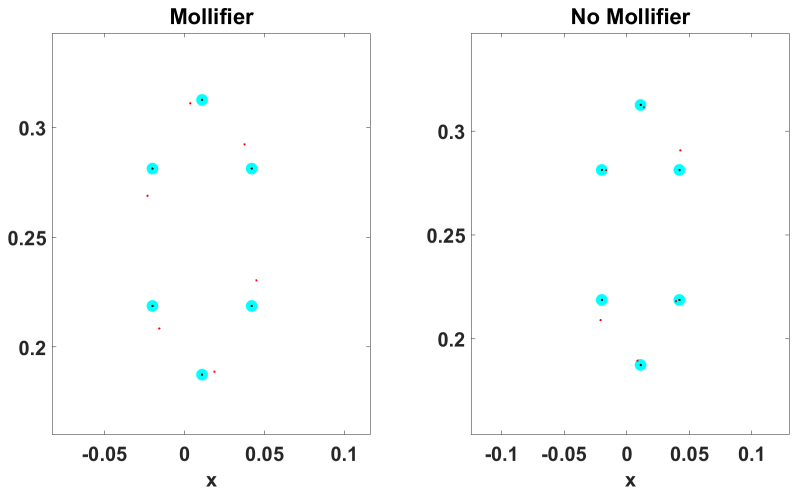


Figure 21: The δ-balls are colored cyan.

We furthermore give plots for a configuration of four δ-balls in B. One can note that

when the radius of the balls is larger that each ball does not have its vortex number change. We also note that as the radius decreases, the number of points in each of the balls fluctuates, but in a decreasing fashion.

4.5 Comparison between Code with and without Implemented Mollifiers

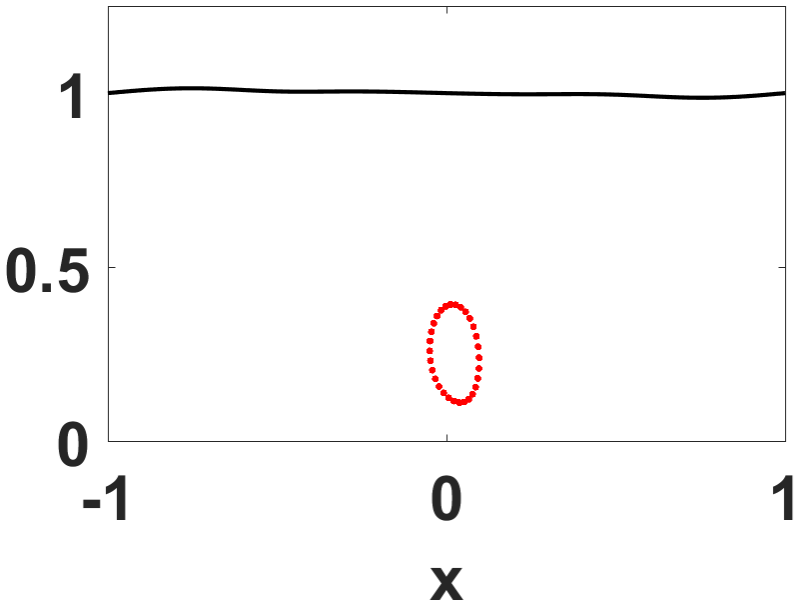


Figure 22: Mollified ellipse at frame 25. Compare to 12, and notice that the deformation is less noticeable with added mollification.

