

UNIVERSITY OF SOUTHAMPTON

FEEG3003

INDIVIDUAL PROJECT

---

## Information Extraction and Visualisation from Vortical Flow Fields

---

*Author:*

Carlos Losada de la Lastra

*Supervisor:*

Dr. Gabriel D. Weymouth

This report is submitted in partial fulfillment of the requirements for the MEng Ship Science, Faculty of Engineering and the Environment, University of Southampton.

1<sup>st</sup> of May, 2015

Total number of words: 9356

## Declaration

I, Carlos Losada de la Lastra declare that this thesis and the work presented in it are my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a degree at this University;
2. Where any part of this thesis has previously been submitted for any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;

There's a way to do it better. Find it, *Thomas Edison*

## Acknowledgements

I would like to thank my supervisor Dr. Gabriel D. Weymouth, for his support throughout this project. Without his constant support via email, his meetings week after week, his approach to problems, and many suggestions, this project would have never accomplished what it has.

Big thank to my family and friends, for their support, even when I was returning home late at night, they managed to cheer me up and encourage me to keep on going. And for listening, while I excited attempted to explain what I was doing all those evenings at uni. A big thank to Jack Gradus for reading over my work, without his many comments, this document would have probably been written in Spanglish.

## Abstract

Information extraction and visualisation of vortical flow fields is a complex task. When attempting to study vortical flows the number of unknown parameters typically exceeds the number of equations. The need for coherent and informative visualisation is key to the understanding of the flow. In this research project the mathematical definition of a vortex core proposed by *Jeong* and *Hussain* [1] was combined with geometrical integration to successfully approximate the circulation field. In order to simplify the problem, the analysed velocity fields were also generated. This provided some knowledge regarding the variables of the flow, and how to isolate them within the field. The process was successfully carried out for simple and more realistic flow fields. A complete information extraction and visualisation algorithm is proposed as a future work suggestion which could lead to further research in this field.

## Contents

<b>Declaration</b>	<b>2</b>
<b>Acknowledgements</b>	<b>4</b>
<b>Abstract</b>	<b>5</b>
<b>List of Figures</b>	<b>8</b>
<b>1 Introduction</b>	<b>9</b>
1.1 Current visualisation approaches to vortical flows . . . . .	9
1.2 Scope for improvements in visualisation techniques . . . . .	10
1.3 Aim of this Research Project . . . . .	10
1.4 Risk Assesment . . . . .	10
1.5 Project Structure . . . . .	11
<b>2 Literature Review</b>	<b>11</b>
2.1 Vortical Velocity Fields . . . . .	11
2.2 Vortex definition and Identification . . . . .	11
2.3 Vortex Geometry . . . . .	12
2.4 Circulation, Impulse and Force generated by a Vortex . . . . .	12
2.5 Particle Visualisation . . . . .	13
2.6 Visualisation of Information . . . . .	14
<b>3 Visualisation Problem</b>	<b>15</b>
3.1 Simplifying a Difficult Problem . . . . .	15
3.2 Forwards and Backwards Problems . . . . .	16
<b>4 Velocity Field Generation Algorithm</b>	<b>16</b>
4.1 Velocity Field Generation . . . . .	16
4.1.1 Rankine Vortex . . . . .	18
4.2 Velocity field <i>.vtu</i> Image Reading and Writing . . . . .	19
<b>5 Visualisation Algorithm</b>	<b>19</b>
5.1 $\lambda_2$ Definition of a Vortex Core . . . . .	19
5.2 Analytical derivations of $\lambda_2$ . . . . .	20
5.2.1 Inside the core . . . . .	20
5.2.2 Outside the core . . . . .	22
5.2.3 Shear layer . . . . .	23
5.3 Generation of Seeding Points . . . . .	24
5.4 Evaluation of Seeds with Time . . . . .	24
5.5 Circulation Integrals and Field . . . . .	26
5.6 General Visualisation Algorithm . . . . .	27
<b>6 Implementation of the General Algorithm</b>	<b>28</b>

<b>7 Results</b>	<b>28</b>
7.1 Velocity Field Generation . . . . .	28
7.2 $\lambda_2$ implementation . . . . .	30
7.3 Circulation Intergration . . . . .	33
<b>8 Future Work Recommendations</b>	<b>35</b>
8.1 Generalisation of the Visualisation Solution . . . . .	35
8.1.1 Seeding Point Selection and Preliminary Circulation Integrals . . . . .	35
8.1.2 Geometric Properties and Numerical Integration . . . . .	36
8.1.3 Impulse of the Vortex Rings . . . . .	37
8.1.4 Force Applied on the Body . . . . .	38
8.2 General Algorithm . . . . .	39
8.3 Language improvements . . . . .	40
8.4 Unstructured Grid Reconstruction . . . . .	40
<b>9 Conclusion</b>	<b>41</b>

## List of Figures

1	Two dimensional visualisation of the wake behind a cylinder. Obtained with the aid of the LilyPad software.[2]	9
2	Three dimensional visualisation of the wake behind a stingray	10
3	Velocity profile in a section through a rankine vortex.	18
4	Example of an initially developed 2 dimensional velocity. The field is a section through a vortex ring. This field was produced with the Matlab software.	29
5	Velocity field induced by a single vortex ring. The magnitude of the velocity is represented by the size and colour of the vectors.	29
6	Figure showing the velocity induced by a chain of vortex rings with finite vortex core thickness.	30
7	Comparison between the visualisation of the flow field and the $\lambda_2$ surface for the single vortex ring.	31
8	Figure comparing the informativeness of the visualisation of the chain of vortices.	32
9	Comparison of the information displayed in the visualisation of a single vortex ring.	33
10	Comparison of the information displayed in the visualisation of the fish-like wake.	34
11	Table showing a speed comparison of various programming languages when carrying out standardised numerical computatons. Figure taken from JuliaLang [4].	40

## 1 Introduction

The interaction between a fluid and a structure is a phenomenon very relevant to the maritime world. The leg of an oil rig, a high speed hydrofoil ferry, or a fish like robot, constitute examples of structures where the interaction with the fluid is key for their optimal design and operation. It should be noted that this interaction is also relevant to many other fields, such as the aerospace and transport. In this project, for simplicity, the interaction will be constrained to the maritime world. During the study of this interaction, if a detailed analysis or simulation is needed, the assumption of a perfect flow is no longer appropriate. The presence of vortical structures will be expected when modelling said interaction.

The definition and behaviour of vortical structures has been extensively covered by the literature. The importance of having the ability to fully understand and visually display these vortices is key to understanding the behaviour of the flow around the structure. Furthermore to comprehend the effect the vortex causes on both the structure and the wake behind it.

### 1.1 Current visualisation approaches to vortical flows.

At present, showing the velocity or vorticity field around a body where vortices are present is the most common technique is to display the information of the flow.

This can be useful in a two-dimensional field, since the direction of the vorticity plane is normal to the plane on which the flow is located. Since the direction of the vorticity is always normal to the plane, it can be treated as a scalar field, providing good visual information regarding the behaviour of the flow.

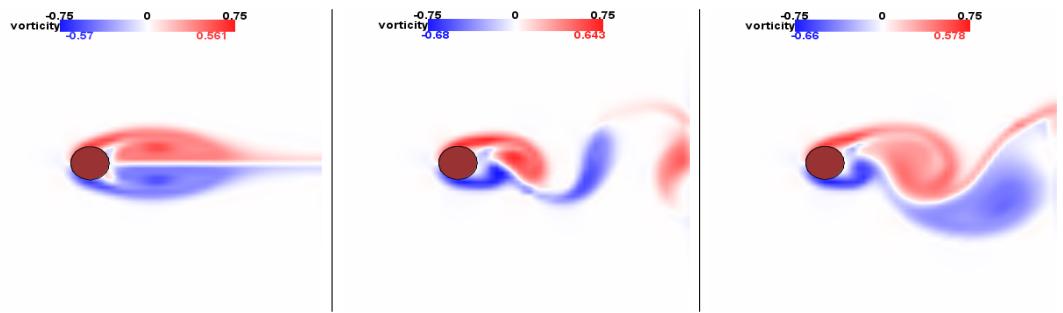


Figure 1: Two dimensional visualisation of the wake behind a cylinder. Obtained with the aid of the LilyPad software.[2]

When our interest focuses on a three-dimensional field, vorticity is no longer a useful parameter to display. The direction of the swirl vector, normal to the plane which the vortex rotates about, is as relevant as the magnitude of the vorticity of the flow. Displaying the vorticity of the flow will not be an informative way of showing the influence of the vortices in a three dimensional flow field.

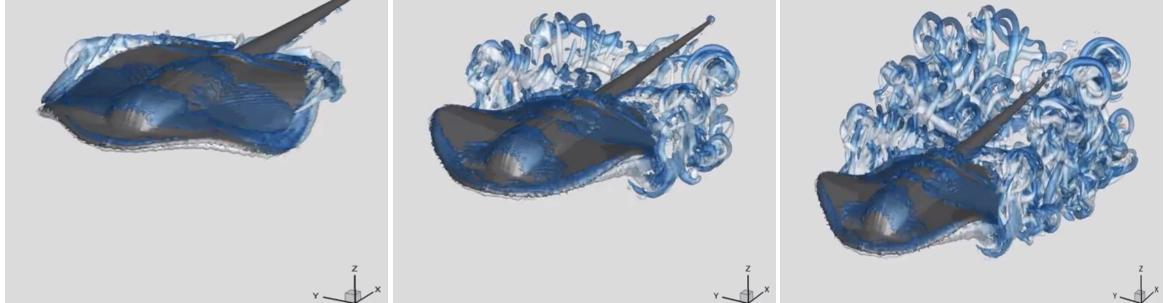


Figure 2: Three dimensional visualisation of the wake behind a stingray

## 1.2 Scope for improvements in visualisation techniques

As discussed previously in this section, visualisation of vortical structures is of great importance in order to fully understand the effect of the vortices on the structure. If our interest lies in three dimensional fields, visualising the vorticity field will not show the flow in an informative way.

Although the vorticity itself is not a valid option for visualising vortical flows, it can provide a good starting point. This is due to the fact that vorticity is local and inherent property of the flow surrounding a vortex. If other inherent properties of the vortex could be displayed in the same informative way as the vorticity in a two dimensional field, the visualisation of the vortices would provide a neat and clean way of presenting the information of the flow.

## 1.3 Aim of this Research Project

This project has the aim of investigating the known relationship between the vorticity of a vortex, the impulse with which the vortex travels through the fluid, and the force that the vortex imposes on the body that generated the vortex. Moreover to implement this relationship in an information extraction and visualisation algorithm. The algorithm should have the ability to be applied to any velocity field obtained throughout experimental or computational methods .

The algorithm will be developed in the Python programming language. The reasons behind choosing Python as the programming language are its easy to write syntax and its ability to work with fields. These qualities will be of great use during both the development of the algorithms, and with the generation and processing of the velocity field where the vortices will be present.

## 1.4 Risk Assesment

The project will be carried out on a computer without any model experimentation or human interaction. Therefore the is no need to carry out an experimental or ethical risk assessment.

## 1.5 Project Structure

The project is structured as follows: in section 2 the literature around which the project is based is explored and discussed. The problem of visualising vortical flow fields and the simplifications made in this project are discussed in section 3. The general characteristics of the visualisation algorithm are defined in section 5. The detailed developed and tested algorithm is introduced and its results are shown in section 6 and section 7, respectively. Section 8 discusses the future work recommendations, and section 9 covers the final conclusions drawn from this project.

## 2 Literature Review

This section covers the research of the literature around which the project is based. The main principles governing velocity fields, vortex core identification, and geometry of vortices are investigated. Furthermore the geometry of vortices, the inherent vortex properties such as circulation, impulse and force applied are covered. Finally particle and information visualisation are investigated.

### 2.1 Vortical Velocity Fields

*Biot* and *Savart*, as shown in *Saffman*'s book *Vortex Dynamics* [6], discovered the relationship governing the magnitude and direction of the magnetic field induced by an electric conductor. This field can be calculated at a point by applying equation 1.

$$\vec{B} = \frac{\mu_0}{4\pi} \oint \frac{\vec{r} \times I dI}{|\vec{r}|^3} \quad (1)$$

Where  $\vec{B}$  is the magnetic field at the position defined by the vector  $\vec{r}$  from the segment of the conductor to the point of interest.  $I$  is the current through and  $dI$  a small segment of the conductor.  $\mu_0$  is the magnetic constant.

