Viscous Aerodynamics Modeling Using a Mesh-Free Approach

Damyn Chipman

A senior thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Bachelor of Science

Andrew Ning (Mechanical Engineering) and David Neilsen (Physics), Advisors

Department of Physics and Astronomy

Brigham Young University

Copyright © 2019 Damyn Chipman

All Rights Reserved

ABSTRACT

Viscous Aerodynamics Modeling Using a Mesh-Free Approach

Damyn Chipman
Department of Physics and Astronomy, BYU
Bachelor of Science

Computational models that accurately describe the physics in fluid domains are essential in the design process of aircraft. These models are used as tools to quickly design, model, test, and optimize aircraft. However, many current mathematical models are too computationally taxing. Optimization of many of these Eulerian, or mesh-based, methods is infeasible, due to the high wall time for each call to the model. Our approach is to use Lagrangian methods, which have the potential of reduced wall time and increased numerical accuracy. The Vortex Particle Method (VPM) is a Lagrangian formulation of the vorticity form of the Naiver-Stokes (NS) equations put together by Alvarez et al [1]. While the VPM is capable of solving the NS equations in a field, boundary conditions must be imposed separately. The purpose of this thesis is to present an approach to introducing boundary information through the calculation of a vortex sheet on a surface required to induce the no-slip boundary condition. The resulting vorticity is then introduced to the vorticity field through the solution of a diffusion equation. This Vortex Sheet Boundary (VSB) Method is presented and verified, and the results are discussed.

Keywords: Vortex Particle Method, Lagrangian Scheme, CFD, Vortex Sheet, Vortex Methods

ACKNOWLEDGMENTS

I'd like to first and foremost thank my wife for her ever constant support, encouragement, and love. None of this would have been possible without your support Alyssa.

Thank you to my family and friends for your support. You all make me who I am today.

In addition, a huge thank you to Dr. Andrew Ning and Eduardo Alvarez from the BYU Mechanical Engineering FLOW Lab for their funding, support, guidance, motivation, and resources. I have learned from the best on what it means to be a good researcher.

Finally, thank you to the many talented physics professors at BYU and with whom I have enjoyed my undergraduate career. Thank you to Dr. David Neilsen, Dr. Grant Hart, Dr. Nathan Powers, Dr. John Colton, and Mrs. Cheryl Davis. And thank you to Dr. Matthew Memmott and Jack Webster of the BYU Chemical Engineering Department.

Contents

Chapter 1

Introduction

1.1 Computational Modeling: The Motivation

An efficient physicist or engineer these days relies heavily on computers for a large number of tasks. From taking data to modeling complex systems, computers significantly improve the productivity of the user. We can use computers to model complicated systems and predict the performance of a particular design. Computational methods, then, are the key to productivity for an analysis team. The purpose of a design and analysis team is to determine the best parameters that result in an optimal set up. Often, we will combine the use of computer models within optimization routines that aid in determining the optimal parameter space. Optimization routines include gradient based methods such as Gradient Descent or Conjugate Gradient Methods, as well as gradient free methods such as genetic algorithms. In these optimization routines, the computer model for the specific system is often called hundreds to thousands of times. So, efficient and fast models are necessary in the use of optimization routines. If a model takes even a minute to run, a thousand calls would take about 16 hours. And that escalates as models take longer to run and return results.

Within computational fluid dynamics (CFD), there are a plethora of methods used to solve the

governing equations on all scales. On one side, there are very computationally taxing methods such as direct numerical simulation, or DNS. These employ finite difference, finite volume, and finite element methods to solve the Navier-Stokes equations, which govern the forces and stresses in a fluid. DNS is very computationally demanding. However, because no approximations are made, besides numerical discretization, in DNS or no additional methods are coupled with DNS, it can be very accurate. On the other side, there are much simpler models to model fluid dynamics, though they often compromise accuracy with computation. An example of this would be potential flow. In potential flow, a scaler potential field, called the velocity potential, is used to approximate the flow around objects. They follow the mathematical solutions of Laplace's equation. While much simpler and easier to implement than DNS, potential flow completely disregards viscous effects in the fluid, leading to oversimplifications. Thus, there exists a high demand for high fidelity and high efficiency physics based models of fluid dynamics. This is the primary motivation for the Vortex Particle Method (VPM) explored below, and the subsequent issues of the VPM, which is the topic of this thesis.

1.2 Eulerian Versus Lagrangian

In computational modeling, methodologies can be (in general) broken into two distinct formulations: Eulerian and Lagrangian. In Eulerian formulations, the information in question— such as velocity, temperature, pressure, etc.— is represented on a static grid and the governing equations are solved at each point on the grid. These are typically associated with mesh methods, as computational meshes are required to be created. The field in question is often a function of space and time and can be represented as

$$\mathbf{u}(\mathbf{x},t) \tag{1.1}$$

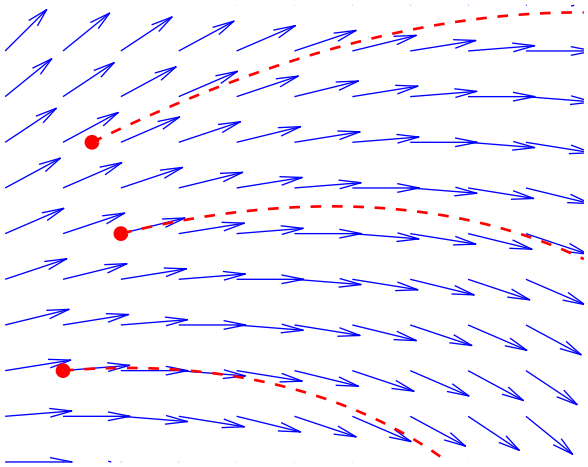

Figure 1.1 Eulerian versus Lagrangian visualization. Eulerian schemes (blue) evaluate a function at each point within a mesh. Lagrangian schemes (red) track the information through particles and update the position and information of the particle.

where $\mathbf{u}$ can represent any scalar or vector field. This can be imagined as observing a body of flowing water and identifying the velocity at every point in the flow.

Alternatively, in Lagrangian formulations, information is represented as computational nodes that move throughout the field. In the water flow example, this would be like placing a leaf in the fluid and observing the motion of the leaf. Each node has an associated value to it, representing the information in question, and a velocity that dictates how the node moves throughout the field. Governing equations are solved as a summation of nodes and their associated values. The field is represented as nodes of the form

$$\mathbf{X}(\mathbf{x}_i, t) \tag{1.2}$$

where $\mathbf{x}_i$ is the location of node i and $\mathbf{X}$ is the associated vector or scalar value of the node.

The two formulations of a field are related through the expression

$$\mathbf{u}(\mathbf{X}(\mathbf{x}_i, t), t) = \frac{\partial \mathbf{X}}{\partial t}(\mathbf{x}_i, t). \tag{1.3}$$

On the left hand side we have the velocity at every point in space (Eulerian), and on the right hand

side we have the change of the position of each node with respect to time (Lagrangian), so both sides come to represent the velocity field through the entire domain. When one wishes to consider differential changes in the field (such as the acceleration of a point in space), due to this relationship and subsequent chain rule, the material derivative is introduced

$$\frac{D\mathbf{F}}{Dt} = \frac{\partial \mathbf{F}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{F}. \tag{1.4}$$

The material derivative gives differential changes in a field **F** that is represented as an Eulerian formulation. The differential change is represented as a sum of the local rate of change (first term) and the convective rate of change in **F** (second term). The material derivative also expresses the transformation from an Eulerian scheme to a Lagrangian scheme.

When applied to computational methods, there are advantages and disadvantages to both formulations. Eulerian formulations can handle complex geometries, as any grid defining the domain can easily be adapted and refined to better approximate the field. In order to do so, numerical complexity increases rapidly. The need to generate a mesh for the domain is computationally taxing and, combined with the need to solve governing equations at each point on the grid, contributes to the major downfall of Eulerian methods. Eulerian formulations are computationally demanding. Lagrangian formulations do not suffer from the need to generate a mesh, thus improving the computation time for a given simulation. In both cases, a very large number of grid points or nodes can be considered and the large number of nodes adds to the computational complexity. However, Lagrangian methods can often be scaled down and contributions from each point can be lumped with other local nodes for reduced complexity [2]. Lagrangian formulations outperform Eulerian when it comes to field calculations. As the evolution of vorticity in trailing wakes (the high concentration of vorticity in the field behind the object) are of particular interest in aerodynamics, a Lagrangian formulation is the best candidate.

In addition, the physics being modeled can also dictate which scheme is more appropriate to use. For example, an Eulerian scheme would be better equipped to solve problems such as

structural analysis and materials modeling. The use of a mesh that can be reformed and refined makes modeling structures simpler than in mesh-less methods (as in finite element analysis). In the case we explore here (aircraft immersed in fluid), a Lagrangian scheme is better fit to model the complex dynamics of the vorticity. Vorticity is a conserved quantity. Lagrangian schemes, due to the particle nature of the field, better conserve the vorticity. Eulerian schemes suffer from more numerical error, which results in a loss of vorticity. Thus, a Lagrangian scheme becomes better fit for modeling the governing equations [3].

1.3 The Vortex Particle Method

The Vortex Particle Method (VPM) is a method used to solve the vorticity formulation of the Navier-Stokes equations. Known for their high complexity, the Navier-Stokes equations represent how fluid reacts to forces within and around the fluid. In the VPM, the vorticity form of the Navier-Stokes equations are used. Vorticity is defined as the curl of the velocity

$$\omega = \nabla \times \mathbf{u} \tag{1.5}$$

and represents rotation within the velocity field. Upon taking the curl of the velocity form the Navier-Stokes equation, and after some development, we arrive at the vorticity form

$$\frac{D\omega}{Dt} = (\omega \cdot \nabla)\mathbf{u} + \nu\nabla^2\omega. \tag{1.6}$$

The presence of the material derivative primes this method to use a Lagrangian scheme. The vorticity is represented as a collection of particles that advect following the governing equations. Each particle is represented with a position $\mathbf{x}_p$, the velocity $\mathbf{u}_p$, and the vectorial circulation $\Gamma_p \equiv \int_{vol_p} \omega_p dV \approx \omega_p V_p$ (where ω_p is the vorticity associated with particle p). Each particle then comes to represent the volume of fluid that travels with the local vorticity. These particles are governed by the following governing equations that describe each particles motion and viscous diffusion:

$$\frac{d}{dt}\mathbf{x}_p(t) = \mathbf{u}(\mathbf{x}_p(t),t) \tag{1.7}$$

$$\frac{d}{dt}\Gamma_p(t) = (\Gamma_p(t)\cdot\nabla)\mathbf{u}(\mathbf{x}_p(t),t) + \frac{d}{dt}\Gamma_p(t)\Big|_{visc}. \tag{1.8}$$

Through the solution of these equations for each particle in the field, the vorticity field is approximated as the summation of each individual particle's vorticity. The curious reader can further study the VPM from Eduardo Alvarez of the BYU Mechanical Engineering Departmenet in [1], or other vortex method textbooks in [4].

