APPLYING SOURCE PANEL METHOD TO REAL WORLD VESSEL PLANNING

by

Thomas Sjåstad

Thesis

for the degree of

Master of Science

Faculty of Mathematics and Natural Sciences
University of Oslo

August 2020

Abstract

Acknowledgements

I would like to thank both my supervisors Morten- Hjorth Jensen and Thomas Mestl for their guidance and insight. This thesis would not have happened without you guys.

Contents

1 Introduction **6**

1.0.1 A brief history of navigation at sea 8
1.0.2 Problem statement 9
1.0.3 Constraints 10

2 Theory **11**

2.0.1 Vector 11
2.0.2 Vector field 12
2.0.3 Navier-Stokes equations 13
2.0.4 Stream functions 14
2.0.5 Velocity potential 16
2.0.6 Properties of the laplace equation 17
2.0.7 Solutions of the Laplace equation 17

3 AIS-data and map projection **47**

3.1 AIS-data 47
3.1.1 Preprocessing AIS-data 48
3.2 Maps and coordiante systems 50
3.2.1 Geographic coordinate system 50
3.2.2 Map selection 52
3.2.3 Projections 52

4 Implementation **55**

4.0.1 Dependencies 56
4.0.2 Creating routes with Pandas 56
4.0.3 Basemap 59
4.1 Analytic approach 63
4.2 Computing the source strengths 63

4.2.1 Computing velocities 65

5 Results **68**
5.1 Classifying routes 68
5.2 Analytic flows around circular objects 70
5.3 Setting the individual strength coefficients 70

Chapter 1

Introduction

Before computers and computation, data was logged manually. While traveling across the worlds seas, logging wind and current information enabled ships to decrease their travel time (Lewis 1927).

Transportation on sea has become, and is still to this day one of the biggest businesses within the global trade. Maritime shipping amounts to 90 % of the world's trade transportation method (UN-Business Action Hub, United Nations 2020). Route planning is a comprehensive guide developed and used to determine the most favorable route, raise awareness of potential problems and ensure the vessel's safe passage. Nowadays, constructing a route depends in most cases on knowledge of past voyages or surroundings of a route e.g. map data. After the development of the Automatic Identification System data (AIS), the maritime industry has used these data to make predictions and estimates for vessels. It's applications have been estimating streamline cargo transfers at harbors, helping companies take better decisions and predicting traffic flow of ships in given region and time period (Wang, Li, and Zhang 2019). The approaches and methods used in research base their predictions on previous voyages (Meijer 2017).

However, AIS data is not always reliable or available in every possible area wanted to traverse. The data could be limited by a malfunction on land or on the vessel. For instance; a vessel could be hiding its location, turning of their transmitter. The signal could be bad which can result in loss of data. The computation time required are often quite large as the amount of data needed to make a good prediction needs to be significantly large (Pallotta, Vespe, and Bryan 2013). The research area constituting route predicting include multiple disciplines such as, mathematics, physics and informatics. Also, in the past several years the application of artificial intelligence, more specifically, machine learning has become quite popular in maritime vessel tracking, because of it's suitability for data treatment in multiple fields within the industry (Pallotta, Vespe, and Bryan 2013), (Meijer 2017), (Århus and Salen 2018).

Depending on method some sort of input to model is required. The input often leads to an output of certain character. Here, mainly inputs will have the type of start and end points or AIS-data. The models dependent on start and end-point

(Hvamb 2015) presents three approaches for route planners. These methods requires a map. The first one using a Voronoi diagram. Making connecting lines between vertices of polygons or points describing obstacles e.g islands. Then, perpendicular lines at their half distance are drawn and connected creating Voronoi lines. Each line not intersecting the obstacles are suggested routes.

The next method is based on Rapidly-exploring Random Trees (RRT). Searching an area by sampling points a graph is produced. A bias toward to the goal makes the model possible for convergence.

The final method presented in (Hvamb 2015) is applies a Probabilistic roadmap and uses this for planning routes for marine vessels.

(Pedersen and Fossen 2012) makes the first route planner by applying potential flow theory on marine vessels. Setting up potentials with different flow characteristics, a flow pattern emerges. From this, following certain streamlines lead to the desired goal. In addition, a vessel guidance scheme is derived making sure the vessel gets to its destination.

(Besse et al. 2015) analyzes taxi-drivers' patterns in San Francisco with GPS-data. Predicting the destination of the taxi's by their choice of route with a method derived from (Besse et al. 2015) a Symmetrized Segment-Path Distance (SSPD).

(Pallotta, Vespe, and Bryan 2013) implements a unsupervised learning scheme (Traffic Route Extraction and Anomaly Detection) to predict routes in different locations. By clustering AIS-data into waypoints for certain areas in a route. Vessels are then classified with state when entering a waypoint.

In addition, when working with the data it is crucial to pre-process the data before making models based on it (Besse et al. 2015). Also, there might be places where no vessel has traversed. How does one make prediction without data?

Instead of using AIS data, conducting voyages is done based on map information. Algorithms search through a space given start and end points. The goal is to maneuver through this space and create a route without colliding with obstacles i.e land (Hvamb 2015). Algorithms developed vary from applications in sciences such as informatics, mathematics and physics (Hvamb 2015), (Besse et al. 2015), (Pedersen and Fossen 2012). The latter has motivated the basis for the research done in this thesis.

In this thesis, a framework for creating and predicting routes for marine vessels will be attempted. The framework will be based on principles and theory in the field of fluid mechanics, more specifically potential theory. By assuming an irrotational and incompressible fluid we can describe the velocity of the fluid by a velocity potential. By setting up an environment given map characteristics the route will be produced by calculating velocity vectors thus finding streamlines. From the findings of (Pedersen and Fossen 2012), the potential flow solution was a good approach for a vessel motion planner. We will take these findings one step further, testing this approach on a geographical area. We will implement both an analytic and numerical solver in this thesis. The reason for both solvers is that the analytic does not take into account the map characteristics which is crucial when finding the fluid velocities surrounding arbitrary geometries. The numerical method is called Source Panel Method (Anderson 1991). By using boundary conditions and setting constraints in specified areas of the landscape makes it possible to give the map as input to the model and thus computing the flow field in the area of interest.

1.0.1 A brief history of navigation at sea

A subtle section is presented here show as to why navigation at sea is important and how far the technology has taken us.

Before, navigating by sea was done by observing celestial bodies and determining one's position. This was before the invention of the compass. One of the first tools used in navigation was a magnetic compass. The compass feels a force by the earth's magnetic field and points to what is known as the "magnetic north". This created some confusion and inconsistent measures at sea due to unawareness of magnetic variance (difference between magnetic north and geographic north).

Jumping centuries forward in time, the twentieth century established navigational tools such as radio beacons, radar, the gyroscopic compass, and the global positioning system (GPS). Some of the tools' characteristics are:

Gyroscopic compass introduced in 1907, always points at true north keeping a point of reference for navigators.

Radar(short for "radio detection and ranging") system was produced in 1935. It was used to locate objects beyond range of vision by projecting radio waves against them. This was, and still is, very useful on ships to locate other ships and land when visibility is reduced.

The *LORAN* (LOng RAnge Navigation) uses low frequency radio transmissions stations to determine a ship's position. Used in U.S navigation with accuracy up to a hundred meters.
In the late twentieth century, replacing the *LORAN* was the global positioning system (GPS). GPS uses the same principle of time difference from separate signals as Loran, but the signals come from satellites. As of 2002, the system consisted of 24 satellites, and gave the mariner a position with accuracy of 9 meters (30 feet) or less [1].

1.0.2 Problem statement

The goal of this thesis will be to create routes from a to b for marine vessels to traverse based on potential theory. Some information about the map is required to use as input when making this type of model. We will try to determine how to the model behaves in a simulated environment together with a realistic one, also looking at the model can be generalized in created several routes at a time.

In addition to making a vessel planner, we will extract routes from AIS-data which will give insight into how the vessel actually travel and we will be able to

[1] Literature in this section is gathered from :http://www.waterencyclopedia.com/Mi-Oc/Navigation-at-Sea-History-of.html

compare the simulated routes with real ones. The thesis will be split up into two parts, an AIS-data and fluid mechanical. The first part be to analyze AIS-data. Given a data set containing vessel information, how is it possible to construct a route. Analyzing the data will be the first step. Looking into how to structure the data in order to find patterns which the model can be based on. An attempt to classify routes into their direction of travel will be conducted. The routes will be divided into vessels travelling ”To”, ”From” or ”Through” the designated destination. This will help getting an overview of the data as the amount of data is often to large to plot up at the same, and not systematized in any way.

The second part of the thesis: fluid mechanical approach. The goal is to Make a velocity vector field given the characteristics of the map and construct a route in which the vessels can base their travel on. Application of the analytic solutions will be simulated following a numerical solution for a more computational heavy simulation. The necessary representations of a map will be objects which needs to be avoided together with a end and/or start point of route.

1.0.3 Constraints

In the thesis when it analyzing the data set, we will assume land is the only obstacle for each vessel. In reality there could be several vessels traversing the same area at the same time which could lead to a change in course. Could be useful to mention shipping rules on the sea!!

Chapter 2

Theory

- -
Worth mentioning what the criterias are for setting up a potential. From Navier stokes equation and conservation of mass to streamfunction. No curl $->$ Potentialflow
- -

In this section the mathematics used in the thesis is presented. The theory will mostly be from the field of fluid mechanics. A brief introduction to the equation governing the properties and motion of fluids will be given, namely Navier-Stokes equations. From the general Navier-Stokes equations, some assumptions and simplifications will be made yielding new equations which will be the used in this thesis. These equations are part of what constitute potential theory in fluid mechanics.
Visualization is a key component the research on which we will conduct. Therefore, tools for using vector fields and plotting are a necessity. A more detailed description can be found in (Gjevik 2009) and (Acheson 2003). The data used will also be described in this section.

2.0.1 Vector

A vector is used to describe a quantity with direction and magnitude. Vectors are useful in navigation for representing quantities such as positions and velocities. From a point of reference this can be represented in a coordinate system. Mathematically this can be written as:

$$\vec{r} = [x, y, z]$$

The arrow indicates that r is of the type vector. The first component moves along the x-axis, second along the y-axis and the third along the z-axis. The three components indicate that the vector $\vec{r}$ exists in a three-dimensional space. This can visualized as seen in the figure below:

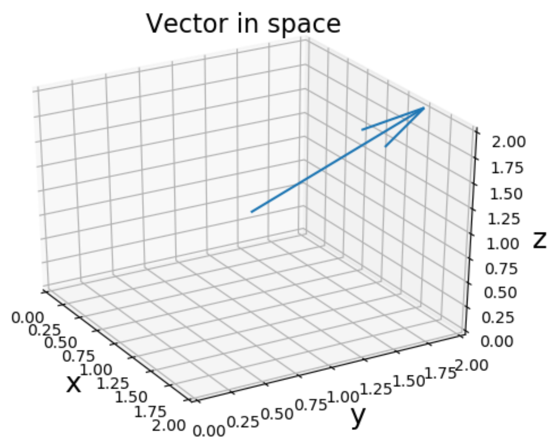

Figure 2.1: shows a typical vector representation. The arrow shows a vectors direction, it's length represents it's magnitude.

2.0.2 Vector field

As well as vectors, vector fields are useful when wanting to describe velocities, forces or accelerations within a domain. Vector-field is a set of vectors assigned to point within a space. Since we are interested in describing the motion of fluids within a domain, these quantities are important to the understanding of what is happening to the system. The field are represented with vectors which are functions of spacial coordinates and time. Mathematically this is written as:

$$\vec{F} = \vec{F}(x, y, z, t)$$

This can further be written as:

$$\vec{F}(x, y, z, t) = (F_x(x, y, z, t), F_y(x, y, z, t), F_z(x, y, z, t))$$

F_x, F_y, F_z are scalar fields. The field is called stationary if it is independent of time. See figure below for a stationary vector field.

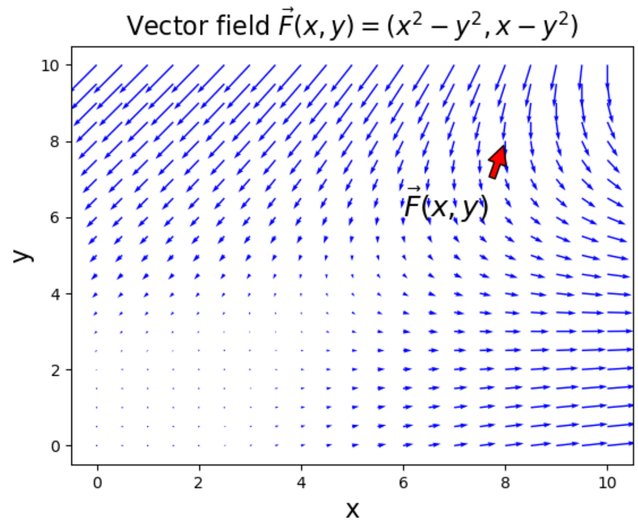

Figure 2.2: Here a vector field is visualized F(x, y) taking in two parameters x and y and calculates the field from [0-10] in both x and y-direction. The arrow size is the magnitude at that specific point, and also indicating the fields direction..

2.0.3 Navier-Stokes equations

The Navier-Stokes equations are derived using the laws of several quantities which are conserved. The quantities include mass, momentum and energy.
The first, the continuity equation arise from the law of conservation of mass which says that mass can not be created or disappear. The general equation of continuity is stated below:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0$$

where ρ is the fluid density, $\nabla = [\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}]$ the gradient and $\vec{u}$ the velocity. We will assume that the properties of fluids looked at are incompressible. This means the fluid's density is constant. The continuity equation for incompressible fluid states:

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

Where both density terms are excluded first by the statement above and dividing by the density ρ.
This leads up to the next equation, the momentum equation. It states that a fluid's density times acceleration is proportional to the forces acting upon the fluid. Since

we are only worried about incompressible fluids the momentum equation for such a fluid is:

$$\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla)\vec{u} = -\frac{\nabla p}{\rho} + \mu \nabla^2 \vec{u} + f_v$$

This is the second Navier-Stokes equation for an incompressible fluid. The left side hand states the inertia term. The terms $\frac{\partial \vec{u}}{\partial t}$ and $(\vec{u} \cdot \nabla)\vec{u}$ are the local and convective acceleration respectively. The right hand side is pressure gradient $\frac{\nabla p}{\rho}$, viscous forces $\mu \nabla^2 \vec{u}$ and external forces f_v. As for the continuity equation we will make some assumption which in turn will produce equation which will be used here. The fluid which will be focused on are stationary and inviscous $\mu = 0$. These assumptions leads further to potential flow theory which will be covered next. The next section will be the main focus of the physics model of the thesis.

2.0.4 Stream functions

Streamlines (see fig 2.3, 2.4 and 2.5), also called field lines are useful to visualize the flow in a fluid. By assuming that the flow of the fluid is steady, that is, the velocity does not change with time at a fixed point in space, makes it possible to find these streamlines. Mathematically the velocity can be written as:

$$\vec{u} = \frac{d\vec{x}}{ds} \tag{2.1}$$

Where $\vec{u} = [u(x,y,z), v(x,y,z), w(x,y,z)]$, $d\vec{x} = [dx, dy, dz]$ and ds is a small change in the curve s. Streamline curves are defined as having same direction as the velocity vector at each point. Then the cross product between the velocity vector and curve s is zero:

$$\vec{u} \times d\vec{x} = 0 \tag{2.2}$$

Writing out the components from the cross product gives:

$$(vdz - wdy)\vec{i} = 0$$

$$(wdx - udz)\vec{j} = 0$$

$$(udy - vdx)\vec{k} = 0$$

This gives the equations:

$$\frac{dx}{u} = \frac{dy}{v} = \frac{dz}{w} = ds \tag{2.3}$$

For a 2-dimensional description the equation for an incompressible flow is :

$$\nabla \cdot \vec{u} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{2.4}$$

$\vec{u}$ is the velocity vector is a function of u and v, $\vec{u} = [u, v]$. u and v are the velocity components in x and-y direction, respectively. $\nabla = [\frac{\partial}{\partial x}, \frac{\partial}{\partial y}]$ is the differential operator nabla. We can relate the velocity vector to the stream functions as:

$$u = \frac{\partial \psi}{\partial y} \tag{2.5}$$

$$v = -\frac{\partial \psi}{\partial x} \tag{2.6}$$

ψ is the stream function. Putting (2.5) and 2.6) into (2.4) yields:

$$\nabla \cdot \vec{u} = \frac{\partial^2 \psi}{\partial x \partial y} - \frac{\partial^2 \psi}{\partial x \partial y} = 0 \tag{2.7}$$

Also assuming the field is irrotational $\nabla \times \vec{u} = 0$, the laplace equation is obtained:

$$\nabla \times \vec{u} = \frac{\partial^2 \psi}{\partial y^2} + \frac{\partial^2 \psi}{\partial x^2} = \nabla^2 \psi = 0 \tag{2.8}$$

Where eq (2.5) and eq (2.6) are substituted in as expressions for the velocity components.