This principle also applies to vortical velocity fields. The Biot-Savart Law also defines the induced velocity field due to a vortex line. Both *Newman* and *Saffman*, respectively in their books *Marine Hydrodynamics*[5] and *Vortex Dynamics*[6] show this relationship, shown in equation 2.

$$\vec{U} = \frac{\Gamma}{4\pi} \oint \frac{\vec{r} \times \vec{ds}}{|\vec{r}|^3} \quad (2)$$

Where  $\vec{U}$  is the velocity field at the position defined by the vector  $\vec{r}$  from the segment of the vortex line to the point of interest, and  $\Gamma$  is the circulation of the vortex line.

These relationships will be applied in the generation of the velocity field. Equation 2 provides the velocity induced by a vortex segment  $d\vec{s}$ , which will be looped through all the points in the field in order to calculate the induced velocity field.

### 2.2 Vortex definition and Identification

*Jeong* and *Hussain* [1], proposed a definition of a vortex as the eigenvalues of the velocity gradient tensor  $\nabla u$ . The definition takes, as a starting point, the pressure minimum criterion,

*Robinson* [18], who showed that this criterion captured vortical structures in a boundary layer. *Jeong* and *Hussain* [1] took *Robinson*'s findings and discarded effects due to viscosity and unsteady straining, defining a vortex core as a connected region with negative second largest eigenvalue  $\lambda_2$ .

$$S^2 + \Omega^2 = \lambda_2 < 0 \text{ (excluding surfaces where } \lambda_2 = 0) \quad (3)$$

Where  $S^2$  and  $\Omega^2$  are respectively the symmetric and antisymmetric parts of the velocity gradient tensor  $\nabla u$ .

*Jiang*, *Machiraju*, and *Thompson* [7] carried out a summary of detection and visualisation techniques for swirling features. An outline of each detection algorithm was provided together with an analysis of its strengths and weak points. The outcome of their research was that geometric verification is a necessary part of detection and visualisation of vortices. This was concluded by pointing out to the research carried out by them on geometric verification of vortices.

The  $\lambda_2$  definition of the vortex core will be used as a starting point for the extraction of information from the flow field. Although the vortex cores will not require verification due to the robustness of the definition, the verification concept will be used in order to extract the vortical properties from the flow.

### 2.3 Vortex Geometry

*Jiang*, *Machiraju*, and *Thompson* [8], proposed a technique to geometrically verify swirling features in flow fields. They used the geometric properties of the streamlines around the vortex cores to distinguish the false positives from the true swirling features. The concept of the 'probe vector', which is normal to the streamline and points towards the vortex core, is key in this geometric verification. Whether the probe vectors along streamline cover an angle bigger or smaller than  $2\pi$  is the criterion that distinguishes the false positives from true swirling motions according to *Jiang et al.* [8].

The concept of the  $2\pi$  criterion will be implemented in the extraction and calculation of the vortex properties. It will be used to ensure the complete integration around the section of the vortex ring.

### 2.4 Circulation, Impulse and Force generated by a Vortex

*Saffman* in his book *Vortex Dynamics* [6] gathers the theory on behaviour of vortices in two and three dimensions. Section 1.3 covers circulation, where the circulation  $\Gamma$  is defined as the line integral of the velocity, as shown in equation 4.

$$\Gamma = \oint_C \vec{u} \cdot d\vec{s} \quad (4)$$

Where  $C$  is a closed curve around the vortex core,  $\vec{u}$  is the velocity field, and  $d\vec{s}$  is a differential of a vortex line.

Furthermore, if the closed curve is reducible, the circulation will be equal to the flux of vorticity through an open surface, as can be seen in equation 5.

$$\Gamma = \oint_C \vec{u} \cdot d\vec{s} = \oint_A \vec{\omega} \cdot \vec{n} \, dS \quad (5)$$

Where  $A$  is the the surface reduced from the closed curve  $C$ ,  $dS$  is the differential of area  $A$ ,  $\vec{n}$  is the vector normal to  $A$ , and  $\omega$  the vorticity.

Impulse of isolated vortices is covered in section 3.7, in the derivation *Saffman* assumes an isolated vortex surrounded by irrotational fluid in an external velocity field. Where the vortex occupies a volume  $V_V$  which has a surface associated with it  $S_V$ .

The impulse of the vortex  $\vec{I}_V$  is defined, according to *Saffman*, as a function of the location of the centroid and the vorticity of the vortex, as shown in equation 6.

$$\vec{I}_V = \frac{1}{2} \int \vec{x} \times \vec{\omega} \, dV_V \quad (6)$$

Where  $\vec{x}$  is the location of the centroid,  $\omega$  is the vorticity along the vortex, and  $V_V$  is the volume occupied by the vortex which produces the impulse  $\vec{I}_V$ .

The velocity field surrounding the vortex  $\vec{U}$ , according to *Saffman* [6], can be split into the part induced by itself  $\vec{U}_v$  and the one induced by external factors  $\vec{U}_e$ . As shown by equation 7, the induced velocity  $\vec{U}_v$  is given by integrating over the vortex volume  $V_V$  and the velocity due to external factors  $\vec{U}_e$  is caused by other vortices or by the movement of bodies.

$$\vec{U} = \vec{U}_v + \vec{U}_e \quad (7)$$

According to *Saffman* [6], the rate of change on the impulse of a vortex, defined in equation 6, can also be related to the force exerted by the vortex in its surroundings. Equation 8 takes into consideration that the change with time of the impulse of the vortex is affected by the body who generated the vortex, which applied the force  $\vec{F}$  to the fluid, and by any other features that contribute to generate the local velocity field due to external factors  $\vec{U}_e$ .

$$\frac{d\vec{I}_V}{dt} = \int \vec{F} \, dV + \int \vec{U}_e \times \vec{\omega} \, dV_V \quad (8)$$

The equations showed above define the main vortical properties. These are the main pieces of information which are key to vortical flow fields. They will be introduced to the algorithm in order to provide an informative visualisation of the vortical flow field.

## 2.5 Particle Visualisation

*Hin and Post* (1993) [9] studied a three dimensional turbulent field by analysing the evolution of a number of particles in time. They applyed their method in a turbulent channel flow and showed the motion of the particles in the field.

In order to evaluate the particles in time the velocity field was integrated for every particle and time step. During the integration of the velocity, its value at the various points was interpolated between the nodes of the mesh.

$$\vec{x}_{i+1} = \vec{x}_i + \int_{t_i}^{t_{i+1}} \vec{u}(\vec{x}, t) dt \quad (9)$$

Where the initial position is  $\vec{x}_i$ , the velocity field is  $\vec{u}$ , and the final position once the integration has been carried out is  $\vec{x}_{i+1}$ .

This integration approach used to calculate the evolution of the particles with time will be used to calculate the vortical properties, which will be obtained by integrating over the velocity flow field.

## 2.6 Visualisation of Information

*Tufte* (2001) published his book *The Visual Display of Quantitative Information* [10], where he took a scientific approach to analysing the visual display of data. According to *Tufte* [10] graphical competence demands three skills:

- The substantive
- The statistical
- The artistic

He also stated that '*Allowing artist-illustrators to control the design and content of statistical graphics if almost like allowing typographers to control the content, style and editing of prose*', emphasising the importance of the information displayed in graphs.

*Tufte* [10] defined a number of parameters which could be extracted from any visual display of data. These parameters provide key information in order to carry out an analysis of the approach taken in the visualisation of the data.

According to *Tufte* [10] the level distortion of the data shown is measured by the Lie Factor, which accounts for the difference between the data and the visual representation of it. The Lie Factor is defined as the ratio between the visual change of the data and the change in the data itself, as can be seen in equation 10.

$$\text{Lie Factor} = \frac{\text{size of effect shown in graphic}}{\text{size of effect in data}} \quad (10)$$

Lie factors higher than 1.05% or lower than 95% are an indication of distortion in the display of the data.

*Tufte* [10] defends in his book that a large part of the ink used in the graph should be used in displaying the data. This part of the ink can not be erased without losing information about the data. *Tufte* [10] stated in his book 'Above all show the data', emphasising this idea, which constitutes the fundamental concept behind graphs. The ratio of the portion of the ink which presents the data with respect to the overall ink used in the graph is defined as the Data-Ink ratio, as shown in equation 11.

$$\text{Data-ink ratio} = \frac{\text{data-ink}}{\text{total ink used to print the graphic}} \quad (11)$$

It measures how effective the main building block of a graph, the ink, is useful in carrying out the graph's main purpose, visually display the data. The Data-Ink ratio is a measure of the excellence of the graphic.

The idea of maximising the Data-ink ratio, according to Tufte, is the key to the theory behind graphic display of information. This concept will be embraced and adopted when developing the visualisation algorithm. The Lie Factor will also be kept in mind and used to check the graphical integrity of the vortex information displayed.

### 3 Visualisation Problem

In the preface of a well known book on turbulence, ‘*A first course in turbulence*’ by Tennekes and Lumley [11], it can be read ‘*In the customary description of turbulence, there are always more unknowns than equations.*’ This statement presents a problem which is known to both to the identification and visualisation of vortices. The combination of environments, vortex arrangements, and flow regimes, make the task of coherent vortical visualisation far from trivial. It is made even more challenging since it should be approached in an elegant and simple way, numerically, computationally, and visually.

It is worth mentioning that taking a very complex velocity field as a starting point would make the extraction of information from the vortices extremely difficult. If a simple field can be analysed successfully, the basic and effective concepts developed can be taken as a starting point for developing methods that can be applied to more complex flows.

However, if the vortical structures in the field are trivial, the methods developed to visualise them would not be applicable to complex fields, thus defeating the point of their development.

The question is then presented as how to simplify the field in order to reduce the number of initial unknowns whilst maintaining a reasonable resemblance with more complex flows. This field could then be used for the initial stages of development of the visualisation algorithm. Its simplicity would allow to easily carry out and implement, basic numerical methods.

#### 3.1 Simplifying a Difficult Problem

The complexity of vortical flows leads us to search for a simplified velocity field. The simplification of the field needs to be done in such a way that it maintains geometric and vortical properties which resemble the more complex vortical flows. This is key for the application of the visualisation algorithm to more realistic and complex velocity fields.

In order to meet these requirements, the option of a self generated velocity field was chosen. The ability to have prior knowledge about the velocity field is a useful tool in the development of the algorithm. One must realise the need of the algorithm to be ‘field-agnostic’, so if an unknown field is given, the algorithm would be capable of carrying out the same calculations.

The wake of a fish in deep water was chosen as a field to generate and subsequently analyse. The reasoning behind this decision is the geometric simplicity of the wake and the clearness of its vortical structures. The wake is constituted by a number of alternating vortex rings. By taking an individual ring, the field is simple enough so that initially the methodology and knowledge can be quickly gained. The wake constitutes in itself a realistic example of a vortical flow field, meeting the condition of applicability to more realistic examples.

### 3.2 Forwards and Backwards Problems

Having prior knowledge of the velocity field gives an advantage in the development and testing of the visualisation tools. But the main advantage is the knowledge of the interaction between the variables that generate the vortical structures. For example, knowing the expressions that govern the magnitude and direction of the velocity field make it easy to find an estimate for the circulation along the vortex ring. Similarly, knowing the shape of the vortex ring makes very easy to find the direction of the impulse generated by the vortex ring.

The modelling and visualisation problem addressed in this project has two well defined parts. Firstly, to generate the velocity field through empirical methods. Once the velocity field has been generated, the second part is to obtain the vortex information ‘hidden’ in the velocity field. If the relation between the two parts is considered, the problem is a well defined forward and backward problem.

Taking a closer look at the forward part, the generation of the velocity field. The whole task can be divided in various parts. The development and testing of the various algorithms to generate, firstly the rings, and after that the velocity field induced by them. Once the field has been produced, it is key to maintain the forward problem independent from the backwards one. In order to do so the velocity field will need to be written to an image file. This will ensure both the ability to clearly split both parts of the overall problem and to read any ‘unknown’ field once the algorithm has been strengthened to accommodate for more complex vortical flow fields.

