UCLA
COMPUTATIONAL AND APPLIED MATHEMATICS

Fast Numerical Computation of 2-D Free Surface Jet Flow with Surface Tension

N. Anders Petersson

November 1994

CAM Report 94-32

Department of Mathematics
University of California, Los Angeles
Los Angeles, CA. 90024-1555

Fast Numerical Computation of 2–D Free Surface Jet Flow with Surface Tension

N. Anders Petersson [1]

October 27, 1993
Revised November 3, 1994

[1]Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, New Mexico, USA. Present address: Dept. of Mathematics, UCLA, Los Angeles, California, USA.

Abstract

A system of partial differential equations that approximate the governing equations for inviscid free surface flow subject to surface tension is presented. The approximation is based on linearizing the velocity together with a small scale approximation of the perturbation of the velocity. Two Dirichlet problems must be solved to form the approximate system, after which it can be evolved without solving Dirichlet problems. The accuracy of the solution is determined by how often the velocity term is linearized. This time-interval is called ΔT. We show that the error in the solution of the approximate system at a fixed time T is of the order $\mathcal{O}(\Delta T^2)$. We demonstrate numerically that the error is closely correlated to the size of the normal velocity, and that there is a stability limit of the form $\Delta T \leq C/\left(|\bar{u}\cdot\bar{N}|_\infty\right)^\gamma$, where $\bar{u}\cdot\bar{N}$ denotes the normal velocity and $\gamma \approx 2.6$. Importantly, C is independent of the resolution, so the time-step ΔT can be chosen independently of the number of grid points, N. This is in contrast to the time-step when the original system is integrated, where the stability limit is $\Delta t \leq \mathcal{O}(N^{-3/2})$ and four Dirichlet problems have to be solved per Δt (for the four stage Runge–Kutta scheme). By numerical experiments, we demonstrate that the approximate system solves the problem very accurately and requires less than 10% of the CPU-time used by the original system.

Keywords: surface tension, free surface, jet, potential flow, Dirichlet problem, boundary integral, pseudo–spectral method.

AMS classifications: 76B10 (primary), 65M70

1 Introduction

When surface tension is included in the potential flow model of free surface dynamics, the corresponding linearized problem gets eigenvalues with imaginary part proportional to $\omega^{3/2}$, where ω is the spatial frequency. If an explicit time-integrator is used to integrate the solution in time, the time step must for stability reasons be very small. A Dirichlet problem must be solved for every evaluation of the time-derivatives, and this is found to constitute the bulk of the computation. Despite the stiffness problem, these type of equations have been simulated numerically by many investigators, see for example [1, 2]. To speed up the computation, we derive an approximate system in which the Dirichlet problem can be solved less frequently. The idea behind the approximate system can be applied to inviscid free surface problems in general, but in this paper we restrict the presentation to the two-dimensional approximation of a slender non-axisymmetric three-dimensional jet subject to surface tension, where the evolution of the flow in the cross-section of the jet is governed by the two-dimensional Euler equations inside of the free surface, cf. [3].

We assume the liquid to be inviscid and incompressible, and the velocity field to be irrotational, so there exists a velocity potential. The velocity potential is governed by Laplace's equation in the interior of the jet subject to Bernoulli's equation and the kinematic condition on the free surface. The physical problem is scaled by the length scale L such that the area of the scaled initial domain equals π, and by a time-scale $T = \sqrt{\rho L^3/\tau}$. Here, ρ is the density of the liquid and τ is the surface tension. This choice of time-scale makes the scaled surface tension equal to 1, i.e. the Weber number is set to one. Henceforth, only the scaled problem will be considered.

Let x and y be the Cartesian coordinates in the cross-sectional plane of the jet, and let t be time. We assume that the liquid occupies the simply connected time-dependent domain $\Omega(t)$ with a smooth boundary $\Gamma(t)$. It is convenient to describe the motion in the positively oriented Lagrangian coordinate $0 \leq \alpha \leq 2\pi$, such that the boundary at time $t \geq 0$ is given by $x = X(\alpha, t)$, $y = Y(\alpha, t)$ and the velocity potential on the boundary is $\phi = \phi(\alpha, t)$. The governing equations for ϕ, X, Y on $\Gamma(t)$ are

$$\phi_t = \frac{1}{2}(u^2 + v^2) - \kappa, \tag{1}$$

$$X_t = u, \tag{2}$$

$$Y_t = v, \tag{3}$$

for $t \geq 0$ subject to the 2π-periodic initial conditions $\phi(\alpha, 0) = \phi_0(\alpha)$, $X(\alpha, 0) = X_0(\alpha)$ and $Y(\alpha, 0) = Y_0(\alpha)$. The curvature of the boundary is

$$\kappa = \sigma^3(X_\alpha Y_{\alpha\alpha} - Y_\alpha X_{\alpha\alpha}), \quad \sigma = 1/\sqrt{X_\alpha^2 + Y_\alpha^2}, \tag{4}$$

and the velocity components satisfy

$$u(\alpha, t) = \psi_x(X(\alpha, t), Y(\alpha, t)), \tag{5}$$

$$v(\alpha, t) = \psi_y(X(\alpha, t), Y(\alpha, t)), \tag{6}$$

where $\psi = \psi(x, y)$ is the solution of

$$\Delta\psi = 0, \qquad \text{in } \Omega(t), \tag{7}$$

$$\psi(X(\alpha,t), Y(\alpha,t)) = \phi(\alpha,t), \quad 0 \le \alpha \le 2\pi. \tag{8}$$

The linearized motion of the related problem governing inviscid water waves subject to surface tension was analyzed by [4]. By projecting the surface coordinates into the local normal and tangential components, it was found that the linearized equations have a tractable form which enabled the authors to prove well-posedness of the solution in a Sobolev space of finite order. This work was extended by [5] to the discrete case, were stability was proven also for the non-linear problem. It was found that the particular type of discretization is very important in order for certain singular operators to cancel to highest order. A similar result was found by [6] who considered the linearized stability properties of different numerical schemes for a vortex sheet subject to surface tension close to equilibrium.

The focus of the present work is to reduce the great computational cost of integrating the governing equations numerically, which is caused by the stiffness of the discrete system due to surface tension. In section 2, we will carry out a simplified version of the general linearization performed in [4] and [7] to show that because of surface tension, the linearized equations have eigenvalues with imaginary part of the order $\mathcal{O}\left((\max_\alpha \sigma(\alpha,t)N)^{3/2}\right)$, where N is the number of grid points in the spatial discretization of the free surface. The time step is therefore restricted by the stability constraint $\Delta t \le \mathcal{O}\left((\max_\alpha \sigma(\alpha,t)N)^{-3/2}\right)$. Furthermore, the stability limit of the time-step can become increasingly restrictive with time because σ depends on time, i.e. the Lagrangian grid points can cluster. The time-step restriction together with the clustering of grid points make the time-integration expensive, because the Dirichlet problem (7), (8) must be solved every time the right hand side of (1)–(3) is evaluated. If a boundary integral formulation is applied, the cost of solving one Dirichlet problem is of the order $\mathcal{O}(N^2)$ operations if direct summation is used or $\mathcal{O}(CN)$ operations, where C is a large constant, if the fast multipole method [8] is used.

In the recent paper [9], a method was proposed to alleviate the stiffness of the more general equations governing the motion of two immiscible ideal fluids subject to surface tension. The governing equations were reformulated using the arclength and tangent-angle variables instead of the Cartesian coordinates X and Y to describe the position of the interface, and the vortex sheet strength was used instead of the velocity potential to describe the velocity field. It is shown that the curvature-term becomes linear in terms of these variables. This property together with a small-scale approximation of the velocity enabled the authors to device a fast implicit time-integration method to solve the reformulated equations in the case of zero density stratification between the fluids. In the presence of a finite density stratification, an integral equation for the time derivative of the vortex sheet strength must be solved. This extension of the method was not treated, so the method can not be applied directly to solve the present problem, where the density outside of the jet is assumed to be zero, which makes the density stratification substantial.

As in noted in [9], the tangential velocity of the grid points is only dictated by the parameterization of the solution. It would therefore be possible to partially reduce the stiffness of the equations by using a non-Lagrangian parameterization of the solution and

prescribing a tangential velocity that for example distributes the grid points uniformly with respect to arclength. This possibility has not been exploited in the present work.

In the present paper, we propose to approximate the governing equations such that the resulting approximate system can be integrated much faster than the original system, while still retaining a very accurate solution. The underlying idea is to take advantage of the leading order structure of the linearized equations derived by [4] and [7]. The approximate system is derived in the following way. Let $\bar{T}$ and $\bar{N}$ denote the unit tangent and normal vectors of the boundary. The velocity can be decomposed as

$$\bar{u} \equiv \begin{pmatrix} u \\ v \end{pmatrix} = (\bar{u} \cdot \bar{T})\bar{T} + (\bar{u} \cdot \bar{N})\bar{N}.$$

The tangential component of the velocity equals the derivative of the velocity potential with respect to arclength, i.e. $\bar{u} \cdot \bar{T} = \sigma\phi_\alpha$. However, the computation of the normal component $\bar{u} \cdot \bar{N}$ requires the solution of a Dirichlet problem, which we want to avoid. The velocity $\bar{u}$ depends on ϕ, X and Y. By linearizing $\bar{u}$ around $\phi^{(0)}$, $X^{(0)}$ and $Y^{(0)}$ and letting $\phi = \phi^{(0)} + \epsilon\phi'$, $X = X^{(0)} + \epsilon X'$, $Y = Y^{(0)} + \epsilon Y'$, we get

$$\bar{u} = \bar{u}^{(0)} + \epsilon\bar{u}' + \mathcal{O}\left(\epsilon^2\right), \tag{9}$$

$$\bar{u}' = \frac{\partial\bar{u}}{\partial\phi}\phi' + \frac{\partial\bar{u}}{\partial X}X' + \frac{\partial\bar{u}}{\partial Y}Y'. \tag{10}$$

After some analysis, which we defer to section 3, we show that the tangential component of the perturbation of the velocity is

$$\bar{u}' \cdot \bar{T}^{(0)} = \sigma^{(0)}\left(\phi'_\alpha - u^{(0)}X'_\alpha - v^{(0)}Y'_\alpha\right). \tag{11}$$

We use the theory in [4] and [7] to motivate the approximation of the normal component of the perturbation of the velocity:

$$\bar{u}' \cdot \bar{N}^{(0)} \approx H(\bar{u}' \cdot \bar{T}^{(0)}), \tag{12}$$

where the operator H has Fourier symbol $i\,\mathrm{sgn}(\omega)$. In those papers, it is shown that the error $\bar{u}' \cdot \bar{N}^{(0)} - H(\bar{u}' \cdot \bar{T}^{(0)})$ is a smoothing operator of the perturbations ϕ', X' and Y'. This means that the contribution to the error is dominated by the low frequency components of ϕ', X' and Y' and that the contribution from the high frequency components tend to zero as the frequency of the perturbation tends to infinity. The velocity is therefore approximately

$$\bar{u} \approx \bar{u}^{(0)} + \epsilon\bar{T}^{(0)}(\bar{u}' \cdot \bar{T}^{(0)}) + \epsilon\bar{N}^{(0)}(\bar{u}' \cdot \bar{N}^{(0)}).$$

Because the tangential component of the velocity is known exactly, we will only use the above expression for the normal component of the velocity. We have $\bar{T}^{(0)} \cdot \bar{N} = \mathcal{O}(\epsilon)$ and $\bar{N}^{(0)} \cdot \bar{N} = 1 + \mathcal{O}(\epsilon^2)$, so it is consistent with the linearization to take the approximate velocity $\tilde{\bar{u}} \approx \bar{u}$ to be

$$\tilde{\bar{u}} \equiv \begin{pmatrix} \tilde{u} \\ \tilde{v} \end{pmatrix} = \bar{T}(\sigma\phi_\alpha) + \bar{N}\left(\bar{u}^{(0)} \cdot \bar{N} + \epsilon H(\sigma^{(0)}(\phi'_\alpha - u^{(0)}X'_\alpha - v^{(0)}Y'_\alpha))\right). \tag{13}$$

In the approximate system we replace the exact velocity by the approximation (13). To compensate for the smooth error in (13), we also introduce a time-dependent forcing which will be derived in section 3. We arrive at

$$\tilde{\phi}_t = \frac{1}{2}(\tilde{u}^2 + \tilde{v}^2) - \kappa + (t - t_0)G_1, \tag{14}$$

$$\tilde{X}_t = \tilde{u} + (t - t_0)G_2, \tag{15}$$

$$\tilde{Y}_t = \tilde{v} + (t - t_0)G_3, \tag{16}$$

for $t \geq t_0$ subject to the initial conditions $\tilde{\phi}(\alpha, t_0) = \phi^{(0)}(\alpha)$, $\tilde{X}(\alpha, t_0) = X^{(0)}(\alpha)$ and $\tilde{Y}(\alpha, t_0) = Y^{(0)}(\alpha)$. We remark that (13) is evaluated by taking $\epsilon\phi' = \tilde{\phi} - \phi^{(0)}$, $\epsilon X' = \tilde{X} - X^{(0)}$ and $\epsilon Y' = \tilde{Y} - Y^{(0)}$.