Several attempts have been made at coupling boundary information into vortex methods. Much of this work is based off of the efforts by Cooper et al. [5]. In their conference paper, they present a method of coupling a vortex method similar to the VPM with a vortex sheet solver. They also use radial basis function approximations to solve the governing equations. However, they lack the mathematical derivation of the governing equations, leaving the reader confused and the results difficult to reproduce. The physical intuition and a more in-depth mathematical explanation (though still heavily lacking for beginning readers) can be found in [4]. Much of the physical intuition regarding the use of a vortex sheet for boundary methods is found therein. Koumoutsakos et al. [6] also offer improved boundary integral methods which form as some of the foundation for the methods found in [4].

1.4 Radial Basis Function Interpolations

Radial Basis Function (RBF) Interpolations are a method of interpolating sets of data through a summation of RBF functions. The primary property of these functions is local influence; they have a high relative value at an origin (or shifted origin) and decay to zero far from the origin. They have an infinite domain. Other useful properties include their smoothness and their ease in use with linear operators such as derivatives and integrals. There are several different RBF that can be

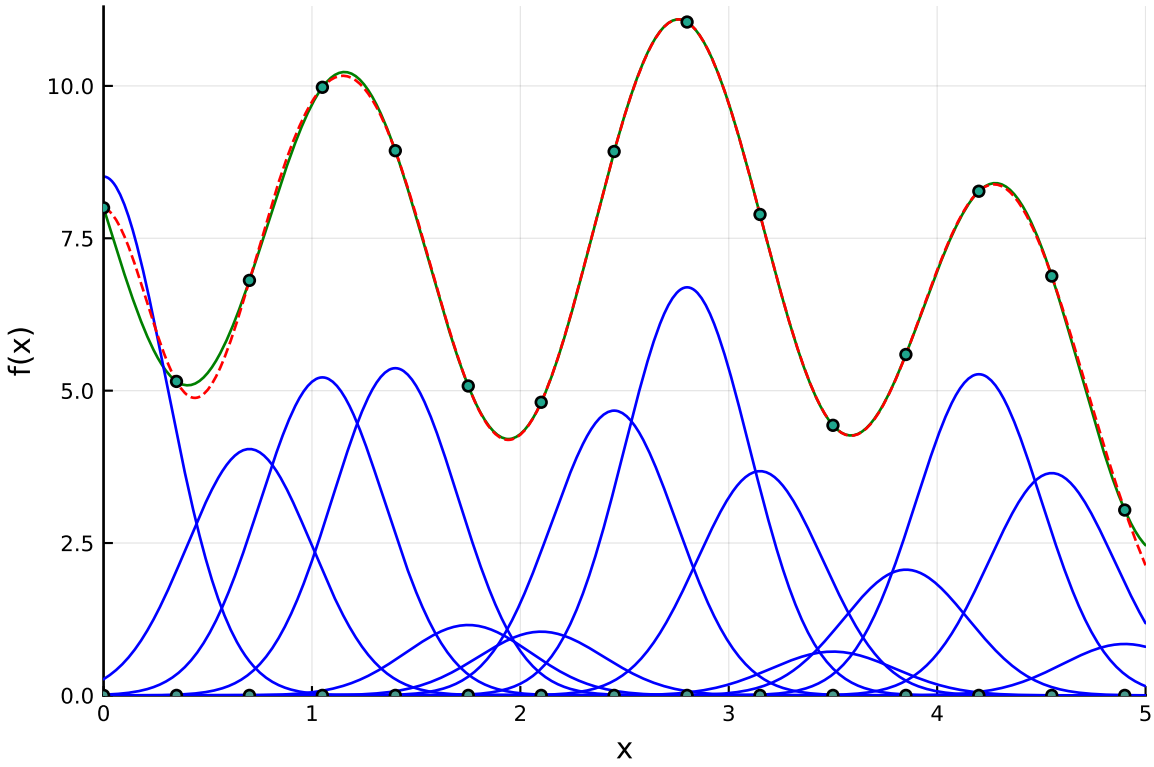

Figure 1.2 Radial Basis Functions (RBF) Interpolations. A function (green) can be interpolated by a summation of RBF (blue) centered around collocation points (green points). After solving for the correct scaling coefficients through the solution of a linear system formed by the RBFs and the function evaluated at the collocation points (green points on green function), the combined RBFs add together to create the interpolate (dashed red).

employed, such as the Gaussian, Multi-quadric, and a Polymorphic Spline [7]. In this presentation, the Gaussian is used and is defined as

$$\phi(x) = \exp\left(\frac{-(x-x_0)^2}{2\sigma^2}\right), \tag{1.9}$$

or, in multiple dimensions,

$$\phi(\mathbf{x}) = \exp\left(\frac{-|\mathbf{x}-\mathbf{x}_0|^2}{2\sigma^2}\right), \tag{1.10}$$

where σ represents the width of the Gaussian. This is a Gaussian function centered about x_0 or $\mathbf{x}_0$ depending on the order of dimension.

Using RBF to interpolate functions is simple. It involves collocating several functions at various points within the domain to set up a linear system for the appropriate scaling coefficients. Let's say we wish to approximate some function $f(x)$. Referring to Fig. 1.2, the solid green function is what we wish to approximate. We do this with a summation of RBFs and a scaling coefficient α_i, according to the following:

$$f(x) \approx \sum_{i=1}^{N} \alpha_i \phi(x - x_i). \tag{1.11}$$

If we can determine the correct α coefficients for each Gaussian at the correct location, we have an approximation to f. We create a set of points in the domain

$$\mathbf{x}_j = \{x_j \big| j = 1, ..., N\} \tag{1.12}$$

and evaluate them at $f(x)$. In Fig. 1.2, these are the green points on the x-axis and the corresponding evaluated points on f. Evaluating f at each point results in a linear system

$$f(x_j) = \sum_{i=1}^{N} \alpha_i \phi(x_j - x_i), \tag{1.13}$$

or in matrix form

$$[f_j] = [\alpha_i][\phi_{ji}] \tag{1.14}$$

where

$$[f_j] = f(x_j) = [f_1 \quad f_2 \quad ... \quad f_N]^T \tag{1.15}$$

$$[\alpha_i] = [\alpha_1 \quad \alpha_2 \quad ... \quad \alpha_N]^T \tag{1.16}$$

$$[\phi_{ji}] = \phi(x_j - x_i) = \begin{bmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1N} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{N1} & \phi_{12} & \cdots & \phi_{NN} \end{bmatrix}. \tag{1.17}$$

Solving this linear system solves for the correct α coefficients that correctly scale each RBF at each point. Summing each of these together results in the correct approximation to f. In Fig. 1.2, the approximation is given by the dashed red line.

RBF really shine in their ability to solve difficult equations such as partial differential equations (PDE) and integral equations (IE). An unknown function is represented as a summation of RBF multiplied by a series of coefficients. Upon substitution of the RBF into the PDE or IE, boundary conditions or other known nodal location information is used to set up a linear system of equations for which the RBF coefficients can be solved for. Again, linear operators such as derivatives and integrals can commute directly to the RBF, leaving the coefficients uniform across all terms

$$f(x) = \sum_i^N \alpha_i \phi_i(|\mathbf{x} - \mathbf{x}_i|) \tag{1.18}$$

$$\nabla f(x) = \sum_i^N \alpha_i \nabla \phi_i(|\mathbf{x} - \mathbf{x}_i|) \tag{1.19}$$

$$\nabla^2 f(x) = \sum_i^N \alpha_i \nabla^2 \phi_i(|\mathbf{x} - \mathbf{x}_i|) \tag{1.20}$$

$$\int f(x)dx = \sum_i^N \alpha_i \int \phi_i(|\mathbf{x} - \mathbf{x}_i|)dx. \tag{1.21}$$

The linear system is set up by collocating the known information around $\mathbf{x}$ and $\mathbf{x}_i$. The application of RBF into IE will be used here to solve for unknown functions and for interpolating fields such as the vorticity field.

1.5 Overview

As information regarding any solid boundaries is lost in the formulation of the VPM, it must be included by coupling additional methods. In this thesis, we explore a Lagrangian formulation that fits well with the Lagrangian VPM. A vortex sheet is introduced at each boundary, and the goal is to solve for the strength of the vortex sheet that cancels the slip velocity on the boundary. From the vortex sheet strength, we calculate the vorticity and diffuse that vorticity into the field, where it becomes particles that follow the VPM. The vortex sheet is defined in chapter 2, and the Vortex Sheet Boundary (VSB) Method is explained in detail in chapter 2. Results, verifications, and

animations are provided from the VSB method in chapter 3.

Chapter 2

Methodology

2.1 Introduction

The methodology of the Vortex Sheet Boundary (VSB) Method is presented in this chapter. It is intended to extend and couple directly with the Vortex Particle Method (VPM) [1]. We start with the mathematical derivation of the method. Following the general formulation, we look directly at the 2D implementation and discretization.