The backwards problem takes the previously generated and written velocity field, and retrieves from it the properties of the vortical structures. The backwards problem has the disadvantage of this is the the information loss during the generation of the field. The main source of loss of information is the possibility of noise present in the velocity field. This noise could be due to various factors, such as inaccurate dynamic measurements, human or calibration errors in tank testing, or numerical inaccuracies during computation, depending on the way the data has been generated. The information loss will contribute to increasing the difficulty of developing the information extraction part within the visualisation algorithm; but can be mitigated in various ways. The most important way is that the effect the vortex is a local feature in the flow field. This concept can be used to the advantage of the information extraction part by keeping the algorithm local to the vortex ring or a section within the ring.

## 4 Velocity Field Generation Algorithm

This section covers the development of the algorithms present in the forward part of the visualisation problem. That is the velocity field generation, and the writing of the field in a *.vtu* image file.

### 4.1 Velocity Field Generation

The generation of the velocity field is constituted by a number of parts:

1. Definition of the velocity field where the rings will be located.

2. Geometric definition of the vortex rings.
3. Induced velocity calculations.
4. Integration of the induced velocities for all the rings in the field at every point.
5. Visually display the vectors corresponding to the velocity field.

The field is defined in Python by six, three dimensional matrices, each of them containing the  $X$ ,  $Y$ , and  $Z$  coordinates and the three velocity components  $U$ ,  $V$ , and  $W$ .

The vortex rings are defined in space by a vector normal to the plane, the radius, and number of points defining the ring.

The induced velocity calculations are carried out by treating each of the segments constituting each vortex ring as a vortex line. By applying the Biot-Savart Law at each of the segments the contribution to the induced velocity of the segments is calculated.

Integrating all the contributions of all the rings to each of the points of the field the induced velocity field due to the vortex rings is obtained.

An outline of the overall velocity generation general algorithm is given below, in algorithm 1.

```

Input: Number, location, radius, and normal vector of vortex ring/s.
Output:  $X, Y, Z, U, V, W$  three dimensional fields.

1 rings = []
2 for n in ring definition data do
3   | ring [n] = function(vortex ring from geodefinition)
4 end
5 vortex segments = []
6 for n in rings do
7   | vortex segments[n] = function(vortex segments from ring)
8 end
9  $\vec{X} = [X, Y, Z]$ 
10  $\vec{U} = [U, V, W]$ 
11 for i do
12   for j do
13     for k do
14       |  $\vec{p}_{temp} = \vec{X}(i, j, k)$ 
15       |  $\vec{u}_{temp} = 0$ 
16       for r in rings do
17         for s in segments do
18           |  $\vec{u}_{temp} += \text{function(induced velocity from vortex segment)}$ 
19         end
20       end
21       |  $\vec{U} = \vec{u}_{temp}$ 
22     end
23   end
24 end
25 return [ $\vec{X}, \vec{U}$ ]

```

**Algorithm 1:** Algorithm that generates the velocity field given the number, location, radius, and normal vector of vortex ring/s.

#### 4.1.1 Rankine Vortex

The algorithm above combined with the Biot-Savart Law for the induced vortical velocity field produces a velocity field. However, if the point where the velocity field is calculated falls too close to the vortex segment, the induced velocity will have an infinite magnitude. Furthermore, at a later stage, where the  $\lambda_2$  definition of a vortex is implemented, it will present difficulties with finding the vortex core because it is of an infinitesimally small thickness. For this reason, a new approach for defining the vortex core had to be taken, where the core is given a finite thickness.

The rankine vortex is a very good approximation of a real vortex with a finite core thickness. Figure 3 shows the velocity distribution along a section of a Rankine Vortex.

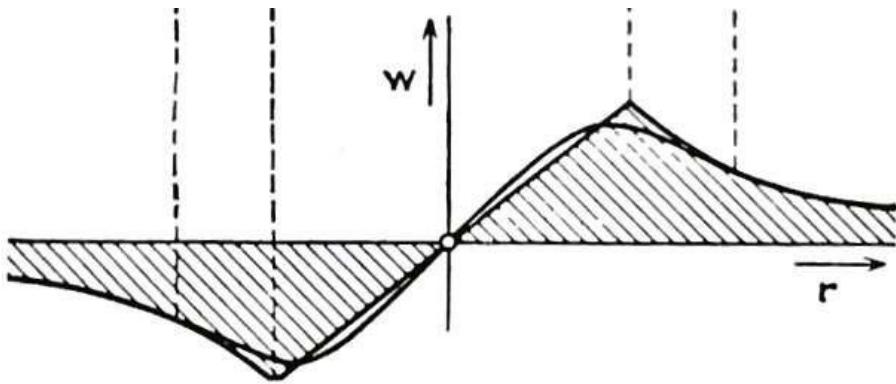


Figure 3: Velocity profile in a section through a rankine vortex.

In figure 3 the line shows the velocity distribution on a real vortex, whereas the shaded area is the induced velocity due to the rankine vortex. This figure is taken from figure 140 in *Fundamentals of Hydro- and Aero-Mechanics* [3].

As it can be seen from figure 3, the velocity field induced by a rankine vortex has two different parts, inside and outside the core of the vortex. The velocity field inside the vortex core is a linear distribution of velocity. The field outside the vortex core experiences a maximum at the edge of the core radius and decays with the ratio of  $1/|\vec{r}|$ . Detailed equations on the velocity field for both inside and outside the core, in a 2- and 3-dimensional field are shown below, in equations 12 and 13.

$$\vec{u}_{inside\ 2d} = \frac{\Gamma}{2\pi} \frac{\vec{r} \times d\vec{s}}{R} \quad \vec{u}_{outside\ 2d} = \frac{\Gamma}{2\pi} \frac{R}{|\vec{r}|^2} \vec{r} \times d\vec{s} \quad (12)$$

$$\vec{u}_{inside\ 3d} = \frac{\Gamma}{4\pi} \oint \frac{\vec{r} \times d\vec{s}}{R} \quad \vec{u}_{outside\ 3d} = \frac{\Gamma}{4\pi} \oint \frac{R}{|\vec{r}|^2} \vec{r} \times d\vec{s} \quad (13)$$

Where all the symbols have their usual and previously defined meaning, and  $R$  is the radius of the vortex core.

By using the rankine vortex definition instead of the the Biot-Savart Law for the induced vortical velocity field, the definition of the velocity field is significantly improved. Furthermore, the  $\lambda_2$  definition of a vortex core is significantly more informative and the differences with real vortex tubes is reduced.

## 4.2 Velocity field .vtu Image Reading and Writing

Writing the velocity field to an image file is a key part in ensuring the forward part remains separate from the backwards one. Furthermore, if the visualisation algorithm was able to read image files, the requirement of being able to use it a ‘field-agnostic’ way is simplified.

The *.vtu* format is the unstructured grid format within the VTK toolkit. The VTK toolkit is part of the ‘back-end’ of visualisation programs such as ParaView, commonly used in OpenFoam visualisations, and Mayavi, widely used in Python 3d visualisations.

Being able to read an unstructured grid provides a good starting point in the algorithm’s ability to be ‘field-agnostic’, but the lack of data points could rise the need for a partial estimated reconstruction of the field.

## 5 Visualisation Algorithm

This section covers a detailed description of the development of the visualisation algorithm, its various parts, and its methods. The description starts once the data has been read from the *.vtu* image file and can be split in the following components:

1. Calculations concerning the obtention of the  $\lambda_2$  field.
2. Generation of seeding points in the vortex core regions.
3. Evaluate seeds in time.
4. Circulation integrals around the vortices.
5. Calculation and display of the approximate circulation field.

### 5.1 $\lambda_2$ Definition of a Vortex Core

The first step in the visualisation of the vortical flow field is to find the vortex cores themselves. In order to accurately locate the cores in the flow field, a strong mathematical definition is needed.

*Jeong* and *Hussain* defined a vortex core as ‘*a connected region with two negative eigenvalues of  $S^2 + \Omega^2$* ’ [1]. Where  $S$  and  $\Omega$  are the symmetric and antisymmetric parts of the velocity gradient tensor  $\nabla u$ . That is to say  $\nabla u_{ij} = S_{ij} + \Omega_{ij}$ , where  $S_{ij} = \frac{1}{2}(u_{ij} + u_{ji})$  and  $\Omega_{ij} = \frac{1}{2}(u_{ij} - u_{ji})$ . So if  $\lambda_1$ ,  $\lambda_2$ , and  $\lambda_3$  are the three eigenvalues of  $S^2 + \Omega^2$ ,  $\lambda_2$  is negative within the vortex core.

This definition was chosen among many others for the taking advantage of the local aspect of the vortex. Furthermore the definition is only dependant on whether the flow does or not present swirling motion. The  $\lambda_2$  field is formed by the value of the second eigenvalue in magnitude ( $\lambda_1 \geq \lambda_2 \geq \lambda_3$ ),  $\lambda_2 < 0$  in a vortex core.

The algorithm calculates for every point the gradient of the field, decomposes it in symmetric and antisymmetric parts, calculates the  $S^2 + \Omega^2$  matrix, finds its eigenvalues and takes the second eigenvalue in magnitude. An outline of the algorithm is shown in algorithm 2.

```

Input:  $X, Y, Z, U, V, W$  three dimensional fields.
Output:  $\lambda_{2field}$ .
1  $\vec{X} = [X, Y, Z]$ 
2  $\vec{U} = [U, V, W]$ 
3 for  $i$  do
4   for  $j$  do
5     for  $k$  do
6        $p_{temp} = \vec{X}(i,j,k)$ 
7        $\nabla \vec{U}$  (at  $p_{temp}$ )
8        $S_{matrix}$ 
9        $\Omega_{matrix}$ 
10       $(S^2 + \Omega^2)_{matrix}$ 
11       $\lambda_{2field}(i,j,k) = \lambda_2$  of  $(S^2 + \Omega^2)_{matrix}$ 
12    end
13  end
14 end
15 return  $\lambda_{2field}$ 

```

**Algorithm 2:** The lambda 2 algorithm which calculates the  $\lambda_2$  scalar for every point in the velocity field.

## 5.2 Analytical derivations of $\lambda_2$

One of the merits of the  $\lambda_2$  definition of a vortex core is its ability to distinguish between swirling motion of the fluid and vorticity. This is key since there is a direct relationship between vortices and vorticity; but whenever in a field there is swirling motion, it will be due to the presence of a vortex.

This section covers the analytical derivation of the  $\lambda_2$  criterion for the identification of vortices. The derivation is carried out for three cases in a 2-dimensional field. The first two cover the velocity field induced by a Rankine vortex. The first derivation is inside and the second one is outside the core of the vortex. The last derivation covers the field induced by a shear layer.

On the first two derivations, the value of the  $\lambda_2$  scalar is negative, due to the presence of swirling. In the third case the field is only a shear layer, where vorticity is present but there is no swirling motion. It is then expected that the  $\lambda_2$  scalar would be zero.

### 5.2.1 Inside the core

The velocity field inside the core can be defined as:

$$\vec{u} = \frac{\Gamma}{2\pi} \frac{\vec{r} \times d\vec{s}}{R} \quad (14)$$

Where  $\Gamma$  is the strength of the vortex,  $\vec{r}$  is the vector from the center of the vortex core to the point where the velocity is induced,  $\hat{s}$  is the tangent vector along the vortex line at the point where the velocity field is induced from, and  $R$  is the radius of the vortex core.