Furthermore, eq (2.2) can give us the conditions on what the stream function must look like in order for this to be true. Putting eq (2.5) and eq (2.6) into eq (2.2) and assuming this is for a 2 dimensional flow the stream function yields:

$$udy - vdx = \frac{\partial \psi}{\partial y} dy + \frac{\partial \psi}{\partial x} dx - = 0 \tag{2.9}$$

$$\partial \psi = 0 \tag{2.10}$$

$$\psi = Constant \tag{2.11}$$

ψ is constant along the streamline. All stream functions satisfying eq (2.7) are valid choices for ψ. Stream functions ψ solving the laplace equations (2.8) are many because of its linearity property. Sinks, sources and doublets are fundamental solutions. The velocity components represented in polar coordinates are:

$$v_r = -\frac{1}{r}\frac{\partial \psi}{\partial \theta} \tag{2.12}$$

$$v_\theta = \frac{\partial \psi}{\partial r} \tag{2.13}$$

and the laplacian in polar coordinates:

$$\nabla^2 \psi = \frac{\partial^2 \psi}{\partial r^2} + \frac{1}{r}\frac{\partial \psi}{\partial r} + \frac{1}{r^2}\frac{\partial^2 \psi}{\partial \theta^2} = 0 \tag{2.14}$$

$r = \sqrt{x^2 + y^2}$ is the radius and θ is the angle from origo.

2.0.5 Velocity potential

There is another way of relating a scalar function to the laplace equation is what is known as the velocity potential. The velocity potential is a scalar function denoted $\Phi(x, y, z, t)$. Here, we will be concerned about a 2-dimensional scalar function in a steady state. The scalar function will take in two parameters $\Phi(x, y)$. We mention this here because the numerical method which we will derive later, the derivations and basis will assume there exists a velocity potential. However, these two scalar functions lead to the laplace equation. We will use the same equation as for the stream function, but with a slightly different approach. Instead of starting with the continuity equation (2.4), but assuming an irrotational flow $\nabla \times \vec{u} = 0$ leads to the existence of a scalar function $\Phi(x, y)$ satisfies the irrotational property. The velocity is said to be derived from a velocity potential. This can be written as:

$$\vec{u} = \nabla \Phi \tag{2.15}$$

$\vec{u}$ is the fluid velocity, $\nabla = [\frac{\partial}{\partial x}, \frac{\partial}{\partial y}]$ is the differential operator and Φ the scalar function. Putting the above equation into eq (2.4) we find:

$$\nabla \cdot \vec{u} = \nabla \cdot (\nabla \Phi) = \nabla^2 \Phi = 0 \tag{2.16}$$

We have now related a scalar function Φ to the laplace equation as we did with the stream function ψ. The velocity components for the vector $\vec{u} = [u, v]$ are then for the velocity potential Φ:

$$u = \frac{\partial \Phi}{\partial x}, \quad v = \frac{\partial \Phi}{\partial y} \tag{2.17}$$

2.0.6 Properties of the laplace equation

The laplacian has some important properties which come in handy when wanting to build flow characteristics. The laplacian operator is a linear operator which says that if two different stream functions are solutions to eq (2.8) then the sum is also a solution. This is known as the superposition principle.

2.0.7 Solutions of the Laplace equation

Here, analytic solutions of the Laplace equation are derived which serve the role as separate flow characteristics: a source solution representing a starting point for vessels, a source representing an end point and a doublet representing circle objects. These flow characteristics can be combined by applying the superposition principle thus simulating basic flow patterns.

Uniform

A uniform flow has constant velocity components u and v in x and y direction respectively. By the relation from eq (2.5) and eq (2.6) this satisfies eq (2.7) and is therefore a solution.

Sink and source

Consider a source flowing radially outwards from a point with a steady rate. Finding how much is flowing outwards, the velocity components of this type of flow can be found. By integrating the flow over a circle with radius r this yields:

$$Q = \int_0^{2\pi} v_r r d\theta = 2\pi r v_r \tag{2.18}$$

Q is the flow strength. A flow which emits fluid isotropically i.e is uniform in all directions. Substituting (2.12) in 2.19 for the radial velocity and integrating. This type of flow has the characteristics of a source and has the stream function:

$$\psi_{source} = \frac{Q}{2\pi}\theta \tag{2.19}$$

This is also a solution of the Laplace equation (2.8). The corresponding velocity components take the form:

$$u_{source} = \frac{Qx}{2\pi(x^2+y^2)} \tag{2.20}$$

$$v_{source} = \frac{Qy}{2\pi(x^2+y^2)} \tag{2.21}$$

Where eq(2.12) is equated with eq (2.18) for the radial velocity. The stream function satisfies the laplace equation eq(2.8). A flow which absorbs fluid isotropically is called a sink and has the same stream function as a source but with a positive sign:

$$\psi_{sink} = -\frac{Q}{2\pi}\theta \tag{2.22}$$

The velocity components share the same form as the source but with opposite sign which is expected.

$$u_{sink} = -\frac{Qx}{2\pi(x^2+y^2)} \tag{2.23}$$

$$v_{sink} = -\frac{Qy}{2\pi(x^2+y^2)} \tag{2.24}$$

Doublets

Here we derive the stream function equation for a doublet. This is where a sink and source coincide at the same point. By placing a sink in a point, a is small distance, (a, 0) with angle θ_2 and a source in (-a,0) with angle θ_1. This gives the stream function using the superposition principle:

$$\psi = -\frac{Q}{2\pi}(\theta_1 - \theta_2) \tag{2.25}$$

Now, take the tangent on each side:

$$\tan\Big(-\frac{2\pi\psi}{Q}\Big) = \tan(\theta_1-\theta_2) = \frac{\tan(\theta_1)-\tan(\theta_2)}{1+\tan(\theta_1)\tan(\theta_2)} \tag{2.26}$$

Further we rewrite $\tan(\theta_1)$ as $\frac{y}{x-a}$ together with $\tan(\theta_1)$ as $\frac{y}{x+a}$ and insert this into eq (2.22).

$$\tan\Big(-\frac{2\pi\psi}{Q}\Big) = \frac{\frac{y}{x-a}-\frac{y}{x+a}}{1+\frac{y^2}{(x^2-a^2)}} \tag{2.27}$$

After cleaning up the equation reads:

$$\tan\left(-\frac{2\pi\psi}{Q}\right) = \frac{2ya}{x^2+y^2-a^2} \tag{2.28}$$

Taking the inverse tangent on both sides of and remembering that a is small distance gives small values in the argument of arctan. For small values of a gives:

$$-\frac{2\pi\psi}{Q} = \arctan\left(\frac{2ya}{x^2+y^2-a^2}\right) \approx \frac{2ya}{x^2+y^2-a^2} \tag{2.29}$$

Solving for the stream function yields:

$$\psi = -\frac{Qay}{\pi(x^2+y^2-a^2)} \tag{2.30}$$

Now, as a tends toward 0, Q will tend to infinity. In this case, let a $->$ 0 and Q $->$ ∞ so that there product is constant. This can be written as:

$$\lim_{\substack{a\to 0\\ Q\to\infty}} -\frac{Qay}{\pi(x^2+y^2-a^2)} = -\frac{ky}{\pi(x^2+y^2)} \tag{2.31}$$

k is the strength of the doublet.
The stream function equation for a doublet reads finally:

$$\psi_{doublet} = -\frac{ky}{\pi(x^2+y^2)} \tag{2.32}$$

The velocity components for a doublet is found from eq (2.5) and eq (2.6):

$$u = \frac{k(y^2-x^2)}{(x^2+y^2)^2} \tag{2.33}$$

$$v = -\frac{kyx}{(x^2+y^2)^2} \tag{2.34}$$

Figures (2.3 - 2.5) visualize different streamline behaviors from different solution of laplace equation (2.8).

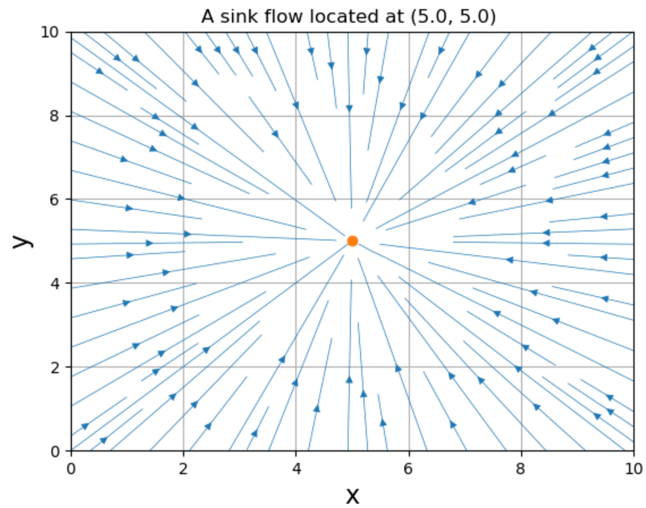

Figure 2.3: A solution of the laplace equation produces streamlines directed radially inward to it's origin (orange point). The velocity decreases inversely proportional from it's distance from the origin. This flow characteristic is called a sink flow. This flow is useful when representing end point of routes as the sink absorbs the fluid.

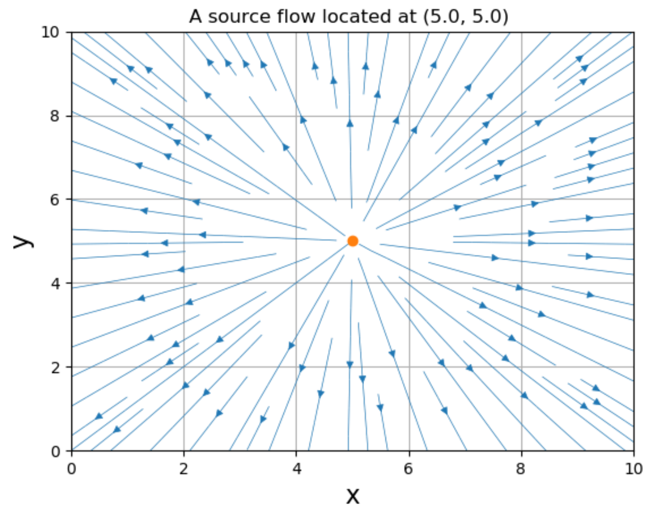

Figure 2.4: This flow characteristic is called a source flow. The velocity is points outward from it's origin (orange point). This flow characteristic emis fluid making it a good representation for initial positions.

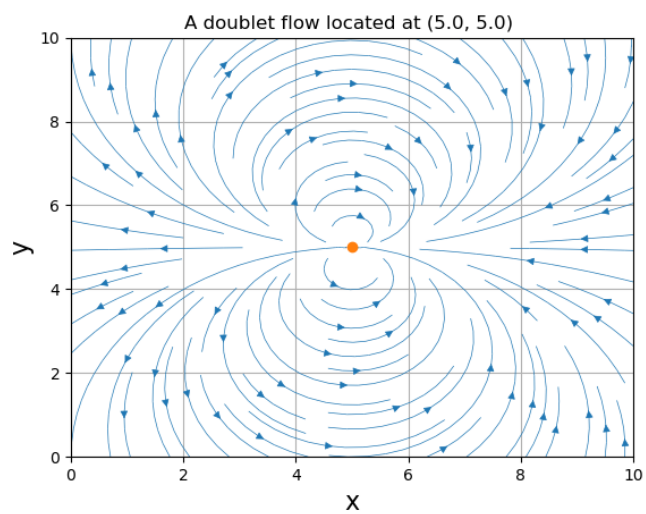

Figure 2.5: A doublet emerges from a sink and source coinciding in the point. A doublet represents objects i.e islands and land. From the figure we see the streamlines flow around the defined point of origin (orange).

A derivation of three (four with uniform) fundamental flow types have been presented. We have shown how we can represent and how the equations emerge from the main equations of Navier-Stokes. These analytic solutions can build flow characteristics by combining the flow and placing them at different points. In order to build complex flows defined the geometry i.e. a numerical approach is needed. This will be the next topic.

Numerical approach

Taking one step further into the simulation of streamline flow, a numerical approach is presented here. The source panel method approximates arbitrary geometric shapes with a set of panels. By using boundary conditions at the surface and placing a small source potential a system of equations is set up and solved. The sections below explain in detail how the source panel method is set up with boundary conditions and geometry , how to calculate the source strengths and how to calculate the velocity flow at every grid cell in the defined area leading us to a flow characteristic.

Boundary conditions

From the latter section, considering an incompressible and irrotational flow produced the Laplace equation. This equation has different solutions leading to various flow characteristics such as uniform, source, sink and doublet flow. It was also found that the superposition principle could be applied to the Laplace equation because of it's linearity property. The superposition principle allows for more realistic flow types than elementary solutions of the Laplace equation. One can simulate objects placed in the flow-field region by restricting the flow at certain points in the region. More commonly known as boundary conditions. Boundary conditions give rise to certain criteria and information needed at crucial points, such as coastlines. The information about the problem at hand is argued by the physical aspect of the scenario. It addresses with what happens physically and converts it into equations which take care of that notion. Solving specific flows for different geometric shapes, boundary conditions are a must.

For a solid- body, a fluid is not able to flow through the body. Mathematically, the normal velocity must then be zero at the surface. In addition, for inviscid fluids the tangential velocity component at the surface of the body is finite. The equations which arise from these boundary conditions can be written mathematically as :

$$\vec{V} \cdot \vec{n} = 0 \tag{2.35}$$

$\vec{V}$ is the flow velocity and $\vec{n}$ is the normal vector pointing out of the body.

$$\vec{V} \cdot \vec{t} = c \tag{2.36}$$

$\vec{V}$ is the flow velocity and $\vec{t}$ is the tangential vector. c indicts a constant value for the tangential velocity component. For flow very far away from the body we assume the velocity to be uniform. This can be written as:

$$u = V_\infty \tag{2.37}$$

u is the velocity in the x-direction and V_∞ is a constant value resulting in a uniform flow in the x-direction.

$$v = 0 \tag{2.38}$$

The y component of the velocity vector is zero. Note, this depends on the problem at hand.

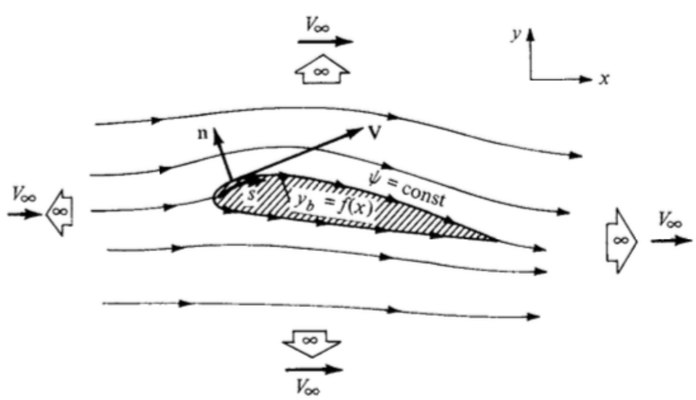

Figure 2.6: The figure shows flow around an airfoil. Together with vector for boundary conditions on solid body. ψ are streamlines. The uniform flow is the same as the equation above i.e. only contribution in the x-direction. Image from Anderson, 2007

Flow around a cylinder

Numerical approach

Navier Stokes' equations are nonlinear, integral equations which do not solve analytically. It is therefore commonly applied two approaches in order to solve Navier Stokes' equations. One, is to simplify the problem at hand to extent that the equations become linear and therefore can be solved analytically. The second is, numerical discipline/approach used to solve a set of equations as analytic solutions do not exist. This allows for a more realistic simulations and solutions. This is more commonly known as Computational Fluid Dynamics (CFD). The nonlinear integral terms are replaced by a discretized approach yielding solutions at specified points (or grid).