2.2 Physical Motivation

The physics for the VSB method is presented first. The goal is to determine a way to incorporate boundary information to a field of particles. In essence, how does one tell the vortex particles that there is a boundary it needs to go around? To answer this, analysis is performed in the reference frame of the object that is moving. The fluid moves relative to the object (i.e. an aircraft or other body). Any object within a flow field will displace the fluid around it. The fluid must flow around the solid boundary of the object (Fig. 2.1A). Looking at the surface of the object, the fluid relative to the object cannot be moving on the surface. The fluid velocity on the surface of the object must

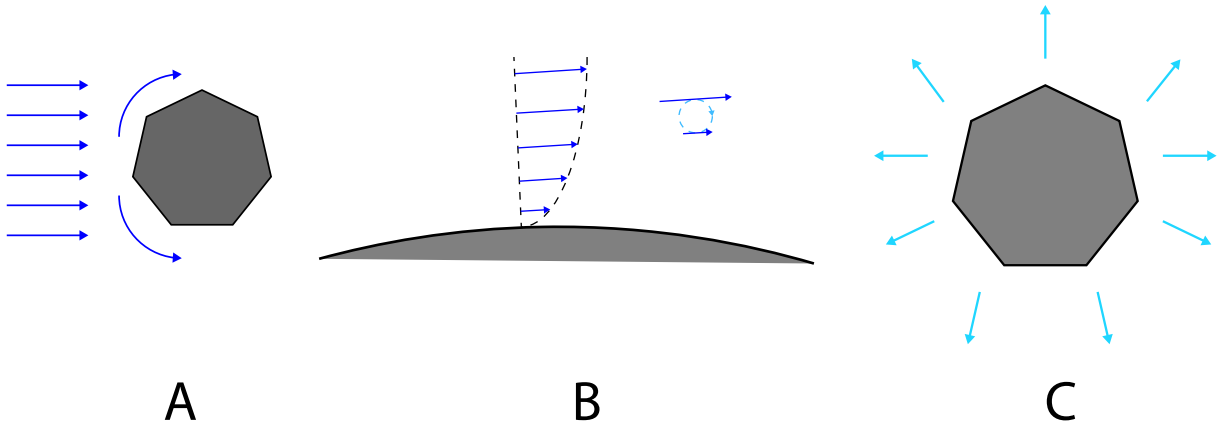

Figure 2.1 Physics of the Vortex Sheet Boundary Method. An object displaces fluid around it (A). As the fluid flows around the object, the no-slip boundary condition dictates that the fluid velocity near the object is zero and is larger further from the object. This results in a shear force on the fluid (B). The shear force adds circulation to the fluid field. The object then acts as a source of circulation (C).

be equal to the velocity of the surface (in this reference frame, it is zero on the surface). This is called the no-slip boundary condition. Further from the surface, the fluid moves faster relative to the surface. Thus, there is a velocity gradient between the surface of the object and a point far from the surface (this is called the free-stream). This velocity gradient is referred to as the boundary layer (Fig. 2.1B). Because there is high (relative) velocity far from the surface, and a low velocity near the surface, a shear force is created by the presence of the boundary. This shear force adds vorticity to the flow field (Fig. 2.1C). So, if the presence of the boundary adds vorticity to the field, the boundary itself acts *as a source of vorticity*.

Now, if one can determine just the right amount of vorticity to add to the particle field such that any boundary conditions are satisfied, then the particles will flow around the surface. To do so, a mathematical formulation called a vortex sheet can be collocated over any boundary or surface one wishes to consider in a vortex method. The vortex sheet (as will be explained in more detail later on) is an infinitesimally small distribution of vorticity over a given surface. It represents a discontinuity in the tangential component of velocity, and it has a characteristic strength, given by γ.

Thus, the correct strength of a vortex sheet (as a very small distribution of vorticity) can impose a boundary condition in the vorticity field. Once this vortex sheet strength is found, the vorticity associated with it must be introduced into the vorticity field. As a vortex sheet is the same as a vorticity flux, and the particles diffuse from the surface, this is done by solving a diffusion equation with the vortex sheet strength contributing to the boundary condition. Much of this intuition can also be found in [4] and [5].

These ideas are implemented below through a mathematical derivation of the equations involving the vortex sheet strength and the needed boundary conditions.

2.3 Mathematical Derivation

Much of the results this work have been presented before, for example in [4] and [5], but they lack the mathematical derivation required to 1) implement the methods numerically (i.e. code the solution) and 2) to understand the results without taking them for granted. A part of the contribution of this work is to present the derivation of the results presented. The other contribution is to present a numerical scheme for solving the resulting equations to be used as part of a computer simulation.

2.3.1 The Vortex Sheet

The first step is to consider the vortex sheet, how it relates to the vorticity field, and how it contributes to the boundary conditions. To account for surface interactions, we consider the effect of a solid surface immersed in an oncoming flow. The surface causes the fluid to flow around the body, introducing vorticity into the flow. This additional vorticity can be represented as coming from some source of vorticity, namely a vortex sheet. Upon representing the surface of the body as a vortex sheet, the strength required by the vortex sheet to induce the no-through and no-slip boundary conditions on the surface must be determine. By imposing these boundary conditions, the vortex

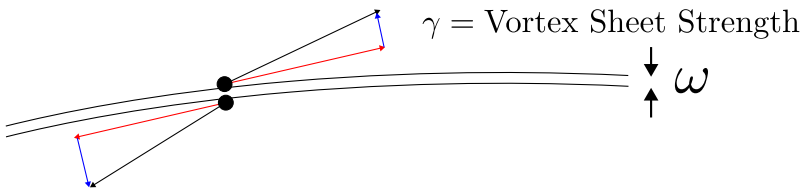

Figure 2.2 Vortex Sheet. The vortex sheet γ is represented as an infinitesimally small distribution of vorticity ω. It represents a discontinuity in the tangential component of velocity (shown here in red).

sheet strength can be found.

As graphically depicted in Fig. 2.2, a vortex sheet represents a discontinuity of fluid velocity across a boundary. The tangential component of velocity is discontinuous, while the normal component is continuous across the boundary. This discontinuity implies an infinite vorticity across the boundary, given as a vortex sheet. The vortex sheet has a characteristic strength given as γ along the length of the sheet. A vortex sheet can be placed on the surface of any boundary in a fluid field and, with the appropriate vortex sheet strength γ, can behave as a solid boundary. This is, of course, just a mathematical formulation. The vortex sheet is not actually physical and not actually placed on a boundary, but assists in the mathematical formulation of the required vortex sheet strength to induce the no-slip and no-through boundary conditions.

A formal definition of the vortex sheet can be found in [4]. The velocity difference across the interface (vortex sheet) is viewed as an infinitesimally thin vorticity distribution along the surface of the body and is defined as

$$\gamma(\mathbf{x}) = \hat{\mathbf{n}}(\mathbf{x}) \times [\mathbf{u}(\mathbf{x}_i) - \mathbf{u}(\mathbf{x}_e)] = \hat{\mathbf{n}} \times [\mathbf{u}_i - \mathbf{u}_e] \tag{2.1}$$

where i and e refer to the internal and external sides of the vortex sheet (which is inside and outside the domain). Interpreting γ again as a very small distribution of vorticity, continuously normal to $\hat{\mathbf{n}}$, we can get through Kelvin's Circulation Theorem [8] for a loop around the surfaces of the vortex sheet another relation between the vortex sheet and the velocity *on the surface* of the vortex sheet:

$$\gamma \times \hat{\mathbf{n}} = \mathbf{u}_i - \mathbf{u}_e. \tag{2.2}$$

This result only applies to the vorticity field on the surface of the boundary, i.e. on the vortex sheet. The domain is divided into the internal and external domains, where the vorticity in-between the two domains is given by eq. (2.2).

2.3.2 Velocity Fields

The goal is to determine the strength of a vortex sheet required to induced the no-slip boundary condition on the surface. To do so, the various contributions for the velocity vector field must be considered.

Start with a vector field $\mathbf{u}$, which represents the velocity through domain Ω and boundary $\partial\Omega$ (a solid surface representing an airfoil, sphere, cylinder, etc.). Through the Helmholtz Decomposition Theorem, the vector field is decomposed into two components, a rotational component $\mathbf{u}_\omega$ and a solenoidal component $\mathbf{u}_\phi$

$$\mathbf{u} = \mathbf{u}_\omega + \mathbf{u}_\phi.$$

These components are defined as

$$\mathbf{u}_\omega = \text{Rotational Component} \rightarrow \omega = \nabla \times \mathbf{u}_\omega$$

$$\mathbf{u}_\phi = \text{Solenoidal Component} = \nabla\phi$$

where ω is the vorticity field. The incompressibility of the velocity field gives

$$\nabla \cdot \mathbf{u} = 0.$$

Upon combining the above, taking the curl of the vorticity, and through the use of a vector identity $\nabla \times \omega = \nabla \times (\nabla \times \mathbf{u}) = \nabla(\nabla \cdot \mathbf{u}) - \nabla^2 \mathbf{u} = -\nabla^2 \mathbf{u}$, Laplace's and Poisson's equation arise,

$$\nabla^2 \phi = \nabla \cdot \mathbf{u} = \nabla \cdot (\nabla \phi) = 0$$

$$\nabla^2 \mathbf{u} = -\nabla \times \omega,$$

which are the principle equations used to solve for the boundaries given. One will notice that the velocity potential equation used often in panel methods is obtained as well [8].

The solution to a vector Poisson's equation is obtained through the use of a Green's function. The definition of a Green's function is any function that is the solution to the following:

$$\mathscr{L} G(\mathbf{x}, \mathbf{x'}) = \delta(\mathbf{x} - \mathbf{x'})$$

where $\mathscr{L}$ can be any linear operator. Allow $\mathscr{L}$ to be the Laplacian and solve the following:

$$\nabla^2 \mathbf{A} = \mathbf{B}.$$

By multiplying (2.3.2) by $\mathbf{B}$ and integrating over V we get

$$\int_V \left[\nabla^2 G(\mathbf{x}, \mathbf{x'})\right] \mathbf{B} dV = \int_V \left[\delta^3(\mathbf{x} - \mathbf{x'})\right] \mathbf{B} dV$$

$$\int_V \nabla^2 G(\mathbf{x}, \mathbf{x'}) \mathbf{B}(\mathbf{x'}) dV = \mathbf{B}(\mathbf{x})$$

$$\int_V \nabla^2 G(\mathbf{x}, \mathbf{x'}) \mathbf{B}(\mathbf{x'}) dV = \nabla^2 \mathbf{A}$$

$$\int_V G(\mathbf{x}, \mathbf{x'}) \mathbf{B}(\mathbf{x'}) dV = \mathbf{A}$$

where δ^3 is a 3D Dirac Delta function. Replacing $\mathbf{A}$ with $\mathbf{u}$ and $\mathbf{B}$ with $-\nabla \times \omega$ gives us the general relationship between velocity and vorticity

$$\mathbf{u} = \int_V G(\mathbf{x}, \mathbf{x'})(-\nabla \times \omega(\mathbf{x'})) dV \tag{2.3}$$

$$= \int_V \nabla \times G(\mathbf{x}, \mathbf{x'}) \omega(\mathbf{x'}) dV \tag{2.4}$$

$$= \nabla \times \int_V G(\mathbf{x}, \mathbf{x'}) \omega(\mathbf{x'}) dV \tag{2.5}$$

where G for both 2D and 3D cases is defined as

$$G(\mathbf{x},\mathbf{x'}) = \begin{cases} \dfrac{1}{4\pi}\dfrac{1}{|\mathbf{x}-\mathbf{x'}|} & \text{3D} \\ -\dfrac{1}{2\pi}\log|\mathbf{x}-\mathbf{x'}| & \text{2D,} \end{cases} \tag{2.6}$$

which is common for Laplace equation Green's functions representing 2D and 3D domains.