The velocity field can be decomposed in  $x$  and  $y$  components, as can be seen in equations 15a and 15b.

$$u_x = -\frac{\Gamma}{2\pi R} \frac{1}{R} y \hat{i} \quad (15a)$$

$$u_y = \frac{\Gamma}{2\pi R} \frac{1}{R} x \hat{j} \quad (15b)$$

Having decomposed the velocity in its components, the jacobian matrix of the velocity field  $J_{\vec{u}}$  can be calculated:

$$J_{\vec{u}} = \frac{d\vec{u}}{dx} = \begin{pmatrix} \frac{\partial u_x}{\partial x} & \frac{\partial u_x}{\partial y} \\ \frac{\partial u_y}{\partial x} & \frac{\partial u_y}{\partial y} \end{pmatrix} = \begin{pmatrix} 0 & -\frac{\Gamma}{2\pi R} \frac{1}{R} \\ \frac{\Gamma}{2\pi R} \frac{1}{R} & 0 \end{pmatrix} \quad (16)$$

Each element can be decomposed in its symmetric and antisymmetric parts:

$$J_{u_{ij}} = S_{ij} + \Omega_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) + \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) \quad (17)$$

The symmetric part can be defined and calculated as:

$$S = \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix}; S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (18a)$$

$$S = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}; S^2 = SS = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \quad (18b)$$

Similarly, the antisymmetric part can be defined and calculated as:

$$\Omega = \begin{pmatrix} \Omega_{11} & \Omega_{12} \\ \Omega_{21} & \Omega_{22} \end{pmatrix}; \Omega_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) \quad (19a)$$

$$\Omega = \begin{pmatrix} 0 & -\frac{\Gamma}{2\pi R} \\ \frac{\Gamma}{2\pi R} & 0 \end{pmatrix}; \Omega^2 = \Omega\Omega = \begin{pmatrix} -\left(\frac{\Gamma}{2\pi R}\right)^2 & 0 \\ 0 & -\left(\frac{\Gamma}{2\pi R}\right)^2 \end{pmatrix} \quad (19b)$$

Then the matrix  $S^2 + \Omega^2$  can be constructed:

$$S^2 + \Omega^2 = (SS + \Omega\Omega) = \begin{pmatrix} -\left(\frac{\Gamma}{2\pi R}\right)^2 & 0 \\ 0 & -\left(\frac{\Gamma}{2\pi R}\right)^2 \end{pmatrix} \quad (20)$$

Furthermore, the eigenvalues of the  $S^2 + \Omega^2$  matrix can be obtained, hence the value of the second eigenvalue:

$$\lambda_1 = -\left(\frac{\Gamma}{2\pi R}\right)^2, \lambda_2 = -\left(\frac{\Gamma}{2\pi R}\right)^2 \quad (21)$$

### 5.2.2 Outside the core

With a similar approach to the derivation inside the core the region of the field outside the core can be evaluated. The velocity field outside the core can be defined as:

$$\vec{u} = \frac{\Gamma}{2\pi} \frac{R}{|\vec{r}|^2} \vec{r} \times d\vec{s} \quad (22)$$

Where  $\Gamma$  is the strength of the vortex,  $\vec{r}$  is the vector from the center of the vortex core to the point where the velocity is calculated,  $\hat{s}$  is the tangent vector along the vortex line at the point where the velocity field is induced from, and  $R$  is the radius of the vortex core.

Decomposing in  $x$  and  $y$  components the velocity field:

$$u_x = \frac{\Gamma}{2\pi} \frac{R}{|\vec{r}|^2} y \hat{i} \quad (23a)$$

$$u_y = -\frac{\Gamma}{2\pi} \frac{R}{|\vec{r}|^2} x \hat{j} \quad (23b)$$

The jacobian of the velocity field  $J_{\vec{u}}$  is calculated:

$$J_{\vec{u}} = \frac{d\vec{u}}{d\vec{x}} = \begin{pmatrix} \frac{\partial u_x}{\partial x} & \frac{\partial u_x}{\partial y} \\ \frac{\partial u_y}{\partial x} & \frac{\partial u_y}{\partial y} \end{pmatrix} = \begin{pmatrix} 0 & -\frac{\Gamma}{2\pi} \frac{R}{|\vec{r}|^2} \\ \frac{\Gamma}{2\pi} \frac{R}{|\vec{r}|^2} & 0 \end{pmatrix} \quad (24)$$

Each element can be decomposed in its symmetric and antisymmetric parts:

$$J_{u_{ij}} = S_{ij} + \Omega_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) + \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) \quad (25)$$

The symmetric part is defined and calculated:

$$S = \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix}; S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (26a)$$

$$S = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}; S^2 = SS = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \quad (26b)$$

Similarly, the antisymmetric part is defined and calculated as:

$$\Omega = \begin{pmatrix} \Omega_{11} & \Omega_{12} \\ \Omega_{21} & \Omega_{22} \end{pmatrix}; \Omega_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) \quad (27a)$$

$$\Omega = \begin{pmatrix} 0 & -\frac{\Gamma}{2\pi} \frac{R}{|\vec{r}|^2} \\ \frac{\Gamma}{2\pi} \frac{R}{|\vec{r}|^2} & 0 \end{pmatrix}; \Omega^2 = \Omega\Omega = \begin{pmatrix} -\left(\frac{\Gamma}{2\pi} \frac{R}{|\vec{r}|^2}\right)^2 & 0 \\ 0 & -\left(\frac{\Gamma}{2\pi} \frac{R}{|\vec{r}|^2}\right)^2 \end{pmatrix} \quad (27b)$$

Then the matrix  $S^2 + \Omega^2$  can be constructed:

$$S^2 + \Omega^2 = (SS + \Omega\Omega) = \begin{pmatrix} -\left(\frac{\Gamma}{2\pi}\frac{R}{|\vec{r}|^2}\right)^2 & 0 \\ 0 & -\left(\frac{\Gamma}{2\pi}\frac{R}{|\vec{r}|^2}\right)^2 \end{pmatrix} \quad (28)$$

Furthermore, the eigenvalues of the  $S^2 + \Omega^2$  matrix can be obtained, hence the value of the second eigenvalue:

$$\lambda_1 = -\left(\frac{\Gamma}{2\pi}\frac{R}{|\vec{r}|^2}\right)^2, \quad \lambda_2 = -\left(\frac{\Gamma}{2\pi}\frac{R}{|\vec{r}|^2}\right)^2 \quad (29)$$

### 5.2.3 Shear layer

The velocity field induced by a shear layer can be defined by the components:

$$u_x = u_\tau y \nu \hat{i} \quad (30a)$$

$$u_y = 0 \hat{j} \quad (30b)$$

The jacobian of the velocity field  $J_{\vec{u}}$  is calculated:

$$J_{\vec{u}} = \frac{d\vec{u}}{d\vec{x}} = \begin{pmatrix} \frac{\partial u_x}{\partial x} & \frac{\partial u_x}{\partial y} \\ \frac{\partial u_y}{\partial x} & \frac{\partial u_y}{\partial y} \end{pmatrix} = \begin{pmatrix} 0 & u_\tau \nu \\ 0 & 0 \end{pmatrix} \quad (31)$$

Each element can be decomposed in its symmetric and antisymmetric parts:

$$J_{u_{ij}} = S_{ij} + \Omega_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) + \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) \quad (32)$$

The symmetric part is defined and calculated:

$$S = \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix}; \quad S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (33a)$$

$$S = \begin{pmatrix} 0 & \frac{1}{2}u_\tau \nu \\ \frac{1}{2}u_\tau \nu & 0 \end{pmatrix}; \quad S^2 = SS = \begin{pmatrix} 0 & \left(\frac{1}{2}u_\tau \nu\right)^2 \\ \left(\frac{1}{2}u_\tau \nu\right)^2 & 0 \end{pmatrix} \quad (33b)$$

Similarly, the antisymmetric part is defined and calculated as:

$$\Omega = \begin{pmatrix} \Omega_{11} & \Omega_{12} \\ \Omega_{21} & \Omega_{22} \end{pmatrix}; \quad \Omega_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) \quad (34a)$$

$$\Omega = \begin{pmatrix} 0 & \frac{1}{2}u_\tau \nu \\ -\frac{1}{2}u_\tau \nu & 0 \end{pmatrix}; \quad \Omega^2 = \Omega\Omega = \begin{pmatrix} 0 & -\left(\frac{1}{2}u_\tau \nu\right)^2 \\ -\left(\frac{1}{2}u_\tau \nu\right)^2 & 0 \end{pmatrix} \quad (34b)$$

Then the matrix  $S^2 + \Omega^2$  can be constructed:

$$S^2 + \Omega^2 = (SS + \Omega\Omega) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \quad (35)$$

Furthermore, the eigenvalues of the  $S^2 + \Omega^2$  matrix can be obtained, hence the value of the second eigenvalue:

$$\lambda_1 = 0, \lambda_2 = 0 \quad (36)$$

### 5.3 Generation of Seeding Points

Once the  $\lambda_2$  field is calculated, it is taken as a starting point for the searching of seeding points. The purpose of the seeding points is to use them to obtain the properties of the vortex. The calculation of the properties is based on the evaluation of the seeds in time. This is due to the fact that the seeds travel on the velocity field which has been generated by the vortex, so if the seeds are studied in time, the information from the vortex can be retrieved from the field.

Since the vortex is a local feature of the flow the seeds need to be close to the vortex core. This is due to the effect of any other feature in the velocity field will be discarded by the particle, because its influence in the flow field near the vortex is negligible. It was decided that the best location for the seeds would be inside but close to the edge of the vortex core.

In order to locate the edge of the vortex core, the  $\lambda_2$  field was searched for local minima, and the corresponding point in the position field was taken as a seeding point. In order to avoid any fluctuations in the value of the  $\lambda_2$  field, a second requirement of negative  $\lambda_2$  value was introduced. An outline of the algorithm is given below:

```

Input:  $X, Y, Z, \lambda_2$  three dimensional fields.
Output: Seeding points.
1  $\vec{X} = X, Y, Z$ 
2 seeding points = [ ]
3 for  $i$  do
4   for  $j$  do
5     for  $k$  do
6       if  $\lambda_2(i,j,k) < 0$  then
7         if  $\lambda_2(i,j,k)$  is a local min then
8           | add  $\vec{X}(i,j,k)$  to seeding points
9         end
10      end
11    end
12  end
13 end
14 return seeding points

```

**Algorithm 3:** Algorithm that searches for seeding points in the velocity and  $\lambda_2$  fields.

### 5.4 Evaluation of Seeds with Time

Once the seeding points have been obtained, they are evaluated in time in order to capture the effect of the field on the particles. Noting that the field has been altered by the presence

of the vortices, the effect that the particle receives is key in the obtention of the properties of the vortices.

In order to evaluate the seeds in time, the velocity field needs to be integrated. According to *Hin* and *Post* [9], the field is integrated for every time step following equation 37.

$$\vec{x}_{i+1} = \vec{x}_i + \int_{t_i}^{t_{i+1}} \vec{u}(\vec{x}, t) dt \quad (37)$$

This can be implemented numerically by the following second order approximation:

$$\vec{x}_{i+1}'' = \vec{x}_i + \left( \frac{1}{2} \Delta t \vec{u}_i \right) + \left( \frac{1}{2} \Delta t \vec{u}_{i+1} \right) \quad (38)$$

$$\vec{u}_{i+1} = \vec{x}_i + (\Delta t \vec{u}_i) \quad (39)$$

Although the second order estimate is a reasonable approximation of the new position of the particle it was found not accurate enough. Seeded particles were studied in time, within the simple, one ring fields, and the results proved the need for a more accurate approximation. A third order was therefore implemented as it can be seen in equation 40.

$$\vec{x}_{i+1}''' = \vec{x}_i + \left( \frac{1}{3} \Delta t \vec{u}_i \right) + \left( \frac{1}{3} \Delta t \vec{u}_{i+1} \right) + \left( \frac{1}{3} \Delta t \vec{u}_{i+1}'' \right) \quad (40)$$

To ensure the completion of the path around the section of the vortex ring, *Jiang*, *Machiraju*, and *Thompson* [8] developed a criterion based on the completion of a  $2\pi$  angle of the probe vectors as the particle traveled around the vortex. This concept was taken in order to develop a different way to ensure that the particle does travel around the vortex ring. The angle between the tangent vectors along the path of the particle was calculated. If the summation of all the angles was bigger or equal to  $2\pi$ , the particle had traveled around the vortex.

An outline of the evolution of the particle with time, and  $2\pi$  criterion algorithm is presented below, in algorithm 4.

```

Input:  $\vec{X}, \vec{U}$ , seed.
Output: Path and velocity of seed with time.

1 t = 10
2 seed  $2\pi = 0$ 
3 while seed  $2\pi = 0$  do
4   path,vel = function(evaluates seed during time t in  $\vec{U}$ )
5   angles = function(calculates angle of tangent vectors along the path)
6   if angles  $\geq 2\pi$  then
7     | seed  $2\pi = 1$ 
8     | return [path, vel]
9   else
10    | t += 2
11  end
12 end
```

**Algorithm 4:** Algorithm that evaluates a seed in time until it completes a circumference around the vortex.

## 5.5 Circulation Integrals and Field

Once the seeds have been evaluated in time, the circulation can be obtained from the paths of the seeds around the vortices. According to *Saffman* [6], the circulation can be defined as the integral of the velocity around a closed path around the vortex.

$$\Gamma = \oint_C \vec{u} d\vec{s} \quad (41)$$

This is implemented by carrying the summation of the velocities corresponding to the evolution of the seeds in time.