Computing the velocity $\bar{u}^{(0)}$ requires one Dirichlet problem to be solved and calculating the forcing terms G_i, $i = 1, 2, 3$, requires the solution of another Dirichlet problem. Hence, after two Dirichlet problems have been solved, the approximate system can be evolved without solving Dirichlet problems. It should be noted that the stability restriction on the approximative system is similar to the original system, so the explicit time-step limit for (14), (15) and (16) is also of the order $\Delta t \leq \mathcal{O}\left((\max_\alpha \sigma(\alpha, t)N)^{-3/2}\right)$. The significant advantage of the approximate system is that the evaluation of the time-derivatives only require the computation of spatial derivatives followed by the application of H. Both of these tasks can be done very quickly by the pseudo spectral method which require $\mathcal{O}(N \log(N))$ operations.

We start integrating the approximate system at $t_0 = 0$ where we take $\phi^{(0)} = \phi_0$, $X^{(0)} = X_0$, $Y^{(0)} = Y_0$. First we compute $\bar{u}^{(0)}$ and G_i, $i = 1, 2, 3$, to set up the approximate system. The system is then integrated in time until $t = t_0 + \Delta T$, after which the linearization of the velocity is redone around $\phi^{(0)}(\alpha) = \tilde{\phi}(\alpha, t_0 + \Delta T)$, $X^{(0)}(\alpha) = \tilde{X}(\alpha, t_0 + \Delta T)$, $Y^{(0)}(\alpha) = \tilde{Y}(\alpha, t_0 + \Delta T)$, and a new approximate system is formed. This procedure is repeated for as long as the equations need to be integrated. An error is committed by approximating the normal component of the velocity. We show in section 3 that this error is of the order $\mathcal{O}(\Delta T^3)$, which implies that the error at a fixed time T is of the order $\mathcal{O}(\Delta T^2)$. We remark that the time-step between linearizing the velocity, ΔT, is different from the explicit time-step Δt, which is used when the approximate system is integrated in time from t_0 to $t_0 + \Delta T$.

In section 4, we reformulate the Dirichlet problem into a Fredholm integral equation of the second kind for the dipole strength. Thereafter, only boundary values of the velocity potential are explicitly needed in the computation. Once the dipole strength is known, the velocity $\bar{u}$ is readily computed by evaluating an integral. We proceed in section 5 by discretizing both the original and the approximate system by the pseudo spectral method. We will use the "alternating-point-trapezoidal" method to evaluate the discrete velocities. This discretization has been shown by [5] and [6] to be essential for the stability of the discrete system. The resulting systems of ODE's are integrated in time by the fourth order accurate Runge–Kutta scheme. In section 6, we perform numerical experiments on both the original system and its approximate counterpart. We first investigate the role of filtering and how the resolution affects the solution. Thereafter, we study the properties of the approximate method. Some concluding remarks are made in section 7.

2 Analysis

The purpose of the analysis presented here is to estimate the eigenvalues of the linearized operator with frozen coefficients. This information is necessary for estimating how large the time step can be without violating the stability condition of the time integrator. For simplicity, we only consider a special case. We refer to [4] and [7] for a derivation of the linearized operator in the general case.

We begin by linearizing (1)–(3) around a solution $\phi^{(0)}(\alpha,t)$, $X^{(0)}(\alpha,t)$ and $Y^{(0)}(\alpha,t)$. Denote the velocity components by

$$u^{(0)} = u[\phi^{(0)}, X^{(0)}, Y^{(0)}], \tag{17}$$
$$v^{(0)} = v[\phi^{(0)}, X^{(0)}, Y^{(0)}]. \tag{18}$$

Let $\phi = \phi^{(0)} + \epsilon\phi'$, $X = X^{(0)} + \epsilon X'$ and $Y = Y^{(0)} + \epsilon Y'$, where $0 < \epsilon \ll 1$. We have

$$u[\phi, X, Y] = u^{(0)} + \epsilon u' + \mathcal{O}(\epsilon^2), \tag{19}$$
$$v[\phi, X, Y] = v^{(0)} + \epsilon v' + \mathcal{O}(\epsilon^2), \tag{20}$$

with

$$u' = \lim_{\epsilon\to 0} \frac{1}{\epsilon}\left(u[\phi^{(0)} + \epsilon\phi', X^{(0)} + \epsilon X', Y^{(0)} + \epsilon Y'] - u^{(0)}\right), \tag{21}$$
$$v' = \lim_{\epsilon\to 0} \frac{1}{\epsilon}\left(v[\phi^{(0)} + \epsilon\phi', X^{(0)} + \epsilon X', Y^{(0)} + \epsilon Y'] - v^{(0)}\right). \tag{22}$$

Neglecting the $\mathcal{O}(\epsilon^2)$ terms yields the following linear problem for the perturbations:

$$\phi'_t = u^{(0)}u' + v^{(0)}v' - \kappa', \tag{23}$$
$$X'_t = u', \tag{24}$$
$$Y'_t = v'. \tag{25}$$

The perturbation of the curvature is

$$\begin{aligned}\kappa' = {} & (\sigma^{(0)})^3\left(Y'_{\alpha\alpha}X^{(0)}_\alpha - X'_{\alpha\alpha}Y^{(0)}_\alpha + X'_\alpha Y^{(0)}_{\alpha\alpha} - Y'_\alpha X^{(0)}_{\alpha\alpha}\right) \\ & - 3\kappa^{(0)}(\sigma^{(0)})^2\left(X'_\alpha X^{(0)}_\alpha + Y'_\alpha Y^{(0)}_\alpha\right).\end{aligned} \tag{26}$$

In this expression, $\kappa^{(0)}$ is the curvature of the unperturbed boundary and the normalization factor is

$$\sigma^{(0)} = 1/\sqrt{\left(X^{(0)}_\alpha\right)^2 + \left(Y^{(0)}_\alpha\right)^2}. \tag{27}$$

Because u and v are linear in ϕ, it is clear that the perturbations of u and v can be split according to

$$\begin{aligned}u[\phi^{(0)} + \epsilon\phi', X^{(0)} + \epsilon X', Y^{(0)} + \epsilon Y'] = {} & \epsilon u[\phi', X^{(0)}, Y^{(0)}] \\ & + u[\phi^{(0)}, X^{(0)} + \epsilon X', Y^{(0)} + \epsilon Y'] + \mathcal{O}(\epsilon^2),\end{aligned} \tag{28}$$
$$\begin{aligned}v[\phi^{(0)} + \epsilon\phi', X^{(0)} + \epsilon X', Y^{(0)} + \epsilon Y'] = {} & \epsilon v[\phi', X^{(0)}, Y^{(0)}] \\ & + v[\phi^{(0)}, X^{(0)} + \epsilon X', Y^{(0)} + \epsilon Y'] + \mathcal{O}(\epsilon^2).\end{aligned} \tag{29}$$

The first term on the right hand sides of (28) and (29) corresponds to a perturbation of the boundary value only and the second term is the contribution from perturbing the shape of the domain only. For the first term, we have $u[\phi', X^{(0)}, Y^{(0)}] = \psi^I_x(X^{(0)}, Y^{(0)})$ and $v[\phi', X^{(0)}, Y^{(0)}] = \psi^I_y(X^{(0)}, Y^{(0)})$, where ψ^I is the solution of

$$\Delta\psi^I = 0, \qquad \text{in } \Omega^{(0)}, \tag{30}$$

$$\psi^I(X^{(0)}(\alpha,t), Y^{(0)}(\alpha,t)) = \phi'(\alpha,t), \quad 0 \le \alpha \le 2\pi, \tag{31}$$

Here, $\Omega^{(0)}$ corresponds to the domain interior to the boundary $x = X^{(0)}(\alpha,t)$, $y = Y^{(0)}(\alpha,t)$.

The second term of the right hand sides of (28) and (29) can be written as $u[\phi^{(0)}, X^{(0)} + \epsilon X', Y^{(0)} + \epsilon Y'] = \psi^{II}_x(X^{(0)} + \epsilon X', Y^{(0)} + \epsilon Y')$ and a corresponding expression for v, where ψ^{II} is the solution of

$$\Delta\psi^{II} = 0, \qquad \text{in } \Omega, \tag{32}$$

$$\psi^{II}(X(\alpha,t), Y(\alpha,t)) = \phi^{(0)}(\alpha,t), \quad 0 \le \alpha \le 2\pi. \tag{33}$$

Here Ω corresponds to the domain interior to the perturbed boundary $x = X(\alpha,t)$, $y = Y(\alpha,t)$.

In the general case, the term ψ^{II} is difficult to analyze, because it is not trivial to estimate how the normal derivative of ψ^{II} depends on the perturbation of the shape. To circumvent this problem, we will restrict the analysis to the special case when

$$\phi^{(0)}(\alpha,t) = C - \frac{t}{R}, \quad C = \text{const.}, \tag{34}$$

$$X^{(0)}(\alpha,t) = R\cos\alpha, \tag{35}$$

$$Y^{(0)}(\alpha,t) = R\sin\alpha, \tag{36}$$

which is a solution of (1)–(3). Now, $\nabla\psi^{II} = 0$ and it is sufficient to study the contribution from ψ^I.

Let (r,α) be polar coordinates, i.e. $x = r\cos\alpha$, $y = r\sin\alpha$. The Dirichlet problem on a circular domain with radius R,

$$\Delta\hat{\psi} = 0, \qquad 0 \le r \le R, \ 0 \le \alpha \le 2\pi, \tag{37}$$

$$\hat{\psi} = e^{i\omega\alpha}, \quad 0 \le r \le R, \ 0 \le \alpha \le 2\pi, \tag{38}$$

where $\omega = 0, \pm1, \pm2, \ldots$, is solved by

$$\hat{\psi}(r,\alpha) = \left(\frac{r}{R}\right)^{|\omega|} e^{i\omega\alpha}. \tag{39}$$

On the boundary $r = R$, we have $\hat{\psi}_r = |\omega| e^{i\omega\alpha}/R$ and $\hat{\psi}_\alpha = i\omega e^{i\omega\alpha}$. Therefore, the relation between the inward normal and tangential derivatives becomes

$$\hat{\psi}_n = i\,\mathrm{sgn}(\omega)\hat{\psi}_s, \tag{40}$$

where s is the arclength. This formula defines the Fourier symbol for the relation between the inward normal and tangential derivative for general boundary data:

$$\psi_n^I(\alpha) = H(\psi_s^I)(\alpha). \tag{41}$$

Hence, by transforming the tangential and normal derivatives of ψ to the x and y directions, we arrive at

$$\psi_x^I = \sigma^{(0)} \left(X_\alpha^{(0)} - Y_\alpha^{(0)} H \right) \sigma^{(0)} \phi'_\alpha, \quad r = R, \tag{42}$$

$$\psi_y^I = \sigma^{(0)} \left(Y_\alpha^{(0)} + X_\alpha^{(0)} H \right) \sigma^{(0)} \phi'_\alpha, \quad r = R. \tag{43}$$

The system (23), (24), (25) now takes the form

$$\begin{aligned} \phi'_t &= -(\sigma^{(0)})^3 \left(Y'_{\alpha\alpha} X_\alpha^{(0)} - X'_{\alpha\alpha} Y_\alpha^{(0)} + X'_\alpha Y_{\alpha\alpha}^{(0)} - Y'_\alpha X_{\alpha\alpha}^{(0)} \right) \\ &\quad + 3\kappa^{(0)} (\sigma^{(0)})^2 \left(X'_\alpha X_\alpha^{(0)} + Y'_\alpha Y_\alpha^{(0)} \right), \end{aligned} \tag{44}$$

$$X'_t = \sigma^{(0)} \left(X_\alpha^{(0)} - Y_\alpha^{(0)} H \right) \sigma^{(0)} \phi'_\alpha, \tag{45}$$

$$Y'_t = \sigma^{(0)} \left(Y_\alpha^{(0)} + X_\alpha^{(0)} H \right) \sigma^{(0)} \phi'_\alpha. \tag{46}$$