Consider placing an object in a flow environment. Calculating the flow at the walls of the object is done by boundary conditions. This grants the possibility of solving individual flow types for bodies of different geometries. When it comes to solving a flow characteristic at a specific point, i.e. at the surface of an object, a numerical routine is needed.

Source panel methods

When solving for the flow around a solid body i.e. the flow around a circular cylinder, the body is predetermined in it's geometric shape. What about shapes that are not circular with arbitrary shape? The same principles of finding the velocity and strengths apply to those objects as well, and it would be complicated to guess what combinations of flows would constitute that specific shape. It would be useful to only need the shape the object in order to solve the flow around it. This is where the source panel method is introduced. The source panel is based on dividing a solid body into into sets of panels of the discretized object. The panels contain a panel strength along it's line. These strengths can be represented by the source type flow presented the section before. There are now an infinite number of strengths along the panel to solve for. These panel source strengths make up a source sheet and can be represented by a function $\lambda(s)$ where s is the distance along the sheet and λ its corresponding strength value at the point s. The units of the strength is square meters per second for the strength quantities for source.sink and doublets. So by considering a infinitesimal line of source strength, $\lambda\ ds$ yield the same dimension. For a point located at a point P a distance r away from the small piece of the panel segment ds an infinitesimally small potential $d\phi$ occurs at the point P obtained at ds. This can be written as:

$$d\phi = \frac{\lambda ds}{2\pi} \ln r \tag{2.39}$$

$r = \sqrt{x^2 + y^2}$ is the distance from ds inducing a dϕ at point p. λ is the source strength. The full source sheet induced at P is found by integrating over the source sheet where the limits are a to b. The velocity potential can further be written as:

$$\phi = \int_a^b \frac{\lambda ds}{2\pi} \ln r \tag{2.40}$$

To make it clear, the integrand contains the potential equation for a source or sink type flow. The integrand can be negative and positive. Earlier, it was stated that the source strength from a panel was $\lambda(s)$ which gave the equation stated above. Now, by assuming that the source strength from one panel is not a function of the distance, but constant, allocating each panel with an unknown source strength coefficient yields the possibility to approximate the source strengths at each panel. Each panel may have different source strengths λ. For n panels n source strengths occur. One for each panel. By combining this with a uniform flow and solving for the source strength λ the desired streamlines is obtained around the object.

The objective is to solve each individual panel strength λ_i, $i = 1, 2, 3, ..., n$. The boundary condition for the normal component states that the velocity is zero in that direction. Furthermore, the boundary condition at each *panel point* allows for a set of equations to be expressed. The notation for a panel j giving rise to a potential at a point p from eq (2.39) above can be written as:

$$\Delta\phi_j = \frac{\lambda_j}{2\pi} \int_j \ln r_{pj} ds_j \tag{2.41}$$

the distance r_{pj} is now $r_{pj} = \sqrt{(x - x_j)^2 + (y - y_j)^2}$. The point p is located at (x, y). Remember that λ_j is constant over the panel j. For all panels producing a potential at point p the equation above is summed over all panels:

$$\phi(p) = \sum_{j=1}^{n} \Delta\phi_j = \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} \int_j \ln r_{pj} ds_j \tag{2.42}$$

Now, move the point p to a panel i and define the point (x_i, y_i) to be the control point of panel i we can express the influence of every panel to the ith panel as:

$$\phi(x_i, y_i) = \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} \int_j \ln r_{ij} ds_j \tag{2.43}$$

With the distance:

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{2.44}$$

The control point is located at the center of each panel. With the location of each panel defined, apply the normal boundary condition at that point. But first, the freestream velocity which was needed to generate a flow around the object is needed to evaluate the correct orientation of the vectors. Let, $\vec{n}_i$ be the normal vector located the control point of panel i. The freestream vector V_∞ can be generated at an angle α relative to the x-axis. This angle to commonly known as the angle of attack. The vector normal to the freestream velocity vector at the control point at the ith panel can be expressed as:

$$V_{\infty,n} = \vec{V_\infty} \cdot \vec{n}_i = V_\infty \cos \beta_i \tag{2.45}$$

β_i is the angle between $\vec{V_\infty}$ and $\vec{n}_i$. Finding the normal velocity component at panel i produced by every panel is:

$$V_n = \frac{\partial}{\partial n_i}[\phi(x_i, y_i)] \tag{2.46}$$

We want component to point outward relative the body. V_n is then positive. The derivative $\frac{\partial}{\partial n_i}$ acts on the distance variable r_{ij}. Note that when j = i the calculation the same panel is done on itself and the distance variable becomes zero. The term is now located in the denominator leading to a singularity. When j = i the contribution has been shown to give $\frac{\lambda_i}{2}$. This gives an update of the potential equation to further be expressed as:

$$V_n = \frac{\lambda_i}{2} + \sum_{j=1,(j \neq i)}^{n} \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial n_i} \ln r_{ij} ds_j \tag{2.47}$$

This the normal vector component of the source panel contribution at panel i. The boundary states:

$$V_{\infty,n} + V_n = 0 \tag{2.48}$$

The flow cannot penetrate the walls of a solid body. The latter equation can be written as:

$$V_\infty \cos \beta_i + \frac{\lambda_i}{2} + \sum_{j=1,(j \neq i)}^{n} \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial n_i} \ln r_{ij} ds_j = 0 \tag{2.49}$$

Substituting the equations V_n and $V_{\infty,n}$ above. This equation has n unknowns. The task now is to gather n equation which in order to solve the source strengths. Solving the source strength coefficients gives rise to the streamlines over the set of panels. This equation can be expressed in matrix form solving the system of equation using numerical methods. This approach makes it possible to control the amount of accuracy needed for the model as the number of panels is used as input to the model.

Tangential velocity calculation

With the source strengths solved. Next, will be to find the tangential velocity component. The equation are similar to the normal velocity component. The only change is the direction in which to take derivatives and find the freestream vector. The vector will be evaluated at the control of each panel as for the normal vector. Tangent to the surface the freestream velocity will be:

$$V_{\infty,t} = V_\infty \sin \beta_i \tag{2.50}$$

In addition, finding the tangential velocity on the panel is found by taking the derivative of eq (2.43) in the tangential direction. This can be expressed as:

$$V_t = \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial t_i} \ln r_{ij} ds_j \tag{2.51}$$

Adding the above equation the final expression for the tangential velocity is:

$$V_i = V_{\infty,t} + V_t = V_\infty \sin \beta_i + \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial t_i} \ln r_{ij} ds_j \tag{2.52}$$

V_i is the sum over the velocity at the control point i. The extra term for when j = i for the normal component does not occur here as that term is zero when j = i. A sanity check when calculating the source strengths and panels is to sum up the product of all source strengths λ and panels lengths. This sum should be zero. If not then the object would be absorbing mass from the flow.

$$\sum_{j=1}^{n} \lambda_j s_j = 0 \tag{2.53}$$

S_j is the jth panel length, λ_j is the jth source strength coefficient.

Thorough derivation of source panel method

From the previous section an overview of the equations for calculating the flow around an arbitrary body was shown. In this section, a more detailed derivation will be done. The integral terms will be calculated together with the geometry of the problem. Finally, this will be set up into a matrix form $Ax = b$ which can be solved thus leading to a solution, namely the source panel method.

Defining geometry

The equations needed for creating the sufficient amount panels is done is here. In addition, some detailed figures will be included for a overview of each variable listed. Each panel can be of different length and therefore each panel needs to be calculated. The starting point needs to be defined and thereafter calculated each panels property in a orderly fashion. This is necessary because the normal vector orientation will be affected by this choice. The suggested method for calculating panels are clockwise. Below, figure 2.7 shows 8 panels approximating a circle with boundary points.

$$dx = (x_2 - x_1) \tag{2.54}$$

$$dy = (y_2 - y_1) \tag{2.55}$$

$$s_1 = \sqrt{dx^2 + dy^2} \tag{2.56}$$

Equations (2.54, 2.55, 2.56) are then generalized for the ith panels and have the notation:

$$dx_i = x_{i+1} - x_i \tag{2.57}$$

$$dy_i = y_{i+1} - y_i \tag{2.58}$$

$$s_i = \sqrt{dx_i^2 + dy_i^2} \tag{2.59}$$

Where $i = 1, 2, 3..., n$ goes up to the number of panels n. Furthermore, from figure 2.9 the control points for each panels is needed. This is where the source strength is set up. This point has the relation:

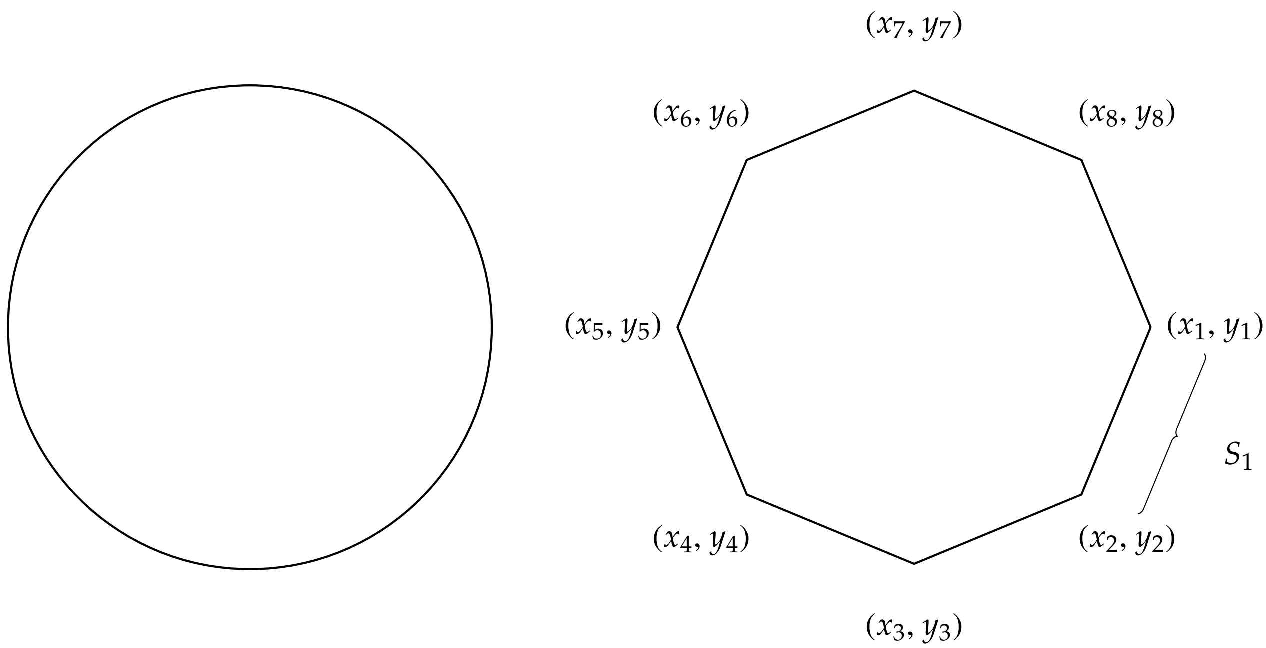

Figure 2.7: Approximating a circle with panels. N panels are created with boundary points representing the end points of each panel (x_b, y_b). From the figure on the right 8 panels s are created with 9 boundary points ((x_1, y_1) are counted twice for the first and last boundary point). More panels can be included for better approximations.

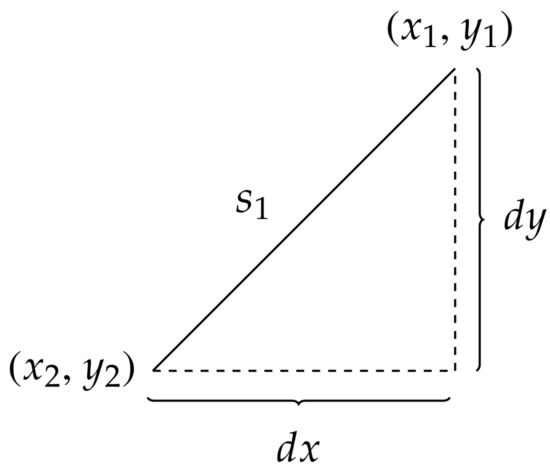

Figure 2.8: A closer view of panel S_1 from figure 2.7. The equations needed for the panel calculation can be seen above.

$$x_{ci} = \frac{x_i + x_{i+1}}{2} \tag{2.60}$$

$$y_{ci} = \frac{y_i + y_{i+1}}{2} \tag{2.61}$$

With the points defined the angle between the panel and x-axis ϕ can be found by:

$$2 = 2 \tag{2.62}$$

$$2 = 21 \tag{2.63}$$

$$\tan \phi_i = \frac{dy}{dx} \tag{2.64}$$

See figure 2.11 for a closer view. Note that the angle ϕ_i does not change for a different placement of the x-axis. Since the boundary points are defined it is convenient to use dy and dx. Furthermore, the normal vector is always perpendicular to the panel. This means that the angle δ_i which is the angle from the x-axis to normal vector and the angle β_i from the free stream vector to normal vector can be expressed as:

$$\delta_i = \phi_i + \frac{\pi}{2} \tag{2.65}$$

$$\beta_i = \delta_i - \alpha \tag{2.66}$$

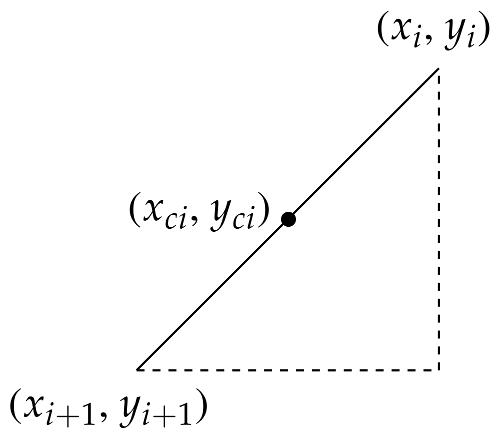

Figure 2.9: The general case for the ith panel with corresponding boundary points and control points.With the boundary points defined, the control points can be calculated by eq's (2.60) and (2.61).

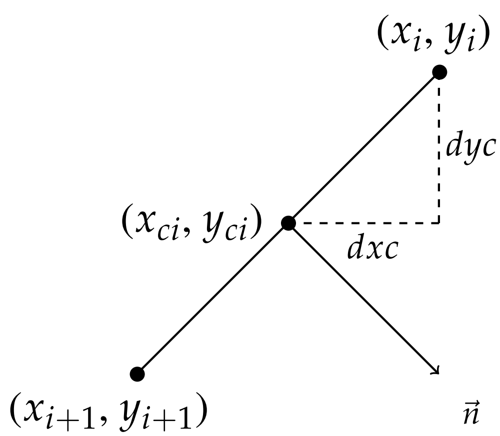

Figure 2.10: In this figure a normal vector is added at the control point together with the change in x and y direction from the control point (dxc and dyc) which is necessary when calculating angles between panels and normal vectors.

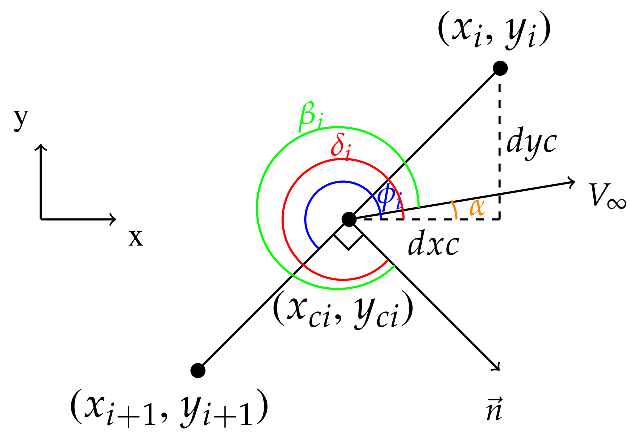

Figure 2.11: This figure shows the angles needed to compute the normal vectors of each panel derived from the boundary and control points.