Once the circulation at the seeded points has been obtained, this information is used to complete the circulation field. This is achieved by combining the information from the  $\lambda_2$  field regarding the location of the vortex cores and the circulation at the seeding points. Since the seeding points are in the vortex core, neighbouring points within the same ring section will have the same circulation. The circulation on the points some distance away from the seed within the same core, could be estimated to have the same circulation. However, if the visualisation algorithm has the need of being ‘field-agnostic’, this last assumption is not valid. Therefore a linear distribution away from the seeded point was assumed.

In order to distinguish neighbouring points outside the core from those inside the core the sign of the  $\lambda_2$  field was investigated. If the sign is negative, the point will be inside the core whereas the positive sign means that the point is outside the core.

To define the in-pane and normal directions of the section of the vortex, a vector tangent to the vortex ring was defined. In order to maintain the method ‘field-agnostic’, the direction was obtained by taking the cross product of the tangent vectors along the path of the seed. By using this method, the flow field, which has been altered and defined by the vortices, dictates the direction tangent to the vortex line defining the ring.

When the vector tangent to the vortex line, which is normal to the swirl plane, has been obtained the points in the plane need to be identified, and the distance from the swirl plane needs to be defined. Geometrically the points that are inside a plane have a zero dot product with a vector normal to the plane, since the angle between them is 90 degrees, and the cosine of 90 degrees is equal to zero.

$$\vec{a} \cdot \vec{b} = |\vec{a}| |\vec{b}| \cos(\alpha) \quad (42)$$

Being  $\alpha$  the angle between the vectors  $\vec{a}$  and  $\vec{b}$ , whose modulus are  $|\vec{a}|$  and  $|\vec{b}|$  respectively.

The distance between a plane and a point can be obtained from the dot product of the vector normal to the plane and a vector from a point in the vector to the point of interest. Rearranging, the expression below can be obtained:

$$distance = \hat{n} \cdot \vec{AB} \quad (43)$$

Where  $A$  and  $B$  are two points in the field,  $A$  is contained in the plane of interest, and  $B$  is the point where the distance is calculated.

Looping through the whole circulation field in order to assign values of the circulation is a very computationally expensive task. Furthermore looping through points where no value of circulation can not be assigned is not a very efficient approach. To overcome this the

loops where limited to 10 cells before and after the seeded point in every direction. In this way the algorithm loops through less number of points, avoiding the possible inefficiency. The outline of the algorithm is shown below:

```

Input: Velocity of a seeds with time,  $\lambda_2$ field.
Output:  $Circulation$ field.
1 circulation of seeds = []
2 for s in seeds do
3   | function(circulation given the velocity with time of the seed)
4 end
5 dist = d
6 for c in circulation of seeds do
7   | for i in ± dist do
8     |   | for j in ± dist do
9       |     |   | for k in ± dist do
10      |       |     |   |  $p_{temp} = \vec{X}(i,j,k)$ 
11      |       |     |   | if  $\lambda_2(p_{temp}) < 0$  then
12        |         |       |   | if  $p_{temp}$  in swirl plane then
13          |           |         |   |   |  $Circ_{field}(p_{temp}) = c$ 
14          |           |         |   |   | else
15            |             |           |   |   |  $Circ_{field}(p_{temp}) =$  linear distribution of c from swirl plane
16            |             |           |   |   | end
17          |           |         |   |   | else
18            |             |           |   |   | nothing
19            |             |           |   |   | end
20        |       |     |   | end
21      |   | end
22    | end
23 end
24 return  $Circulation$ field
```

**Algorithm 5:** Algorithm that calculates the circulation at the seeds and completed the circulation field.

## 5.6 General Visualisation Algorithm

The algorithms described above are combined in the general visualisation algorithm. The general algorithm reads a *.vtu* image file, calculates the  $\lambda_2$  field, obtains and evaluates in time the seeding points, and calculates and completes the circulation field. An outline of the algorithm is shown in algorithm 6

**Input:**  $X, Y, Z, U, V, W$  three dimensional fields.

**Output:** Approximate circulation field

```

1  $\vec{X} = X, Y, Z$ 
2  $\vec{U} = U, V, W$ 
3 L2 field = function(calculates L2 field)
4 Seed points = function(seeding points based on L2 field)
5 Circulation field = []
6 for  $s$  in seeds do
7   | function(evaluates seeds in time until  $2\pi$ )
8 end
9 for  $p$  in Circfield do
10  | function(assigns circulation within Circfield
11 end
12 return Circulation field

```

**Algorithm 6:** Algorithm that generates and completes the circulation field combining the use of the  $\lambda_2$  field and the geometric properties of the flow field.

## 6 Implementation of the General Algorithm

The general algorithms described in the previous section were implemented in the Python programming language. Python was chosen for the large online documentation and its easy to read syntax. Although the easy syntax is gained in detriment of performance, the speed gained in development was thought to be beneficial especially in a short project involving a relatively long and new task. In order to carry out default vector, matrix, and mathematical operations, the Numpy [12] and Math [13] libraries were used.

The implementation was carried out in an object oriented style, splitting the tasks into functions, which again broke up into more subtasks and subfunctions. This way if a function or method has an error or is not producing the desired output it can be modified as a self-contained module. Furthermore, the functions can be called by other methods reducing the amount of rewriting needed.

## 7 Results

This section covers the results obtained throughout the implementation of the general algorithm described in previous sections. The results begin with the generation of the velocity field. Continue with the implementation of the  $\lambda_2$  criterion for vortex identification. And conclude with the visualisation of the circulation field.

### 7.1 Velocity Field Generation

A large number of velocity fields were generated during this research project. At the very initial generation of the velocity field, simple 2-dimensional flow fields were generated. At a later stage in the velocity field generation part, single vortex rings were generated in order to provide a starting point in which to develop the initial visualisation functions. In order to

arrive at more realistic velocity fields, in the final stages of the project, fields with multiple vortex rings where developed.

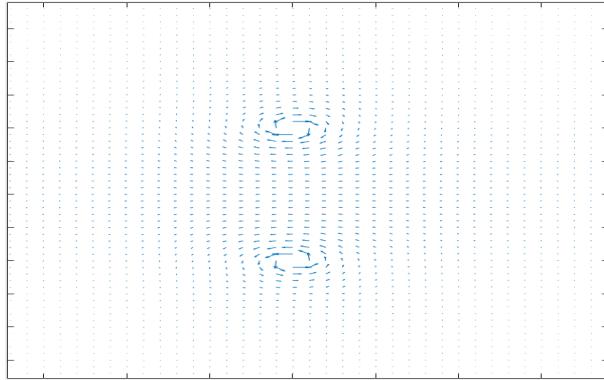


Figure 4: Example of an initially developed 2 dimensional velocity. The field is a section through a vortex ring. This field was produced with the Matlab software.

Figure 4 is of the initial stages of the velocity field generation part of the visualisation problem, when the investigation concerning the generation of the velocity field was being carried out.

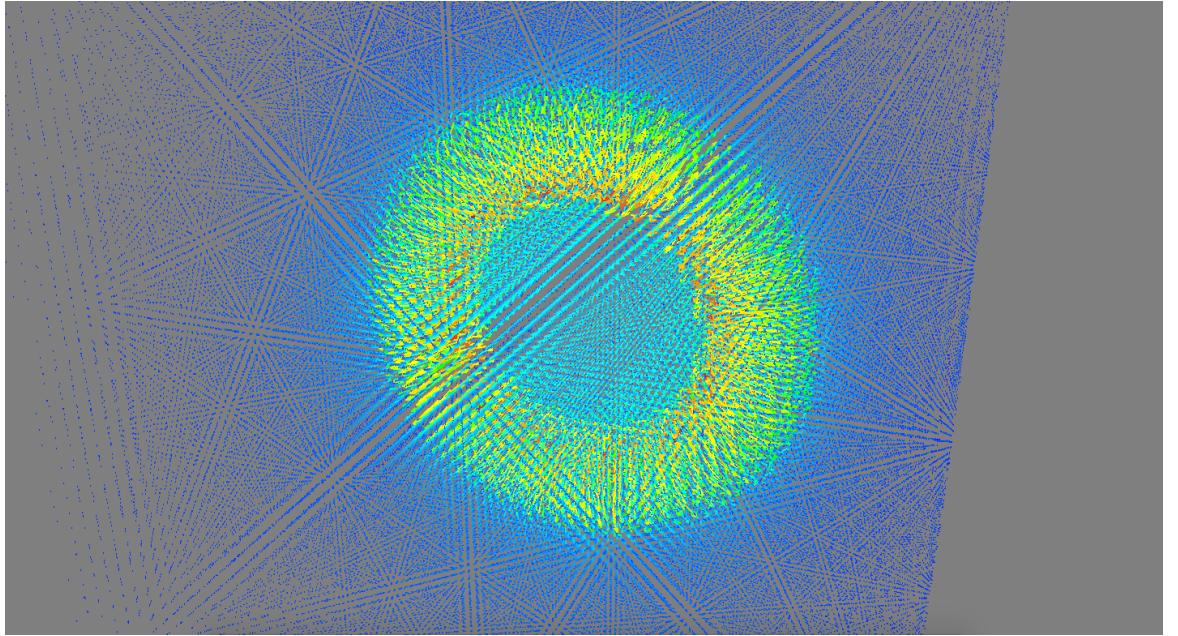


Figure 5: Velocity field induced by a single vortex ring. The magnitude of the velocity is represented by the size and colour of the vectors.

It is worth noting, in the field shown in figure 5 above, that the actual ring can be distinguished easily in the middle of the field. This is made obvious by the size and colour scheme applied to the velocity vectors. The underlying principle for this, is the fact that the vortex is a local feature within the velocity field. This concept, mentioned in earlier sections of this research paper, is the key to the extraction of the vortical information from the flow field.

Figure 6 shows the idealised model of the wake behind a fish. The field was used in the final stages of the development of the visualisation algorithm. This was carried out in order to confirm the robustness of the developed algorithms.