Freezing the coefficients and Fourier transforming the dependent variables yields

$$\frac{\partial}{\partial t} \begin{pmatrix} \hat{\phi}' \\ \hat{X}' \\ \hat{Y}' \end{pmatrix} = A \begin{pmatrix} \hat{\phi}' \\ \hat{X}' \\ \hat{Y}' \end{pmatrix}, \tag{47}$$

where

$$A = (\sigma^{(0)})^2 \begin{pmatrix} 0 & -a_{12}\omega^2 + b_{12} i\omega & -a_{13}\omega^2 + b_{13} i\omega \\ b_{21} i\omega & 0 & 0 \\ b_{31} i\omega & 0 & 0 \end{pmatrix}. \tag{48}$$

The coefficients are

$$\begin{aligned} a_{12} &= \sigma^{(0)} Y_\alpha^{(0)}, \\ a_{13} &= -\sigma^{(0)} X_\alpha^{(0)}, \\ b_{12} &= 3\kappa^{(0)} X_\alpha^{(0)} - \sigma^{(0)} Y_{\alpha\alpha}^{(0)}, \\ b_{13} &= 3\kappa^{(0)} Y_\alpha^{(0)} + \sigma^{(0)} X_{\alpha\alpha}^{(0)}, \\ b_{21} &= X_\alpha^{(0)} - Y_\alpha^{(0)} i \mathrm{sgn}(\omega), \\ b_{31} &= Y_\alpha^{(0)} + X_\alpha^{(0)} i \mathrm{sgn}(\omega). \end{aligned}$$

The eigenvalues of A are the roots of $\det(A - \lambda I) = 0$, where

$$\det(A - \lambda I) = -\lambda^3 + \lambda i\omega (\sigma^{(0)})^4 \left((-a_{12}\omega^2 + b_{12} i\omega) b_{21} + (-a_{13}\omega^2 + b_{13} i\omega) b_{31} \right). \tag{49}$$

One root is $\lambda_1 = 0$ and after some algebra we find that the other two roots are the solutions of

$$\lambda^2 = -(\sigma^{(0)})^3|\omega|^3 - 2\kappa^{(0)}(\sigma^{(0)})^2\omega^2 + (\sigma^{(0)})^2\sigma_\alpha^{(0)} i\omega|\omega|. \tag{50}$$

In our case, $\sigma^{(0)} = 1/R$, so $\sigma_\alpha^{(0)} = 0$ and the remaining roots become purely imaginary,

$$\lambda_{2,3} = \pm i(\sigma^{(0)}|\omega|)^{3/2}\sqrt{1 + \frac{2\kappa^{(0)}}{\sigma^{(0)}}\frac{1}{|\omega|}}. \tag{51}$$

We will use (51) to estimate how large the time step can be without violating the stability criterion of the time integration method. It is our experience from numerical experiments that this expression yields a useful estimate of the spectrum even when the cross-section is far from being circular and when the potential is far from being constant, if the values of $\sigma^{(0)}(\alpha)$ and $\kappa^{(0)}(\alpha)$ that maximize the eigenvalues are used.

3 The approximate system

If the governing equations are integrated in time with the four stage Runge–Kutta scheme, the right hand side of (1)–(3) must be evaluated four times per time step, and each evaluation requires the solution of a Dirichlet problem. Even though the solution of the Dirichlet problem changes very little between each time step, and iterative methods can be constructed that only require a few iterations to find the solution, we have found that the bulk of the computation consists of solving Dirichlet's problem. In this section we derive an approximate system which can be integrated much faster than the original system. The approximation is based on linearizing the velocity terms in the governing equations followed by approximating the solution of Dirichlet's problem.

We will estimate the error in the solution of the approximate system in the L_2-norm, which we define for 2π-periodic vector functions $\bar{F} = (F_1, F_2, F_3)^T$ and $\bar{G} = (G_1, G_2, G_3)^T$ according to

$$\|\bar{G}\|_2 = \langle \bar{G}, \bar{G}\rangle_2^{1/2}, \quad \langle \bar{F}, \bar{G}\rangle_2 = \int_0^{2\pi} \sum_{i=1}^{3} F_i^*(\alpha) G_i(\alpha)\, d\alpha. \tag{52}$$

We proceed by improving the analysis of the linearized velocity (10) to allow for a non-constant $\phi^{(0)}(\alpha)$ and include effects from the term $\nabla\psi^{II}$, defined by (32), (33). As in [4] and [7], we find it convenient to decompose $\bar{u}'$ into its normal and tangential components. Let the unperturbed inward unit normal vector be $\bar{N}^{(0)}$ and the unperturbed unit tangent vector be $\bar{T}^{(0)}$. In terms of $X^{(0)}$ and $Y^{(0)}$,

$$\bar{T}^{(0)} = \sigma^{(0)} \begin{pmatrix} X_\alpha^{(0)} \\ Y_\alpha^{(0)} \end{pmatrix}, \quad \bar{N}^{(0)} = \sigma^{(0)} \begin{pmatrix} -Y_\alpha^{(0)} \\ X_\alpha^{(0)} \end{pmatrix}. \tag{53}$$

We easily get from (21) and (22),

$$\bar{u}' \cdot \bar{T}^{(0)} = \nabla\psi^I(X^{(0)}, Y^{(0)}) \cdot \bar{T}^{(0)} + \lim_{\epsilon \to 0} \frac{1}{\epsilon}\left(\nabla\psi^{II}(X, Y) \cdot \bar{T}^{(0)} - \bar{u}^{(0)} \cdot \bar{T}^{(0)}\right) \tag{54}$$

$$\bar{u}' \cdot \bar{N}^{(0)} = \nabla\psi^I(X^{(0)}, Y^{(0)}) \cdot \bar{N}^{(0)} + \lim_{\epsilon \to 0} \frac{1}{\epsilon}\left(\nabla\psi^{II}(X, Y) \cdot \bar{N}^{(0)} - \bar{u}^{(0)} \cdot \bar{N}^{(0)}\right) \tag{55}$$

The gradient of ψ^{II} can be written as

$$\nabla\psi^{II} = \frac{\partial\psi^{II}}{\partial s}\bar{T} + \frac{\partial\psi^{II}}{\partial n}\bar{N}, \tag{56}$$

with $\partial/\partial s$ being the derivative with respect to arclength and $\partial/\partial n$ being the inward normal derivative. Straightforward algebra yields

$$\sigma = \sigma^{(0)} - \epsilon(\sigma^{(0)})^2\bar{T}^{(0)} \cdot \begin{pmatrix} X'_\alpha \\ Y'_\alpha \end{pmatrix} + \mathcal{O}(\epsilon^2), \tag{57}$$

and

$$\bar{T} \cdot \bar{T}^{(0)} = 1 + \mathcal{O}(\epsilon^2), \tag{58}$$

$$\bar{N} \cdot \bar{T}^{(0)} = -\epsilon\sigma^{(0)}\bar{N}^{(0)} \cdot \begin{pmatrix} X'_\alpha \\ Y'_\alpha \end{pmatrix} + \mathcal{O}(\epsilon^2). \tag{59}$$

We now study the tangential component $\bar{u}' \cdot \bar{T}^{(0)}$. We have

$$\bar{u}^{(0)} \cdot \bar{T}^{(0)} = \sigma^{(0)}\phi^{(0)}_\alpha, \quad \frac{\partial\psi^{II}}{\partial n} = \bar{u}^{(0)} \cdot \bar{N}^{(0)} + \mathcal{O}(\epsilon), \tag{60}$$

and because ψ^{II} satisfies (33),

$$\frac{\partial\psi^{II}}{\partial s} = \sigma\phi^{(0)}_\alpha. \tag{61}$$

Hence, (54) and (56) yield

$$\begin{aligned}
\bar{u}' \cdot \bar{T}^{(0)} &= \nabla\psi^I \cdot \bar{T}^{(0)} + \lim_{\epsilon\to 0}\frac{1}{\epsilon}\left(\sigma\phi^{(0)}_\alpha \bar{T} \cdot \bar{T}^{(0)} + \left(\bar{u}^{(0)} \cdot \bar{N}^{(0)} + \mathcal{O}(\epsilon)\right)\bar{N} \cdot \bar{T}^{(0)} - \sigma^{(0)}\phi^{(0)}_\alpha\right) \\
&= \sigma^{(0)}\left(\phi'_\alpha - \left((\bar{u}^{(0)} \cdot \bar{T}^{(0)})\bar{T}^{(0)} + (\bar{u}^{(0)} \cdot \bar{N}^{(0)})\bar{N}^{(0)}\right) \cdot \begin{pmatrix} X'_\alpha \\ Y'_\alpha \end{pmatrix}\right) \\
&= \sigma^{(0)}\left(\phi'_\alpha - u^{(0)}X'_\alpha - v^{(0)}Y'_\alpha\right).
\end{aligned} \tag{62}$$

We remark that only geometrical arguments were used in the derivation of $\bar{u}' \cdot \bar{T}^{(0)}$. Hence, ψ^I and ψ^{II} being solutions of Laplace's equation has no bearing on the form of $\bar{u}' \cdot \bar{T}^{(0)}$.

The normal component $\bar{u}' \cdot \bar{N}^{(0)}$ is more difficult to analyze, because it requires knowledge of how the normal derivative of the solution of Dirichlet's problem depends on the shape of the domain Ω. Without analysis, but recalling from (41) the exact relation between the inward normal derivative and the tangential derivative when the domain is circular, we will approximate $\bar{u}' \cdot \bar{N}^{(0)}$ by

$$\bar{u}' \cdot \bar{N}^{(0)} \approx H(\bar{u}' \cdot \bar{T}^{(0)}) = H\left(\sigma^{(0)}\left(\phi'_\alpha - u^{(0)}X'_\alpha - v^{(0)}Y'_\alpha\right)\right). \tag{63}$$

It can be shown that the error $\bar{u}' \cdot \bar{N}^{(0)} - H(\bar{u}' \cdot \bar{T}^{(0)})$ is a smoothing operator of ϕ', X' and Y', cf. [4] and [7]. The linearized velocity (10) is therefore approximately

$$\bar{u}' \approx \bar{T}^{(0)}(\bar{u}' \cdot \bar{T}^{(0)}) + \bar{N}^{(0)}(\bar{u}' \cdot \bar{N}^{(0)}).$$

Because the tangential component of the velocity is known exactly, we will only use the above expression for the normal component of the velocity. We have $\bar{T}^{(0)} \cdot \bar{N} = \mathcal{O}(\epsilon)$ and $\bar{N}^{(0)} \cdot \bar{N} = 1 + \mathcal{O}(\epsilon^2)$, so it is consistent with the linearization to take the approximate velocity $\tilde{\bar{u}} \approx \bar{u}$ to be

$$\tilde{\bar{u}} \equiv \begin{pmatrix} \tilde{u} \\ \tilde{v} \end{pmatrix} = \bar{T}(\sigma\phi_\alpha) + \bar{N}\left(\bar{u}^{(0)} \cdot \bar{N} + \epsilon H(\sigma^{(0)}(\phi'_\alpha - u^{(0)}X'_\alpha - v^{(0)}Y'_\alpha))\right). \tag{64}$$

In the approximate system we replace the exact velocity by the approximation (64). To compensate for the smooth error in (64), we also introduce a time-dependent forcing which we will present below. This results in the system (14)–(16).