α is the angle which the free stream vectors occurs. This is constant. The angles are visualized in figure 2.11. The normal vector components can be expressed as:

$$n_{xi} = x_{ci} + s_i \cos \delta_i \tag{2.67}$$

$$n_{yi} = y_{ci} + s_i \sin \delta_i \tag{2.68}$$

$$\vec{n} = n_{xi}\vec{i} + n_{yi}\vec{j} \tag{2.69}$$

Where i and j are unit vectors in x and y - direction respectively.

Building complex flows

With the panels and angles defined from the boundary and control points, the necessary components are thus obtained. This now leads to the ability to build flows which are independent of the analytic potential flow solutions. Recall, the source velocity potential:

$$\phi_1 = \frac{\lambda_1}{2\pi} \ln r_{1p} \tag{2.70}$$

Where λ_1 is the source strength from panel S_1 induced at the point p. $r_{1p} = \sqrt{(x_1 - x_p)^2 + (y_1 - y_p)^2}$. Note that this is the same description as the stream function, but with a velocity potential. With the super position principle we can add together multiple sources. In this case the total amount induced by the panels

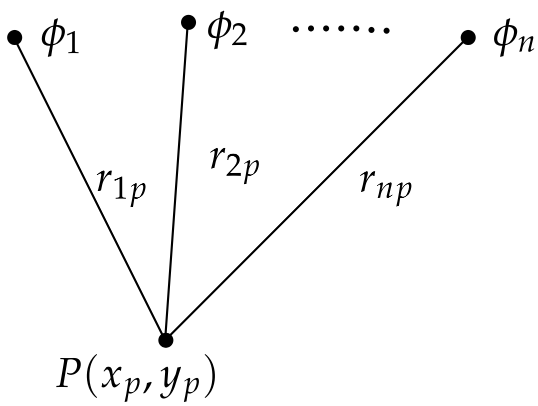

Figure 2.12: n sources ϕ_n induce the potential ϕ_p at a point P.

of the body $\phi_p = \phi_1 + \phi_2 + \phi_3 + ... + \phi_n$. Or expressed in terms of the distance from point P to the panels control point:

$$\phi_p = \frac{\lambda_1}{2\pi} \ln r_{1p} + \frac{\lambda_2}{2\pi} \ln r_{2p} + ... + \frac{\lambda_n}{2\pi} \ln r_{np} \tag{2.71}$$

Which can be written as:

$$\phi_p = \sum_{i=1}^{n} \frac{\lambda_i}{2\pi} \ln r_{ip} \tag{2.72}$$

See figure 2.12 above.

Let's now assume that for a curve with infinite source on it's arc. The expression for calculating the velocity potential at a point is given by the integral:

$$\phi_p = \int_a^b \frac{\lambda(s)}{2\pi} \ln r_p(s) ds \tag{2.73}$$

The integral assumes the source strength over the body' periphery (see figure 2.13) depends on the distance s along the surface S. By breaking the the surface up into straight lines the integral above can be manipulated into the equation:

$$\phi_p = \int_a^b \frac{\lambda(s_{ab})}{2\pi} \ln r_p(s_{ab}) ds_{ab} + \int_b^c \frac{\lambda(s_{bc})}{2\pi} \ln r_p(s_{bc}) ds_{bc} \tag{2.74}$$

Figure 2.14 depicts this integral case. The curve now consists of two lines where each line contains it's own source strength $\lambda(s_{ab}$ and $\lambda(s_{bc})$. The number of panels can be adjusted. For n panels the integral reads:

$$\phi_p = \sum_{j=1}^{n} \int_j \frac{\lambda(s_j)}{2\pi} \ln r_{pj}(s_j) ds_j \tag{2.75}$$

Now assume the source strength $\lambda(s)$ is constant along each panel. This leads to the source strength at panel j $\lambda(s_j) = \lambda_j$ and the distance $r_{pj}(s_j) = r_{pj}$. With this we write the integral equation above as:

$$\phi_p = \sum_{j=1}^{n} \lambda_j \int_j \frac{\ln r_{pj}}{2\pi} ds_j \tag{2.76}$$

See figure 2.14 and 2.15.

The source panel strengths needs to be solved. This constitutes the main part of the source panel method. A quick summary of the main features of this section. The goal is to build a flow around a arbitrary object. We found that the object induces a small potential ϕ at a point P in space. Further, a derivation of how to calculate the total potential at point P. With this, some simplifications have been considered; breaking a curve of arbitrary shape into n panels consisting of straight lines, assuming the source strength parameter λ is constant along a panel j. Combing this with a freestream velocity, we have all the contributions needed in order to calculate the flow around an object of arbitrary shape. In the next section we dive further into the details how to set up the equations needed for solving the panel source strengths.

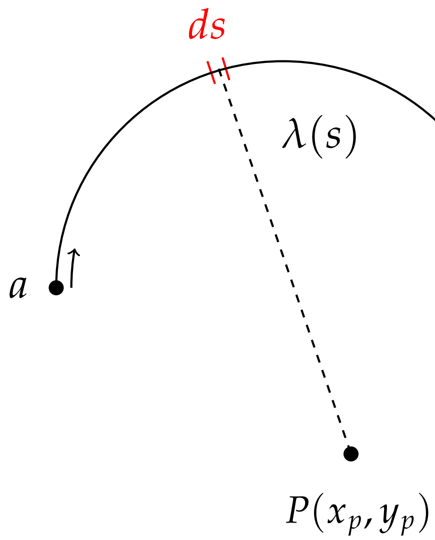

Figure 2.13: A body with a source strength $\lambda(s)$ induces a potential at point P. Integrating from a to b grants thew total contribution of the potential along the periphery of the surface. The arrow indicates the integration direction.

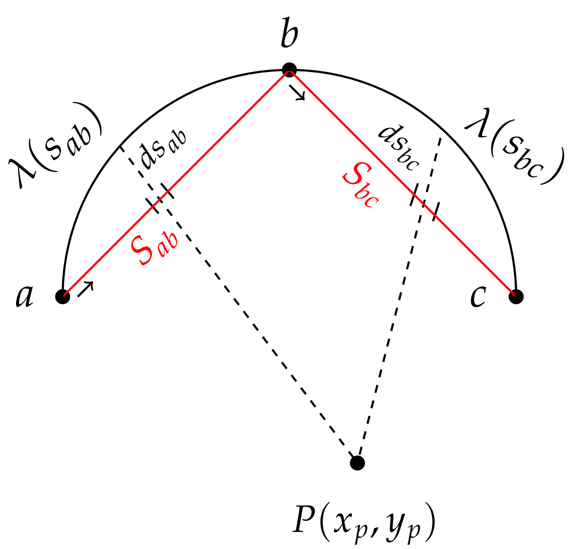

Figure 2.14: Splitting the body into panels thus adding up the contribution from the panel approximation.

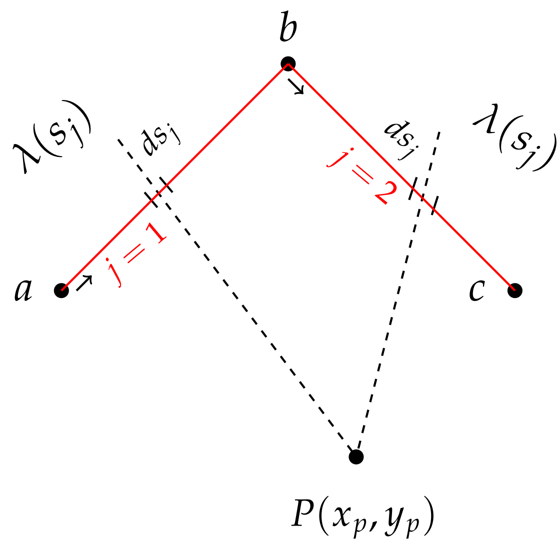

Figure 2.15: This figure shows the angles needed to compute the normal vectors of each panel derived from the boundary and control points.

Calculating flow around an arbitrary shaped object

An insight into how to solve set up system of equations for source strengths is done. The condition that the normal component at the surface of the object is zero will be used to set up the system of equations. Further to calculate the velocities the tangential component will be used. Finally the potential can be solved for at every grid cell for the defined area of interest.

$$\phi_p = V_\infty \cos(\alpha)x + V_\infty \sin(\alpha)y + \sum_{j=1}^{n} \lambda_j \int_j \frac{\ln r_{pj}}{2\pi} ds_j \tag{2.77}$$

The two first terms are the freestream velocity vectors in x and y direction respectively. Taking the derivative of the potential ϕ_p in the x and y direction the velocities can be found:

$$V_{x,p} = \frac{\partial \phi_p}{\partial x} \tag{2.78}$$

$$V_{y,p} = \frac{\partial \phi_p}{\partial y} \tag{2.79}$$

The normal and tangential velocity deal with the physical part of the solution. They can be expressed as:

$$\vec{V} \cdot \vec{n} = 0 \tag{2.80}$$

No fluid can pass through the object

$$\vec{V} \cdot \vec{t} = C \tag{2.81}$$

C is a constant. The flow runs along the surface of the object. By placing the source strengths λ at the control points of each panel the can be done to the normal vector which here we will solve for the source strengths. The velocity potential at panel i can be written using eq (2.77):

$$\phi_p = V_\infty \cos(\alpha)x + V_\infty \sin(\alpha)y + \sum_{j=1}^{n} \lambda_j \int_j \frac{\ln r_{ij}}{2\pi} ds_j \tag{2.82}$$

Where we sum up the contribution of every source strength λ_j induced at panel i. i and $j = 1, 2, 3..., n$ run up to the number of panels. The normal and tangential velocity at panel i:

$$V_{n,i}^{Total} = \frac{\partial \phi_i}{\partial n_i} = V_\infty \cos(\alpha)\frac{\partial x_i}{\partial n_i} + V_\infty \sin(\alpha)\frac{\partial y_i}{\partial n_i} + \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial n_i} \ln r_{ij} ds_j = 0 \tag{2.83}$$

$$V_{t,i}^{Total} = \frac{\partial \phi_i}{\partial t_i} = V_\infty \cos(\alpha)\frac{\partial x_i}{\partial t_i} + V_\infty \sin(\alpha)\frac{\partial y_i}{\partial t_i} + \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial t_i} \ln r_{ij} ds_j \tag{2.84}$$

Let's start by looking at the first terms in both the normal and tangential direction. The normal component' terms:

$$V_{n,i}^{freestream} = \frac{\partial \phi_i}{\partial n_i} = V_\infty \cos(\alpha)\frac{\partial x_i}{\partial n_i} + V_\infty \sin(\alpha)\frac{\partial y_i}{\partial n_i} \tag{2.85}$$

The tangential component:

$$V_{t,i}^{freestream} = \frac{\partial \phi_i}{\partial t_i} = V_\infty \cos(\alpha)\frac{\partial x_i}{\partial t_i} + V_\infty \sin(\alpha)\frac{\partial y_i}{\partial t_i} \tag{2.86}$$

The change in x-direction with respect to the normal derivatives can be written as:

$$\frac{\partial x_i}{\partial n_i} = \cos(\delta_i) \quad , \quad \frac{\partial y_i}{\partial n_i} = \sin(\delta_i) \tag{2.87}$$

and the same with tangential:

$$\frac{\partial x_i}{\partial t_i} = \cos(\delta_i - \frac{\pi}{2}) \quad , \quad \frac{\partial y_i}{\partial t_i} = \sin(\delta_i - \frac{\pi}{2}) \tag{2.88}$$

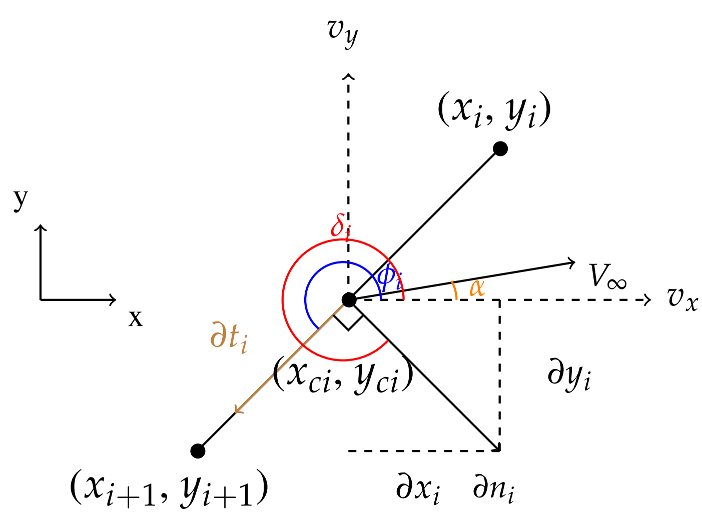

Figure 2.16: Panel i has both tangential and normal components. These two quantities can be expressed with trigonometric functions.

The setup can seen in figure 2.16. Inserting eq (2.88) into eq (2.87) and using the trig identity $\cos(a \pm b) = \cos(a)\cos(b) \mp \sin(a)\sin(b)$ we end up with simpler term for the freestream velocity:

$$V_{n,i}^{freestream} = V_\infty \cos(\beta_i) \tag{2.89}$$

β_i is the angle from angle of attack α to δ_i $\beta_i = \delta_i - \alpha$. Can be seen in figure 2.11. Putting eq (2.89) into eq (2.83):

$$V_{n,i}^{Total} = \frac{\partial \phi_i}{\partial n_i} = V_\infty \cos(\beta_i) + \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial n_i} \ln r_{ij} ds_j = 0 \tag{2.90}$$

The same can be done for the tangential term:

$$V_{t,i}^{Total} = \frac{\partial \phi_i}{\partial t_i} = V_\infty \sin(\beta_i) + \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial t_i} \ln r_{ij} ds_j \tag{2.91}$$

$i = 1, 2, ..., n$

Normal vector geometry integral

In this section we look at the integral part of the total normal velocity induced at panel i. Recall:

$$V_{n,i}^{Total} = \frac{\partial \phi_i}{\partial n_i} = V_\infty \cos(\beta_i) + \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial n_i} \ln r_{ij} ds_j = 0 \tag{2.92}$$

Let's put our attention at the integral that needs to solved and express the terms with the use of variables mentioned above. This will be done more thoroughly below. By only looking at the integral part, we write the integral as:

$$I_{ij} = \int_j \frac{\partial}{\partial n_i} \ln r_{ij} ds_j \tag{2.93}$$

The logarithm term can be manipulated by the chain rule $\frac{d\ln(f(x))}{dx} = \frac{1}{f(x)}\frac{df(x)}{dx}$. Applying this to eq (2.93) gives:

$$I_{ij} = \int_j \frac{1}{r_{ij}} \frac{\partial r_{ij}}{\partial n_i} ds_j \tag{2.94}$$

Also, recall the distance r_{ij} which is the distance between panel j to panel i:

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{2.95}$$

Taking derivative of the distance w.r.t to normal component yields the equation:

$$\frac{\partial r_{ij}}{\partial n_i} = \frac{(x_i - x_j)\frac{\partial x_i}{\partial n_i} + (y_i - y_j)\frac{\partial y_i}{\partial n_i}}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}} \tag{2.96}$$

From figure 11 and the equations in (2.87) we recognize the derivatives in the latter equation as $\cos(\delta_i) = \frac{\partial x_i}{\partial n_i}$ and $\sin(\delta_i) = \frac{\partial y_i}{\partial n_i}$. Substituting eq's(2.96):

$$\frac{\partial r_{ij}}{\partial n_i} = \frac{(x_i - x_j)\cos(\delta_i) + (y_i - y_j)\sin(\delta_i)}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}} \tag{2.97}$$

Gathering eq (2.97), eq (2.95) and inserting this into eq (2.94) the equation now reads:

$$I_{ij} = \int_j \frac{(x_i - x_j)\cos(\delta_i) + (y_i - y_j)\sin(\delta_i)}{(x_i - x_j)^2 + (y_i - y_j)^2} ds_j \tag{2.98}$$

Further, we want to express the integrand in terms of s_j as this is the integration variable. We find that x_j and y_j

$$x_j = X_j + s_j \cos(\phi_j) \ , \quad y_j = Y_j + s_j \sin(\phi_j) \tag{2.99}$$

Here we represent the point on the jth panel as a parameter representation. X_j and Y_j are starting points for the panel, and s_j is the length of the jth panel. ϕ_j is angle between the x-axis and the panel j. The angle ϕ_j can be derived from the identity $\cos(\delta_i) = -\sin(\phi_i)$ and $\sin(\delta_i) = \cos(\phi_i)$ where the relation of δ_i and ϕ_i are $\delta_i = \phi_i + \pi/2$ as seen in figure 11. Inserting this into I_{ij} the integral now reads:

$$I_{ij} = \int_j \frac{(x_i - x_j)(-\sin(\phi_i)) + (y_i - y_j)\cos(\phi_i)}{(x_i - x_j)^2 + (y_i - y_j)^2} ds_j \tag{2.100}$$

For the sake tidyness, the numerator will be considered first. Thereafter, the denominator will be considered. From the above equation the numerator reads:

$$(x_i - X_j - s_j \cos(\phi_j)(-\sin(\phi_i)) + (y_i - Y_j - s_j \sin(\phi_i)\cos(\phi_i)) \tag{2.101}$$

after inserting eq (2.99). Further we arrange the terms such that:

$$(X_j - x_i)\sin(\phi_i) + (y_i - Y_j)\cos(\phi_i) + s_j(\cos(\phi_j)\sin(\phi_i) - \sin(\phi_j)\sin(\phi_i)) \tag{2.102}$$

and use the trig identity $\sin a - b = \sin a \cos b - \cos a \sin b$ on the last term:

$$(X_j - x_i)\sin(\phi_i) + (y_i - Y_j)\cos(\phi_i) + s_j \sin(\phi_i - \phi_j) \tag{2.103}$$

At last we write the numerator in terms of s_j as this is the variable which will be integrated over:

$$s_j A + D \tag{2.104}$$

where $A = \sin(\phi_i - \phi_j)$ $D = (X_j - x_i)\sin(\phi_i) + (y_i - Y_j)\cos(\phi_i)$. The numerator is now simplified and can be integrated.