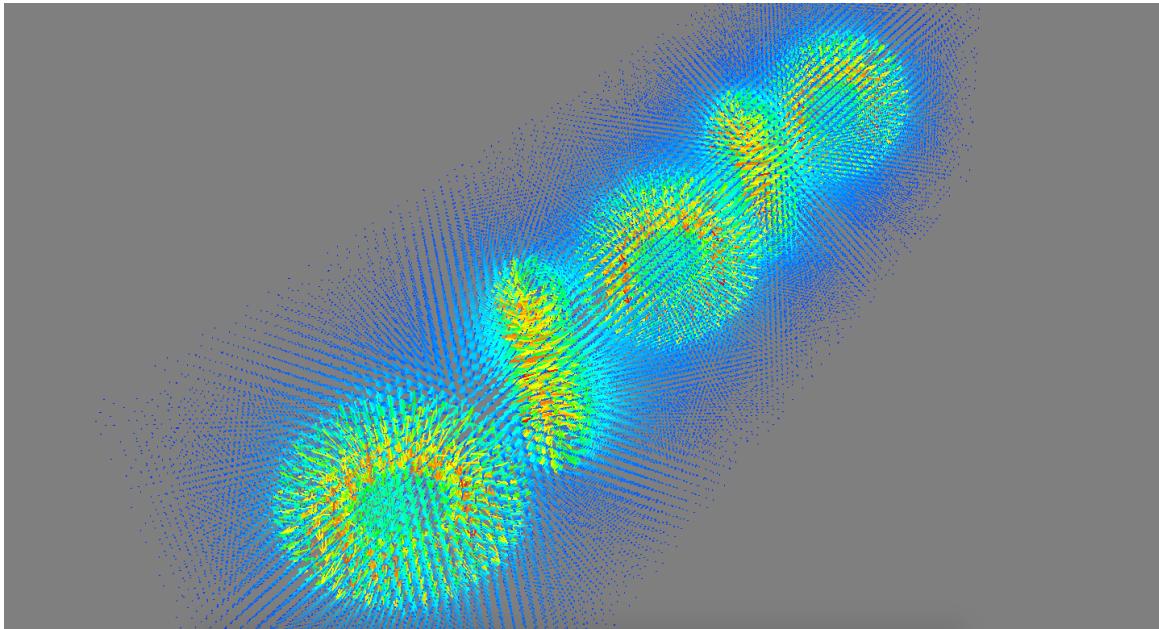
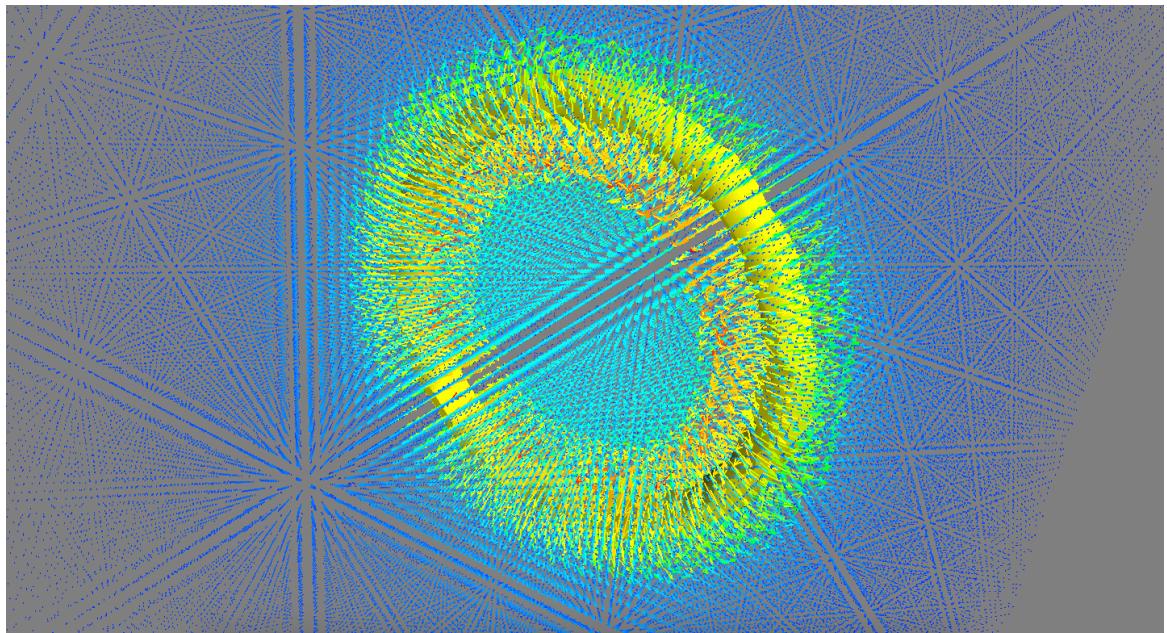
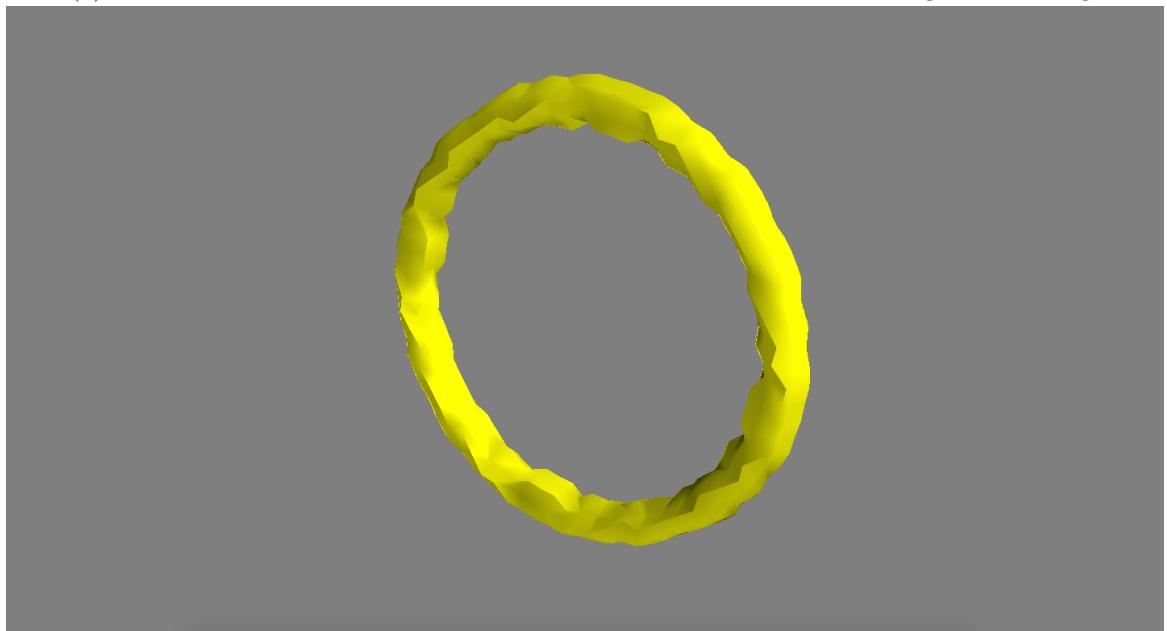
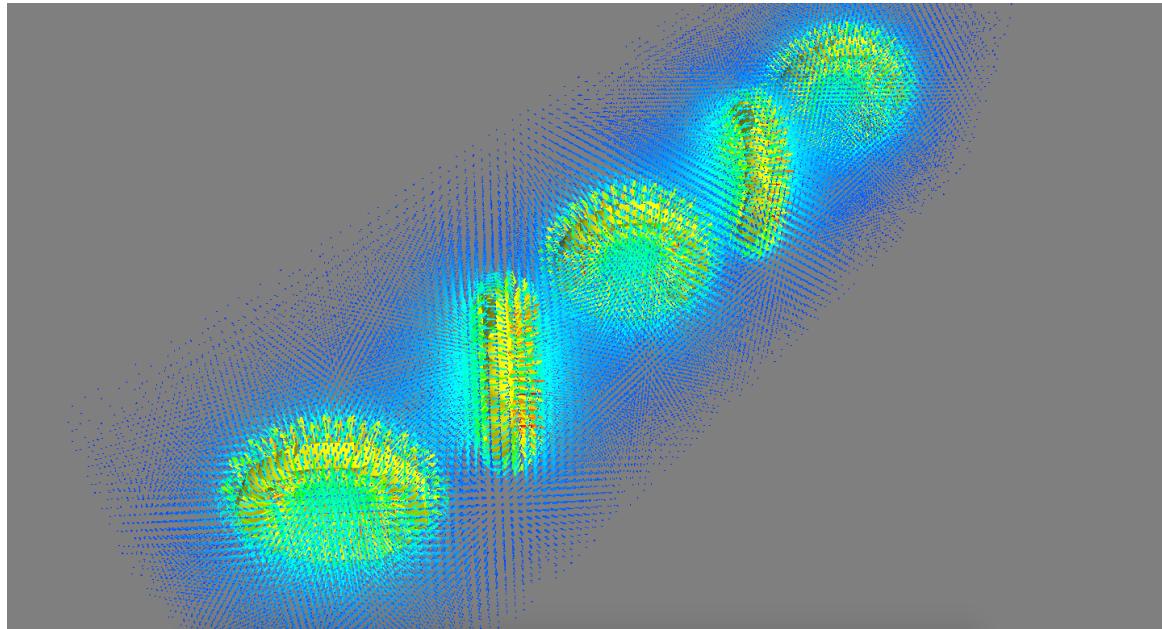


Figure 6: Figure showing the velocity induced by a chain of vortex rings with finite vortex core thickness.

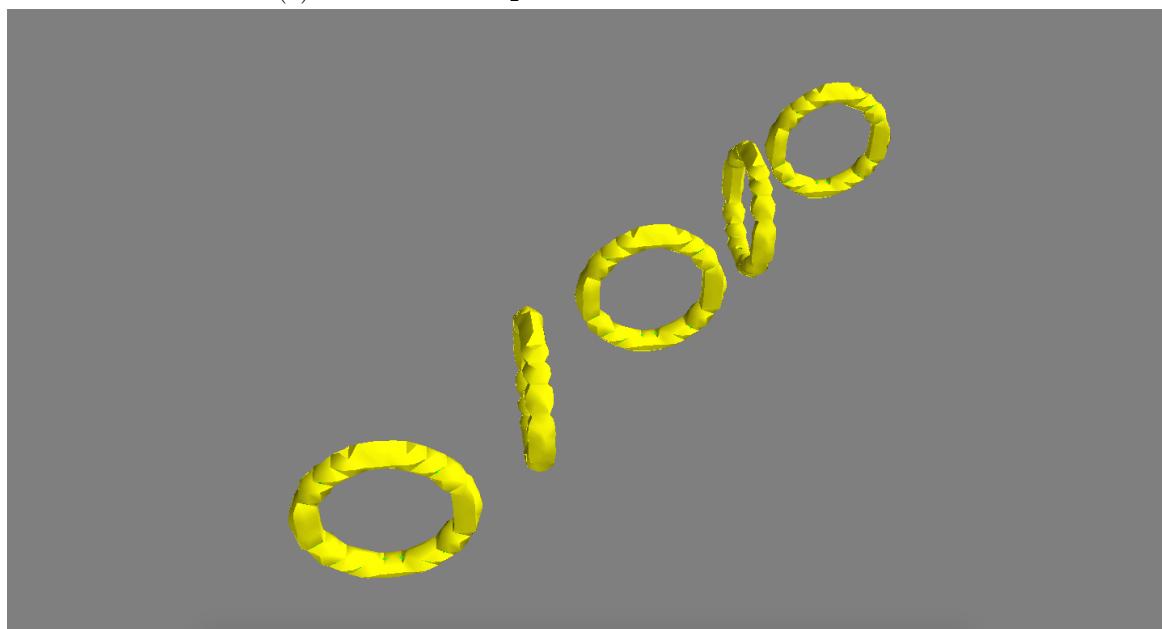
## 7.2 $\lambda_2$ implementation

The implementation of the  $\lambda_2$  definition of a vortex core was the starting point for the development of the visualisation algorithm. The  $\lambda_2$  definition does provide very good information about the location of the vortex core, but lacks information regarding the cores that have been identified. The single ring and the chain of rings flow fields were visualised. Firstly, the combined visualisation of the flow field and the vortex ring is shown. Then the flow field is removed and only the  $\lambda_2$  surface is displayed. This is shown in figure 7.

(a) Combined visualisation of the flow field and the  $\lambda_2$  surface for the single vortex ring.(b)  $\lambda_2$  surface for the vortex ring.Figure 7: Comparison between the visualisation of the flow field and the  $\lambda_2$  surface for the single vortex ring.



(a) Flow field and  $\lambda_2$  surface for the chain of vortices



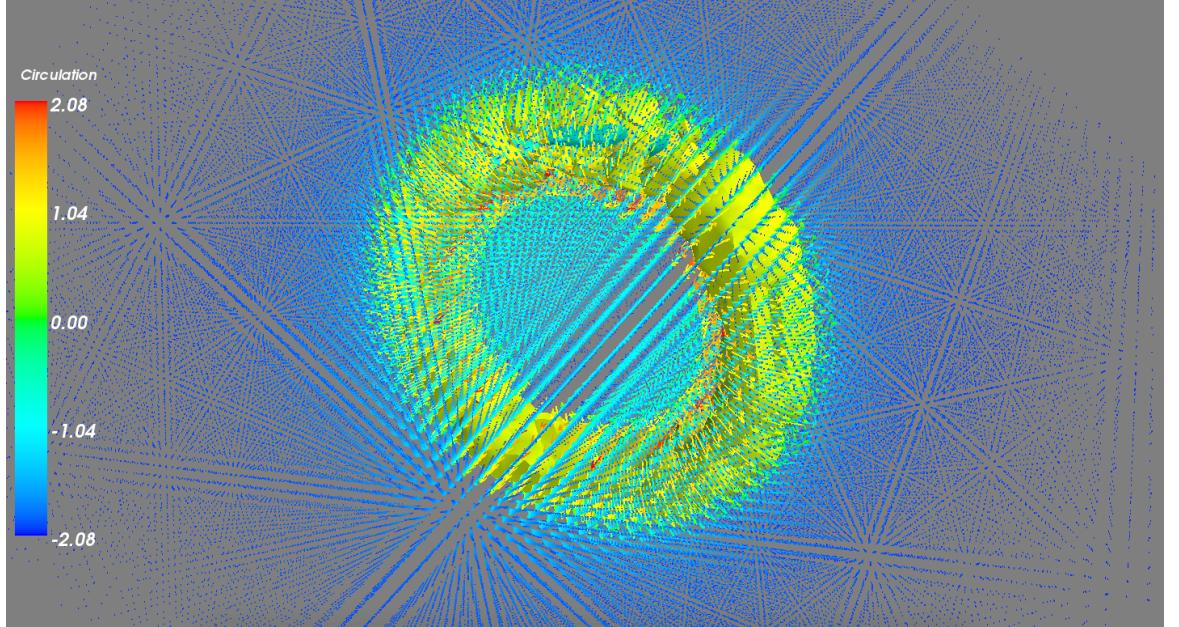
(b)  $\lambda_2$  surface for the chain of vortices.