Concerning the strength of the vortex sheet, because the vortex sheet represents a vorticity flux, there is a similar relationship to (2.5) showing the velocity induced by the vortex sheet

$$\mathbf{u}_\gamma = \nabla \times \int_S G(\mathbf{x},\mathbf{x'})\gamma(\mathbf{x'})dS. \tag{2.7}$$

This expression will be used later in combination with other velocity field components to arrive at an expression for the unknown vortex sheet strength.

The total velocity field is now separated into its several contributions

$$\mathbf{u} = \mathbf{u}_\omega + \mathbf{u}_\gamma + \mathbf{u}_\infty = \mathbf{u}_{ext} + \mathbf{u}_\gamma \tag{2.8}$$

where

$$\mathbf{u}_\omega = \text{Velocity due to vortex particles}$$

$$\mathbf{u}_\gamma = \text{Velocity induced from vortex sheet}$$

$$\mathbf{u}_\infty = \text{Free-stream velocity}$$

$$\mathbf{u}_{ext} = \mathbf{u}_\omega + \mathbf{u}_\infty.$$

The relationships between velocity and vorticity (which includes the vortex sheet) are given by (2.5) and (2.7):

$$\mathbf{u}_\omega = \nabla \times \int_{V_e \bigcup V_i} \omega(\mathbf{x'})G(\mathbf{x},\mathbf{x'})dV(\mathbf{x'})$$

$$\mathbf{u}_\gamma = \nabla \times \int_S \gamma(\mathbf{x'})G(\mathbf{x},\mathbf{x'})dS(\mathbf{x'})$$

In this case, any velocity induced by the vorticity field will be computed using the Vortex Particle Method [1], which calculate the velocity in a different manner.

The first logical approach is to simply take the expressions for the velocity contributions and plug them into the appropriate boundary conditions. As will be shown, this results in an expression that is ill-fit for discretization and computation. The no-through boundary condition is given by

$$\mathbf{u} \cdot \hat{\mathbf{n}} - \mathbf{u}_b \cdot \hat{\mathbf{n}} = 0 \tag{2.9}$$

where $\mathbf{u}_b$ represents any body velocity including rotation. In this study, it *only* represents rotation as any translational velocity will be part of the free-stream velocity. And in all cases presented, there is no rotation, so $\mathbf{u}_b = 0$. By substituting eq. (2.8) into eq. (2.9) and substituting each of the expressions between velocity and vorticity (eq. (2.5) and eq. (2.7)) into what results ultimately gives the following expression

$$\int_S \gamma(\mathbf{x'}) \times \nabla G(\mathbf{x}, \mathbf{x'}) \cdot \hat{\mathbf{n}} \, dS(\mathbf{x'}) = (\mathbf{u}_{ext} - \mathbf{u}_b) \cdot \hat{\mathbf{n}}. \tag{2.10}$$

The strength of the vortex sheet γ is unknown. Everything on the right hand side of the equation is known, either through the VPM or the given operating conditions. This result is a Fredholm Integral equation of the first kind. It is characterized by the unknown expression inside the integrand equal to some given information on the right hand side. According to Fredholm Theory, such equations are often ill-formed [9]. Discretization of this expression results in non-invertible matrices, which make computation impossible. So instead, the goal is to now derive an expression for the vortex sheet strength in the form of a Fredholm Integral equation of the second kind (done in the following section), which will be more fitted for numerical discretization and computation.

2.3.3 Fredholm Integral Equations and the Poincare Identity

To arrive at the desired tangential representation of the boundary conditions, we start with eq. 2.8 and use a vector Poincare Identity [10]. The Poincare Identity extends Helmholtz's Decomposition

Theorem to define the solenoidal and rotational components in the exterior domain and on the surface of the domain. We define domain $\Omega \subset \Re^3$ with smooth boundary $\partial\Omega = S(\mathbf{x})$. The vector field (velocity) is defined in the exterior domain $\Omega' = \Re^3 - \partial\Omega$. The solenoidal and rotational components of vector field $\mathbf{u}$ are defined as

$$\mathbf{u}(\mathbf{x}) = \nabla\phi(\mathbf{x}) + \nabla \times \mathbf{A}(\mathbf{x}) \tag{2.11}$$

$$\phi(\mathbf{x}) = -\int_D g(\mathbf{x},\mathbf{x'})Q(\mathbf{x'})d\mathbf{x'} - V(\hat{\mathbf{n}}\cdot\mathbf{u})(\mathbf{x}) \tag{2.12}$$

$$\mathbf{A}(\mathbf{x}) = \int_D G(\mathbf{x},\mathbf{x'})\xi(\mathbf{x'})d\mathbf{x'} + V(\hat{\mathbf{n}}\times\mathbf{u})(\mathbf{x}) \tag{2.13}$$

where

$$\xi(\mathbf{x}) = \nabla \times \mathbf{u} = \omega$$

$$Q(\mathbf{x}) = \nabla \cdot \mathbf{u}$$

$$V(f)(\mathbf{x}) = \int_S G(\mathbf{x},\mathbf{x'})f(\mathbf{x'})dS(\mathbf{x'}) \quad for\ \mathbf{x} \in \partial\Omega$$

and where g is a Green's function for $\nabla\phi$ (there is no need to derive it as it drops out, as shown below), and where G is given by eq. (2.6).

Plugging these definitions into $\mathbf{u}$ yields

$$\begin{aligned}\mathbf{u}(\mathbf{x}) =& \nabla\Big[-\int_D g(\mathbf{x},\mathbf{x'})\ \nabla\cdot\mathbf{u}(\mathbf{x'})d\mathbf{x'} - \int_S G(\mathbf{x},\mathbf{x'})(\hat{\mathbf{n}}\cdot\mathbf{u})(\mathbf{x'})dS(\mathbf{x'})\Big] + \\ & \nabla\times\Big[\int_D G(\mathbf{x},\mathbf{x'})\omega(\mathbf{x'})d\mathbf{x'} + \int_S G(\mathbf{x},\mathbf{x'})\ (\hat{\mathbf{n}}\times\mathbf{u})(\mathbf{x'})dS(\mathbf{x'})\Big].\end{aligned}$$

The first term disappears due to the incompressibility of $\mathbf{u}$ and the third term is the relationship between velocity and vorticity, thus it is the same as $\mathbf{u}_\omega$. We add a free-stream component and combine the free-stream and vorticity component of velocity to get the external velocity we defined earlier, $\mathbf{u}_{ext}$. We arrive at

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}_{ext} + \nabla\int_S G(\mathbf{x},\mathbf{x'})(\hat{\mathbf{n}}\cdot\mathbf{u})(\mathbf{x'})dS(\mathbf{x'}) + \nabla\times\int_S G(\mathbf{x},\mathbf{x'})\ (\hat{\mathbf{n}}\times\mathbf{u})(\mathbf{x'})dS(\mathbf{x'}),$$

and if we allow this to limit to the velocity near the surface of the body (i.e. on the vortex sheet)

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}_{ext} + \nabla \int_S G(\mathbf{x},\mathbf{x'})(\hat{\mathbf{n}} \cdot (\mathbf{u}_e - \mathbf{u}_i))(\mathbf{x'})dS(\mathbf{x'}) + \nabla \times \int_S G(\mathbf{x},\mathbf{x'})\ (\hat{\mathbf{n}} \times (\mathbf{u}_e - \mathbf{u}_i))(\mathbf{x'})dS(\mathbf{x'}).$$

Because the normal component of velocity is continuous across the vortex sheet, or $\mathbf{u}_e \cdot \hat{\mathbf{n}} = \mathbf{u}_i \cdot \hat{\mathbf{n}}$, we get

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}_{ext} + \nabla \times \int_S (\hat{\mathbf{n}} \times (\mathbf{u}_e - \mathbf{u}_i))(\mathbf{x'})G(\mathbf{x},\mathbf{x'})dS(\mathbf{x'})$$

which, through the vector identity $\nabla \times (f\mathbf{A}) = f(\nabla \times \mathbf{A}) + \nabla f \times \mathbf{A}$ for f being a scalar function, and the limiting case of points on the surface (approaching the vortex sheet from both interior and exterior domains) [4] produces

$$\mathbf{u}_\gamma(\mathbf{x}) = \pm\frac{\gamma \times \hat{\mathbf{n}}}{2} + \int_S \gamma(\mathbf{x'}) \times \nabla G(\mathbf{x},\mathbf{x'})dS(\mathbf{x'}).$$

Finally, applying this to the no-slip condition $\mathbf{u} \cdot \hat{\mathbf{t}} - \mathbf{u}_b \cdot \hat{\mathbf{t}} = 0$ yields the final boundary integral equation we attempt to solve

$$\mathbf{u} \cdot \hat{\mathbf{t}} - \mathbf{u}_b \cdot \hat{\mathbf{t}} = 0 \tag{2.14}$$

$$(\mathbf{u}_\gamma + \mathbf{u}_{ext}) \cdot \hat{\mathbf{t}} - \mathbf{u}_b \cdot \hat{\mathbf{t}} = 0 \tag{2.15}$$

$$\pm\frac{\gamma \times \hat{\mathbf{n}}}{2} \cdot \hat{\mathbf{t}} + \int_S \gamma(\mathbf{x'}) \times \nabla G(\mathbf{x},\mathbf{x'}) \cdot \hat{\mathbf{t}} dS(\mathbf{x'}) = (\mathbf{u}_{ext} - \mathbf{u}_b) \cdot \hat{\mathbf{t}}. \tag{2.16}$$

This result is the general boundary condition we wish to solve. The strength of the vortex sheet is the unknown quantity, so this equation is a non-homogenous Fredholm Integral Equation of the second kind, which was the purpose of this alternate derivation. This type of equation has better formed solutions. We now generalize eq. (2.16) into a form where we determine the general solution through Fredholm Theory.