Moving on the denominator. Similar approach will used as with the numerator. From eq (2.98) write out the terms and insert eq's (2.99):

$$x_i^2 - 2x_i(X_j + s_j \cos\phi_j) + (X_j + s_j \cos\phi_j)^2 + y_i^2 - 2y_i(Y_j + s_j \sin\phi_j) + (Y_j + s_j \sin(\phi_j)^2 \tag{2.105}$$

Now gather all s_j terms with same polynomial degree. This leads to:

$$s_j^2 + 2s_j B + C \tag{2.106}$$

where, $B = X_j \cos\phi_j + Y_j \sin\phi_j - x_i \cos\phi_j - y_i \sin\phi_j$ and $C = (x_i - X_j)^2 + (y_i - Y_j)^2$. The end term with the numerator and denominator put back into eq(2.93) we finally end up with:

$$I_{ij} = \int_0^{S_j} \frac{s_j A + D}{s_j^2 + 2 s_j B + C} ds_j \tag{2.107}$$

Before integrating we need to complete the square in the denominator.

$$s_j^2 = 2Bs_j + C = s_j^2 + 2Bs_j + B^2 + C - B^2 = (s_j + B)^2 + E^2 \tag{2.108}$$

where $E = \sqrt{C - B^2}$. Now the integral reads:

$$I_{ij} = \int_0^{S_j} \frac{s_j A + D}{(s_j + B)^2 + E^2} ds_j \tag{2.109}$$

This is now on the form that can be integrated. Set $u = s_j + B$.

$$I_{ij} = \int_0^{S_j} \frac{(u - B)A + D}{u^2 + E^2} du \tag{2.110}$$

The integral will be split up into two integrals by separating the numerator.

$$I_{ij} = A \int_0^{S_j} \frac{u}{u^2 + E^2} du + (D - AB) \int_0^{S_j} \frac{1}{u^2 + E^2} du \tag{2.111}$$

Using substitution $\gamma = u^2 + E^2$ on the first integral again we obtain the solution as:

$$A \int_0^{S_j} \frac{u}{u^2 + E^2} du = \frac{A}{2} \int_0^{S_j} \frac{d\gamma}{\gamma} = \frac{A}{2} [\ln \gamma]_0^{s_j} \tag{2.112}$$

Now substitute u and γ

$$\frac{A}{2} \Big[\ln (S_j + B^2 + E^2) - \ln (B^2 + E^2) \Big] \tag{2.113}$$

$$\frac{A}{2} \ln \left[\frac{(S_j + B)^2 + E^2}{B^2 + E^2} \right] = \frac{A}{2} \ln \left[\frac{S_j^2 + 2 S_j B + B^2 + E^2}{B^2 + E^2} \right] \tag{2.114}$$

With the final touch of using $E^2 = C - B^2$ we end up with:

$$\frac{A}{2} \ln \left[\frac{S_j + 2 S_j B + C}{C} \right] \tag{2.115}$$

The second integral in eq (2.111) has the solution:

$$(D - AB) \int_0^{S_j} \frac{1}{u^2 + E^2} du = \frac{(D - AB)}{E} \arctan \left[\frac{u}{E} \right] \tag{2.116}$$

As with the first integral we substitute back and end up with:

$$\frac{D - AB}{E} \left(\arctan \left[\frac{S_j + B}{E} \right] - \arctan \left[\frac{B}{E} \right] \right) \tag{2.117}$$

Let's sum up the most important parts of this integration below. The integration gave the solution:

$$I_{ij} = \frac{A}{2} \ln \left[\frac{S_j + 2S_j B + C}{C} \right] + \frac{D - AB}{E} \left(\arctan \left[\frac{S_j + B}{E} \right] - \arctan \left[\frac{B}{E} \right] \right) \tag{2.118}$$

Where the terms A, B, C, D and E are:

$$\begin{aligned} A &= \sin(\phi_i - \phi_j) \\ B &= (X_j - x_i) \cos \phi_j + (Y_j - y_i) \sin \phi_j \\ C &= (x_i - X_j)^2 + (y_i - Y_j)^2 \\ D &= (X_j - x_i) \sin \phi_i + (y_i - Y_j) \cos \phi_i \\ E &= \sqrt{C - B^2} \end{aligned}$$

Finally after an amount of mathematical operations the solution I_{ij} is found which means the normal velocity component can be computed by:

$$V_{n,i} = V_\infty \cos \beta_i + \sum_{j=1}^{N} \frac{\lambda_j}{2\pi} I_{ij} \tag{2.119}$$

This sums up the normal vector computations. Before setting up the system of equations we need to solve the integral in the tangential velocity vector eq (2.91). The same approach will be applied as for the normal velocity vector.

Tangential vector geometry integral

Here we tackle the tangential integral part of eq (2.91) containing the positional derivative w.r.t. the tangential component $\int_j \frac{\partial}{\partial t_i} \ln r_{ij} ds_j$. The approach will be similar to the normal component integral above, but with different angles. Recall $\delta_i = \phi_i + \frac{\pi}{2}$. From figure 11 δ_i was the angle used in the normal geometry integral. Since the tangential and normal component are orthogonal to each other ϕ_i will be the angle used here.

$$J_{ij} = \int_j \frac{\partial}{\partial t_i} \ln r_{ij} ds_j \tag{2.120}$$

The logarithm term can be manipulated by the formula $\frac{d\ln(f(x))}{dx} = \frac{1}{f(x)}\frac{df(x)}{dx}$. Applying this to eq (2.121) gives:

$$J_{ij} = \int_j \frac{1}{r_{ij}} \frac{\partial r_{ij}}{\partial t_i} ds_j \tag{2.121}$$

Also, recall the distance r_{ij} which is the distance between panel j to panel i:

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{2.122}$$

Taking derivative of the distance w.r.t to normal component yields the equation:

$$\frac{\partial r_{ij}}{\partial t_i} = \frac{(x_i - x_j)\frac{\partial x_i}{\partial t_i} + (y_i - y_j)\frac{\partial y_i}{\partial t_i}}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}} \tag{2.123}$$

We recognize the derivatives as $\cos(\phi_i) = \frac{\partial x_i}{\partial t_i}$ and $\sin(\phi_i) = \frac{\partial y_i}{\partial t_i}$. See figure 11. Substituting the derivatives into eq (2.123) gives:

$$\frac{\partial r_{ij}}{\partial t_i} = \frac{(x_i - x_j)\cos(\phi_i) + (y_i - y_j)\sin(\phi_i)}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}} \tag{2.124}$$

Gathering eq (2.124), eq (2.122) and inserting this into eq (2.121) the equation now reads:

$$J_{ij} = \int_j \frac{(x_i - x_j)\cos(\phi_i) + (y_i - y_j)\sin(\phi_i)}{(x_i - x_j)^2 + (y_i - y_j)^2} ds_j \tag{2.125}$$

Recall eq (2.99). Express x_j and y_j in terms of s_j:

$$x_j = X_j + s_j\cos(\phi_j)\ , \quad y_j = Y_j + s_j\sin(\phi_j) \tag{2.126}$$

The denominator in the tangential integral is the same as in the normal integral $(s_j + B)^2 + E^2$, $E = \sqrt{C - B^2}$. The numerator is a bit different. Only looking at the numerator and substituting eq (2.126) into the numerator of eq (2.125):

$$(x_i - X_j - s_j \cos(\phi_j)) \cos(\phi_i) + (y_i - Y_j - s_j \sin(\phi_j)) \sin(\phi_i) \tag{2.127}$$

Writing out the terms in the latter equation and arrange:

$$(x_i - X_j)\cos(\phi_i) + (y_i - Y_j)\sin(\phi_i) - s_j(\cos(\phi_j)\cos(\phi_i) + \sin(\phi_j)\sin(\phi_i)) \tag{2.128}$$

This can be simplified using $\cos(a - b) = \cos a \cos b + \sin a \sin b$:

$$(x_i - X_j)\cos(\phi_i) + (y_i - Y_j)\sin(\phi_i) - s_j \cos(\phi_j - \phi_i) \tag{2.129}$$

The numerator reads:

$$s_j A_t + D_t \tag{2.130}$$

Where $C_t = -\cos(\phi_j - \phi_i)$ and $D_t = (x_i - X_j)\cos(\phi_i) + (y_i - Y_j)\sin(\phi_i)$ The J_{ij} integral now reads:

$$J_{ij} = \int_0^{S_j} \frac{s_j C_t + D_t}{s_j^2 + 2s_j B + C} ds_j \tag{2.131}$$

This can be integrated with the same integral methods as the normal integral. Finally we end up with:

$$J_{ij} = \frac{C_t}{2} \ln\left[\frac{S_j + 2S_j B + C}{C}\right] + \frac{D_t - C_t B}{E}\left(\arctan\left[\frac{S_j + B}{E}\right] - \arctan\left[\frac{B}{E}\right]\right) \tag{2.132}$$

Where the terms C_t, B, C, D_t, and E are:

$$\begin{gathered} C_t = -\cos(\phi_i - \phi_j) \\ B = (X_j - x_i)\cos\phi_j + (Y_j - y_i)\sin\phi_j \\ C = (x_i - X_j)^2 + (y_i - Y_j)^2 \\ D_t = (x_i - X_j)\cos\phi_i + (y_i - Y_j)\sin\phi_i \\ E = \sqrt{C - B^2} \end{gathered}$$

The normal and tangential geometry integral have been expressed in terms of panel s_j thus making it possible to integrate and finding the contributions of every panel λ_j induced onto panel i. Next we set up these equations into a matrix form.

Solving the system of equations

With the I_{ij} integral solved, the next part is to solve for source strength at each panel. Recall the normal component being equal to zero at every control point $x_{ci}, y_{ci}, i = 1, 2, 3..., n$.

$$V_{n,i}^{Total} = V_\infty \cos(\beta_i) + \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} I_{ij} = 0 \tag{2.133}$$

$$\sum_{j=1}^{n} \frac{\lambda_j}{2\pi} I_{ij} = -V_\infty \cos(\beta_i) \tag{2.134}$$

When j = i the I_{ij} has been found to be $\frac{\lambda}{2}$. The equations above now reads:

$$\frac{\lambda_i}{2} + \sum_{\substack{j \neq i \\ j=1}}^{n} \frac{\lambda_j}{2\pi} I_{ij} = -V_\infty \cos(\beta_i) \tag{2.135}$$

Eq (2.133) can further be inserting into a matrix representing a set of linear equations.

$$\begin{bmatrix} \pi & I_{12} & I_{13} & \dots & I_{1n} \\ I_{21} & \pi & \dots & \dots & I_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ I_{n1} & I_{n2} & \dots & \dots & \pi \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \vdots \\ \lambda_n \end{bmatrix} = -2\pi V_\infty \begin{bmatrix} \cos\beta_1 \\ \cos\beta_2 \\ \cos\beta_3 \\ \vdots \\ \cos\beta_n \end{bmatrix} \tag{2.136}$$

Multiplying the matrix equation with 2π simplifies the equation. The set of equation in matrix form grants the source strengths of all n panels which is solved numerically. This approximation can be adjusted by increasing the amounts of panels making the result more accurate. The source strength distribution which is now solved makes it possible to solve for the remaining flow in the defined area.

Streamline geometry integral

There is clearly some repetition in these sections, but the main idea is to make it clear which equations are needed and in what order they must be solved for in order to be able in solving the streamlines around a solid body. Here, with the sources strengths found the velocities at every grid cell can be calculated.

Using eq (2.77) which was the velocity potential induced at a point p and take the derivatives in x and y direction the velocity component are found eq (2.78) and eq (2.79), respectively. The integrand is similar to the ones above and is carried out in the same manner. Note, the only change is the variable in which to take the derivative. The integral has the solution:

$$V_{x,p} = V_\infty \cos(\alpha) + \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} \left[\frac{-\cos\phi_j}{2} \ln\left[\frac{S_j + 2S_jB + C}{C} \right] + \frac{A_{xp}}{E} \Bigg(\arctan\left[\frac{S_j + B}{E} \right] - \arctan \Bigg[\tag{2.137}$$

where $A_{xp} = x_p - X_j + \cos\phi_j B$

$$V_{y,p} = V_\infty \sin(\alpha) + \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} \left[\frac{-\sin\phi_j}{2} \ln\left[\frac{S_j + 2S_jB + C}{C} \right] + \frac{A_{yp}}{E} \Bigg(\arctan\left[\frac{S_j + B}{E} \right] - \arctan \Bigg[\tag{2.138}$$

where $A_{yp} = y_p - Y_j + \sin\phi_j B$.

The integrals are now solved leading to the calculation thus a solution for finding the streamlines around a solid body with arbitrary shape. A summary of the main concepts and motivation for source panel method is mentioned next. Analytic solutions of the Laplace equation gives rise to basic flow characteristics. These solution come to short when wanting to find flows around arbitrary bodies. This motivates the source panel method. The requirements for the source panel method is knowing the shape of the body. The body is then approximated by a set of panels which can be varied for more accurate results. At the center of each panel a small source strength flow is placed. The velocity components at the center of the panel are set with boundary conditions on the normal and tangential velocity. The boundary conditions gives rise to two geometric integrals where the source strength values are needed. The normal geometric is first solved finding the source strength values at the center of each panel. The tangential can then be used to find the flows at the center. At last, the contribution of every strength value from each panel give rise to the remaining velocities to be solved in the area.

Chapter 3

AIS-data and map projection

3.1 AIS-data

A concise introduction of the data used in the thesis is presented. Also, a motivation as to why AIS data was introduced in the first place is mentioned.

Autonomous Identification System data, for short AIS-data, is a self reporting system installed on maritime vessels. It sends out information about a vessels characteristics in real time. Some characteristics are: a vessels position, speed over ground (SOG), course over ground (COG) and vessel identification. The receiver on the other end, typically other ships within a vicinity or vessel traffic services (VTS), use the information for safety measures i.e. avoiding collisions.
The development of AIS-data come after the environmental catastrophe of the Exxon Valdez oil spillage in 1989 (ref: marineinsight, 2019). In 2004 the International Maritime Organization (IMO) made it mandatory for all vessels with a weight over 300 gross tons to install a AIS transceiver (ref: IMO, 2020). The huge amounts of data being logged has made analyzing AIS-data a useful tool within the maritime industry. Being able to predict arrival and waiting times at ports, turnaround times and make accurate decisions in advance regarding risk assessment is a benefit for the parties involved.
Managing the data are important aspects as to how it is going to be used. It is also crucial to keep in mind that in most cases when operating with data-dependent models is being aware of its limitations ref!!!.
In this thesis, an extraction within a specified area will be made of the AIS-data. The data is obtained from Det Norske Veritas (DNV). The AIS-data will help gain

insight and perspective on the route done by vessels as this is actual route done by vessels. As AIS-data is installed by many different vessel types, an interesting view is to view what vessel types take which route depending on the environment. This could help with significance of the fluid mechanical model comparing routes.
References in chapter:
https://www.marineinsight.com/maritime-history/the-complete-story-of-the-exxon-valdez-oil-spill/
http://www.imo.org/en/OurWork/safety/navigation/pages/ais.aspx
https://www.portvision.com/news-events/press-releases-news/how-to-use-ais-data-to-your-advantage

3.1.1 Preprocessing AIS-data

AIS-data provides several variables at a given timestamp. These variables are:

- International Maritime Organization (IMO) number: is a identification number for vessels and is unique for every vessel.