We will derive the optimal form of the forcing G_i, $i = 1, 2, 3$, and estimate the error in the solution of the approximate system by solving both the original and approximate systems by asymptotic expansions in time. We start by considering the original system (1)–(3) for $t \geq t_0$ subject to the initial conditions

$$\phi(\alpha, t_0) = \phi^{(0)}(\alpha), \tag{65}$$
$$X(\alpha, t_0) = X^{(0)}(\alpha), \tag{66}$$
$$Y(\alpha, t_0) = Y^{(0)}(\alpha). \tag{67}$$

For $t:\ 0 \leq t - t_0 \ll 1$, we make the Ansatz

$$\begin{aligned} \phi(\alpha, t) &= \phi^{(0)}(\alpha) + \sum_{k=1}^{p} \frac{1}{k}(t - t_0)^k \phi^{(k)}(\alpha) + \mathcal{O}((t - t_0)^{p+1}), \\ X(\alpha, t) &= X^{(0)}(\alpha) + \sum_{k=1}^{p} \frac{1}{k}(t - t_0)^k X^{(k)}(\alpha) + \mathcal{O}((t - t_0)^{p+1}), \\ Y(\alpha, t) &= Y^{(0)}(\alpha) + \sum_{k=1}^{p} \frac{1}{k}(t - t_0)^k Y^{(k)}(\alpha) + \mathcal{O}((t - t_0)^{p+1}). \end{aligned} \tag{68}$$

The velocity satisfies (9) and by linearizing the curvature, we get

$$\kappa = \kappa^{(0)} + \epsilon\kappa' + \mathcal{O}(\epsilon^2), \tag{69}$$
$$\kappa' = \frac{\partial \kappa}{\partial X} X' + \frac{\partial \kappa}{\partial Y} Y'. \tag{70}$$

Hence, inserting (68) into both (9) and (69) yield

$$\bar{u} = \bar{u}^{(0)} + (t - t_0)\left\{\frac{\partial \bar{u}}{\partial \phi}\phi^{(1)} + \frac{\partial \bar{u}}{\partial X}X^{(1)} + \frac{\partial \bar{u}}{\partial Y}Y^{(1)}\right\} + \mathcal{O}((t - t_0)^2), \tag{71}$$

$$\kappa = \kappa^{(0)} + (t - t_0)\left\{\frac{\partial \kappa}{\partial X}X^{(1)} + \frac{\partial \kappa}{\partial Y}Y^{(1)}\right\} + \mathcal{O}((t - t_0)^2). \tag{72}$$

We determine the functions $\phi^{(k)}$, $X^{(k)}$, $Y^{(k)}$ by inserting (68), (71) and (72) into the original system (1)–(3) and identifying the terms with the same power in $t - t_0$. This results in:

$$\phi^{(1)} = \frac{1}{2}\left((u^{(0)})^2 + (v^{(0)})^2\right) - \kappa^{(0)}, \tag{73}$$

$$X^{(1)} = u^{(0)}, \tag{74}$$

$$Y^{(1)} = v^{(0)}. \tag{75}$$

The $\mathcal{O}\left((t-t_0)^2\right)$ terms are

$$\begin{aligned}\phi^{(2)} &= u^{(0)}\left\{\frac{\partial u}{\partial \phi}\phi^{(1)} + \frac{\partial u}{\partial X}X^{(1)} + \frac{\partial u}{\partial Y}Y^{(1)}\right\} \\ &\quad + v^{(0)}\left\{\frac{\partial v}{\partial \phi}\phi^{(1)} + \frac{\partial v}{\partial X}X^{(1)} + \frac{\partial v}{\partial Y}Y^{(1)}\right\} - \frac{\partial \kappa}{\partial X}X^{(1)} - \frac{\partial \kappa}{\partial Y}Y^{(1)},\end{aligned} \tag{76}$$

$$X^{(2)} = \frac{\partial u}{\partial \phi}\phi^{(1)} + \frac{\partial u}{\partial X}X^{(1)} + \frac{\partial u}{\partial Y}Y^{(1)}, \tag{77}$$

$$Y^{(2)} = \frac{\partial v}{\partial \phi}\phi^{(1)} + \frac{\partial v}{\partial X}X^{(1)} + \frac{\partial v}{\partial Y}Y^{(1)}. \tag{78}$$

For the approximate system, we make the corresponding Ansatz

$$\begin{aligned}\tilde{\phi}(\alpha,t) &= \phi^{(0)}(\alpha) + \sum_{k=1}^{p}\frac{1}{k}(t-t_0)^k\tilde{\phi}^{(k)}(\alpha) + \mathcal{O}((t-t_0)^{p+1}), \\ \tilde{X}(\alpha,t) &= X^{(0)}(\alpha) + \sum_{k=1}^{p}\frac{1}{k}(t-t_0)^k\tilde{X}^{(k)}(\alpha) + \mathcal{O}((t-t_0)^{p+1}), \\ \tilde{Y}(\alpha,t) &= Y^{(0)}(\alpha) + \sum_{k=1}^{p}\frac{1}{k}(t-t_0)^k\tilde{Y}^{(k)}(\alpha) + \mathcal{O}((t-t_0)^{p+1}).\end{aligned} \tag{79}$$

Proceeding in the same way as for the original system, the terms in the asymptotic expansion for the solution of the approximate system become:

$$\tilde{\phi}^{(1)} = \frac{1}{2}\left((u^{(0)})^2 + (v^{(0)})^2\right) - \kappa^{(0)}, \tag{80}$$

$$\tilde{X}^{(1)} = u^{(0)}, \tag{81}$$

$$\tilde{Y}^{(1)} = v^{(0)}. \tag{82}$$

The $\mathcal{O}\left((t-t_0)^2\right)$ terms are

$$\begin{aligned}\tilde{\phi}^{(2)} &= u^{(0)}\left\{\frac{\partial \tilde{u}}{\partial \phi}\phi^{(1)} + \frac{\partial \tilde{u}}{\partial X}X^{(1)} + \frac{\partial \tilde{u}}{\partial Y}Y^{(1)}\right\} \\ &\quad + v^{(0)}\left\{\frac{\partial \tilde{v}}{\partial \phi}\phi^{(1)} + \frac{\partial \tilde{v}}{\partial X}X^{(1)} + \frac{\partial \tilde{v}}{\partial Y}Y^{(1)}\right\} - \frac{\partial \kappa}{\partial X}X^{(1)} - \frac{\partial \kappa}{\partial Y}Y^{(1)} + G_1,\end{aligned} \tag{83}$$

$$\tilde{X}^{(2)} = \frac{\partial \tilde{u}}{\partial \phi}\phi^{(1)} + \frac{\partial \tilde{u}}{\partial X}X^{(1)} + \frac{\partial \tilde{u}}{\partial Y}Y^{(1)} + G_2, \tag{84}$$

$$\tilde{Y}^{(2)} = \frac{\partial \tilde{v}}{\partial \phi}\phi^{(1)} + \frac{\partial \tilde{v}}{\partial X}X^{(1)} + \frac{\partial \tilde{v}}{\partial Y}Y^{(1)} + G_3. \tag{85}$$

By comparing the terms in the asymptotic expansions, we see that $\phi^{(1)} = \tilde{\phi}^{(1)}$, $X^{(1)} = \tilde{X}^{(1)}$ and $Y^{(1)} = \tilde{Y}^{(1)}$. Furthermore, the $\mathcal{O}((t-t_0)^2)$ terms will be identical if we take the forcing to be

$$\begin{aligned} G_1 &= u^{(0)}\left\{\frac{\partial u}{\partial \phi}\phi^{(1)} + \frac{\partial u}{\partial X}X^{(1)} + \frac{\partial u}{\partial Y}Y^{(1)} - \frac{\partial \tilde{u}}{\partial \tilde{\phi}}\phi^{(1)} - \frac{\partial \tilde{u}}{\partial \tilde{X}}X^{(1)} - \frac{\partial \tilde{u}}{\partial \tilde{Y}}Y^{(1)}\right\} \\ &\quad + v^{(0)}\left\{\frac{\partial v}{\partial \phi}\phi^{(1)} + \frac{\partial v}{\partial X}X^{(1)} + \frac{\partial v}{\partial Y}Y^{(1)} - \frac{\partial \tilde{v}}{\partial \tilde{\phi}}\phi^{(1)} - \frac{\partial \tilde{v}}{\partial \tilde{X}}X^{(1)} - \frac{\partial \tilde{v}}{\partial \tilde{Y}}Y^{(1)}\right\}, \quad (86) \\ G_2 &= \frac{\partial u}{\partial \phi}\phi^{(1)} + \frac{\partial u}{\partial X}X^{(1)} + \frac{\partial u}{\partial Y}Y^{(1)} - \frac{\partial \tilde{u}}{\partial \tilde{\phi}}\phi^{(1)} - \frac{\partial \tilde{u}}{\partial \tilde{X}}X^{(1)} - \frac{\partial \tilde{u}}{\partial \tilde{Y}}Y^{(1)}, \quad (87) \\ G_3 &= \frac{\partial v}{\partial \phi}\phi^{(1)} + \frac{\partial v}{\partial X}X^{(1)} + \frac{\partial v}{\partial Y}Y^{(1)} - \frac{\partial \tilde{v}}{\partial \tilde{\phi}}\phi^{(1)} - \frac{\partial \tilde{v}}{\partial \tilde{X}}X^{(1)} - \frac{\partial \tilde{v}}{\partial \tilde{Y}}Y^{(1)}. \quad (88) \end{aligned}$$

By this choice of forcing, the difference will satisfy $(\phi - \tilde{\phi}, X - \tilde{X}, Y - \tilde{Y})^T = \mathcal{O}\left((t-t_0)^3\right)$. We remark that the forcing terms are easily computed by numerical differentiation, i.e.,

$$\frac{\partial u}{\partial \phi}\phi^{(1)} + \frac{\partial u}{\partial X}X^{(1)} + \frac{\partial u}{\partial Y}Y^{(1)} = \lim_{\epsilon \to 0}\frac{1}{\epsilon}\left(u[\phi^{(0)} + \epsilon\phi^{(1)}, X^{(0)} + \epsilon X^{(1)}, Y^{(0)} + \epsilon Y^{(1)}] - u^{(0)}\right),$$

and a corresponding expression for the v component of the velocity. In the practical computation, ϵ is taken to be a small positive number.

The approximate system will be integrated up to time $t = t_0 + \Delta T$, after which the linearization of the velocity is redone and a new approximate system is formed. The error in the approximate solution at time $t_0 + \Delta T$ can be estimated in the following way. Lets denote the solution of the approximate system at time $t_0 + \Delta T$ by $\tilde{\phi}(t_0 + \Delta T)$, $\tilde{X}(t_0 + \Delta T)$ and $\tilde{Y}(t_0 + \Delta T)$. From the previous analysis, we have that the error in that solution is of the order $\mathcal{O}(\Delta T^3)$. Hence, the exact velocity at that time can be approximated by

$$\begin{aligned} \bar{u}(t_0 + \Delta T) &\equiv \bar{u}[\tilde{\phi}(t_0 + \Delta T), \tilde{X}(t_0 + \Delta T), \tilde{Y}(t_0 + \Delta T)] \\ &= \bar{u}[\phi(t_0 + \Delta T), X(t_0 + \Delta T), Y(t_0 + \Delta T)] + \mathcal{O}(\Delta T^3). \quad (89) \end{aligned}$$

In the same way, the curvature satisfies

$$\kappa[X(t_0 + \Delta T), Y(t_0 + \Delta T)] = \kappa[\tilde{X}(t_0 + \Delta T), \tilde{Y}(t_0 + \Delta T)] + \mathcal{O}(\Delta T^3).$$

We also evaluate the approximate velocity

$$\tilde{\bar{u}}(t_0 + \Delta T) \equiv \tilde{\bar{u}}[\tilde{\phi}(t_0 + \Delta T), \tilde{X}(t_0 + \Delta T), \tilde{Y}(t_0 + \Delta T)].$$

At time $t_0 + \Delta T$, the time-derivative of the error therefore satisfies

$$\begin{aligned} (\phi - \tilde{\phi})_t &= \frac{1}{2}|\bar{u}(t_0 + \Delta T)|^2 - \frac{1}{2}|\tilde{\bar{u}}(t_0 + \Delta T)|^2 - \Delta T G_1 + \mathcal{O}(\Delta T^3), \quad (90) \\ (X - \tilde{X})_t &= u(t_0 + \Delta T) - \tilde{u}(t_0 + \Delta T) - \Delta T G_2 + \mathcal{O}(\Delta T^3), \quad (91) \\ (Y - \tilde{Y})_t &= v(t_0 + \Delta T) - \tilde{v}(t_0 + \Delta T) - \Delta T G_3 + \mathcal{O}(\Delta T^3). \quad (92) \end{aligned}$$