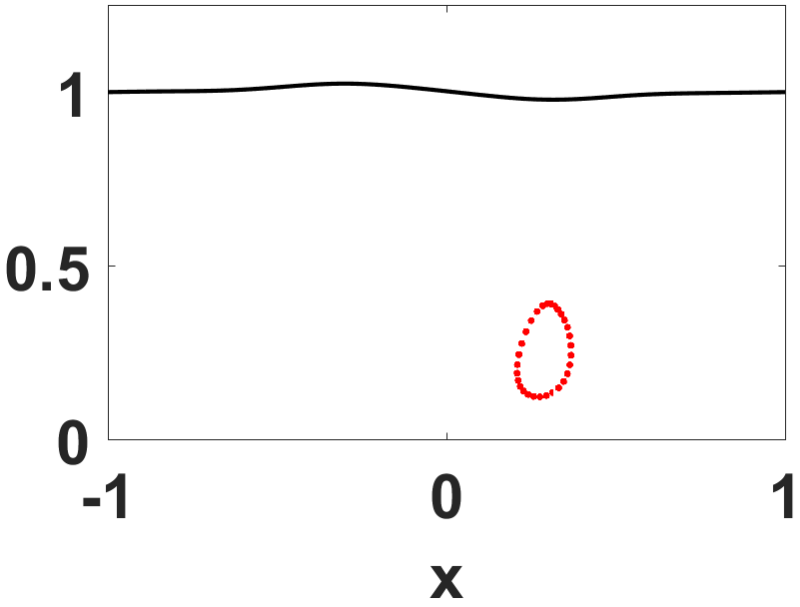


Figure 23: At the final frame, while 16 showcases clustering, the mollified ellipse has still held together remarkably well.

In both simulations, we have 1 row, 30 columns, 128 modes, $\mu = .08$, $\gamma = \sqrt{.2}$, `zpos1` $= .25$, $\Gamma = 1$, $F = .2$, and a time-final value of 30. The simulations are set to be ellipses with the grid spacing $da = \frac{dz}{2}$ and $db = 2da$. The stability parameter for both simulations was .2987, but the elapsed runtime for the mollified simulation was considerably longer than the runtime for the unmollified simulation. (The elapsed runtime for the mollified code was 758.437221 seconds, and the elapsed runtime for the unmollified code was 289.900563 seconds.) There is certainly much more deformity in the ellipse without mollification. The mollified ellipse also seems to rotate on an axis (albeit slowly) while mostly maintaining its shape, while the unmollified ellipse remains upright throughout the simulation and deforms very quickly.

5 Conclusions and Future Research

Future research could take several directions. The specific parameters of the ellipses could be varied and examined further, and refining the deformation metric for the ellipse to become more efficient would be a priority. Additionally, we will continue to fine-tune the mollification by choosing and testing different, more accurate cutoff functions.

A Evaluation of Integrals

Proposition 1. For real coefficients a, b, c with $a < 0$, we have

$$\int_{-\infty}^{\infty} e^{ax^2+bx+c}\,dx = \sqrt{-\frac{\pi}{a}}e^{c-b^2/4a}. \tag{21}$$

Proof. Note first that

$$\begin{aligned} ax^2+bx+c &= c + a\left(x^2+\frac{b}{a}x\right) \\ &= c - \frac{b^2}{4a} + a\left(x^2+\frac{b}{a}x+\frac{b^2}{4a^2}\right) \\ &= c - \frac{b^2}{4a} + a\left(x+\frac{b}{2a}\right)^2; \end{aligned}$$

then

$$\begin{aligned} \int_{-\infty}^{\infty} e^{ax^2+bx+c}\,dx &= e^{c-b^2/4a}\int_{-\infty}^{\infty} e^{a(x+b/2a)^2}\,dx \\ &= e^{c-b^2/4a}\int_{-\infty}^{h} \infty e^{ax^2}\,dx \\ &= e^{c-b^2/4a}\frac{1}{\sqrt{-a}}\int_{-\infty}^{\infty} e^{-u^2}\,du \\ &= \sqrt{-\frac{\pi}{a}}e^{c-b^2/4a}. \end{aligned}$$