2.3.4 Fredholm Theory

Let

$$\gamma(\mathbf{x}) = \pm\frac{\gamma \times \hat{\mathbf{n}}}{2} \cdot \hat{\mathbf{t}} \tag{2.17}$$

$$Q(\mathbf{x}, \mathbf{x'}) = \nabla G(\mathbf{x}, \mathbf{x'}) \cdot \hat{\mathbf{t}} \tag{2.18}$$

$$h(\mathbf{x}) = (\mathbf{u}_{ext} - \mathbf{u}_b) \cdot \hat{\mathbf{t}}. \tag{2.19}$$

The function γ will be an unknown function of a single variable. The function Q will be a function of two variables, $\mathbf{x}$ (a spatial position vector) and $\mathbf{x'}$ (a dummy variable that will integrate away). This will be referred to as the kernel function. Fredholm Theory will apply for both scalar and vector valued functions, so the cross product is properly taken into account. And h is a known function of a single variable that represents any operating or known conditions. This is the non-homogenous part of the equation.

The general non-homogenous Fredholm Integral Equation of the second kind has the following form:

$$\gamma(\mathbf{x}) - \int Q(\mathbf{x}, \mathbf{x'})\gamma(\mathbf{x'})d\mathbf{x'} = h(\mathbf{x}). \tag{2.20}$$

This can easily be adapted into 2D or 3D domains depending on the form of Q and h. We go through Fredholm Theory for the solution in a general sense for robustness. In order to produce equations that are accurate, have solutions, and are able to be discretized in both 2D and 3D, we need to do the following: 1) Breakdown the kernel and eigenfunctions into forms that can be solved and 2) Add an additional equation because (2.20) has a non-unique solution according to Fredholm Theory. We will use Kelvin's Theorem as this additional equation

$$\Gamma(\mathbf{x}) = -\int_S \gamma(\mathbf{x})dS. \tag{2.21}$$

where Γ is the circulation vector [8].

The theory used here is part of what is called spectral theory, which attempts to generalize eigenproblems of matrices to more general linear operators (such as integral and differential operators). For example, the typical eigenvalue problem presented in matrix for is as follows:

$$A\mathbf{v} = \lambda B\mathbf{v}$$

which is the generalized eigenproblem. Here, A and B are square matrices, $\mathbf{v}$ is a vector (and the eigenvector) and λ is an eigenvalue. B is also often the identity matrix, which then reduces the above to another typical eigenproblem:

$$A\mathbf{v} = \lambda \mathbf{v}$$

which is solved by moving everything to one side of the equation and factoring out $\mathbf{v}$ to produce

$$(A - \lambda I)\mathbf{v} = 0$$

The above has a non-zero solution if and only if the determinant of the matrix $A - \lambda I$ is zero, which allows the solution of the eigenvalues through the use of the characteristic equation.

This can be generalized to other linear operators such as integrals and derivatives. For example, consider the differential equation:

$$\frac{d}{dt} f(t) = \lambda f(t)$$

This can also represent an eigenproblem where the linear operator is d/dt and can be solved through integration. The solution is an eigenfunction (which itself is a function of the eigenvalue):

$$f(t) = f(0)e^{\lambda t}$$

Again, in spectral theory, the idea of eigenproblems is extended to linear operators which act as linear transformation of a vector $\mathbf{v}$ from a vector space V to itself times a scalar multiple of itself (i.e. times the eigenvalue). For an indepth study, see ref. [9]. Of particular use in this presentation is the parallel between spectral decomposition of a matrix and a kernel. A square matrix can be factorized

$$A = W\Lambda W^{-1}$$

where W is a square matrix with columns as the column eigenvectors of A and Λ is a diagonal matrix whose entries are the eigenvalues of A. Similarily, for a kernel Q, we can decompose it according to

$$Q(\mathbf{x},\mathbf{x'}) = \sum_{i=1}^{\infty} \lambda_i \rho_i(\mathbf{x}) \xi_i(\mathbf{x'}),$$

where λ is an eigenvalue, ρ is the solution to the eigenproblem

$$\int_S Q(\mathbf{x},\mathbf{x'}) \rho(\mathbf{x'}) d\mathbf{x'} = \lambda \rho(\mathbf{x}), \tag{2.22}$$

and ξ is the solution to the transposed eigenproblem

$$\int_S Q(\mathbf{x'},\mathbf{x}) \xi(\mathbf{x'}) d\mathbf{x'} = \lambda \xi(\mathbf{x}).$$

There is a direct parallel in the spectral decomposition of Q here and the eigendecomposition of a matrix.

From here, we can normalize the eigenfunctions ρ and ξ for the decomposed kernel as:

$$\oint \rho_i(\mathbf{x}) \xi_i(\mathbf{x}) dS(\mathbf{x}) = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{otherwise} \end{cases}$$

For the given kernel Q, for $i = 1$, we get that $\xi_1 = const. = 1/L$ (2D), $\xi_1 = const. = 1/SA$ (3D) and $\lambda = 1$. L is the perimeter of the body and SA is the surface area of the body. ρ needs to be solved explicitly for the shape of the body from the solution to the eigenproblem (2.22).

The solution to (2.20) is non-unique, as it allows for any additional solution if

$$\gamma(\mathbf{x}) = \gamma_0(\mathbf{x}) + \alpha \rho(\mathbf{x})$$

for an arbitrary α and ρ is the solution to the eigenvalue problem. So we supply an additional constraint through Kelvin's Theorem

$$\oint \gamma(\mathbf{x}) dS(\mathbf{x}) = -\Gamma_{wake}$$

We multiply this by the first term in the decomposition of the kernel and get a set of equations to solve in 2D

$$\frac{\rho_1(\mathbf{x})}{L}\oint \gamma(\mathbf{x})dS(\mathbf{x}) = -\frac{\rho_1(\mathbf{x})}{L}\Gamma_{wake} \tag{2.23}$$

$$\gamma(\mathbf{x}) - \oint Q(\mathbf{x},\mathbf{x'})\gamma(\mathbf{x'})dS(\mathbf{x'}) = h(\mathbf{x}). \tag{2.24}$$

Upon subtracting (2.23) and (2.24), we get out final boundary integral equation to solve:

$$\gamma(\mathbf{x}) - \oint \left[Q(\mathbf{x},\mathbf{x'}) - \frac{\rho_1(\mathbf{x})}{L}\right]\gamma(\mathbf{x'})dS(\mathbf{x'}) = h(\mathbf{x}) - \frac{\rho_1(\mathbf{x})}{L}\Gamma_{wake} \tag{2.25}$$

Upon close inspection, one will notice that this is simply another Fredholm Integral Equation of the second kind, where the first term of the decomposition of the kernel has been removed. This eliminates the singularity associated with the original Fredholm Integral equation of the first kind and results in a well posed equation to solve. This equation represents the end of the general derivation of the vortex sheet strength required to induce a no-slip boundary condition on the surface of the body. From here, appropriate kernels can be replaced in both 2D and 3D domains, ρ can be solved for numerically from the eigenproblem (2.22), and γ can be solved for through any method deemed appropriate (an RBF approximation is used in this case).

2.3.5 Vortex Diffusion

Once the strength of the vortex sheet is known, the associated vorticity must be introduced into the vorticity field. The vortex sheet represents spurious vorticity generated to cancel the no-slip boundary condition. For conservation sake (as vorticity is a conserved quantity), "it's strength should account for the modification of the circulation of the flow field" [4]. Each piece of the vortex sheet contributes a small amount of circulation to the flow field over time. By considering the contribution of the entire vortex sheet, a relationship between the vortex sheet and circulation can be formulated as

$$\oint \gamma(S)dS = \int_t^{t+\delta t} \frac{d\Gamma}{dt'}dt'. \tag{2.26}$$

Kelvin's Theorem relates changing circulation to the vorticity field in a form of a vorticity flux:

$$\frac{d\Gamma}{dt} = \nu \oint \frac{\partial \omega}{\partial n}(S) dS. \tag{2.27}$$

Upon substitution of 2.27 into the RHS of 2.26

$$\int_t^{t+\delta t} \frac{d\Gamma}{dt'} dt' = \int_t^{t+\delta t} dt' \oint \nu \frac{\partial \omega}{\partial n}(S) dS \tag{2.28}$$

So the vortex sheet strength and the vorticity flux are related by

$$\begin{aligned} \gamma(S) &= \int_t^{t+\delta t} \nu \frac{\partial \omega}{\partial n}(S) dt' \\ &= \nu \frac{\partial \omega}{\partial n}(S)[t + \delta t - t] \end{aligned}$$

Simplifying and solving for the vortex flux yields

$$\frac{\partial \omega}{\partial n} = \frac{\gamma(S)}{\nu \delta t}. \tag{2.29}$$

By solving for the diffused vorticity from the above expression (after already have solved for the strength of the vortex sheet γ) through a RBF approximation approach as already discussed, the correct amount of vorticity can be diffused to satisfy the no-slip boundary condition. This is the vorticity diffused from the vortex sheet. This vorticity must be added to the field as time advances. This is done by adding the appropriate amount of vorticity to particles collocated on the surface of the vortex sheet and then allowing the VPM to govern the updating of the associated information.

Now having developed the necessary expressions to 1) calculate the strength of a vortex sheet to satisfy the no-slip boundary condition and 2) diffuse the introduced vorticity into the vorticity field, we now look at how to implement these expressions in 2D and how to represent them numerically.

2.4 2D Implementation and Numerical Discretization

In order to solve for the strength of the vortex sheet and then the subsequent diffused vorticity, the goal is to discretize and solve eq. (2.25) for γ and then solve eq. (2.29) for ω. In both cases,

the approach is to approximate the function to be solved for as a radial basis function series with unknown coefficients. By plugging in the RBF approximation into the equations, a linear system is formed that is used to solve for the RBF coefficients.