Hence, the difference between the third order terms in the asymptotic expansions fulfill

$$\phi^{(3)} - \tilde{\phi}^{(3)} = \frac{1}{\Delta T^2}\left(\frac{1}{2}\left|\bar{u}(t_0+\Delta T)\right|^2 - \frac{1}{2}\left|\tilde{\bar{u}}(t_0+\Delta T)\right|^2 - \Delta T G_1\right) := D_1, \tag{93}$$

$$X^{(3)} - \tilde{X}^{(3)} = \frac{1}{\Delta T^2}\left(u(t_0+\Delta T) - \tilde{u}(t_0+\Delta T) - \Delta T G_2\right) := D_2, \tag{94}$$

$$Y^{(3)} - \tilde{Y}^{(3)} = \frac{1}{\Delta T^2}\left(v(t_0+\Delta T) - \tilde{v}(t_0+\Delta T) - \Delta T G_3\right) := D_3, \tag{95}$$

as $\Delta T \to 0$. By neglecting the fourth order terms in the asymptotic expansions, the error at time $t_0 + \Delta T$ approximately satisfies

$$\|\bar{e}\|_2 \approx \frac{\Delta T^3 \|\bar{D}\|_2}{3}, \quad \bar{e} = \bar{\Phi} - \tilde{\bar{\Phi}}, \quad \bar{\Phi} = \begin{pmatrix} \phi \\ X \\ Y \end{pmatrix}, \quad \tilde{\bar{\Phi}} = \begin{pmatrix} \tilde{\phi} \\ \tilde{X} \\ \tilde{Y} \end{pmatrix}, \quad \bar{D} = \begin{pmatrix} D_1 \\ D_2 \\ D_3 \end{pmatrix}.$$

If we assume $\|\bar{D}\|_2$ to vary on a time-scale much slower than ΔT, we can use this estimate to adaptively adjust next ΔT to keep the error in the approximate system on a constant level. We enforce the relative error $\|\bar{e}\|_2 / \|\tilde{\bar{\Phi}}\|_2 \leq \delta$ by taking

$$\Delta T = \delta^{1/3}\left(\frac{3\|\tilde{\bar{\Phi}}\|_2}{\|\bar{D}\|_2}\right)^{1/3}. \tag{96}$$

To integrate the solution from time 0 to T, the approximate system must be formed $N_T = T/\Delta T$ times. The local errors in each time interval will accumulate to a global relative error at time T, $E(T) \equiv \|\bar{\Phi}(T) - \tilde{\bar{\Phi}}(T)\|_2 / \|\tilde{\bar{\Phi}}(T)\|_2$, which can be expected to be of the order

$$E(T) = \mathcal{O}(N_T \delta) = \mathcal{O}\left(T\delta^{2/3}\left(\frac{\|\bar{D}\|_2}{3\|\tilde{\bar{\Phi}}\|_2}\right)^{1/3}\right). \tag{97}$$

To integrate the approximate system up to time T, Dirichlet's problem must be solved $2N_T$ times and the right hand side of the approximate system must be computed of the order $T\mathcal{O}((\max\sigma N)^{3/2})$ times. The time-step in the explicit integration of the approximate system is independent of δ so the effort in solving the approximate system is of the order

$$2TC_D\delta^{-1/3}\left(\frac{\|\bar{D}\|_2}{3\|\tilde{\bar{\Phi}}\|_2}\right)^{1/3} + 4TC_A(\max\sigma N)^{3/2}, \tag{98}$$

where C_D denotes the cost of solving one Dirichlet problem and C_A is the cost of evaluating the right hand side of the approximate system. It is our experience that the first term of (98) dominates the second term. In order to half the error, δ must decrease by a factor $1/\sqrt{8}$, which increases the δ-dependent cost of computing the solution by a factor $\sqrt{2}$. These estimates will be verified by the numerical examples in section 6.

4 The Dirichlet problem

There are several efficient ways of numerically solving the Dirichlet problem (7), (8). It would for instance be possible to employ a composite grid approach [10], but in the present method we utilize the more commonly used boundary integral technique [11] and rewrite the Dirichlet problem into an integral equation. In the remainder of this section, the time-dependence of ϕ, X and Y will be suppressed to make the notation clearer.

In terms of the real-valued dipole strength $\mu(\alpha)$, the complex potential Φ in the interior of Ω can be written as

$$\Phi(z) = \frac{1}{2\pi i}\int_0^{2\pi} \frac{\mu(\alpha')}{z(\alpha') - z} z_\alpha(\alpha')\, d\alpha', \tag{99}$$

where $z = x + iy$ and $z(\alpha') = X(\alpha') + iY(\alpha')$. The real-valued velocity potential is $\phi = \mathrm{Re}\,\Phi$. Using the limit of (99) as z approaches Γ, we deduce that ϕ on the boundary satisfies

$$\phi(\alpha) = \frac{1}{2}\mu(\alpha) + \mathrm{Re}\,\frac{1}{2\pi i} P.V. \int_0^{2\pi} \frac{\mu(\alpha')}{z(\alpha') - z(\alpha)} z_\alpha(\alpha')\, d\alpha'. \tag{100}$$

This is a Fredholm integral equation of the second kind. The integral equation (100) for the dipole strength is not ideally suited for discretization, because the integrand is singular at $\alpha' = \alpha$. We proceed by studying this singularity. Cauchy's theorem yields

$$P.V. \int_0^{2\pi} \frac{z_\alpha(\alpha')\, d\alpha'}{z(\alpha') - z(\alpha)} = P.V. \int_\Gamma \frac{dz'}{z' - z} = \pi i. \tag{101}$$

Hence,

$$P.V. \int_0^{2\pi} \frac{\mu(\alpha')}{z(\alpha') - z(\alpha)} z_\alpha(\alpha')\, d\alpha' = \mu(\alpha)\pi i + P.V. \int_0^{2\pi} \frac{\mu(\alpha') - \mu(\alpha)}{z(\alpha') - z(\alpha)} z_\alpha(\alpha')\, d\alpha'. \tag{102}$$

Inserting (102) into (100) gives

$$\phi(\alpha) = \mu(\alpha) + \mathrm{Re}\,\frac{1}{2\pi i} P.V. \int_0^{2\pi} \frac{\mu(\alpha') - \mu(\alpha)}{z(\alpha') - z(\alpha)} z_\alpha(\alpha')\, d\alpha'. \tag{103}$$

By Taylor expansion we find

$$\lim_{\alpha' \to \alpha} \mathrm{Re}\left\{ \frac{1}{2\pi i} \frac{\mu(\alpha') - \mu(\alpha)}{z(\alpha') - z(\alpha)} z_\alpha(\alpha') \right\} = \lim_{\alpha' \to \alpha} \mathrm{Re}\left\{ \frac{\mu_\alpha(\alpha)}{2\pi i} + \mathcal{O}(\alpha' - \alpha) \right\} = 0. \tag{104}$$

After utilizing this analytical limit, the integral (103) is straight forward to discretize.

By differentiating (99) with respect to z and integrating by parts, we obtain

$$\frac{d\Phi}{dz} = \frac{1}{2\pi i}\int_0^{2\pi} \frac{\mu_\alpha(\alpha')}{z(\alpha') - z}\, d\alpha', \tag{105}$$

and by taking the limit of (105) as z approaches Γ we find that the velocity on the boundary satisfies

$$u(\alpha) - iv(\alpha) = \frac{\mu_\alpha(\alpha)}{2z_\alpha(\alpha)} + \frac{1}{2\pi i} P.V. \int_0^{2\pi} \frac{\mu_\alpha(\alpha')}{z(\alpha') - z(\alpha)} \, d\alpha'. \tag{106}$$

Hence, instead of solving the Dirichlet problem (7), (8) and then differentiating the velocity potential to get the velocity on the boundary, we first solve (103) for the dipole strength μ. Thereafter, the integral (106) is evaluated for the velocity on the boundary.

5 Discretizing the systems

We define a grid on the boundary by $\alpha_j = (j-1)h$, $j = 1, 2, \ldots, N$, $h = 2\pi/N$ and let $f_j = f(\alpha_j)$ denote a grid function. Assume N to be even and define the discrete Fourier transform of f by

$$f(\alpha_j) = \hat{f}_0 + \sum_{\omega=1}^{N/2-1} \hat{f}_s(\omega) \sin(\omega\alpha_j) + \sum_{\omega=1}^{N/2} \hat{f}_c(\omega) \cos(\omega\alpha_j), \tag{107}$$

where

$$\hat{f}_0(\omega) = \frac{1}{N} \sum_{j=1}^{N} f(\alpha_j), \tag{108}$$

$$\hat{f}_s(\omega) = -\frac{2}{N} \sum_{j=1}^{N} f(\alpha_j) \sin(\omega\alpha_j), \tag{109}$$

$$\hat{f}_c(\omega) = \frac{2}{N} \sum_{j=1}^{N} f(\alpha_j) \cos(\omega\alpha_j). \tag{110}$$

Furthermore, let $S_h f_j$ be the spectral approximation of $df/d\alpha(\alpha_j)$, defined by

$$S_h f_j = \sum_{\omega=1}^{N/2-1} \omega \hat{f}_s(\omega) \cos(\omega\alpha_j) - \sum_{\omega=1}^{N/2-1} \omega \hat{f}_c(\omega) \sin(\omega\alpha_j) \tag{111}$$

Also note that the operator H applied to a grid function gives

$$H(f)(\alpha_j) = \sum_{\omega=1}^{N/2-1} \hat{f}_s(\omega) \cos(\omega\alpha_j) - \sum_{\omega=1}^{N/2-1} \hat{f}_c(\omega) \sin(\omega\alpha_j). \tag{112}$$

We refer to [12] and [13] for the theory of quadrature rules for integral equations. To discretize the integral equation (103) we apply the trapezoidal rule. Taking (104) into account yields

$$\phi_j = \mu_j + \frac{h}{2\pi} \sum_{\substack{k=1 \\ j \neq k}}^{N} \mathrm{Re}\left\{ \frac{S_h z_k}{i(z_k - z_j)} \right\} (\mu_k - \mu_j), \quad 1 \leq j \leq N, \tag{113}$$

where $z_k = X(\alpha_k) + iY(\alpha_k)$. Note that this discretization is spectrally accurate, because the integrand is periodic. The velocity integral (106) must be discretized with care to achieve a discrete system that is stable in time. It was shown by [5] and [6] that the "alternating-point-trapezoidal" method has this property. The discretization is given by

$$u_j - iv_j = \frac{S_h \mu_j}{2 S_h z_j} + \frac{h}{\pi i} \sum_{\substack{k=1 \\ (j-k) \text{ odd}}}^{N} \frac{S_h \mu_k}{z_k - z_j}, \tag{114}$$

where we denote the discrete approximation of the velocity components by $u_j \approx u(\alpha_j)$ and $v_j \approx v(\alpha_j)$. Similar to the trapezoidal rule, this formula is spectrally accurate when it is applied to a periodic function, because it can be written as a linear combination of the trapezoidal rule for grid sizes h and $2h$.

In order to compute the discrete velocity components (u_j, v_j) corresponding to a given discrete potential ϕ_j, we first solve the linear system (113) for the discrete dipole strength μ_j. Thereafter, the velocity components are calculated by evaluating (114). The dense non-symmetric matrix in (113) depends on the shape of the domain, and will hence change with time. It is therefore not economical to LU-decompose the matrix every time the time-derivatives need to be evaluated. Instead we solve (113) by the iterative method GMRES [14]. By using the solution from the previous time level as initial guess, the iteration converges to roundoff level in a few iterations. We found by numerical experiments that GMRES converged faster when (103) was discretized by the trapezoidal rule (as in (113)) than when it was discretized by the midpoint rule. We have used direct summation to evaluate the matrix-vector products in the GMRES iterations and for computing the velocity. The operational count for these operations is of the order $\mathcal{O}(N^2)$. We note that for large N, the cost of performing these tasks could be reduced to $\mathcal{O}(CN)$, where C is a large constant, by using the fast multi-pole method [8].

We discretize the system (1)–(3) and the approximate system (14)–(16) by the pseudo spectral method in space, i.e. spatial derivatives of dependent variables are computed by (111) and products are formed pointwise. For the approximate system, $H(f)$ is computed by (112). Applied to the original system (1)–(3), this procedure yields a system of ODE's for $\phi_j(t)$, $X_j(t)$ and $Y_j(t)$, $1 \le j \le N$, $t \ge 0$. For the approximate system (14)–(16) we get a system of ODE's for $\tilde{\phi}_j(t)$, $\tilde{X}_j(t)$ and $\tilde{Y}_j(t)$. We will integrate these systems of ODE's in time by the fourth order accurate four stage Runge–Kutta scheme. It is well known that this scheme is stable for problems with purely imaginary eigenvalues if $\Delta t \max|\lambda| \le C_{\Delta t}$, $C_{\Delta t} \approx 2.8$, where Δt is the time step. We find $\max|\lambda|$ by evaluating (51) for $\omega = \pi/h$, which is the largest spatial frequency on a grid with grid size $h = 2\pi/N$.