- Longitude and latitude: make up vessel positions in a geographic coordinate system. Their units are angles and make up unique points on a sphere. In a cartesian coordinate system the longitude and latitude would represent x and y coordinates respectively.

- Heading: is based on directions on a compass. Used for finding vessels' course measured in angles where 0°and 360 °is true north, 90 °true east, 180 °true south and 270 °true west.

- Speed Over Ground (SOG): is a vessel speed in magnitude relative to the earth's surface. SOG is measure in knots or(1.852 km/h).

- Ship type: specifies the ships characteristic e.g. cargo ship, passenger ship, tug boat, tank ships, coast guard ship, pleasure craft, high-speed craft, towing vessel, fishing boats, sailing vessels, ekranoplans. If the vessel is not of this "other" is registered.

- Length: the vessels maximum length.

- Time stamp position: is the time when data was registered. Measured with an accuracy up to seconds.

Data structure

The key elements in a tidy data set is being consistent, easy to access and work with and exploit. Another element to take into consideration is how the computer will be able to extract values. This will help when it comes to analyzing the data (Wickham 2014).

Each row represent a observation and each column represent a variable. This is the preferred method how to organize data according to(Wickham 2014). Below is an image of how the AIS-data is organized from file:

	IMONumber	TimestampPosition	Longitude	Latitude	Heading	SpeedOverGround	ShipType	Length	unix
1	9536521	2018-06-04 00:00:04	10.270005	63.47954	271	8.3	cargo_ships	89	1528070404
2	9536521	2018-06-04 00:10:05	10.21974	63.4794083333333	268	8.1	cargo_ships	89	1528071005
3	9536521	2018-06-04 00:20:06	10.1665833333333	63.47936	268	9	cargo_ships	89	1528071606
4	9536521	2018-06-04 00:30:14	10.108645	63.4789383333333	269	9.3	cargo_ships	89	1528072214
5	9536521	2018-06-04 00:40:24	10.05038	63.4776783333333	274	9.1	cargo_ships	89	1528072824
6	9536521	2018-06-04 00:50:24	9.99481333333333	63.4795516666667	289	8.9	cargo_ships	89	1528073424
7	9536521	2018-06-04 01:00:04	9.94693	63.489025	307	8.2	cargo_ships	89	1528074004
8	9536521	2018-06-04 01:10:06	9.91273833333333	63.5058183333333	325	8.4	cargo_ships	89	1528074606

Figure 3.1: The first 8 rows of the data set is shown. Each observation is a row number with columns representing the vessel' characteristic/information at a given time.

As seen from figure (3.1) the structure of the data is up to standard with the guide lines from (Wickham 2014). Next, we calculate the velocity components of each observation by using 2 columns from the data set.

Calculating velocities

Velocities are useful units to calculate. This grants the possibility to visualize a vessels direction and heading. Here, we will see which vessels travel to and from Trondheim. Vessels may take detour, traveling through Trondheim to then converge to Trondheim after a while. From figure (3.1) the velocities can be found by using columns: heading and speedoverground. The velocities can be found with these two equations:

$$v_{lon} = SOG \cdot \sin(heading) \tag{3.1}$$

$$v_{lat} = SOG \cdot \cos(heading) \tag{3.2}$$

v_{lon} and v_{lat} are the velocity components in the corresponding longitude and latitude direction, respectively. SOG (Speed Over Ground) is the vessels magnitude of velocity and heading is the vessels angle in degrees.

Route extraction

Visualizing the data can be done in many ways. Often as the data set is big, plotting every data point can be messy and misleading. Here, some information about how one route emerges and what classifies a route is of interest. We want to see where each vessel travels and which path each individual vessel takes. By looking at the timestamp column in the data set, there is often a trend of logged data from a certain time interval. This could indicate the start and end of a route. Keeping in mind that it could also mean some else e.g. turning of the tracking system, malfunction in the power etc. More details on this in the implementation chapter.

A check can be done with the velocities v_{lon}, v_{lat}. If the direction for each time series has the same direction they follow the same route.

3.2 Maps and coordiante systems

The knowledge of the earth's shape has been known by cartographers for the past 2000 years (Snyder 1987). When traveling far it was necessary to know where and distance of travel. How to measure distances on the globe has been needed for many years when people needed to navigate from one place to the other.

This is mentioned here, as the data we will be working with are registered as longitude and latitude coordinates. A short motivation why these quantities are used on maps will be mentioned in the coming sections. Additionally, how convert a spherical surface onto a rectangular surface we be focus in the following section.

3.2.1 Geographic coordinate system

A coordinate system is useful tool when working with positional data. This gives insight distances in multiple dimensions and also from the use of mathematics grants the possibility of calculating positions from a given reference point or relative to given points.

When determining positions on a spherical shaped object e.g. the earth, one must take into account the curvature and how distances are measured. The measures of longitude and latitude have been implemented and defined in such a way to make a structured way to split up the globe. These are angular measurements. There are several ways of defining latitude and longitude: geocentric , astronomical and geographic. The latter is most widely used.

Latitude, also known parallels are measured from north to south direction with an interval of 180 °. Equator is located at 0 °. The earth is not a perfect sphere. This affects the relative lengths of latitudes. 1 °of latitude amounts to 110.567km at Equator and 111.699 km at the poles (Encyclopedia britannica 2020).

Longitude, also known as meridians are measured from west to east with an interval of 180 °. The prime meridian, which is where 0 °is defined, runs through Greenwich, London The relative distances are equal for 1 °of longitude and equals 111.32 km (Encyclopedia britannica 2020). Longitude and latitude coordinates make up a geographic coordinate system. Positional information can be determined in terms of the prime meridian and the Equator. See figure below:

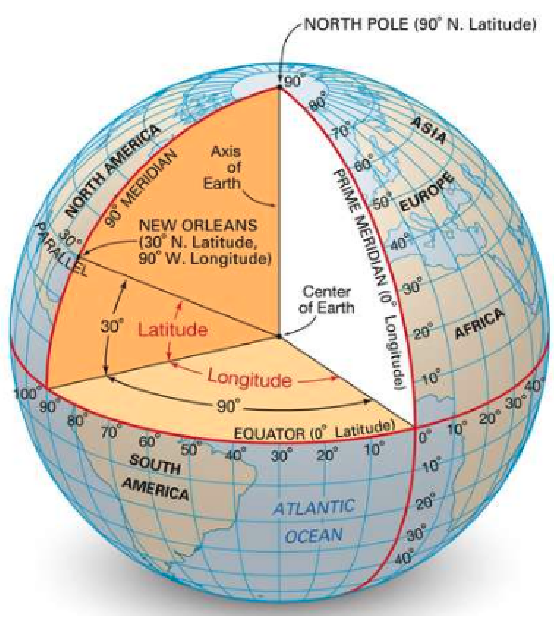

Figure 3.2: Here we see how each angle og longitude and latitude is defined from the earth's center. Figure gathered from (Encyclopedia britannica 2020).

3.2.2 Map selection

The data used in the thesis are logged from the surface of a sphere i.e. the earth. When it comes time to visualize the data, this is done on flat surface. Some background is needed in order to find the appropriate representation. In this section, an introduction to maps and the different ways to represent maps in two dimensions are presented. As we know the earth has a spherical shape, but in geography class the earth was usually shown to us on a two dimensional map. This is known as a map projection. If you tried to flatten the earth onto a rectangular plane also trying to maintain properties such as geometry, area size and distance you would not succeed (see figure 3.3). An example is trying to peel an orange and make it's shape rectangular. Converting from three to two dimensions leads to distortion (Snyder 1987). What properties to preserve is chosen by the way am projection is done.

Figure 3.3: Here, the earth flattened out without any distortion. Linking together the pieces comes at the price of distortion. Figure gathered from (Šavrič and Kennedy, Melita 2020).

3.2.3 Projections

When deciding which projection to use, it is useful to consider what it is most important quantity to preserve. Below we state some projection properties which are commonly considered:

Equal-area preserves the area-size of objects. This is useful when wanting to see the relative size of multiple objects.

Conformal preserves the angle between points. This is important when navigating at sea and when using large- scaled maps.

Equidistant maintains the distance between points. Preserving this quantity comes in handy when wanting to compare several distances.

Compromise considers some distortion from all properties, but attempting to balancing it.

There are three types of methods commonly used when it comes to map projection: cylinder, cone and plane.

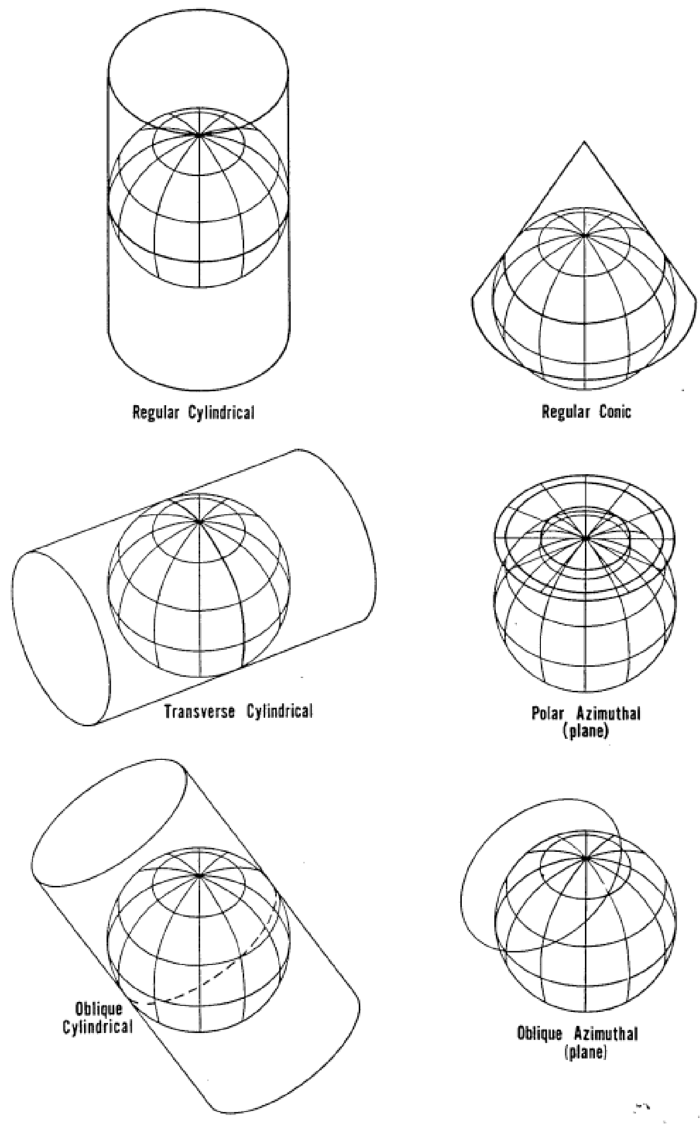

Figure 3.4: Here we see how the different projections are made from a spherical surface to a flat surface. Figure gathered from (Snyder 1987).

Chapter 4

Implementation

This section presents the methods and tools needed for visualizing and calculating the flow field in a given domain. A route classification based on raw AIS-data will also be carried out dividing this chapter into two parts.

First of, the method used in the thesis was inspired and done by a self-study searching in the literature for path planning methods. A method within the field of physics was sought after. Applying potential theory to build up an artificial motion planner came to light from (Pedersen and Fossen 2012) making the assumption that streamlines could represent real vessel routes

The analytic solutions does not take into account the geometry of land of arbitrary shapes which in this case is crucial as we wanted to simulate the flow given map data. This approach resulted in a guessing game where it was attempted to represent the flow field by placing different potential in the different in order to fit the map profile. A new approach was necessary.

The source panel method seemed as an appropriate and more applicable approach. The idea here is to produce flow-fields given information about the map as input and calculate and thus solve for the velocity components needed to visualize a flow-field. There was not found any research regarding using real maps in the literature. The closest research was from (Pedersen and Fossen 2012) where an artificial map was produced.

The reason for this was that the solution relied on objects places in the area and thereafter finding the flow-field. Also, the source panel method approximates objects with straight lines (panels). This was also an important property because if it was possible to separate and set a constraint on the boundaries of land and sea then the boundary conditions and panels could then represent a coastlines on the selected area. The method would now need the coastlines of the selected map.

This was one of reasons to choose Basemap as a map creator as their is a built-in attribute of the Basemap class instance returning the coastlines as line segments. For more details see below.

In order to study the data and the environment around, a visualisation tool is necessary. This grants insight into how the vessel traverse and how much data is necessary for extracting wanted routes in area of interest. This will be done with a map present. In addition, the flow fields need to be plotted in order to understand it's motion. This section presents a framework in how to create a flow field using the source panel method applied for a real life scenario. From the works of (Pedersen and Fossen 2012) the method showed promising results for vessels. Thereafter, a map can be created showing the most interesting areas and in addition saving the computer from unnecessary and excessive calculations. This is done with a package from the matplotlib toolkit called Basemap (explained in more detail below).

With the map created with boundaries the next step will be to plot up some of the AIS-data. Reading the data into a python program is done with pandas.

4.0.1 Dependencies

The main program requires some libraries: Pandas and Basemap. Furthermore, the main program calls additional functions which also need to be present in the folder in order to run properly. These functions calculate the source strengths (sourcepanel.py) and the potential at each grid cell (sourcepoint.py).

4.0.2 Creating routes with Pandas

Pandas is a library where it's sole purpose is intended for data analysis. This software was created for usage in Python. A Dataframe (df) type object is created which let's the user manipulate a dataset by indexing, slicing, subsetting, merging, accessing rows and columns to only name a few operations. Pandas requires Python 3.6.1 or higher. Pandas can be installed with anaconda[1]:

```
conda install -c anaconda pandas
```

or if using Python Package Index (PyPI) [2]:

[1] https://anaconda.org/anaconda/pandas

[2] https://pandas.pydata.org/pandas-docs/stable/getting_started/install.html

```
pip install pandas
```

Pandas supports different types of files format. Here, the file type will be comma-separated-value, more commonly known as a csv file. The CSV files are read with pandas as:

```
import pandas as pd
df = pd.read_csv('Routing/aisToTrondheim.csv', index_col = '
    TimestampPosition', parse_dates = True)
```

Listing 4.1: The data lies in the folder "Routing". Pandas allows for several ways to index the data. We choose to index on the column "TimestampPosition" column. This is read in as an argument when reading in the file

The Euclidean metric is a useful tool for measuring and keeping track of distances between points. We create a new column in the df and calculate the absolute distance between the wanted end point and every registered positional coordinate. This is calculated by:

$$d_i = \sqrt{(x_i - x_{final})^2 + (y_i - y_{final})^2} \tag{4.1}$$

x_i and y_i are the ith longitude and latitude values, respectively. x_{final} and y_{final} are the end points longitude and latitude values, respectively. We can calculate velocities and store the values and preserve the index in new columns in the df:

```
df['Heading'] = np.radians(df['Heading'])
df['VelocityLongitude'] = df['SpeedOverGround']*np.sin(df['
    Heading'])
df['VelocityLatitude'] = df['SpeedOverGround']*np.cos(df['
    Heading'])
df['DistTrondheim'] = np.sqrt((df['Longitude'].values -
    trond_lon)**2 + (df['Latitude'].values - trond_lat)**2)
```

Create and add new columns Velocity Longitude, Velocity Latitude and Dist-Trondheim and use the existing ones to convert degrees to radians.