In the third line we use the change of variables $u = \sqrt{-a}x$, and in the final line we use the fact that

$$\int_{-\infty}^{\infty} e^{-u^2}\,du = \sqrt{\pi}.$$

□

Proposition 2. For real coefficients a, b, c, d with $a < 0$, we have

$$\int_{-\infty}^{\infty} e^{ax^2+bx+c+d|x|}\,dx = \sqrt{-\frac{\pi}{a}}e^{c}[E_1+E_2], \tag{22}$$

where

$$E_1 = \begin{cases} \exp\left[-\frac{(b-d)^2}{4a}\right]\left(\frac{1}{2} - \frac{\operatorname{erf}|(d-b)/2\sqrt{-a}|}{2}\right) & \text{if } \frac{d-b}{2\sqrt{-a}} < 0 \\ \exp\left[-\frac{(b-d)^2}{4a}\right]\left(\frac{1}{2} + \frac{\operatorname{erf}|(d-b)/2\sqrt{-a}|}{2}\right) & \text{if } \frac{d-b}{2\sqrt{-a}} > 0, \end{cases}$$

$$E_2 = \begin{cases} \exp\left[-\frac{(b+d)^2}{4a}\right]\left(\frac{1}{2} + \frac{\operatorname{erf}|(-d-b)/2\sqrt{-a}|}{2}\right) & \text{if } \frac{-d-b}{2\sqrt{-a}} < 0 \\ \exp\left[-\frac{(b+d)^2}{4a}\right]\left(\frac{1}{2} - \frac{\operatorname{erf}|(-d-b)/2\sqrt{-a}|}{2}\right) & \text{if } \frac{-d-b}{2\sqrt{-a}} > 0, \end{cases}$$

and where the error function "erf" is defined by

$$\operatorname{erf}(x) = \frac{1}{\sqrt{\pi}} \int_{-x}^{x} e^{-u^2}\, du$$

for $x \geq 0$.

Proof. First, write

$$\int_{-\infty}^{\infty} e^{ax^2+bx+c+d|x|}\, dx = \int_{-\infty}^{0} e^{ax^2+(b-d)x+c}\, dx + \int_{0}^{\infty} e^{ax^2+(b+d)x+c}\, dx.$$

For the first integral on the right-hand side, we complete the square in the exponent to obtain

$$\begin{aligned}
\int_{-\infty}^{0} e^{ax^2+(b-d)x+c}\, dx &= \exp\left[c - \frac{(b-d)^2}{4a}\right] \int_{-\infty}^{0} \exp\left[a\left(x + \frac{b-d}{2a}\right)^2\right] dx \\
&= e^c \exp\left[-\frac{(b-d)^2}{4a}\right] \int_{-\infty}^{(b-d)/2a} e^{ax^2}\, dx \\
&= e^c \exp\left[-\frac{(b-d)^2}{4a}\right] \frac{1}{\sqrt{-a}} \int_{-\infty}^{(d-b)/2\sqrt{-a}} e^{-u^2}\, du \\
&= \sqrt{-\frac{\pi}{a}} e^c \exp\left[-\frac{(b-d)^2}{4a}\right] \begin{cases} \frac{1}{2} - \frac{\operatorname{erf}|(d-b)/2\sqrt{-a}|}{2}, & \text{if } \frac{d-b}{2\sqrt{-a}} < 0, \\ \frac{1}{2} + \frac{\operatorname{erf}|(d-b)/2\sqrt{-a}|}{2}, & \text{if } \frac{d-b}{2\sqrt{-a}} > 0, \end{cases}
\end{aligned}$$

where in the third line we use the change of variables $u = \sqrt{-a}x$ and in the final line we use the identity

$$\frac{1}{\sqrt{\pi}} \int_{-\infty}^{L} e^{-u^2}\, du = \begin{cases} \frac{1}{2} - \frac{\operatorname{erf}|L|}{2} & \text{if } L < 0 \\ \frac{1}{2} + \frac{\operatorname{erf}|L|}{2} & \text{if } L > 0. \end{cases}$$

A similar computation for the second integral on the right-hand side then yields the desired formula. □

B Clustering Graphs

We give plots for a configuration of four δ-balls, such that the first delta ball exists at the very topmost point of the elliptic frame (here we start with 20 vortex points), the second at the very bottom, the third at the very left point, and fourth at the very right most point.