6 Numerical experiments

We will perform numerical experiments for the family of initial data given by:

$$\phi_0(\alpha) = 0, \qquad 0 \le \alpha \le 2\pi, \tag{115}$$

$$X_0(\alpha) = a\cos\alpha, \quad 0 \le \alpha \le 2\pi, \tag{116}$$

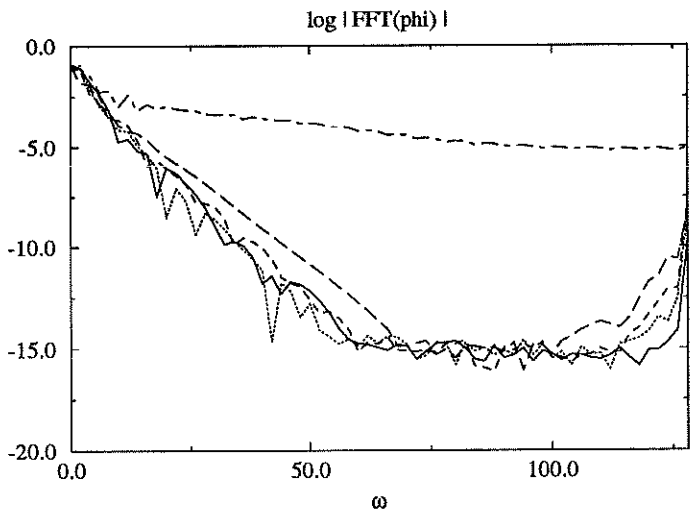

Figure 1: The spectrum of ϕ without filtering for initial data with $AR = 1.5$. The different lines correspond to times: 1.00 (solid), 2.00 (dotted), 2.37 (dashed), 2.69 (long-dashed) and 2.84 (dot-dashed).

$$Y_0(\alpha) = \frac{1}{a}\sin\alpha, \quad 0 \leq \alpha \leq 2\pi. \tag{117}$$

Hence, the initial velocity is zero and the initial cross section is elliptical with area π and aspect ratio $AR = a^2$. If $a = 1$, the solution of (1)–(3) is trivially given by $\phi(\alpha, t) = -t$, $X(\alpha, t) = \cos\alpha$ and $Y(\alpha, t) = \sin\alpha$. The symmetry between the x and y directions makes it sufficient to study $a > 1$.

We begin by integrating the system (1)–(3) to study its properties before we attempt the approximate system (14)–(16). All computations presented below were performed at $C_{\Delta t} = 2.5$, in 64 bits precision. The Fourier transforms were computed by the FFT-package for real-valued functions in the SLATEC library.

6.1 Stability and filtering of the discrete solution

We start with the case $a = \sqrt{1.5}$, i.e. $AR = 1.5$. We took $N = 256$ and integrated the discrete version of (1)–(3) up to time $t \approx 2.84$, after which the solution blew up. To discuss the instability it is instructive to study the time-evolution of $|\hat{\phi}(\omega)| =: \max(|\hat{\phi}_s(\omega)|, |\hat{\phi}_c(\omega)|)$. Because of symmetries in the initial data, $|\hat{\phi}(\omega)| = 0$, for $\omega = 1, 3, 5, \ldots$. Therefore, only even frequencies will be presented in the graphs of the spectrum for ϕ. In figure 1 we present $|\hat{\phi}(\omega)|$ as function of ω at different times. The component of the Nyquist frequency $\omega = N/2$ grows rapidly in time and it appears that once it has reached a critical level, the components of the lower frequencies become polluted through aliasing and this makes the solution blow up. We therefore introduce a filter by which the components of the Nyquist frequency ($\hat{\phi}_c(N/2)$, $\hat{X}_c(N/2)$, $\hat{Y}_c(N/2)$) are set to zero after every time step; this filter will be called the Nyquist filter. The time evolution of $|\hat{\phi}(\omega)|$ when the Nyquist filter is

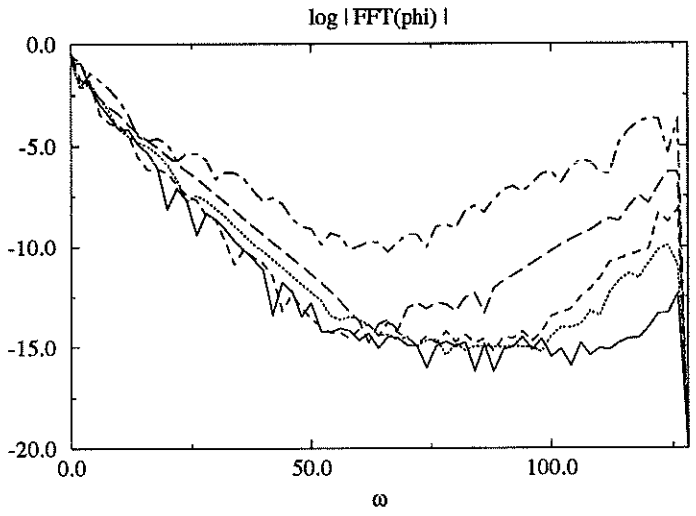

Figure 2: The spectrum of ϕ with Nyquist filtering for initial data with $AR = 1.5$. The different lines correspond to times: 2.00 (solid), 4.00 (dotted), 6.00 (dashed), 8.00 (long-dashed) and 9.62 (dot-dashed).

used is presented in figure 2. Clearly, the situation is improved because the solution can be integrated much longer in time. However, the components of the higher frequencies, which for a long time are at roundoff level, eventually start growing and make the solution blow up at time $t \approx 9.62$. We conjecture that the blow-up is caused by aliasing effects.

Several filtering techniques for stabilizing the solution have been proposed in the literature. One remedy of the instability problem, which first was proposed by Krasny [15] and more recently discussed by Shelley [16], is to set every Fourier component of ϕ, X and Y smaller than ϵ to zero after every time step; this filter will be referred to as the Krasny filter. Naturally, ϵ should be some small number greater than the machine precision. Note that when this filter is used, the highest frequency present in the solution will in general be smaller than $N/2$, which is taken into account when the time step is computed. We present the time-evolution of $|\hat{\phi}(\omega)|$ with Krasny filtering at $\epsilon = 10^{-14}$ in figure 3. Now the solution stays smooth for long times.

Our practical experience is that the Krasny filter stabilizes the solution if the resolution is sufficiently good to allow the spectrum to decay below the ϵ-level before $\omega = N/2$. If this condition is not satisfied, the filter will not take any action and the components of the high frequencies will start growing, which eventually destroys the solution. This behavior makes the Krasny filter somewhat fragile, because it is difficult to estimate how fast the spectrum will decay before the problem has been solved. To circumvent this problem we propose a different filter, by which the Fourier components corresponding to frequencies larger than $\omega = N/3$ are set to zero after every time step. The time step can therefore be calculated with $\omega = N/3$. This filter will be denoted the $N/3$-filter. In table 1 we compare solutions obtained with the $N/3$-filter to the solution computed with the Krasny filter with $\epsilon = 10^{-14}$ and $N = 384$. We conclude that the Krasny and $N/3$ filters have a

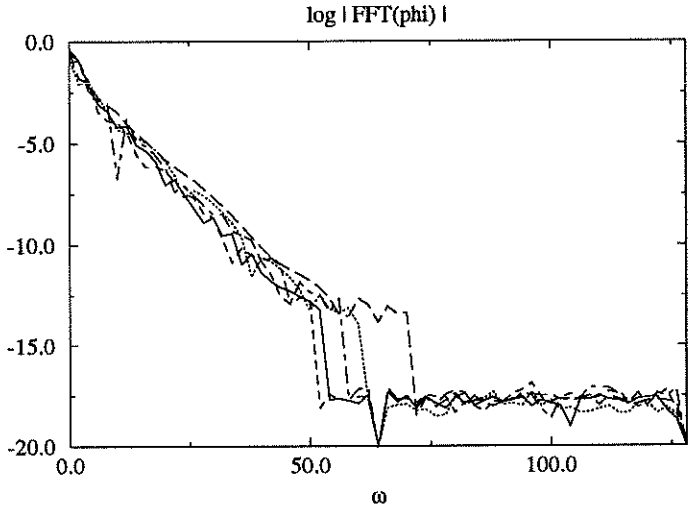

Figure 3: The spectrum of ϕ with Krasny filtering at $\epsilon = 10^{-14}$ for initial data with $AR = 1.5$ and $N = 256$. The different lines correspond to times: 2.00 (solid), 4.00 (dotted), 6.00 (dashed), 8.00 (long-dashed) and 10.00 (dot-dashed).

Rel. Diff.	N	CPU $[s]$
$8.85 \cdot 10^{-3}$	64	64
$1.17 \cdot 10^{-4}$	96	262
$3.16 \cdot 10^{-5}$	128	669
$3.25 \cdot 10^{-6}$	192	2696
$5.11 \cdot 10^{-7}$	256	7462
$8.86 \cdot 10^{-8}$	384	28459

Table 1: Relative difference measured in L_2-norm at time $t = 5.0$ between the solution obtained with Krasny filtering at $\epsilon = 10^{-14}$ and solutions obtained with the $N/3$ filter. The initial data had $AR = 2.0$. Note the savings in CPU-time when the resolution is decreased. The CPU-timings were done on a SUN IPX workstation.

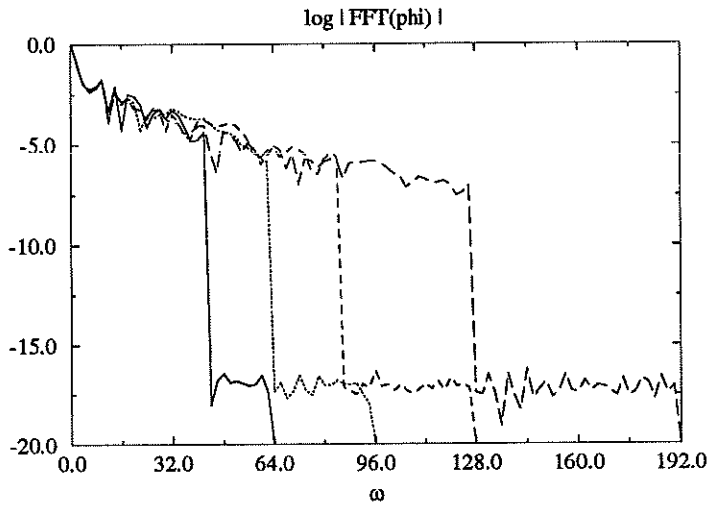

Figure 4: The spectrum of ϕ at time $t = 5.00$ for initial data with $AR = 3.0$ using the $N/3$ filtering with different resolutions: $N = 128$ (solid), $N = 192$ (dotted), $N = 256$ (dashed), $N = 384$ (long-dashed).

similar effect on the solution when the resolution is sufficiently good for the Krasny filter to take any action.

To demonstrate how slowly the spectrum decays for a larger initial AR, we present the spectrum at time $t = 5.0$ for the case with $AR = 3.0$ for different resolutions in figure 4. The shape of the free surface as function of time for the same case with the resolution $N = 256$ is presented in figure 5. This simulation required 9827 seconds of CPU-time on a SUN IPX workstation.

6.2 Accuracy of the approximate system

From the last example in the previous section, it is clear that simulating only the first oscillation of a jet with high initial aspect ratio, which calls for a large N to resolve the solution, requires a substantial amount of CPU-time. In these cases we are interested in speeding up the computation by replacing the original system by the approximate system. One oscillation of the jet yields important information for engineering applications [3], and we will therefore restrict our study to the time interval $0 \leq t \leq 3$, which approximately corresponds to the first oscillation of the jet. Similar to the original system, a filter must be introduced in the time-integration of the approximate system in order to achieve a smooth solution. We will for this purpose use the $N/3$ filter.