The next step is to take a look at the data. As mentioned above (AIS-data Extracting routes) the timestamp column can be used to sample routes from the data set. The data set has dimension $71694x12$. Each column represent a variable for specific timestamp (see figure of AIS-data example plot above, 3.1). Working with all that data becomes complex and computational heavy. To get a clearer view, a subset of the data is drawn out. Here we it is based on the vessel type. We create a new df consisting only of one single vessel type:

```
df_subset = df[df['ShipType'].isin(['tugboats'])].copy()
```

Listing 4.2: All tugboat type vessels are stored in a new variable which can further be analyzed

The subset now consists of only one vessel type (in this case tugboats) from the main data set. The objective now is to extract routes from the data subset. From the timestamp column it can be seen that on the same day the data contains a fixed interval of logged AIS-data varying from seconds to minutes. One could argue that this is the same vessel being tracked. A sanity check is to make sure the IMO number is the same for the suggested route.

A variable holding all unique dates in the subset is needed for separating the data even more. This is based on the assumption that every data point holding the same date constitute one unique route. The number of data points vary from date to date. We need to keep track of the number of data points in each route. At last, each set of routes made, we want to find vessels travelling to some end point. Each defined route does not necessarily share the same end point. There are various reasons for this some of which include: vessels turn off their AIS-sender, there could be a malfunction, vessels are not necessarily bound to travel to the same destination. We choose to separate each route into three categorise: vessels travelling to, through and from the destination. This classification is of interest as when using the flow field we want every vessel to converge to the same point.

For each route we check if the distance is decreasing or increasing and make the assumption: If the last absolute distance value is larger than the first, we conclude that the vessel is travelling from the destination.

Vessels travelling through is measured by counting the number of times the distance increase occurs. If the count is larger then some number, we conclude the route is travelling through. And vice versa for route travelling to. By making this classification, we can easily discard routes not of interest. Below we show the implementation:

```python
for i in range(unique_dates.shape[0]):
    route = df_subset.loc[str(unique_dates[i])]
    last_dist = route['DistTrondheim'][-1]
    first_dist = route['DistTrondheim'][0]

    if last_dist < first_dist:
        df_subset.loc[route.index, 'Direction'] = 'To'
        count = 0

        # check route going to if starting to deviate from dest.
        for j in range(route.shape[0] - 1):
            next_dist = route['DistTrondheim'][j + 1]
            prev_dist = route['DistTrondheim'][j]
            if next_dist < prev_dist:
                #dist decreasing
                count = 0
            else:
                #dist increasing
                count += 1
            if count == 3:
                #Ship could be travelling through
                df_subset.loc[route.index, 'Direction'] = 'Through'
                break
    else:
        df_subset.loc[route.index, 'Direction'] = 'From'
```

To access the classified routes, we create df and collect all vessel classified as 'To' from the 'Direction' column and gather the corresponding variables needed to plot and visualize:

```python
df_ToTrondheim = df_subset.loc[df_subset['Direction'] == 'To',
                               ['Longitude', 'Latitude', 'VelocityLongitude', 'VelocityLatitude', 'DistTrondheim']]
```

With the use of Pandas we have been able to extract subsets of data which in this case makes it less chaotic when visualizing the data. We have been able categorize and produce routes from the data. In addition, a basic classification determining the vessels' direction and end points is made.

4.0.3 Basemap

Creating the map in python is done with the matplotlib basemap toolkit. Together with matplotlib Basemap can represent geospatial data onto different map projections. By sorting the AIS-data in terms of some vessel characteristics the array containing vessel information can be used to read the longitude and latitude values needed for setting up the boundaries of the map. The necessary requirements for Basemap [3] are:

- Matplotlib 1.0.0 (or later, download)
- Python 2.6 (or later, including Python 3)
- Matplotlib 2.2 LTS requires Python 2.7 or later
- Matplotlib 3.0 requires Python 3.5 or later
- NumPy 1.2.1 (or later)
- Array support for Python
- PROJ4 Cartographic Projections Library

If using anaconda [4] environment Basemap can be installed by the following command in the terminal: When creating a map with Basemap an class object instance

```
conda install -c anaconda basemap
```

is made. The object as attributes and methods which are treated equally as any other class object in python. The arguments passed into the Basemap object the boundary points, projection type, resolution. Plotting the map is done using matplotlibs standard plotting commands. A Basemap class instance is created with some formatting:

```python
from mpl_toolkits.basemap import Basemap
m = Basemap(llcrnrlon = minlon, llcrnrlat = minlat , urcrnrlon =
    maxlon , urcrnrlat = maxlat,
            resolution = 'i', projection = 'cyl', lat_0 = lat0,
    lon_0 = lon0)
m.drawmapboundary(fill_color = 'white')
m.fillcontinents(color = 'lightgrey', lake_color = 'white',
    zorder=1)
```

Listing 4.3: Boundary points for the map can be found from simple max and min function in the standard python library. See details below in listing 5.4

[3] https://matplotlib.org/basemap/users/installing.html

[4] https://anaconda.org/anaconda/basemap

An object m is created with parameters lower left corner longitude ($llcrnrlon$), lower left corner latitude ($llcrnrlat$), upper right corner longitude ($urcrnrlon$) and upper right corner latitude ($urcrnrlat$). Basemap allows to choose 4 different resolutions: c (crude), l (low), i (intermediate), h (high), f (full) or None. The projection type . lon_0 and lat_0 set the center of the map.

The Basemap object has multiple attributes which can be used to extract characteristics of the area in the form arrays or to make plots more realistic and understandable.

Creating map

Given the AIS-data we now can limit our map to the maximum and minimum values of our the data. The range of longitude and latitude degrees are (-180, 180) and (-90, 90) respectively. 0 degrees of longitude is the Prime Meridian which goes through Greenwich, England. negative values represent the western hemisphere and positive values represent the eastern hemisphere. For degrees of latitude, the 0 degree is placed at equator. Therefore, negative values represent the southern hemisphere and positive values the northern hemisphere.

The maximum and minimum values can found by:

```
minlon = max(-180, min(df['Longitude']))
minlat = max(-90, min(df['Latitude']))
maxlon = min(180, max(df['Longitude']))
maxlat = min(90, max(df['Latitude']))
```

Listing 4.4: Boundary points for the map can be found from simple max and min function in the standard python library

These values are used as arguments for the Basemap class instance. The projection chosen is cylindrical making the representation rectangular. See figure 4.1 below.

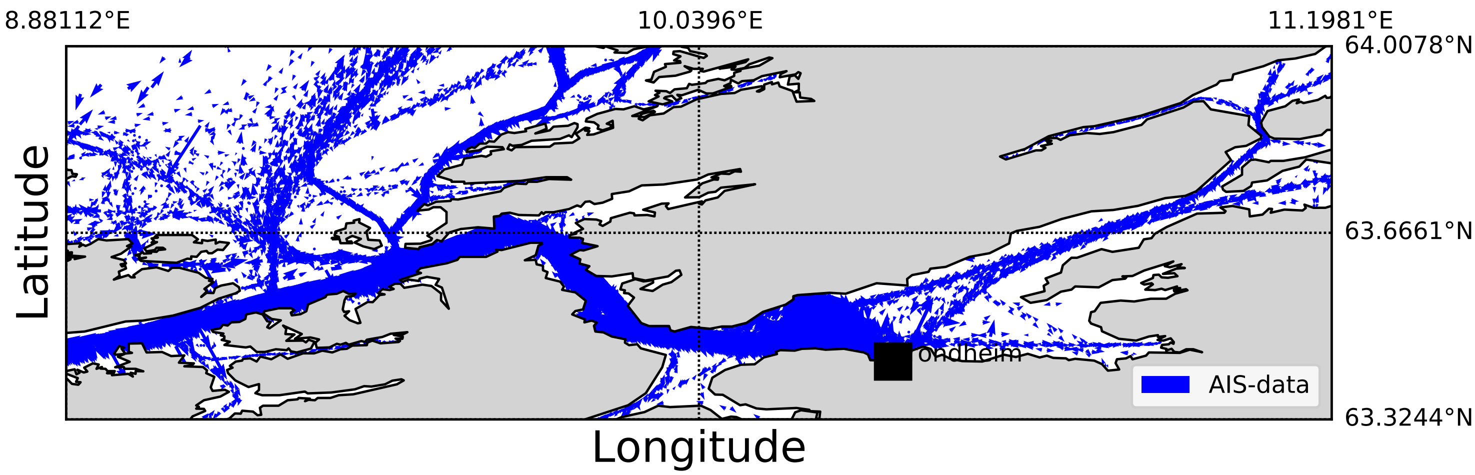

Figure 4.1: An overview of all AIS-data logged from 4/6/2018 to 13/10/2019 in the fjord of Trondheim. The blue arrows indicate the vessels direction of travel. The resolution in Basemap is set high. Format: grey is land white is ocean.

The map instance has several attributes. In order to apply the sourcepanel method, information regarding the coastline is crucial. After creating a map instance, the coastlines are found using the method:

```
coast = m.drawcoastlines()
```

The data is gathered from a geography data set[5]. This object holds information about the coastlines which can be retrieved further by the class attribute:

```
coordinates = coast.get_segments()
```

This holds a list of arrays where each array contains vertices making up the polygon representation of each connected land area. Lists can be converted easily into numpy arrays making it more convenient to work when computing mathematical operations.

[5]GSHHG(https://www.soest.hawaii.edu/pwessel/gshhg/)

4.1 Analytic approach

For the analytic solution it was chosen to create a class instance for each flow characteristics initializing the class with a flow strength value and location of origin in space $(x0, y0)$. For each method the defined grid is inputted and the corresponding velocities are returned. The velocity contributions are then added together for the final flow-field to visualized. Below is a flow chart for the class instance:

4.2 Computing the source strengths

With the use of Basemap' attributes, the necessary information needed to compute the source strengths which then allows for a flow-field to emerge based on the environment. The idea is to place a set of unknown source strengths λ at the center of each segment of coastlines. The vertices are collected from the $get\_segments()$ attribute and make up the boundary points (see fig 2.7). Furthermore, the control points (x_{ci}, y_{ci}) can be calculated from eq (2.60) (2.61), respectively given the boundary points. Also, with the boundary points given we can calculate the angles needed for the normal and tangential component and freestream velocity angle in normal direction. The angles ϕ_i, δ_i and β_i are acquired using first eq 2.64, then eq's (2.65) and (2.66). The angles are stored in arrays with dimension $nx1$ where n is the number of panels.

From the boundary points, we have calculated the necessary components of what is needed for computing the geometric integral. Thereafter, solving the system of equations for the source strengths λ is what we will go through next.

For solving the geometric integrals for the normal and tangential component eq (2.93) and eq(2.120) two empty matrices, each with dimension nxn where n is the number of panels, are set up. As we saw from the derivation of the normal and tangential geometric integral, they share very much the same variables. Therefore, these are calculated at the same time. The I_{ij} geometric integral is calculated by the terms from eq (2.118) where A,B,C,D and E is listed below . S_j is the length of panel j. For all $j \neq i$ eq (2.118) fills every element of the matrix, leaving us with the diagonal elements. As stated before it was found that when $j = i$ the I_{ij} has the value of $\frac{\lambda_i}{2}$. The tangential part is computed in the same manner using eq (2.132) for computing each element in the matrix J_{ij}. We are now ready to solve the system of equations, which means finding the source strengths λ at each control point (XC,YC).

This is rather short process as we will use the numpy module np.linalg.solve

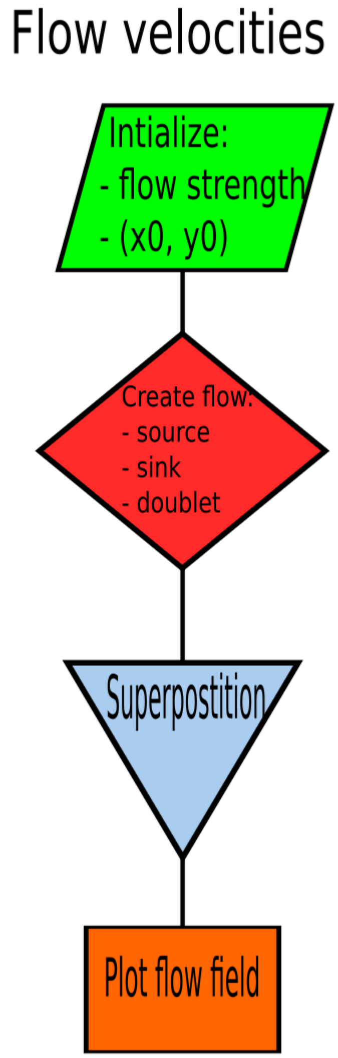

Figure 4.2: Flow chart using the class instance for calculating the velocities at every grid point.

which solves a set of linear equations. In order to calculate the matrix must be a square matrix, have full rank i.e the columns or rows must linearly independent G. Strang, Linear Algebra and Its Applications, 2nd Ed., Orlando, FL, Academic Press, Inc., 1980, pg. 22. First, the diagonal elements of the matrix needs to filled with π from 2.136 for the case when $j = i$ which means physically when we calculate the total contribution of potential induced at itself. With the matrix produced from the python program $sourcepanel.py$ where the function $solveGeometricIntegrals()$ is implemented, we use the array of β values found from $flowboundary.py$ and thus solve $I\lambda = b$ where I is the geometric integral values, b are the array β multiplied with $-2\pi V_\infty$. The full matrix equation can be seen in 2.136.

4.2.1 Computing velocities

After computing the source strengths, the job is now to set up an area in which we can compute and see how the source strengths will affect this area and thereafter the velocities vectors. Define a number of grid cells. For an increase amount demands more computing power, naturally.

In order to help the streamlines converge to the wanted location, we have here decided to place a sink potential at the desired point, namely Trondheim. After calculating the velocities with the defined grid points we just use the superposition principle adding this contribution to every grid cell.

Finally, we compute the velocity components at every defined grid cell in the area in interest. This is done with eq's (2.137) and (2.138). Note that from the latter equations the sink potential located at the end points is not included. Therefore, the final equation for computing the velocities at a point p due the potential induced at that point reads:

$$V_{x,p} = V_\infty \cos(\alpha) - \frac{\lambda_s x}{2\pi(x^2+y^2)} + \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} \left[\frac{-\cos\phi_j}{2} \ln\left[\frac{S_j+2S_jB+C}{C}\right] + \frac{A_{xp}}{E}\left(\arctan\left[\frac{S_j+B}{E}\right] - \arctan\left[\frac{B}{E}\right]\right)\right] \tag{4.2}$$

$$V_{y,p} = V_\infty \sin(\alpha) - \frac{\lambda_s y}{2\pi(x^2+y^2)} + \sum_{j=1}^{n} \frac{\lambda_j}{2\pi} \left[\frac{-\sin\phi_j}{2} \ln\left[\frac{S_j+2S_jB+C}{C}\right] + \frac{A_{yp}}{E}\left(\arctan\left[\frac{S_j+B}{E}\right] - \arctan\left[\frac{B}{E}\right]\right)\right] \tag{4.3}$$