As this will be done in 2D at first, start by taking the generalized eq. (2.25) and substituting in Q, h, and Γ_{wake} for the 2D case

$$\begin{aligned} Q(\mathbf{x},\mathbf{x'}) &= \frac{1}{2\pi}\frac{\partial}{\partial n}\left[\log|\mathbf{x}-\mathbf{x'}|\right] \\ h(\mathbf{x}) &= \left[(\mathbf{u}_{ext}-\mathbf{u}_b)\cdot\hat{\mathbf{n}}\right] \\ \Gamma_{wake} &= 0 \end{aligned}$$

which gives the following expression for γ

$$\gamma(\mathbf{x}) - \frac{1}{2\pi}\int_S \left[\frac{\partial}{\partial n}\left[\log|\mathbf{x}-\mathbf{x'}|\right] - \frac{\rho_1(\mathbf{x})}{L}\right]\gamma(\mathbf{x'})dS(\mathbf{x'}) = \mathbf{u}_{ext}\cdot\hat{\mathbf{t}}. \tag{2.30}$$

The first order of business is to determine ρ_1, which is a function of the geometry of the boundary. The eigenfunction ρ_1 is the first term in the eigendecomposition of the kernel Q. It's solution is found by integrating eq. (2.22):

$$\int_S Q(\mathbf{x},\mathbf{x'})\rho(\mathbf{x'})d\mathbf{x'} = \lambda\rho(\mathbf{x}).$$

where $\lambda = 1$. This is also done through a RBF approximation. The function ρ_1 is approximated as a series of RBFs and unknown coefficients

$$\rho_1(\mathbf{x}) \approx \sum_{i=1}^{N}\eta_i\phi(|\mathbf{x}-\mathbf{x}_i|) \tag{2.31}$$

where $\mathbf{x}_i$ is a set of points collocated on the geometry of the surface (vortex sheet) and where ϕ is the Gaussian RBF with Gaussian spreading σ

$$\phi(|\mathbf{x}-\mathbf{x}_i|) = \exp\left(\frac{-|\mathbf{x}-\mathbf{x}_i|^2}{2\sigma^2}\right). \tag{2.32}$$

By substituting eq. (2.31) into eq. (2.22) (and replacing Q with it's 2D counterpart)

$$\int_S \frac{1}{2\pi}\frac{\partial}{\partial n}\big[\log(|\mathbf{x}-\mathbf{x'}|)\big]\sum_{i=1}^{N}\eta_i\phi(|\mathbf{x}-\mathbf{x}_i|)dS' = \sum_{i=1}^{N}\eta_i\phi(|\mathbf{x}-\mathbf{x}_i|)$$
$$\sum_{i=1}^{N}\eta_i\int_S \frac{1}{2\pi}\frac{\partial}{\partial n}\big[\log(|\mathbf{x}-\mathbf{x'}|)\big]\phi(|\mathbf{x}-\mathbf{x}_i|)dS' = \sum_{i=1}^{N}\eta_i\phi(|\mathbf{x}-\mathbf{x}_i|).$$

At this point, a coefficient A is introduced to simplify the notation

$$A_i = \frac{1}{2\pi}\frac{\partial}{\partial n}\big[\log(|\mathbf{x}-\mathbf{x'}|)\big]\phi(|\mathbf{x}-\mathbf{x}_i|).$$

Because the goal is to solve for the unknown coefficients η, commute everything on the right hand side to the left (with the other unknown coefficients)

$$\sum_{i=1}^{N}\eta_i\int_S A_i dS' - \sum_{i=1}^{N}\eta_i\phi_i = 0.$$

One more coefficient, Ξ, is introduced that represents the integral piece of the first term

$$\Xi_i = \int_S A_i dS'.$$

The integral can be evaluated using any method deemed convenient. For the purpose of this project, a simple quadrature is used, and the details can be found in Appendix A. Plugging this into the above simplifies the expression to

$$\sum_{i=1}^{N}\eta_i\Xi_i - \sum_{i=1}^{N}\eta_i\phi_i = 0.$$

At this point, $\mathbf{x}$, which is inside both Ξ_i and ϕ_i, is evaluated at each point on the surface. The index j is used to distinguish between i. Doing this allows Ξ and ϕ to be written as double indexed values given as

$$\Xi_{ij} = \int_S A_{ij}dS' = \frac{1}{2\pi}\frac{\partial}{\partial n}\big[\log(|\mathbf{x}_j-\mathbf{x'}|)\big]\phi(|\mathbf{x}_j-\mathbf{x}_i|) \tag{2.33}$$
$$\phi_{ij} = \phi(|\mathbf{x}_j-\mathbf{x}_i|) = \exp\Big(\frac{-|\mathbf{x}_j-\mathbf{x}_i|^2}{2\sigma^2}\Big) \tag{2.34}$$

This now represents a linear system of equations for η that can be solved using matrix methods

$$\left[\Xi_{ij} - \phi_{ij}\right]\left[\eta_i\right] = \left[0\right] \tag{2.35}$$

$$\begin{bmatrix} \Xi_{11}-\phi_{11} & \Xi_{12}-\phi_{12} & \dots & \Xi_{1N}-\phi_{1N} \\ \Xi_{21}-\phi_{21} & \Xi_{22}-\phi_{22} & \dots & \Xi_{2N}-\phi_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \Xi_{N1}-\phi_{N1} & \Xi_{N2}-\phi_{N2} & \dots & \Xi_{NN}-\phi_{NN} \end{bmatrix} \begin{bmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_N \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{2.36}$$

Once ρ_1 is found, it can be substituted into eq. (2.30).

The vortex sheet strength γ is found using a similar RBF approximation method. Approximate the vortex sheet with a radial basis function

$$\gamma(\mathbf{x}) \approx \sum_{i=1}^{N} \alpha_i \phi(|\mathbf{x}-\mathbf{x}_i|). \tag{2.37}$$

Thus by substitution

$$\sum_{i=1}^{N} \alpha_i \phi(|\mathbf{x}-\mathbf{x}_i|) - \frac{1}{2\pi}\int_S \frac{\partial}{\partial n}\left[\log|\mathbf{x}-\mathbf{x}\text{'}|\right] \sum_{i=1}^{N} \alpha_i \phi(|\mathbf{x}-\mathbf{x}_i|) dS' + \int_S \frac{\rho_1(\mathbf{x})}{L} \sum_{i=1}^{N} \alpha_i \phi(|\mathbf{x}-\mathbf{x}_i|) dS' = \mathbf{u}_{ext}\cdot\hat{\mathbf{t}} \tag{2.38}$$

We introduce a set of coefficients F and G

$$F_i = \frac{1}{2\pi}\frac{\partial}{\partial n}\left[\log(|\mathbf{x}-\mathbf{x}\text{'}|)\right]\phi(|\mathbf{x}-\mathbf{x}_i|) \tag{2.39}$$

$$G_i = \frac{\rho_1}{L}\phi_i(|\mathbf{x}-\mathbf{x}_i|) \tag{2.40}$$

And introducing one more set of coefficients

$$\Theta_i = \int_S F_i dS' \tag{2.41}$$

$$\Lambda_i = \int_S G_i dS' \tag{2.42}$$

where these integral are also evaluated using a simple quadrature. By again evaluating $\mathbf{x}$ at each point on the surface, another set of double indexed coefficients arise, leading to another linear

system of equations

$$\sum_{i=1}^{N} \alpha_i \phi_{ij} - \sum_{i=1}^{N} \alpha_i \Theta_{ij} + \sum_{i=1}^{N} \alpha_i \Lambda_{ij} = \mathbf{u}_{ext} \cdot \hat{\mathbf{t}}_i \tag{2.43}$$

$$\begin{bmatrix} \phi_{11} - \Theta_{11} + \Lambda_{11} & \phi_{12} - \Theta_{12} + \Lambda_{12} & \dots & \phi_{1N} - \Theta_{1N} + \Lambda_{1N} \\ \phi_{21} - \Theta_{21} + \Lambda_{21} & \phi_{22} - \Theta_{22} + \Lambda_{22} & \dots & \phi_{2N} - \Theta_{2N} + \Lambda_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{N1} - \Theta_{N1} + \Lambda_{N1} & \phi_{N2} - \Theta_{N2} + \Lambda_{N2} & \dots & \phi_{NN} - \Theta_{NN} + \Lambda_{NN} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{bmatrix} = \begin{bmatrix} \mathbf{u}_{ext} \cdot \hat{\mathbf{t}}_1 \\ \mathbf{u}_{ext} \cdot \hat{\mathbf{t}}_2 \\ \vdots \\ \mathbf{u}_{ext} \cdot \hat{\mathbf{t}}_N \end{bmatrix} \tag{2.44}$$

Once these coefficients are solved for, they can be plugged back into eq. (2.37) to get an expression for γ everywhere on the surface of the object.

Finally, now knowing the vortex sheet strength γ, use eq. (2.29) to solve for the diffused vorticity from the boundary

$$\frac{\partial \omega}{\partial n} = \nabla \omega \cdot \hat{\mathbf{t}} = -\frac{\gamma}{\nu \delta t}$$

Again, this is done through a RBF approximation

$$\omega(\mathbf{x}) \approx \sum_{i=1}^{N} \beta_i \phi(|\mathbf{x} - \mathbf{x}_i|) \tag{2.45}$$

By plugging the RBF approximation into the diffusion expression

$$\sum_{i=1}^{N} \beta_i \nabla \phi(|\mathbf{x} - \mathbf{x}_i|) \cdot \hat{\mathbf{n}}_i = -\frac{\gamma_i}{\nu \delta t} \tag{2.46}$$

A simplifying coefficient Π is introduced

$$\Pi_i = \nabla \phi(|\mathbf{x} - \mathbf{x}_i|) \cdot \hat{\mathbf{n}}_i \tag{2.47}$$

and upon evaluating $\mathbf{x}$ at each point on the surface gives the final double indexed coefficient and final linear system

$$\sum_{i=1}^{N} \beta_i \Pi_{ij} = -\frac{\gamma_i}{\nu \delta t} \tag{2.48}$$

$$\left[\Pi_{ij}\right]\left[\beta_i\right] = \left[-\frac{\gamma_i}{\nu \delta t}\right] \tag{2.49}$$

$$\begin{bmatrix} \Pi_{11} & \Pi_{12} & \dots & \Pi_{1N} \\ \Pi_{21} & \Pi_{22} & \dots & \Pi_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \Pi_{N1} & \Pi_{N2} & \dots & \Pi_{NN} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_N \end{bmatrix} = \begin{bmatrix} -\frac{\gamma_1}{v\delta t} \\ -\frac{\gamma_2}{v\delta t} \\ \vdots \\ -\frac{\gamma_N}{v\delta t} \end{bmatrix} \tag{2.50}$$

These coefficients can now be plugged into eq. (2.45) in order to arrive at the vorticity that is diffused from the vortex sheet at each time step. The vorticity that leaves the surface does so in the form of vortex particles with the associated vorticity, after which the VPM dictates how that particle evolves over time.

2.5 Code Structure

The Vortex Sheet Boundary Method can be implemented in the following manner. There are two primary structures: 1) How the VSB itself is built, and 2) How the VSB couples with the VPM.