We begin by studying how the error in the approximate system depends on the time-interval between linearizing the velocity, ΔT. In table 2, we present the relative error at time $t = 3.0$ for the case $AR = 3.0$ with $N = 256$. The solution of the original system with the same resolution was taken as reference solution. From this example we see that the error is approximately proportional to ΔT^2. Also note that the estimated relative

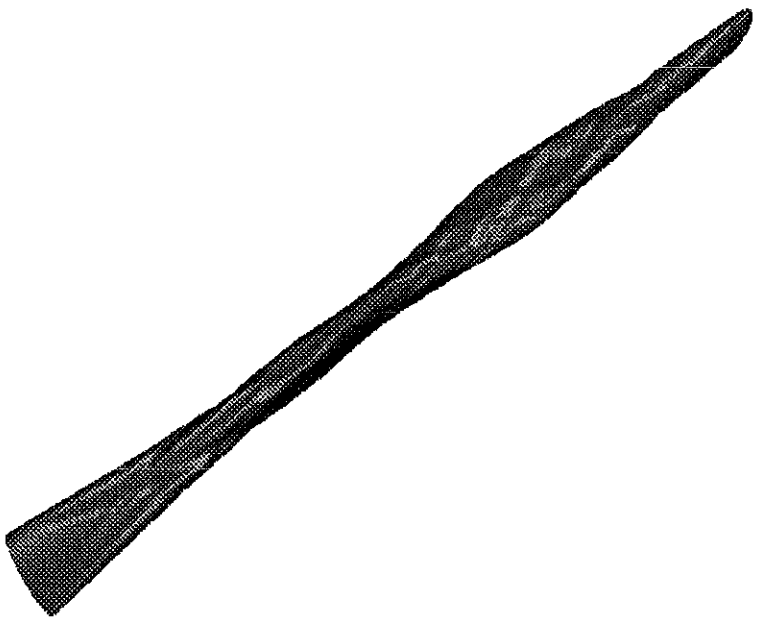

Figure 5: A rendered image of the free surface as function of time. The initial cross-section with $AR = 3.0$ is in the lower left corner. Time increases diagonally upwards and reaches $t = 5.0$ in the upper right corner. Note the chain-like appearance of the free surface which is observed in slender three-dimensional jets [3].

ΔT	Est. Error	Rel. Error	CPU $[s]$	N_T	$\Delta T/\Delta t$
$1.0 \cdot 10^{-2}$	$5.61 \cdot 10^{-3}$	$1.77 \cdot 10^{-3}$	61.1	300	5.03
$5.0 \cdot 10^{-3}$	$1.36 \cdot 10^{-3}$	$4.25 \cdot 10^{-4}$	103.5	600	2.52
$2.5 \cdot 10^{-3}$	$3.38 \cdot 10^{-4}$	$1.04 \cdot 10^{-4}$	194.1	1200	1.28

Table 2: Results from integrating the approximate system with a fixed time-interval ΔT between linearizing the velocity. Note that the error is approximately proportional to ΔT^2. In this example, $AR = 3.0$ and $N = 256$. The CPU-timings refer to one processor on a CRAY YMP. The original system required 610.7 seconds of CPU time.

error

$$E_{est} = \sum_{k=1}^{N_T} \frac{\Delta T^3 \|\bar{D}(t_k)\|_2}{3\|\bar{\tilde{\Phi}}(t_k)\|_2}, \tag{118}$$

consistently over-predicts the actual relative error by a factor ≈ 3. To indicate how much CPU-time is saved by using the approximate system, we also present the average number of explicit time-steps per ΔT, defined by

$$\frac{\Delta T}{\Delta t} = \frac{1}{N_T} \sum_{k=1}^{N_T} \frac{\Delta T}{\Delta t(t_k)}.$$

Twice the $\Delta T/\Delta t$ ratio gives a rather good estimate of how much faster it is to integrate the approximate system compared to the original system, because two Dirichlet problems are solved every time the velocity term is linearized and four Dirichlet problem are solved per explicit time-step when the original system is integrated.

The solutions of the approximate system are very accurate. For example, spending about 10% of the CPU-time resulted in an error of 0.177%. It is likely that the modeling errors caused by neglecting physical phenomena as viscosity and vorticity are much larger than this.

According to (93)–(95), the error in the approximation satisfies $\bar{D}\Delta T^3/3 + \mathcal{O}(\Delta T^4)$. To show that $\bar{D}$ is essentially independent of ΔT we present in figure 6 the L_2-norm of $\bar{D}$ as function of time for $\Delta T = 0.01$ and $\Delta T = 0.0025$. Again, the initial data had $AR = 3.0$ and the resolution was $N = 256$. Because the approximation is based on simplifying the normal component of the velocity, the error in the approximation can be suspected to depend on the properties of that quantity. To enable a close comparison, we also present the max-norm of the normal velocity in figure 6. We conclude that there is a clear correlation in time between the normal velocity and the error term $\bar{D}$. From this conclusion, we are lead to investigate how the size of the normal velocity affects the stability of the time-integration. In particular, we are interested in how large ΔT can be before the time-integration goes unstable. We therefore took initial data with increasing AR, which correspond to an increasing normal velocity, and increased ΔT until the solution would blow up. The largest ΔT where the solution did not blow up as function of the maximum norm of the normal velocity can be found in figure 7. In these computations, AR was in the range 1.5–3.5 and the resolution was in the range 128–512. We found that $N = 128$ only provided adequate resolution for $AR \leq 2.5$ and $N = 256$ was only good for $AR \leq 3.0$. The resolution $N = 384$ was adequate to resolve all AR and the $N = 512$ resolution was used to verify the $N = 384$ computations. The data-points fall on an almost straight line with slope ≈ -2.6 in the log-log graph, so we are lead to the estimate

$$\Delta T \leq \frac{C}{\left(|\bar{u} \cdot \bar{N}|_\infty\right)^{\bar{\gamma}}}, \quad \gamma \approx 2.6, \tag{119}$$

where C is independent of N. Therefore, as long as ΔT satisfies (119), it can be chosen to meet accuracy restrictions instead of stability restrictions.

We note that $\|\bar{D}\|_2$ varies by an order of magnitude in figure 6. From the error estimate (118), we see that a large portion of the error is committed when $\|\bar{D}\|_2$ is large. To better

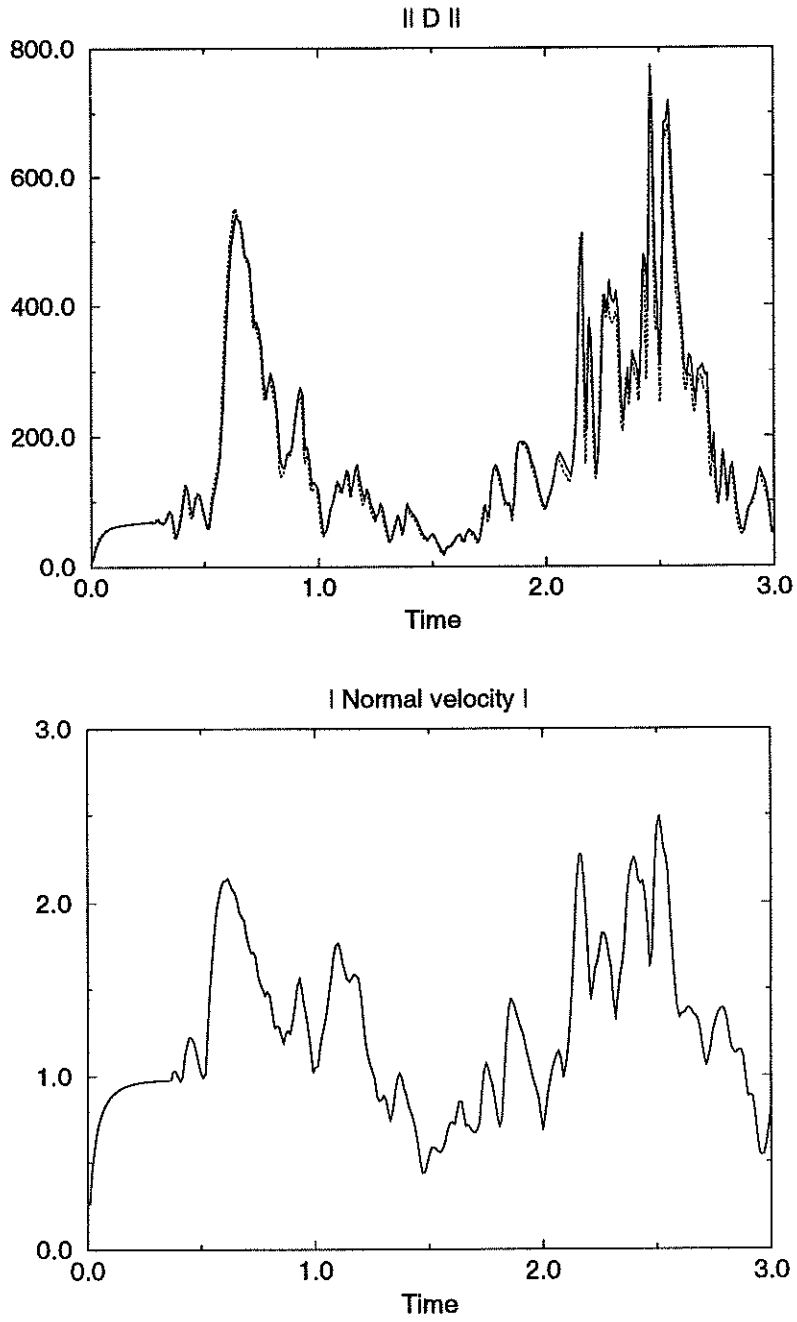

Figure 6: Top graph: The L_2-norm of the error term $\bar{D}$ as function of time for $\Delta T = 0.01$ (solid) and $\Delta T = 0.0025$ (dotted). Bottom graph: The max-norm of the normal component of the velocity as functions of time. In both graphs the aspect ratio was $AR = 3.0$ and $N = 256$.

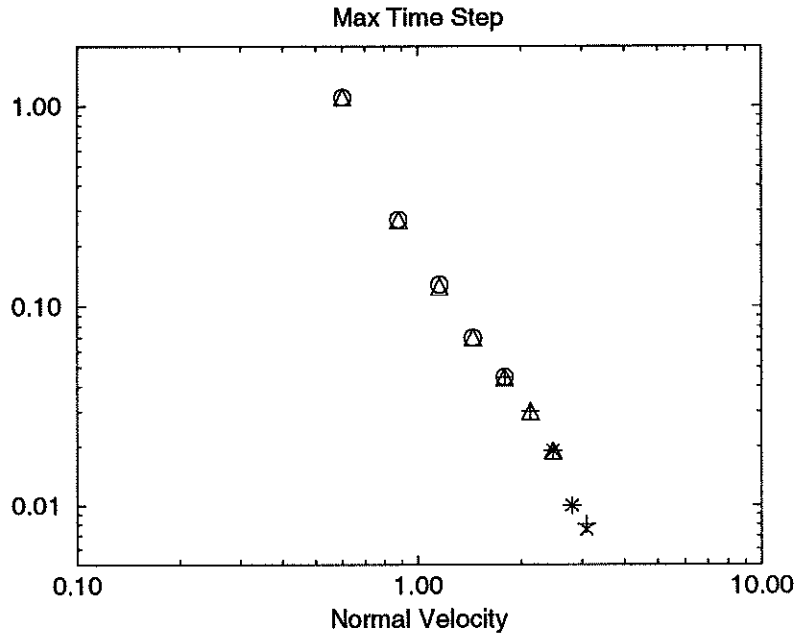

Figure 7: The maximum ΔT as function of the maximum norm of the normal velocity on a log-log scale. The initial data had $AR = 1.5 - 3.5$. The resolution was in the range $N = 128 - 512$ where $N = 128$ are circles, $N = 256$ are triangles, $N = 384$ are pluses, and $N = 512$ are denoted by X's. Note that ΔT is essentially independent of N.

δ	Est. Error	Rel. Error	CPU [s]	N_T	$\Delta T/\Delta t$
$1.25 \cdot 10^{-4}$	$2.28 \cdot 10^{-2}$	$7.73 \cdot 10^{-3}$	41.2	173	8.84
$4.42 \cdot 10^{-5}$	$1.01 \cdot 10^{-2}$	$3.45 \cdot 10^{-3}$	49.1	220	6.93
$1.56 \cdot 10^{-5}$	$4.77 \cdot 10^{-3}$	$1.64 \cdot 10^{-3}$	60.0	299	5.09
$5.52 \cdot 10^{-6}$	$2.34 \cdot 10^{-3}$	$8.09 \cdot 10^{-4}$	76.6	418	3.63
$1.95 \cdot 10^{-6}$	$1.15 \cdot 10^{-3}$	$4.00 \cdot 10^{-4}$	101.8	588	2.58

Table 3: Results from integrating the approximate system with a fixed δ. The CPU-timings refer to one processor on a CRAY YMP. In this case, $AR = 3.0$ and $N = 256$. The original system required 610.7 seconds of CPU-time.

optimize the computational resources, we will employ the adaptive time-step control (96). In table 3, we compare the solution of the approximate system with the solution of the original system at time $t = 3.0$. In this case, $AR = 3.0$ and $N = 256$. In this table, the estimated relative error (118) is compared to the actual relative error, where the solution of the original system with the same resolution was taken as reference solution. Note that (97) predicts that the relative error should half when δ is decreased by a factor $1/\sqrt{8}$. Our numerical results confirm this estimate. Also note that the estimated error is consistently over-predicting the actual error by a factor ≈ 3.