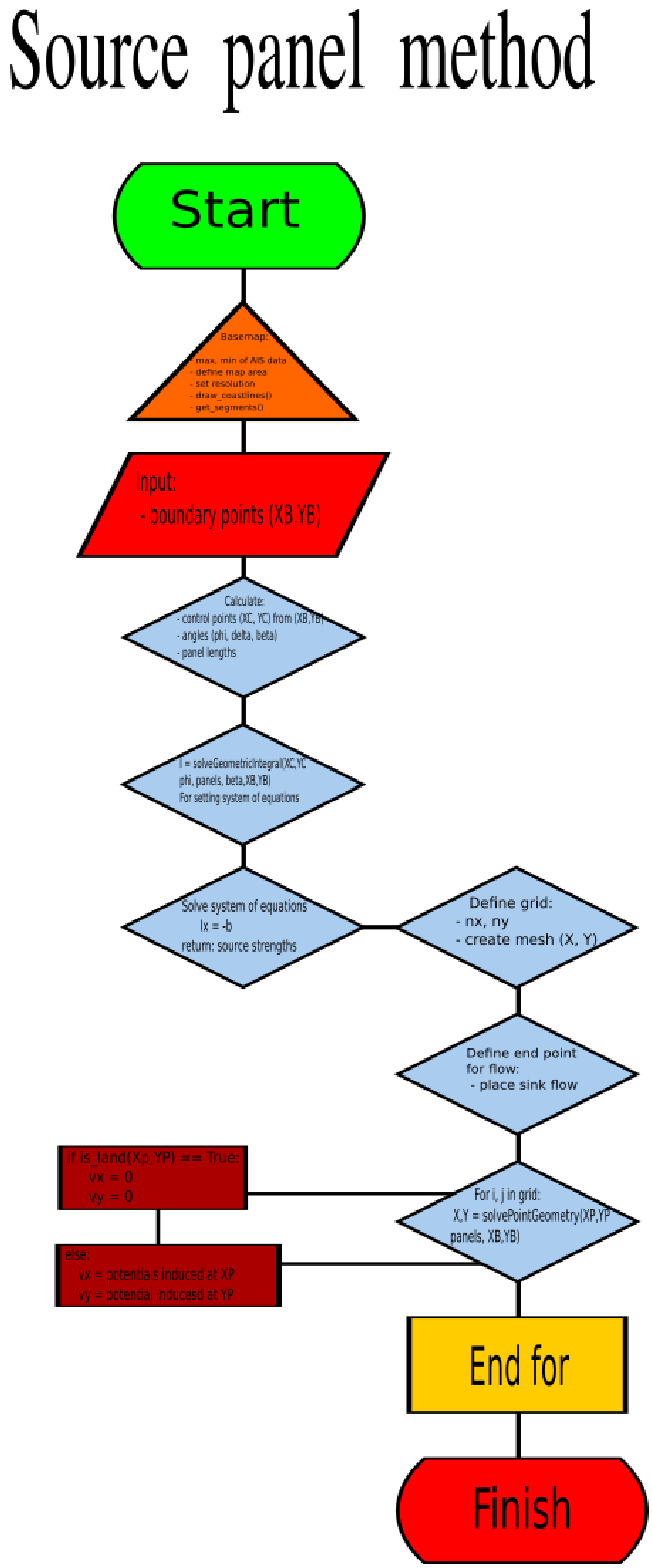

Figure 4.3: Flow chart for the source panel method

To summarize: We have extracted AIS-data using Pandas Dataframe. With this, a classification of routes travelling ”To”, ”From” or ”Through” was performed to get an overview of how the vessel traverse. Next, a map was created from the max and min longitude and latitude values of the data set setting boundary points of the map area. We obtained a crucial parameter from Basemap, the coastlines making it possible to set out the control points at mid point of each coastline segment. Furthermore, at the control points source strengths are placed giving n equations for n panels. The source strengths were then solved for by representing the n equation in matrix form eq (2.136). An intermidiate step was to set a sink flow in the destination making sure the flow converges. With source strengths computed and the sink set out, the final task was finding the velocity at a point p induced by the source strengths yielding the velocity components of the flow at every point in the defined grid.

Chapter 5

Results

Our main assumption and approach for creating motion planner is representing vessels routes as streamlines in the flow of a fluid. The streamlines calculated serves as possible routes for a vessel to follow. We present the analytic and numerical solution for velocity potential and stream functions which lead to the representation of nonuniform flow profiles in a domain. We will attempt to create a flow around a circle using the analytic solutions of the laplace equations. The cylinder representing an object which cannot be traversed through. To check the validity we add another cylinder and see if the solution still holds and can be generalized to an increase of objects.

Furthermore, we apply the source method to a real map generated from Basemap, namely the fjord of Trondheim. AIS-data from this area will be shown for some vessel types. We will be interested in studying how the model generalizes, meaning how the model works on several route. By extracting routes from the AIS-data we can see how each vessel traverses through the area.

5.1 Classifying routes

The presentation of data is a key factor to understand how vessel traverse as the raw AIS-data is not ordered based on any metric. Therefore, a classification is done making is more straightforwardly to collect routes. From 4.1 every data point is plotted. This is of course not an optimal setting and is excessive. The reason for this was to make it clear how chaotic the data looks like without any metric. For this case, we would also like to point out that not all classification done are with the equal amount of significance or validity. The reason being the lack of data which could be caused by malfunction at either sender or receiver, turning of sender or others.

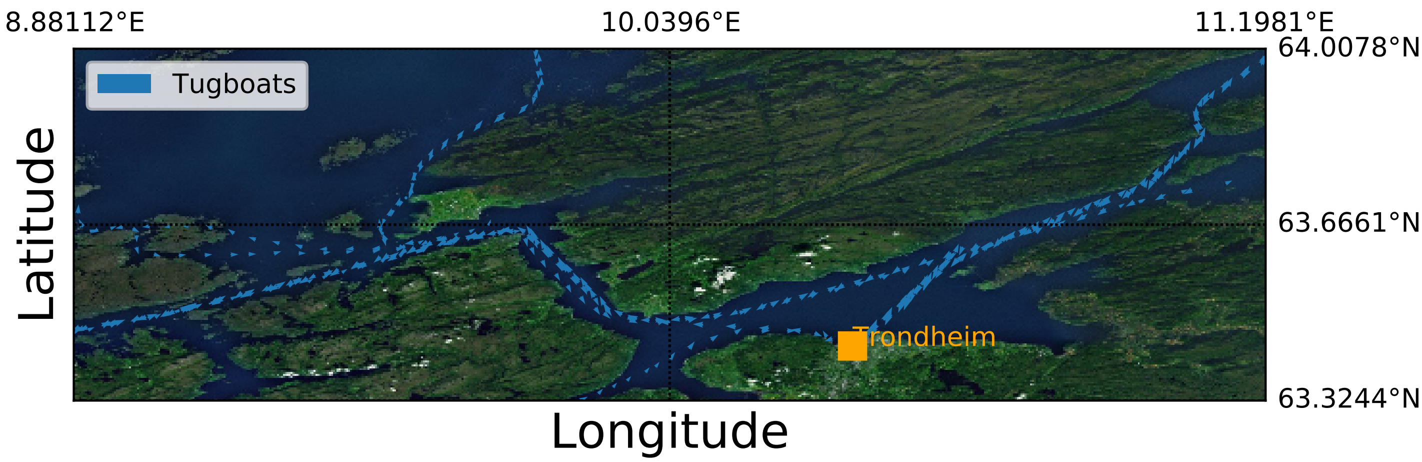

Figure 5.1: A subset of the data containing all tugboats are seen here. Some similar traversing is done going to and from Trondheim. As we also can see are tugboats going straight past Trondheim.

The projection style chosen is equidistant cylindrical. This projection divides the earth up into equal rectangular pieces. Preserving the distance between points and thereafter comparing them is what is mainly focused on here. As seen in figure 5.2 at upper part we see only two AIS-data points. This underlines the mentioning of data not being logged for equal amount of timestamps and do not guarantee a convergence toward Trondheim only the direction.

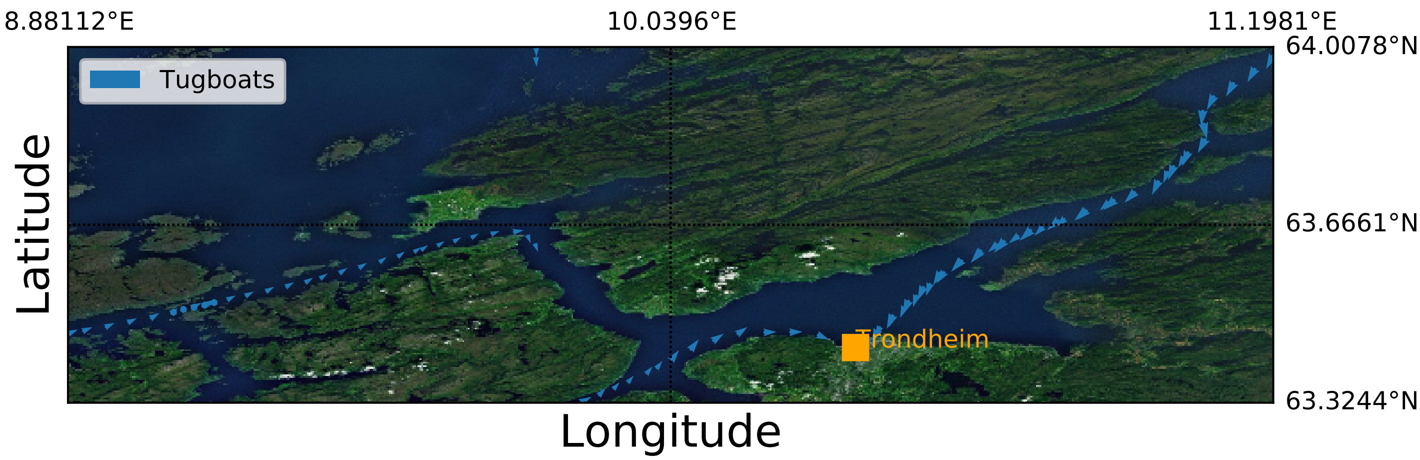

Figure 5.2: Here, all "To" Trondheim vessel are plotted. As we see, the amount of AIS-data contained in the individual routes vary from route to route.

5.2 Analytic flows around circular objects

For given circular objects and geometries we found that doublet was a good candidate for creating flows around objects. Note, not every choice for a streamline is optimal. Some streamlines lead too close to the object in front and could in worst case lead to collision with the object. This becomes an instance of optimizing w.r.t safety or economy. Travelling closer objects leads to a shorter path to goal, but the risk of collision is larger. The analytic solutions do have some important properties. Analytic solution are have a physical aspect to them, meaning they yield a good representation of the real world. Sharing some flexibility in where the flows are placed, the flow will nearly almost lead to the end point. The geometry though needs to easy to represent with the analytic solutions. Increasing the complexity of the flow, it might not be possible to use the elementary flow types. If it was possible the combinations of potentials or stream functions from laplace equations needed to be tested a many time in order to fit the stream line profile.

5.3 Setting the individual strength coefficients

When placing out the individual flow types, the flow strengths or strengths coefficients needs to be set. The strength coefficients can be regulated and adjusted.

This will affect the streamline paths. This is the case for analytic solution. One could argue that together with the flow characteristics the flow strengths determines the streamlines generated because of its . From fig 5.3 and 5.4 they chosen in such a way that there exist path between the start and end point.

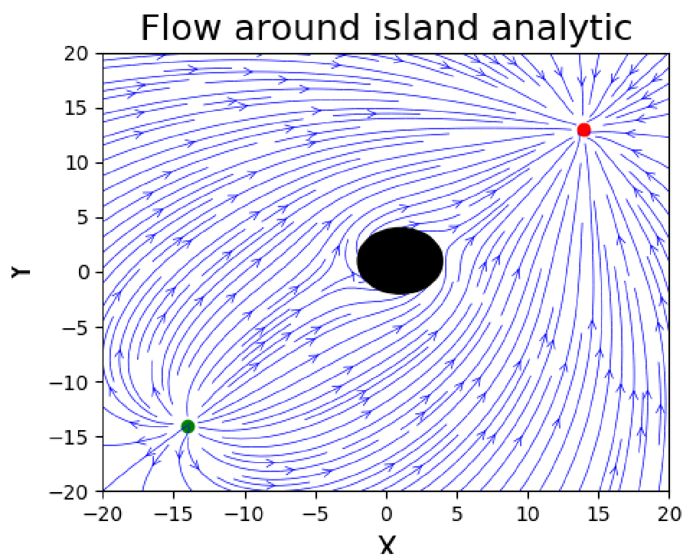

Figure 5.3: Here, streamlines (blue) shows a fluid flowing from Green (starting points), past the a circular object, converging at red (end point). The strengths chosen here are: 1, 4, 0.5

This solution is very computational light. The computational complexity can be increased by the amount of grid points defined in the domain. Note that the streamline does begin or end at the exact location of the source and sink. The reason being that a singularity occurs in those points see eq(2.20), (2.21), (2.23), (2.24).

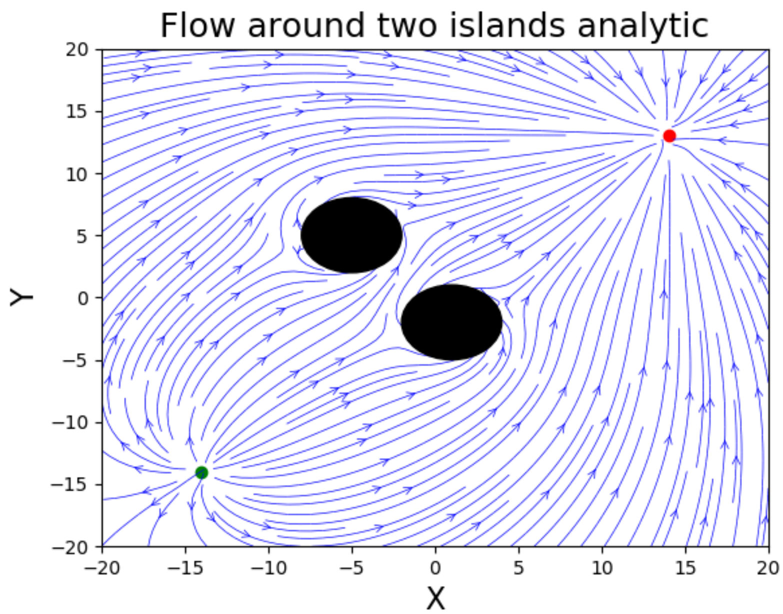

Figure 5.4: Caption

With this approach limitations occur. The complexity of the domain is not very applicable to real world map or environments. The adjusting the flow strength could be never ending if the goal is to replicate one specific vessel route with a streamline. In addition, when the objective comes to solving for distinct flow-field in a domain, information about the domain is needed. However, using the source panel method the complexity can

Bibliography

Acheson, D.J (2003). *Elementary Fluid Dynamics*. Oxford.

Anderson, John D. (1991). *Fundamentals of Aerodynamics*. en. 2nd ed. McGraw-Hill Series in Aeronautical and Aerospace Engineering. New York: McGraw-Hill. ISBN: 978-0-07-001679-8.

Arhus, Gisle Hoel and Stian Røyset Salen (2018). "Predicting Shipping Freight Rate Movements Using Recurrent Neural Networks and AIS Data". en. In: p. 78.

Besse, Philippe et al. (Aug. 2015). "Review and Perspective for Distance Based Trajectory Clustering". en. In: *arXiv:1508.04904 [cs, stat]*. arXiv: `1508.04904 [cs, stat]`.

Encyclopedia britannica (2020). *Latitude and Longitude*. https://www.britannica.com/science/latitude.

Gjevik, Bjørn (2009). *Innføring i Fluidmekanikk*.

Hvamb, Knut (2015). "Motion Planning Algorithms for Marine Vehicles". en. In: p. 153.

Lewis, Charles Lee (1927). *Matthew Fontaine Maury, the Pathfinder of the Seas*. United States naval institute.

Meijer, Ricardo (2017). "ETA Prediction Predicting the ETA of a Container Vessel Based on Route Identification Using AIS Data". PhD thesis.

Pallotta, Giuliana, Michele Vespe, and Karna Bryan (June 2013). "Vessel Pattern Knowledge Discovery from AIS Data: A Framework for Anomaly Detection and Route Prediction". en. In: *Entropy* 15.12, pp. 2218–2245. ISSN: 1099-4300. DOI: `10.3390/e15062218`.

Pedersen, Morten D. and Thor I. Fossen (2012). "Marine Vessel Path Planning & Guidance Using Potential Flow". en. In: *IFAC Proceedings Volumes* 45.27, pp. 188–193. ISSN: 14746670. DOI: `10.3182/20120919-3-IT-2046.00032`.

Šavrič, Bojan and Kennedy, Melita (2020). *Map Projections*. https://www.google.com/url?sa=i&url=https%3A
G4borMyEPS8um7oLQP&ust=1582711305852000&source=images&cd=vfe&ved=0CAIQjRxqFwoTCM

Snyder, John (1987). *Map Projections- a Working Manual*.
UN-Business Action Hub, United Nations (2020). *Business.Un.Org*. https://business.un.org/en/entitie
Wang, Xuantong, Jing Li, and Tong Zhang (Dec. 2019). "A Machine-Learning Model for Zonal Ship Flow Prediction Using AIS Data: A Case Study in the South Atlantic States Region". en. In: *Journal of Marine Science and Engineering* 7.12, p. 463. ISSN: 2077-1312. DOI: 10.3390/jmse7120463.
Wickham (2014). "Tidy Data". In: *Journal of statistical software*, p. 24.