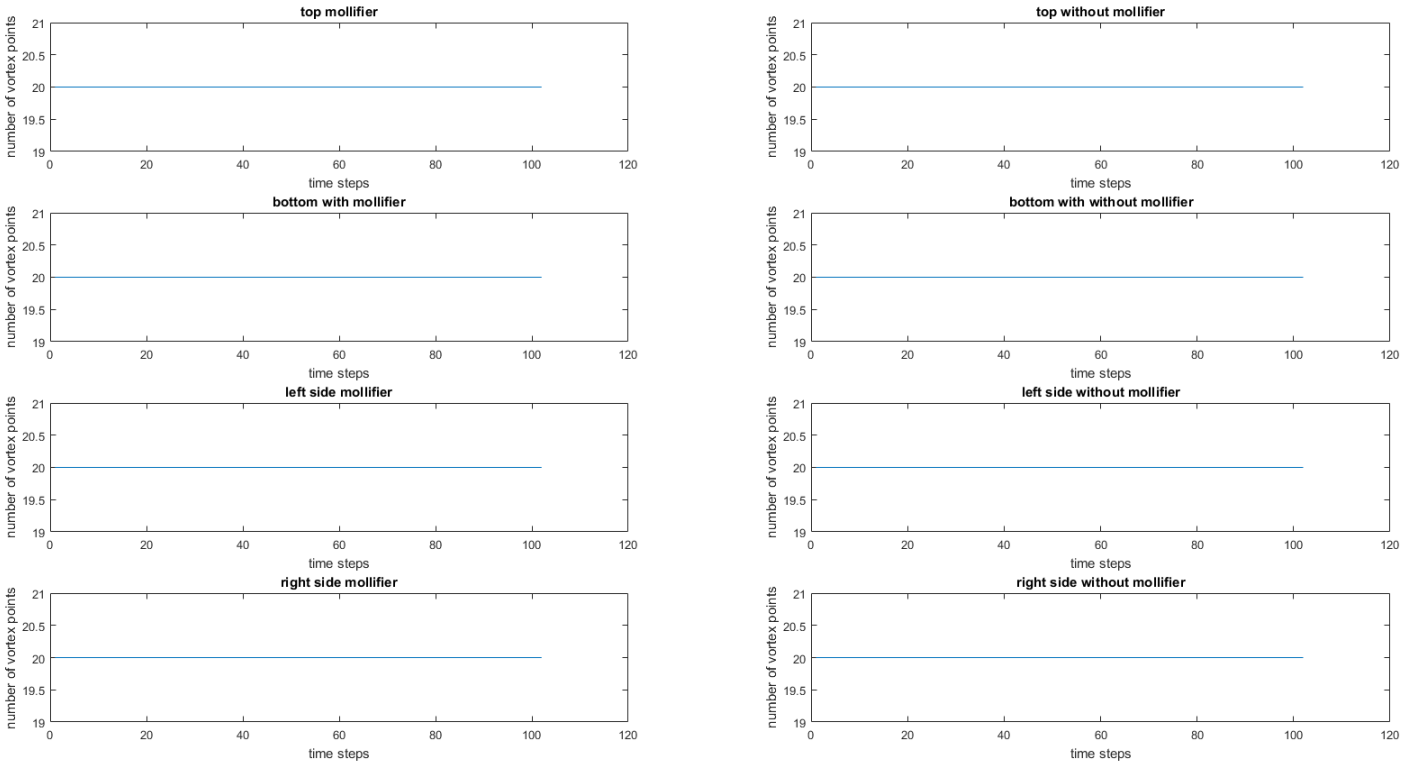


Figure 24: $\delta = 0.5$

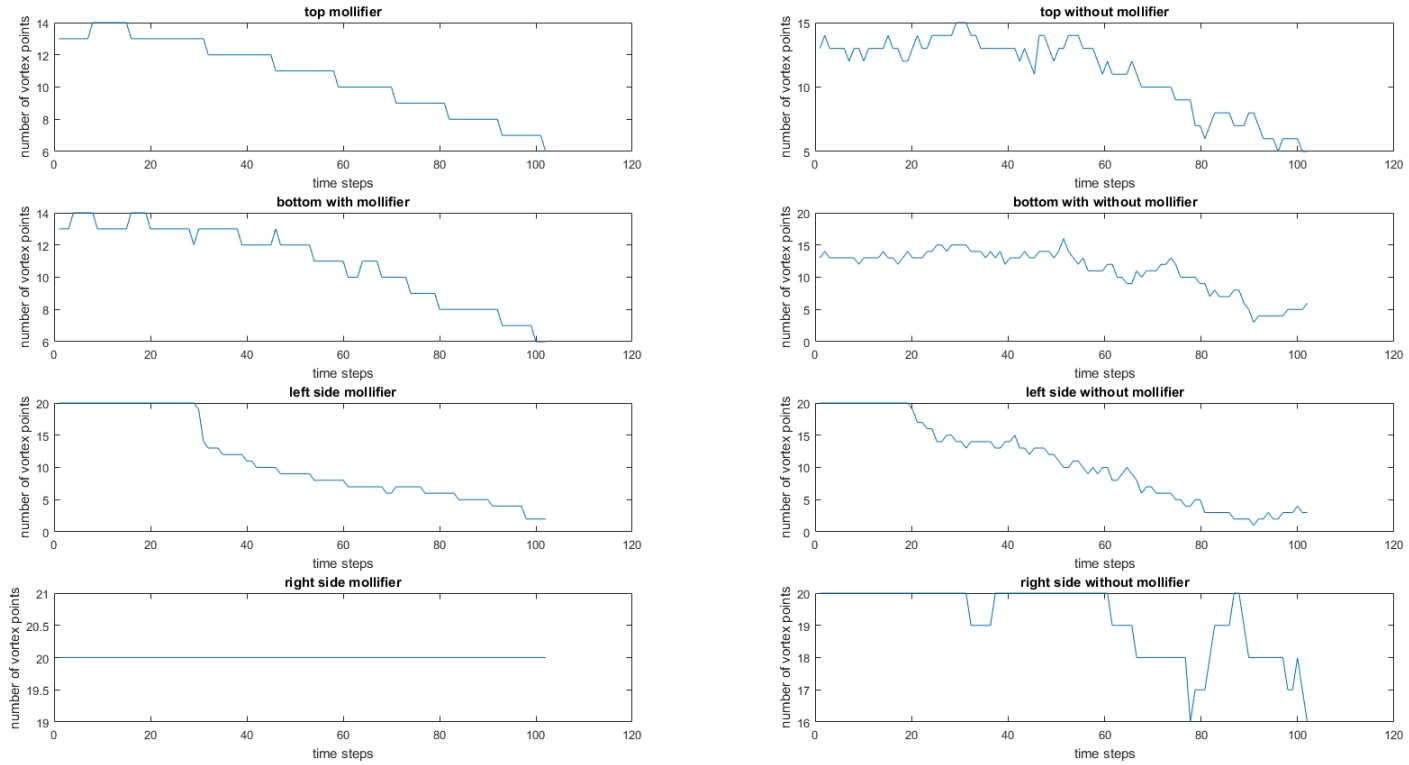


Figure 25: $\delta = 0.1$

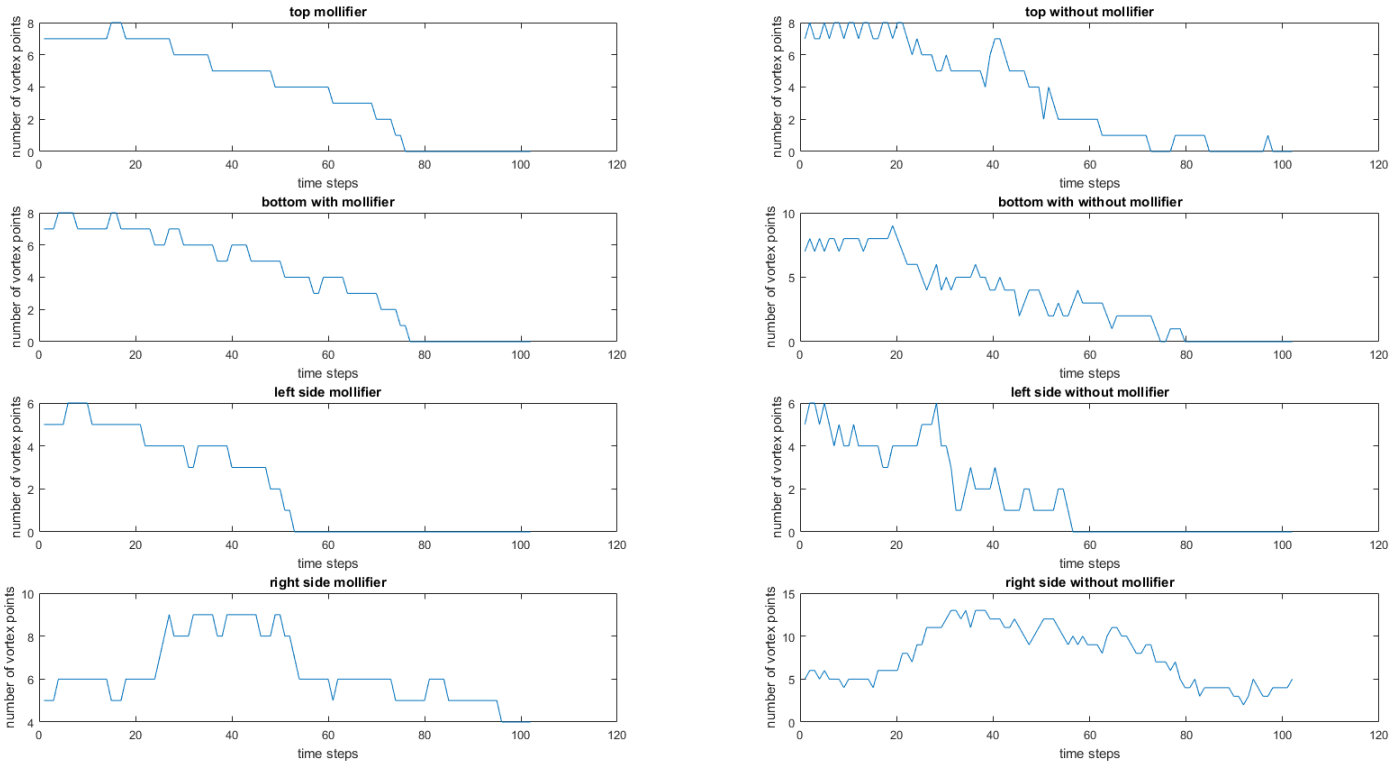


Figure 26: $\delta = 0.05$

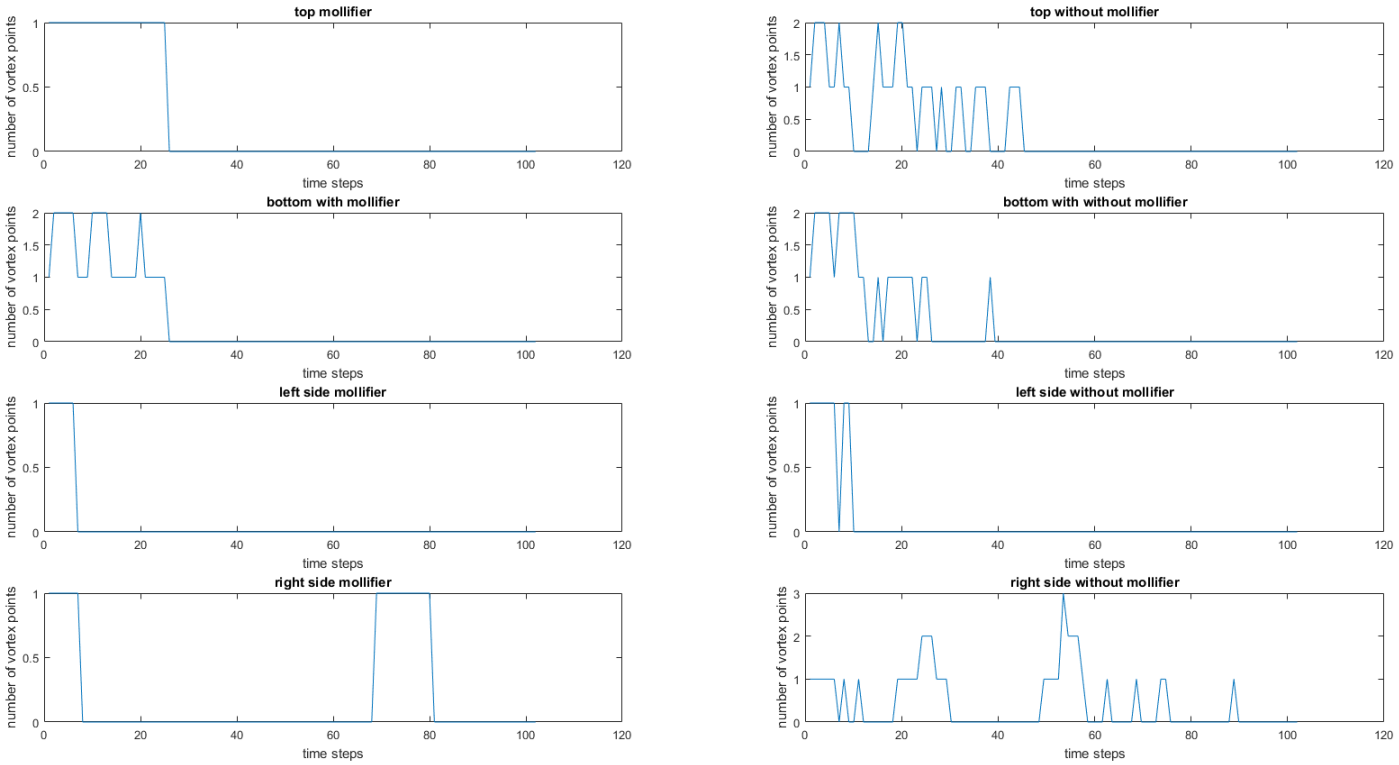


Figure 27: $\delta = 0.01$

C Acknowledgments

We would like to thank Professor Curtis for his direction and contributions to our research this summer. We would also like to acknowledge Eunji Yoo and Robert Insley for their support and assistance throughout our research project.