The first step is to generate the geometry of the boundary. The geometry for any boundary must be supplied as a set of points that define the surface. In order to allow for generality in both 2D and 3D, both the normal unit vectors and tangent unit vectors to the surface of the boundary must also be supplied. The normal and tangent unit vectors are given for each of the given points that define the surface. In addition, operating conditions such as the free stream component of the velocity and the angle of attack (the angle between the chord of an airfoil and the direction of the surrounding free stream). Once these sets of vectors (position vectors, normal unit vectors, and tangent unit vectors) and the operating conditions are supplied, calculation of the geometric parameters can begin.

For every boundary and for every velocity field (whether just a free stream or a combination of vorticity particle contributions and free stream), there are two sets of radial basis function (RBF) coefficients that need to be calculated. They correspond to the equations (2.31) and (2.37). Once these equations are solved and a set of coefficients are found for each RBF approximation, they are

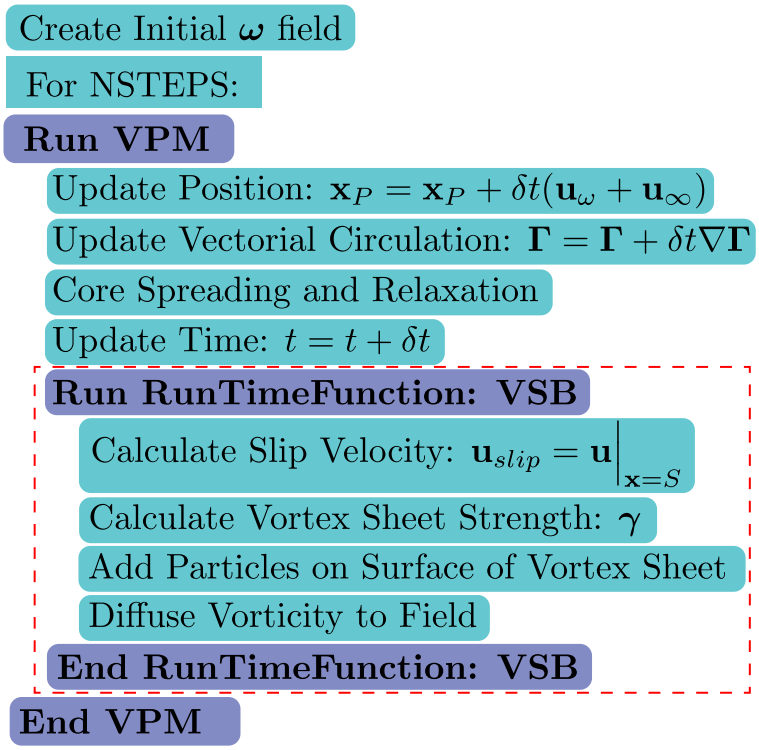

Figure 2.3 Algorithm layout for coupling of Vortex Particle Method (VPM) and Vortex Sheet Boundary Method (VSB). After creating an initial vorticity field, the VPM steps through time, updating and calculating several quantities. These include the position, circulation, and core spreading of each particle. At every time step, the VPM calls a run time function in order to add particles, remove particles, debug, etc. It is here where the VSB is implemented. At every time step, the VSB calculates the slip velocity due to all contributions of velocity. Based on that slip velocity, the strength of the vortex sheet required to induce the no-slip boundary condition is found. Particles are added on the surface of the vortex sheet according to the required vorticity field. These particles are then diffused into the particle field, where the VPM handles the updating of their information.

stored in a mutable struct called Parameters. This object contains the Boundary object, the velocity field as a function of position, and the RBF coefficients α_i and η_i.

With the Parameters object, which again contains the information regarding the geometry and operating conditions, the VSB is ready to couple with the VPM. The first step is to define the initial vorticity field. When using the VSB, the initial vorticity field is the vorticity field produced by

the vortex sheet (found to induce the no-slip boundary condition). Next, the VPM calls a runtime function every time step. This is used for adding particles to the field, viewing conditions throughout the simulation, and debugging. This is where the VSB is coupled. After running through the initial steps of the VPM (see Fig. 2.3), the RunTime Function is called. The VSB calculates the slip velocity (the velocity on the surface of the boundary, i.e. the no-slip velocity), and calculate the required vortex sheet strength γ needed to enforce the no-slip boundary condition. Once that is known, particles are added to the surface with the corresponding vorticity found from the diffusion equation. These particles are then diffused into the vorticity field, where the VPM continues to update their information (position vector, circulation vector, etc.). This process is repeated for a given number of time steps until the simulation is terminated.

Chapter 3

Results

The verifications and results of the Vortex Sheet Boundary (VSB) Method are presented hereafter. The numerical methodology is verified against a known analytical solution for the case of an infinite cylinder in free-stream flow. Next, we look at a dynamic visualization of the VPM coupled with the Vortex Particle Method (VPM) for the same infinite cylinder case. The cylinder is immersed in a flow characterized by creeping flow. Together with the analytical solution and visualization, we have a quantitative and qualitative verification of the methods presented in this thesis.

3.1 Analytical Verification - Cylinder

For the case of an infinite cylinder in free flow (infinite in the plane normal to the flow field), there exists an analytical solution for the strength of the vortex sheet to induce the no-slip boundary conditions. We wish to solve for γ from the expression

$$\gamma(\mathbf{x}) - \oint \left[Q(\mathbf{x}, \mathbf{x'}) - \frac{\rho_1(\mathbf{x})}{L} \right] \gamma(\mathbf{x'}) dS(\mathbf{x'}) = h(\mathbf{x}) - \frac{\rho_1(\mathbf{x})}{L} \Gamma_\Omega$$

where

$$Q(\mathbf{x},\mathbf{x'}) = \frac{1}{2\pi}\frac{\partial}{\partial n}[\log|\mathbf{x}-\mathbf{x'}|]$$
$$h(\mathbf{x}) = -(\mathbf{u}_{ext}-\mathbf{u}_{body})\cdot\hat{\mathbf{t}}$$
$$\Gamma_\Omega = \int_{V_e}\Omega(\mathbf{x})dV(\mathbf{x}) = 0.$$

The term Γ_Ω represents the circulation produced by a rotating body. For all cases studied here, there is no circulation, thus is equals zero. For elliptic bodies, the eigenfunction ρ_1 can be found analytically by integration of (2.22), resulting in

$$\rho_1(\theta) = [1-\varepsilon\cos^2\theta]^{-1/2}$$

where ε is the eccentricity of the elliptical body. For a cylinder, $\varepsilon = 0$, thus $\rho_1(\theta) = 1$. After some work, we arrive at the strength of the vortex sheet γ given as the simple expression

$$\gamma(\mathbf{x}) = h(\mathbf{x}) = -(\mathbf{u}_{ext}-\mathbf{u}_{body})\cdot\hat{\mathbf{t}} = -u_\infty\sin\theta$$

for θ being the standard polar angle measured from the positive x-axis.

We compare the results of the analytical solution to the numerical evaluation of the vortex sheet strength and present them in Fig. 3.1

3.2 Qualitative Verification - Cylinder in Creeping Flow

Having verified that the VSB accurately calculates the required vortex sheet strength to induce the no-slip boundary conditions, we turn to coupling the VSB and VPM.

Following the methodology presented in chapter 2 for coupling the VSB and VPM, the case of a circle in free stream is implemented. Due to the current limitations of the implementation of the method, a Reynold's number of 0.1 is used. The Reynold's number is a ratio of the inertial to

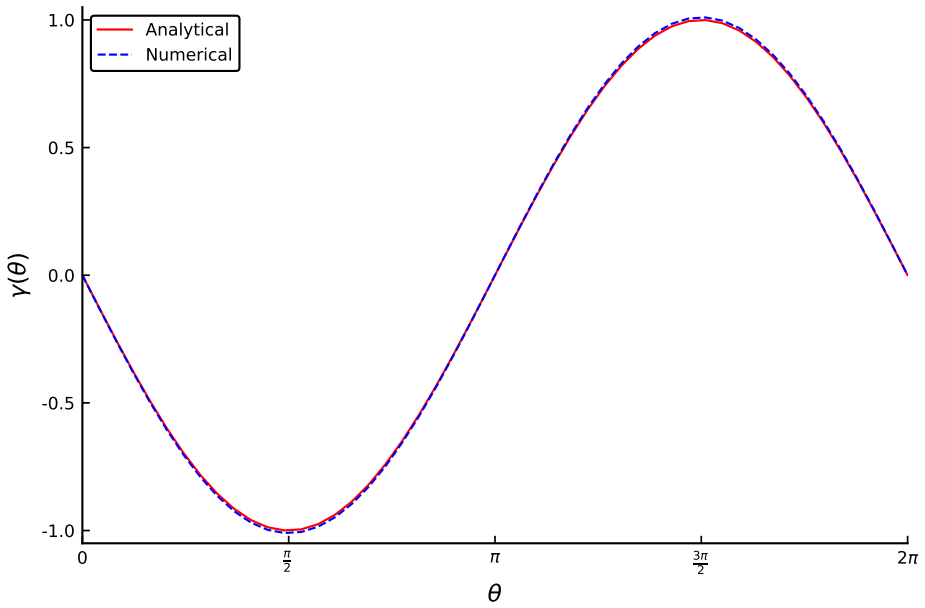

Figure 3.1 Comparison of analytical and numerical solutions for the strength of the vortex sheet γ in free-stream. As can be observed, the numerical solution correctly calculates the required vortex sheet strength as compared to the analytical solution derived in chapter 3.

viscous forces acting within a fluid, and is defined as

$$Re = \frac{\text{Inertial Forces}}{\text{Viscous Forces}} = \frac{u_\infty D}{\nu} \tag{3.1}$$

where u_∞ is the free stream, D is the diameter of the circle, and ν is the kinematic viscosity. The Reynold's number is the parameter that describes fluid properties such as creeping flow, laminar flow, and turbulent flow. Aircraft fly at high Reynold's numbers so the flow becomes turbulent. This model hopes to be able to model high Reynold's numbers to be more applicable in the aircraft design process. To increase the Reynold's number, the kinematic viscosity must decrease. However, this leads to numerical instabilities that are introduced in the diffusion sub-step. Eq. 2.29 shows how the vorticity flux is related to the strength of the vortex sheet and kinematic viscosity. The vorticity flux is inversely related to the kinematic viscosity, so a decrease in ν leads to an increase in the vorticity flux. For reasons currently unknown, the simulation is stable when the denominator of eq. 2.29 is on the order of one and unstable if not. So, this current limitation only allows for