By comparing the number of linearizations, N_T, to the CPU-time in table 3, we see that the CPU-time/N_T ratio decreases as N_T increases. This indicates that the CPU-time is not only spent setting up the approximate system, but also used for integrating the approximate system. Therefore, the CPU-time/N_T ratio is larger when N_T is smaller. The estimate (98) predicts that N_T would increase by a factor $\sqrt{2}$ when δ is decreased by a factor $1/\sqrt{8}$. In the present example, N_T increases slightly slower than that.

To study how the resolution affects the solution of the approximate system, we doubled the number of grid points to $N = 512$. The result is presented in table 4. In this case the reference solution was taken to be the solution of the original system with $N = 512$. It can be seen that the relative error is essentially the same as for $N = 256$. This is explained by the fact that the relative difference between the solutions of the original systems with $N = 256$ and $N = 512$, respectively, is only $6.22 \cdot 10^{-5}$. The relative errors reported in tables 3 and 4 are therefore dominated by errors committed by simplifying the velocity in the approximate system.

The estimate (98) indicates that the cost of solving the approximate system would increase by a factor between 4 and $4\sqrt{2}$ when N is doubled, because C_D increases by a factor 4, C_A increases by a factor close to 2 and the number of time-steps for integrating the approximate system increases by a factor $2\sqrt{2}$. This estimate is verified by comparing the CPU-timings of table 3 and 4. This should be compared to the cost of solving the original system, which increases by a factor $8\sqrt{2}$ when N is doubled, because the cost of

δ	Est. Error	Rel. Error	CPU $[s]$	N_T	$\Delta T/\Delta t$
$1.25 \cdot 10^{-4}$	$2.45 \cdot 10^{-2}$	$8.72 \cdot 10^{-3}$	189.2	184	23.86
$4.42 \cdot 10^{-5}$	$1.03 \cdot 10^{-2}$	$3.46 \cdot 10^{-3}$	205.4	224	19.24
$1.56 \cdot 10^{-5}$	$4.79 \cdot 10^{-3}$	$1.66 \cdot 10^{-3}$	250.1	300	14.34
$5.52 \cdot 10^{-6}$	$2.35 \cdot 10^{-3}$	$8.15 \cdot 10^{-4}$	317.0	419	10.25
$1.95 \cdot 10^{-6}$	$1.15 \cdot 10^{-3}$	$4.02 \cdot 10^{-4}$	408.6	589	7.28

Table 4: Results from integrating the approximate system with a fixed δ. The CPU-timings refer to one processor on a CRAY YMP. In this case, $AR = 3.0$ and $N = 512$. The original system required 6218.8 seconds of CPU-time.

solving a Dirichlet problem increases by a factor 4 and the number of time-steps increases by a factor $2\sqrt{2}$. The CPU-timings for the original system also confirm this estimate. For this reason the relative saving in CPU-time by using the approximate system increases when the resolution increases. For example, at this resolution, we only have to spend 3% of the CPU-time required by the original system to achieve a solution with the relative error 0.34%.

We conclude by using the approximate system to simulate the complicated dynamics starting from the initial data with $AR = 5.0$. In figure 8, we show the time-evolution of the cross-sections of the jet. This simulation was done with the resolution $N = 1536$ and $\Delta T = 1.0 \cdot 10^{-3}$. It required 7,322 seconds of CPU-time on one processor on a CRAY YMP. The average $\Delta T/\Delta t$ ratio was 7.68, which indicates that it would have taken approximately 15 times longer to compute this solution if the original system had been used. The error estimate (118) predicted that the relative error at time $t = 1.25$ was $9.0 \cdot 10^{-4}$.

7 Conclusions

We have presented a system of partial differential equations that approximate the governing equations for inviscid free surface flows subject to surface tension. The approximation is based on repeated linearization of the velocity term together with a small scale approximation of the perturbation of the velocity. Two Dirichlet problems must be solved to form the approximate system, after which it can be evolved without solving Dirichlet problems. The accuracy of the solution of the approximate system is determined by how often the velocity term is linearized. This time-interval is denoted ΔT. We have shown that the error in the solution of the approximate system at a fixed time T is of the order $\mathcal{O}(\Delta T^2)$. We have exemplified the use of the approximate system by integrating the equations governing a slender non-axisymmetric three-dimensional jet subject to surface tension, where the evolution of the cross-section of the jet is governed by the two-dimensional Euler

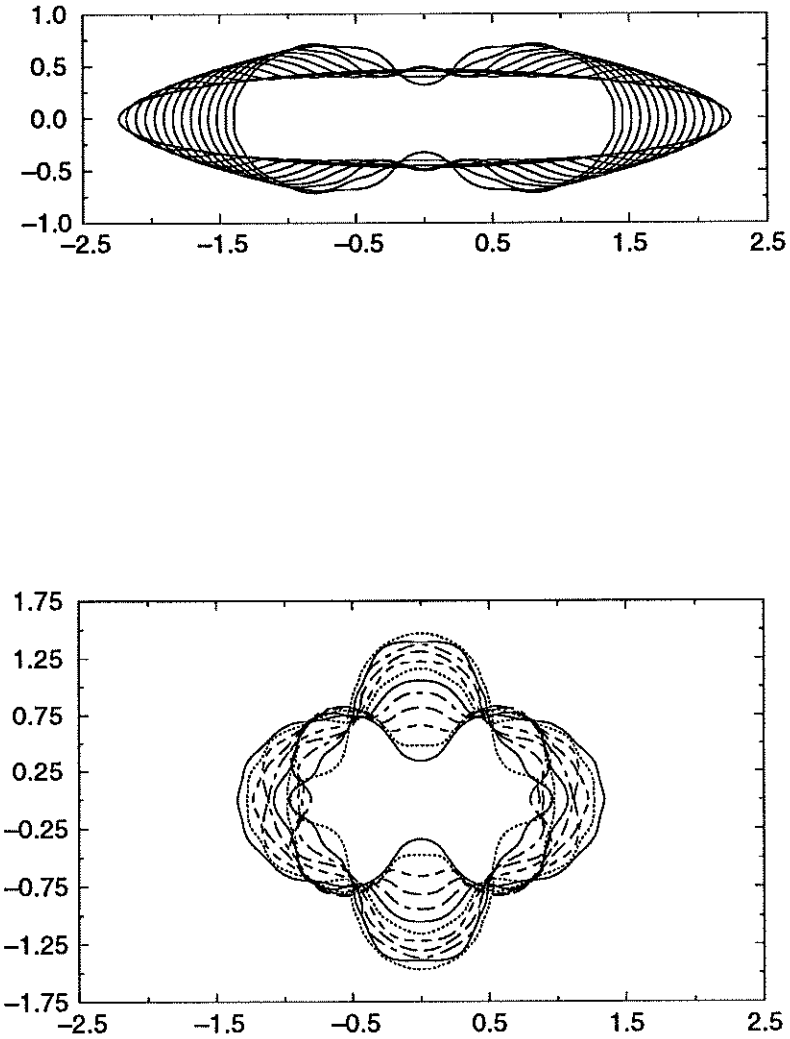

Figure 8: Time-evolution of the cross-sections of the jet with initial aspect ratio $AR = 5.0$. Top graph: t=0.00 to t=0.65 with spacing 0.05. Bottom graph: t=0.70 to t=1.25 with spacing 0.05. Here, t=0.70 is solid, t=0.75 is dotted, t=0.80 is dashed, t=0.85 is long-dashed, t=0.90 is dot-dashed, t=0.95 is solid, t=1.00 is dotted, t=1.05 is dashed, t=1.10 is long-dashed, t=1.15 is dot-dashed, t=1.20 is solid, and t=1.25 is dotted.

is closely correlated to the size of the normal velocity, and that there is a stability limit of the form $\Delta T \leq C/\left(|\bar{u}\cdot\bar{N}|_\infty\right)^\gamma$, where $\bar{u}\cdot\bar{N}$ denotes the normal velocity and $\gamma \approx 2.6$. Importantly, C is independent of the resolution. Hence, the time-step ΔT can be chosen independently of the number of grid points, N. This is in contrast to the time-step when the original system is integrated, where the stability limit is $\Delta t \leq \mathcal{O}(N^{-3/2})$ and four Dirichlet problems have to be solved per Δt (for the four stage Runge–Kutta scheme). Therefore, the relative saving of CPU-time by using the approximate system increases when the resolution increases. For example, in the case with elliptical initial cross-section with aspect ratio 3.0 and 256 grid points, we compared the solution of the original and approximate systems at time $t = 3$, which approximately corresponds to one oscillation of the jet. Spending 8% of the CPU-time necessary to solve the original system yielded an error in the solution of 0.34%; spending 12.5% of the CPU-time gave the error 0.08%. When the resolution was doubled to 512 grid points we only needed to spend 3% of the CPU-time to achive the error 0.34% and 5% of the CPU-time to get the error 0.08%. We conclude that the approximate system yields a powerful tool for simulating free surface problems with surface tension, especially when the spatial grid size must be small to accurately resolve the solution.

8 Acknowledgement

This work was supported by the U. S. Department of Energy through Los Alamos National Laboratory under contract W–7405–ENG–36.

References

[1] G. R. Baker and D. W. Moore. The rise and distortion of a two–dimensional gas bubble in an inviscid liquid. *Phys. Fluids*, A1:1451–1459, 1989.

[2] D. I. Pullin. Numerical studies of surface-tension effects in nonlinear Kelvin-Helmholtz and Rayleigh-Taylor instabilities. *J. Fluid Mech.*, 119:507–532, 1982.

[3] S. E. Bechtel. The oscillation of slender elliptical inviscid and Newtonian jets: Effects of surface tension, inertia, viscosity and gravity. *J. Applied Mech.*, 56:968–974, 1989.

[4] J. T. Beale, T. Y. Hou, and J. S. Lowengrub. Growth rates for the linearized motion of fluid interfaces away from equilibrium. *Comm. Pure Appl. Math.*, XLVI:1269–1301, 1993.

[5] J. T. Beale, T. Y. Hou, and J. S. Lowengrub. Convergence of a spectrally accurate boundary integral method for water waves. Preprint, Courant Institute of Mathematical Sciences, 1993.

[6] G. R. Baker and A. Nachbin. Stable methods for vortex sheet motion in the presence of surface tension. Technical Report 92–23, Dept. of Mathematics, Ohio State University, 1992.

[7] N. A. Petersson and H.-O. Kreiss. Well–posedness of 2–D free surface flows subject to gravity or surface tension. LA–UR 94-2978, Los Alamos National Laboratory, NM, 1994. (Submitted to J. Part. Diff. Eqn's).

[8] L. Greengard and V. Roklin. A fast algorithm for particle summations. *J. Comput. Phys.*, 105:267, 1987.

[9] T. Y. Hou, J. Lowengrub, and M. J. Shelley. Removing the stiffness from interfacial flows with surface tension. Preprint, Courant Institute of Mathematical Sciences, 1993.

[10] G. Chesshire and W. D. Henshaw. Composite overlapping meshes for the solution of partial differential equations. *J. Comput. Phys.*, 90(1):1–64, 1990.

[11] G. R. Baker. Generalized vortex methods for free-surface flows. In R. Meyer, editor, *Waves on Fluid Interfaces*, pages 53–81. Univ. Wisc. Press, 1983.

[12] C. T. H. Baker. *The numerical treatment of integral equations.* Clerendon Press, Oxford, 1977.

[13] A. Sidi and M. Israeli. Quadrature methods for periodic singular and weakly singular Fredholm integral equations. *J. Sci. Comp.*, 3:323, 1988.

[14] Y. Saad and M. H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7:856–869, 1986.

[15] R. Krasny. A study of singularity formation in a vortex sheet by the point-vortex approximation. *J. Fluid Mech.*, 167:65–93, 1986.

[16] M. J. Shelley. A study of singularity formation in vortex–sheet motion by a spectrally accurate vortex method. *J. Fluid Mech.*, 244:493–526, 1992.