References

[1] P.A. Tyvand. On the interaction between a strong vortex pair and a free surface. *Phys. Fluids A*, 2:1624–1634, 1990.

[2] P.A. Tyvand. Motion of a vortex near a free surface. *J. Fluid Mech.*, 225:673–686, 1991.

[3] S. Fish. Vortex dynamics in the presence of free surface waves. *Phys. Fluids A*, 3:504–506, 1991.

[4] D.L. Marcus and S.A. Berger. The interaction between a counter-rotating vortex pair in vertical ascent and a free surface. *Phys. Fluids A*, 1:1988–2000, 1989.

[5] J. G. Telste. Potential flow about two counter-rotating vortices approaching a free surface. *J. Fluid Mech.*, 201:259–278, 1989.

[6] W.W. Willmarth, G. Tryggvason, A. Hirsa, and D. Yu. Vortex pair generation and interaction with a free surface. *Phys. Fluids A*, 1:170–172, 1989.

[7] E.A. Kuznetsov and V.P. Ruban. Cherenkov interaction of vortices with a free surface. *JETP*, 88:492–505, 1999.

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

[9] M.J. Ablowitz, A.S. Fokas, and Z.H. Musslimani. On a new non-local formulation of water waves. *J. Fluid Mech.*, 562:313–343, 2006.

[10] C.W. Curtis and H. Kalisch. Vortex dynamics in nonlinear free surface flows. *Phys. Fluids*, 29:032101, 2017.

[11] P.G. Saffman. *Vortex Dynamics*. Cambridge University Press, Cambridge, 1992.

[12] H. Lamb. *Hydrodynamics*. Dover, New York, N.Y., 1945.

[13] Pijush K. Kundu, Ira M. Cohen, and David R. Rowling. *Fluid Mechanics*. Academic Press, Waltham, MA, 2012.