Figure 8: Figure comparing the informativeness of the visualisation of the chain of vortices.

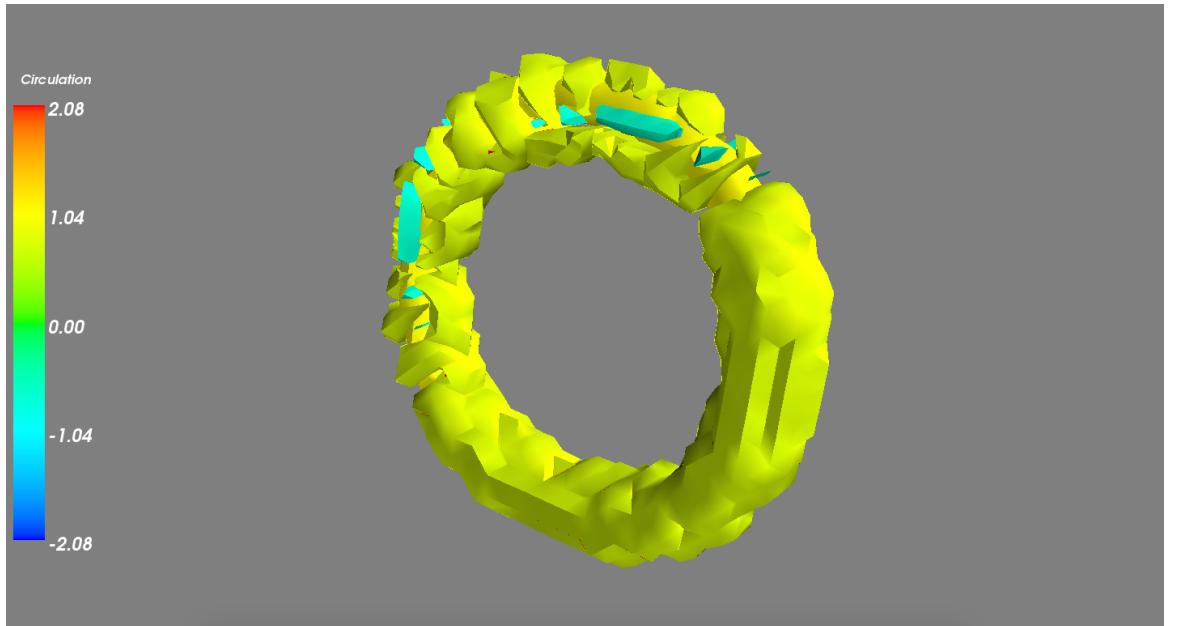
### 7.3 Circulation Intergation

The circulation integrals were the main developed and tested visualising algorithm. They combined the information regarding the location of the vortex core with the over time study of the seeded particles. The circulation field was approximated by integrating the velocity of the particle with respect to time. This added more information to the shape originally provided by the  $\lambda_2$  criterion.

The circulation integrals were implemented in the visualisation algorithm and applied to a number of flow fields. Initially, implemented to singe vortex rings for validation and debugging purposes. In a later stage in the development, the fish-like wake was analysed.

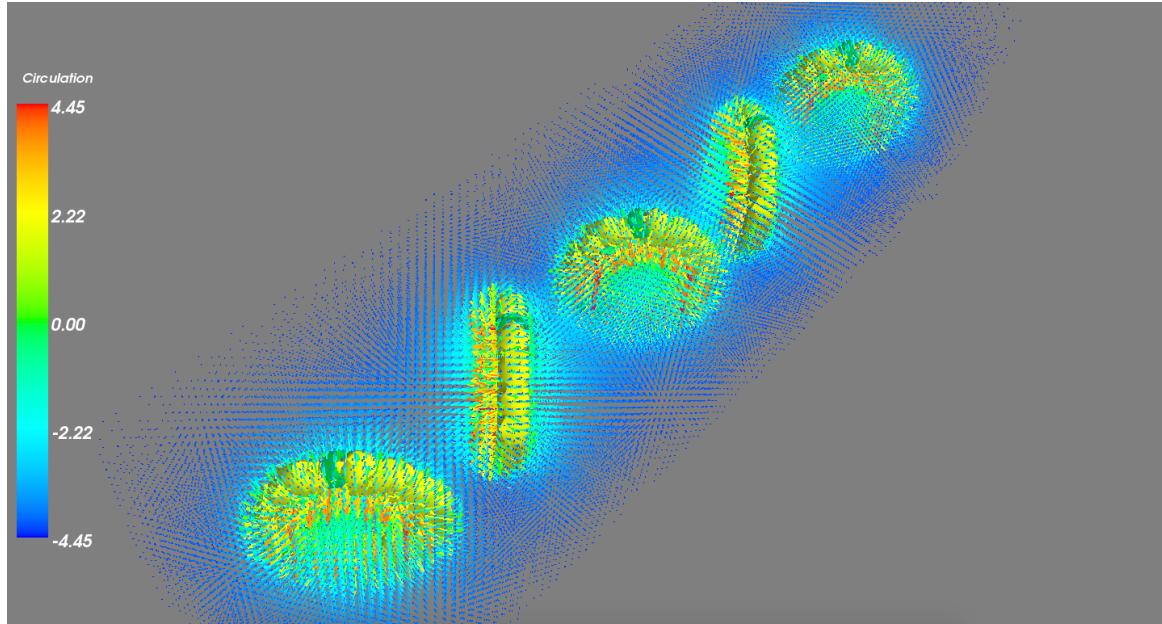


(a) Flow field together with the surface from the circulation field.

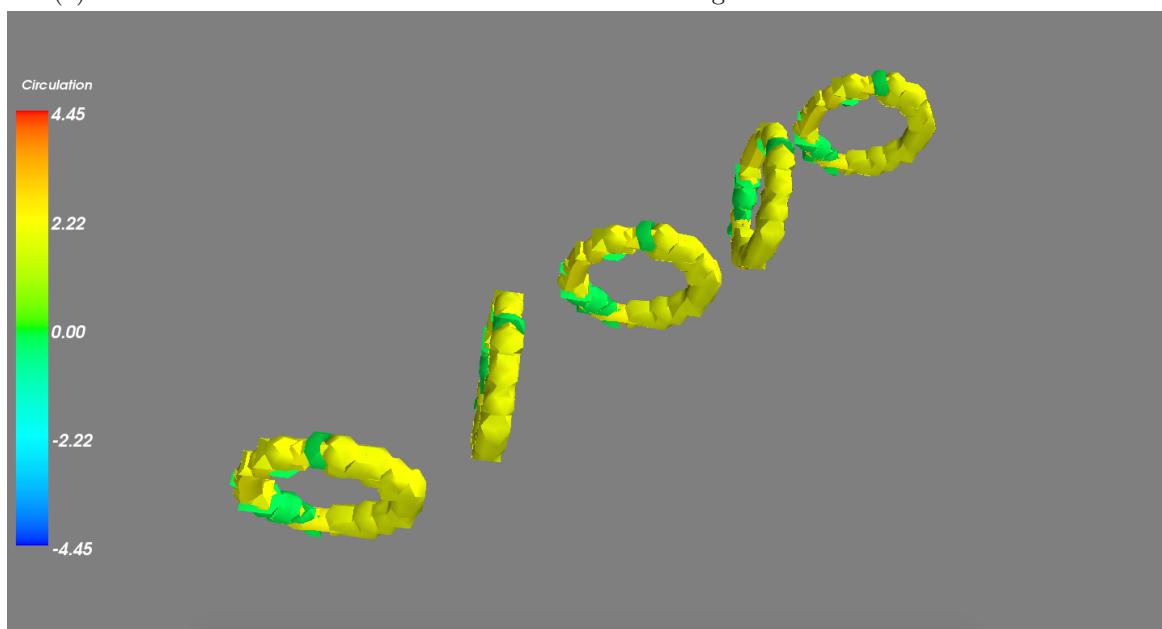


(b) Surface of the circulation field.

Figure 9: Comparison of the information displayed in the visualisation of a single vortex ring.



(a) Visualisation of the fish-like vortex wake flow field together with the circulation surface.



(b) Circulation surface for the fish-like wake flow field.

Figure 10: Comparison of the information displayed in the visualisation of the fish-like wake.

## 8 Future Work Recommendations

The algorithms and results shown in earlier sections of this report are only the fully implemented and successfully tested visualisation solutions.

A more robust and ‘field-agnostic’ algorithm was conceptually developed and, if successfully implemented, would be capable of carrying the relevant calculations for any flow field. It would not only more accurately calculate the circulation, but would also obtain the impulse and force applied by the vortical structure. This section outlines the algorithm and any other relevant future work suggestions.

### 8.1 Generalisation of the Visualisation Solution

In order to arrive at a complete ‘field-agnostic’ solution, it is essential to disregard any other information that has not been extracted from the velocity field itself. The two proposed tools to carry out the information extraction are the  $\lambda_2$  definition of a vortex core and the geometric properties obtained from the evaluation in time of the seeded particles.

The  $\lambda_2$  definition of a vortex core is used to provide the starting point from where to identify the points of interest in the flow field. Since the vortex, as shown previously, is a local feature, the only region of interest within the field is the inside of the vortex core. Once the region inside the core has been identified, geometric calculations and numerical integrations can be carried out in order to extract the vortical properties from the flow field.

The question is then presented as how to carry out the geometrical and numerical approaches in a systematic and organised way, taking as a starting point the  $\lambda_2$  field.

The proposed solution is based on two main concepts: the adequate selection of the seeding points, and the integration between those seeding points. Both concepts are carried out within the vortex cores, where any influence, both from other vortical structures or noise is negligible.

#### 8.1.1 Seeding Point Selection and Preliminary Circulation Integrals

The selection of the seeding points is the initial step after obtaining the  $\lambda_2$  field. Following steps are based on the location and number of seeds, making this part quite relevant to the algorithm.

The main input to the seeding selection process is the  $\lambda_2$  field and there are two conceptually developed options. On the one hand, a local approach could be chosen by carrying out a search for negative, local minimum points in the  $\lambda_2$  field. This option is valid and has been implemented and tested in the visualisation algorithm proposed in this research project. On the other hand, a global approach could be taken by dividing the whole field cells. The possible seeding points would be the corners of the cells. Any seed whose value of  $\lambda_2$  is not negative would be discarded, obtaining the final set of seeding locations. Alternatively, a combination both the global and local methods could be taken.

In order to finalise the preliminary seeding process, the selected points are evaluated in time. After the evaluation of the points, the circulation field can be completed for the points within the swirl plane where each seed is contained.

The purpose of this part of the algorithm is to quickly gain the location of points of interest within the flow field. Furthermore to obtain in every preliminarily seeded point,

the initial geometrical variables, such as the vector normal to the swirl plane, which are necessary in the next part of the algorithm.

### 8.1.2 Geometric Properties and Numerical Integration

Once the circulation has been preliminarily evaluated and the geometric variables have been obtained, integration between the seeding points is carried out.

The algorithm can start at any seeding point, agreeing with the concept of the algorithm being ‘field-agnostic’, and integrates following the  $\lambda_2$  negative field, until it finds a point where the circulation has been evaluated. This point can be found for two reasons: the integration has continued until it has returned to itself, in the case of a closed vortex tube, or it has found a different seeding point, which could be the case of any vortical structure where multiple seeding points have been previously found.

The obtention of the next point in the integration path is key to the successful integration between the preliminary seeded points. The direction of the integration is defined by the swirl vector, which was previously defined as the average of the cross product of the vectors tangent to the path of the seeded particle where the integration is starting from. The next point is then obtained by a numerical integration of the swirl vector field. The method of integration is the same as the one used for the seeding of a particle, with the particularity that instead of searching the velocity field for the corresponding velocity vector, the swirl vector is calculated. The swirl vector is obtained in the same way as before, averaging the cross product of the vectors tangent to the path of the particle. Algorithm 7 outlines this process.