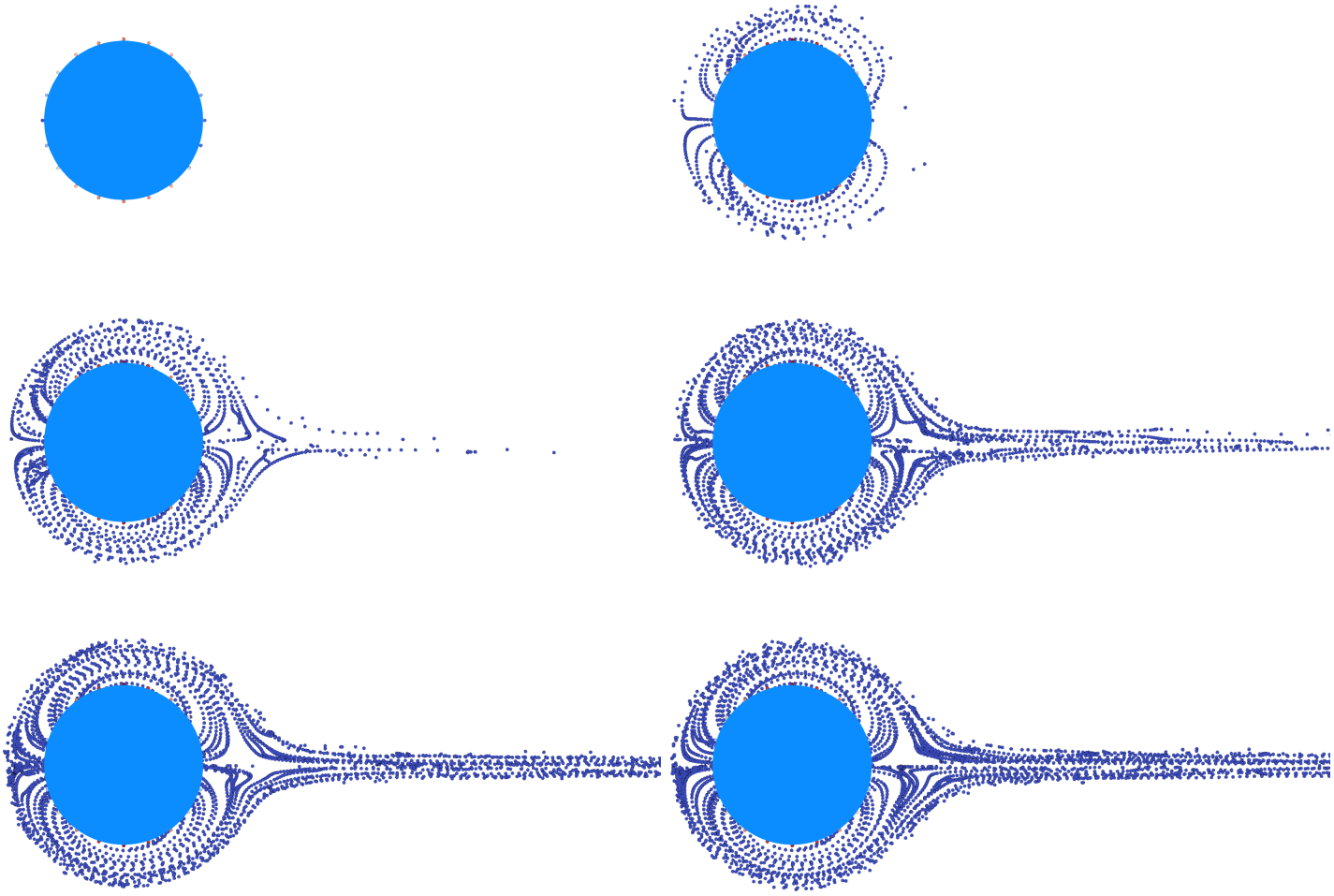

Figure 3.2 Snapshots of a cylinder in creeping flow (Re = 0.1). Various qualitative verifications can be made including the presence a stagnation point on the leading edge of the cylinder, no boundary layer, and a long, thin trailing wake behind the cylinder.

low Reynold's number calculations. Future studies of the stability, time-stepping, and operation conditions hope to show how to improve this obviously large limiting factor.

Snapshots of the cylinder in free-stream are presented in Fig. 3.2. This serves as a preliminary and qualitative verification of both the coupling of the VSB and VPM and as a verification of the diffusion sub-step of the VSB. Again, the flow is characterized by creeping flow. Creeping flow has little or no boundary layer, as there is not a large velocity gradient between the object and the fluid.

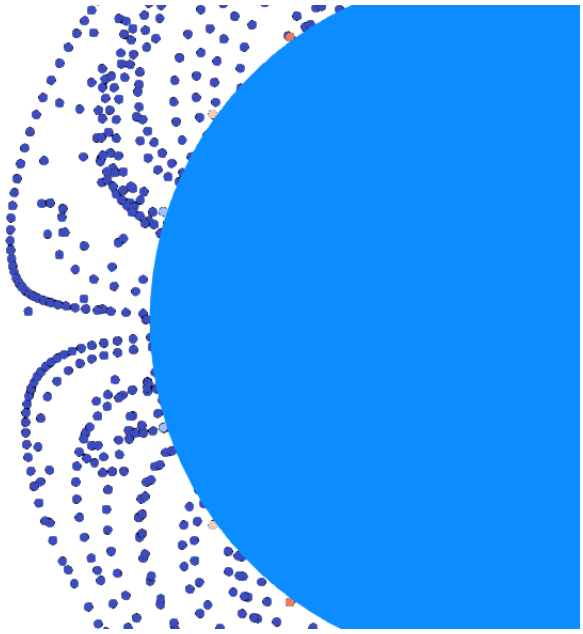

Figure 3.3 Close-up of cylinder in free-stream. This is the front of the cylinder. The stagnation point is highlighted here. The particles diffusing off of the surface follow the contours of the cylinder around towards the rear of the cylinder. The stagnation point is where the pressure is zero and any streamlines (lines of velocity, or paths of the particles) divide and go around the surface. The stagnation point on a cylinder is on its leading edge.

This can be seen in 3.2 close to the surface of the circle. A stagnation point, a point of zero pressure where the flow is forced around an object, can been seen as well in Fig. 3.3. The thin trailing wake behind the circle in the simulation is also reminiscent of an object in creeping flow. These various qualitative verifications offer addition evidence of the correctness of the presented method.

3.3 Further Work

A lot can be done to improve upon the methods presented here. Various modifications can be made to increase both the speed of the computation and the scope of the application.

To expand the scope of the VSB and VPM coupling, these methods can be implemented in 3D. They are inherently 3D methods in the first place. The same process that was done in 2D can be

done in 3D to derive the numerical versions needed to implementation. The geometry would need to be well defined, along with the appropriate tangent and normal unit vectors.

To increase the speed and memory usage of the VSB, the numerical method for solving for the RBF coefficients can be adjusted. Due to the symmetrical nature of RBFs, their smoothness, and their local influence, the matrices formed by the evaluation of the RBFs can be solved through iterative methods. Indeed, the RBF coefficients can be solved for without ever having to actually form a matrix at all. One method for doing this is called the Conjugate Gradient method, and it is an iterative method for solving systems with specific properties. The linear systems formed by the RBF approximations fit these properties and allow for a faster and less memory extensive method. This will greatly enhance the VSB to allow for much more refined geometry through the use of many more particles on the surface of a boundary.

Appendix A

Quadrature Integration

Numerical integration is a topic that has a wide expanse of solutions and approaches. Throughout the course of the derivation of the equations, and especially the numerical discretization, the need for a way to integrate general surface and volume integrals became apparent. The method I settled on is a simple quadrature that I detail below. In every case where an integration is performed numerically, this is the function that does the work.

Let's say I wish to evaluate the following

$$\int_S \mathbf{F} \cdot \hat{\mathbf{n}} \, dS = \int_S G \, dS \tag{A.1}$$

where $G = G(x,y,z) = \mathbf{F}(x,y,z) \cdot \hat{\mathbf{n}}(x,y,z)$. This is a generalized surface integral. There is a vector function $\mathbf{F}$ that I wish to evaluate along a surface that has a unit normal vector function $\hat{\mathbf{n}}$. The surface is approximated as a collection of panels. The integral can then be approximated as a summation of the function evaluated at each panel times the area of each panel, or

$$\int_S G(x,y,z) \, dS \approx \sum_{j=1}^{N} G(x_j, y_j, z_j) \Delta S_j. \tag{A.2}$$

As $N \to \infty$, the exact integral is obtained once again.

Appendix B

Code Listings

The code can be publicly accessed, downloaded, and used (with acknowledgements) via Github at https://github.com/byuflowlab/VSB.jl.

Index

Bibliography

[1] E. J. Alvarez and A. Ning, "Development of a Vortex Particle Code for the Modeling of Wake Interaction in Distributed Propulsion," 2018 Applied Aerodynamics Conference (2018).

[2] J. P. David J. Willis and J. K. White, "A Combined pFFT-Multipole Tree Code, Unsteady Panel Method with Vortex Particle Wakes," International Journal for Numerical Methods in Fluids **00,** 1–6 (2000).

[3] C. F. G. v. B. A. Palha, L. Manickathan, "A Hybrid Eulerian-Lagrangian Flow Solver," (2015).

[4] P. D. K. Georges-Henri Cottet, *Vortex Methods Theory and Practice* (Cambridge University Press (CUP), 2000).

[5] L. B. Christopher D. Cooper, *19th AIAA Computational Fluid Dynamics Conference* (2009).

[6] F. P. P. Koumoutsakos, A. Leonard, "Boundary Conditions for Viscous Vortex Methods," Journal of Computational Physics **113,** 52–61 (1994).

[7] Y. Hon and R. Schaback, in *Scientific Computing with Radial Basis Functions*,

[8] A. P. Joseph Katz, *Low-Speed Aerodynamics*, 2nd ed. (Cambridge University Press (CUP), 2001).

[9] H. Hochstadt, *Integral Equations* (John Wiley Sons, 1989).

[10] R. P. P. Bassanini, M.R. Lancia, "Numerical Approximation of Boundary Integral Equations in Three Dimensional Aerodynamics," In *Boundary Element Methods: Fundamentals and Application*, N. N. S. Kobayashi, ed., pp. 40–48 (Springer, Berlin, Heidelberg, 1992).