This process is repeated until the algorithm has evaluated all the seeding points initially obtained from the  $\lambda_2$  field.

**Input:**  $\vec{X}, \vec{U}, \lambda_{2field}$ , previous seeded point.

**Output:** Next point in the integration path within the vortex core region.

```

1  $\vec{X} = X, Y, Z$ 
2  $\vec{U} = U, V, W$ 
3  $p_0$  = previous seeded point.
4  $s_0$  = swirl vector corresponding to previous seeded point.
5 dist = distance away from  $p_0$  where to search for the next point.
6  $p_{found} = 0$ 
7 while  $p_{found} == 0$  do
8    $p_1 = s_{s0} * dist$ 
9   if  $\lambda_2(p_1) < 0$  and similar  $\lambda_2(p_0)$  then
10    |  $p_{found} = 1$ 
11    |  $s_1 = \text{function}(\text{evaluates } p_1 \text{ in time until } 2\pi, \text{ and returns its swirl vector})$ 
12   else
13    | dist *= 0.8
14   end
15 end
16  $p_2 = p_0 + (dist * 0.5 * s_0) + (dist * 0.5 * s_1)$ 
17  $s_2 = \text{function}(\text{evaluates } p_2 \text{ in time until } 2\pi, \text{ and returns its swirl vector})$ 
18  $p_3 = p_0 + (dist * 0.33 * s_0) + (dist * 0.33 * s_1) + (dist * 0.33 * s_2)$ 
19 return  $p_3$ 
```

**Algorithm 7:** Proposed algorithm to estimate the next integration point.

### 8.1.3 Impulse of the Vortex Rings

In the case of the vortex rings, an accurate circulation field can lead to the calculation of the impulse of each ring. According to Saffman [6], the impulse of an isolated vortex is defined by the following equation:

$$\vec{I}_V = \frac{1}{2} \int \vec{x} \times \vec{\omega} dV_V \quad (44)$$

Where  $\vec{x}$  is the location of the centroid of the ring,  $V_V$  the volume enclosed by it, and  $\vec{\omega}$  is the vorticity, which could be easily obtained from its relationship with the circulation, as can be seen from equation 45.

$$\Gamma = \oint_C \vec{u} \cdot d\vec{s} = \oint_A \vec{\omega} \cdot \vec{n} dS \quad (45)$$

The centroid of the ring, where the impulse from the ring acts from, can be obtained by calculating the centre of the area enclosed by integration paths within the same  $\lambda_2$  negative surface. This constitutes the circular ring shown in earlier sections, but should be approached in such a way that the impulse for asymmetric vortex rings will also be successfully obtained.

The direction on which the impulse will act could be calculated from the average of the cross product of the tangent vectors of the integrated  $\lambda_2$  surface.

Adding the impulse of the vortex ring to the visualisation algorithm would provide information regarding the interaction of the vortex ring with the fluid, which will be useful in applications such as the analysis of vortex ring wakes.

#### 8.1.4 Force Applied on the Body

Returning to the field of ‘field-agnostic’ information extraction, a key piece of information when analysing vortical flow fields is the force the vortical structures exert. This force is applied on the body that has generated such field as a result of its interaction with the flow. According to Saffman [6], the force applied by the body to the fluid is defined by equation 46.

$$\vec{D} = \int \vec{u} \times \vec{\omega} \, dV \quad (46)$$

Where the force  $\vec{D}$  is dependant on the velocity field  $\vec{u}$ , the vorticity  $\vec{\omega}$ , and the volume over which it is integrated  $V$ .

This could be easily implemented numerically, since the direction in which the force is applied is the same as the impulse but in the opposite sense, and the point in which it is applied is the centroid of the shape.

## 8.2 General Algorithm

An outline of the algorithm described in earlier parts of this section is given below:

```

Input: Velocity field image file, .vtu or similar.
Output: Visualisation of force applied by the body to the fluid, and in the case of
vortex rings also the impulse of the rings.

1  $\vec{X} = X, Y, Z$ 
2  $\vec{U} = U, V, W$ 
3 L2 field = function(calculates L2 field)
4 Seed points = function(seeding points based on L2 field)
5 Circulation field = [ ]
6 for  $s$  in seeds do
7   | function(evaluates seeds in time until  $2\pi$ )
8 end
9 for  $p$  in  $Circ_{field}$  do
10  | function(assigns circulation within  $Circ_{field}$  in seeded swirl planes)
11 end
12 for  $s$  in seeds do
13   | function(numbers the  $\lambda_2$  surface section where the seed is present and the
integration is carried allong)
14   | while next point no circulation do
15     |   next point =function(calculates next point)
16     |   if next point has circulation then
17       |     | go to next seed
18     |   else
19       |     | function(evaluates seed in time until  $2\pi$  criterion is met)
20       |     | function(calculates circulation)
21       |     | function(clculates swirl vector)
22       |     | function(adds circulation to  $Circ_{field}$ )
23   | end
24 end
25 end
26 function(Takes the numbered  $\lambda_2$  surfaces joins neighbouring sections)
27 function(Takes the closed  $\lambda_2$  surfaces and integrates along them to obtain the
impulses and forces)
28 function(Visually displays all the previously calculated.)
29 return Force and Impulse fields

```

**Algorithm 8:** Proposed algorithm to accurately extract the information form the vortical flow field.

This may appear very computationally expensive, but in order to accurately extract information from any vortical flow field it is expected to amount to at least 4 times the time spent in numerically generating the field.

### 8.3 Language improvements

In order to reduce computational time, the option of changing implementation language should be investigated. The Python programming language has the advantage of the initial development speed however computational speed is lost. The figure below shows a comparison of a variety of commonly used programming languages.

	<b>Fortran</b>	<b>Julia</b>	<b>Python</b>	<b>R</b>	<b>Matlab</b>	<b>Octave</b>	<b>Mathematica</b>	<b>JavaScript</b>	<b>Go</b>	<b>LuaJIT</b>	<b>Java</b>
	gcc 4.8.2	0.3.7	2.7.9	3.1.3	R2014a	3.8.1	10.0	V8 3.14.5.9	go1.2.1	gsl-shell 2.3.1	1.7.0_75
<code>fib</code>	0.57	2.14	95.45	528.85	4258.12	9211.59	166.64	3.68	2.20	2.02	0.96
<code>parse_int</code>	4.67	1.57	20.48	54.30	1525.88	7568.38	17.70	2.29	3.78	6.09	5.43
<code>quicksort</code>	1.10	1.21	46.70	248.28	55.87	1532.54	48.47	2.91	1.09	2.00	1.65
<code>mandel</code>	0.87	0.87	18.83	58.97	60.09	393.91	6.12	1.86	1.17	0.71	0.68
<code>pi_sum</code>	0.83	1.00	21.07	14.45	1.28	260.28	1.27	2.15	1.23	1.00	1.00
<code>rand_mat_stat</code>	0.99	1.74	22.29	16.88	9.82	30.44	6.20	2.81	8.23	3.71	4.01
<code>rand_mat_mul</code>	4.05	1.09	1.08	1.63	1.12	1.06	1.13	14.58	8.45	1.23	2.35

**Figure:** benchmark times relative to C (smaller is better, C performance = 1.0).

Figure 11: Table showing a speed comparison of various programming languages when carrying out standardised numerical computations. Figure taken from JuliaLang [4].

There are a number of possibilities in the language selection. The simplest of the solutions would consist of continuing with the use of Python and combine it with the use of the C language and the Numba package [14] for compiler optimisation. The optimum solution was found to be Julia which offers the optimum balance between the development and computational speed. Julia is a newly developed language, which combines easy to write syntax and computational speed. This is achieved through an optimal compiler design. In order to further increase the computational speed, the C language could be used in obvious slow parts of the computation such as for-loops.

The combination of Julia and the C language has achieved speeds similar to the C language, with the great advantage of the quick development of the basic algorithms due to its easy syntax.

### 8.4 Unstructured Grid Reconstruction

As a brief section, and in order to ensure the complete ‘field-agnosticity’ of the method, the supposition of a flow field with not enough data points is covered. This case may be a result of lack of data, from an experiment, or a measurement during operation. In order to accurately carry out the information extraction from the flow field, and the calculation of the vortical properties a determined density of data points is required.

In order to solve this potential deficit in the number of data points, an inverse process, similar to the approach used in the third order approximation of the position of the particle would be used. The interpolation or approximation of the velocity field would be done if the number of vortical structures found within the field was large compared with the density of data points. This would need to be carried out in a recurrent way once the  $\lambda_2$  definition of a vortex has been initially applied and an initial idea of the number of features in the field has been obtained.

The mathematical tools were not investigated, but conceptually the mathematical and numerical approach has been set to solve the issue of the data point deficit.

## 9 Conclusion

This project has successfully investigated the relationships governing the generation of, and information extraction from, vortical velocity fields. A range of velocity fields were successfully generated, both for algorithm development, and application of the developed algorithm to more realistic cases. The developed and tested algorithm incorporated, as a initial step, the  $\lambda_2$  definition of a vortex core. Said algorithm, combined the application of seeding techniques and geometric properties of the flow to approximately extract the circulation field from the flow. Improvements in the robustness of the circulation integrals could be made, but an estimation of the circulation field was obtained in a simple and elegant way, both numerically and computationally. The further work section provided an outline of a very robust and ‘field-agnostic’ solution. This proposed, but not implemented solution, could be the starting point to further research on information extraction and visualisation from vortical flow fields.

## References

- [1] J. Jeong and F. Hussain, “On the identification of a vortex,” *Journal of Fluid Mechanics*, vol. 285, pp. 69–94, 2 1995.
- [2] G. D. Weymouth, “Lily-pad: Test platform for real-time two-dimensional fluid/structure interaction simulations written in processing. <https://github.com/weymouth/lily-pad>.”
- [3] O. Tietjens, L. Prandtl, and L. Rosenhead, *Fundamentals of hydro- and aeromechanics: based on lectures of L. Prandtl*. Engineering societies monographs, Dover Publ, 1934.
- [4] “Julialang: Julia is a high-level, high-performance dynamic programming language for technical computing, with syntax that is familiar to users of other technical computing environments. [www.julialang.org](http://www.julialang.org).”
- [5] J. N. Newman, *Marine hydrodynamics / J. N. Newman*. MIT Press Cambridge, Mass, 1977.
- [6] P. Saffman, *Vortex Dynamics*. Cambridge Monographs on Mechanics, Cambridge University Press, 1992.
- [7] M. Jiang, R. Machiraju, and D. Thompson, “Detection and visualization of vortices,” in *The Visualization Handbook*, pp. 295–309, Academic Press, 2005.
- [8] M. Jiang, R. Machiraju, and D. Thompson, “Geometric verification of swirling features in flow fields,” in *Proceedings of the Conference on Visualization '02*, VIS '02, (Washington, DC, USA), pp. 307–314, IEEE Computer Society, 2002.
- [9] A. Hin and F. H. Post, “Visualization of turbulent flow with particles,” in *Proceedings Visualization '93* (G. Nielson and R. Bergeron, eds.), IEEE Computer Society Press, Los Alamitos, CA, 1993. 46–52.
- [10] E. R. Tufte, *The Visual Display of Quantitative Information*. 2001.
- [11] H. Tennekes and J. L. Lumley, *A first course in turbulence*. Cambridge (Mass.), London: M.I.T. Press, 1972.
- [12] “Numpy: Numpy is the fundamental package for scientific computing with python. <http://www.numpy.org>.”
- [13] “The python satandard library: 9.2 math - mathematical functions. <https://docs.python.org/2/library/math.html>.”
- [14] “Numba: Numba generates optimized machine code from pure python code using the llvm compiler infrastructure. <http://numba.pydata.org>.”
- [15] D. Thompson, R. Machiraju, M. Jiang, J. Nair, G. Craclun, and S. Venkata, “Physics-based feature mining for large data exploration,” *Computing in Science Engineering*, vol. 4, pp. 22–30, Jul 2002.
- [16] D. Sujudi and R. Haimes, “Identification of swirling flow in 3-d vector fields,” 1995.

- [17] A. Nicolle and I. Eames, “Numerical study of flow through and around a circular array of cylinders,” *Journal of Fluid Mechanics*, vol. 679, pp. 1–31, 7 2011.
- [18] S. Robinson, *The kinetics of boundary layer structure*. PhD thesis, Stanford University, 1991.