

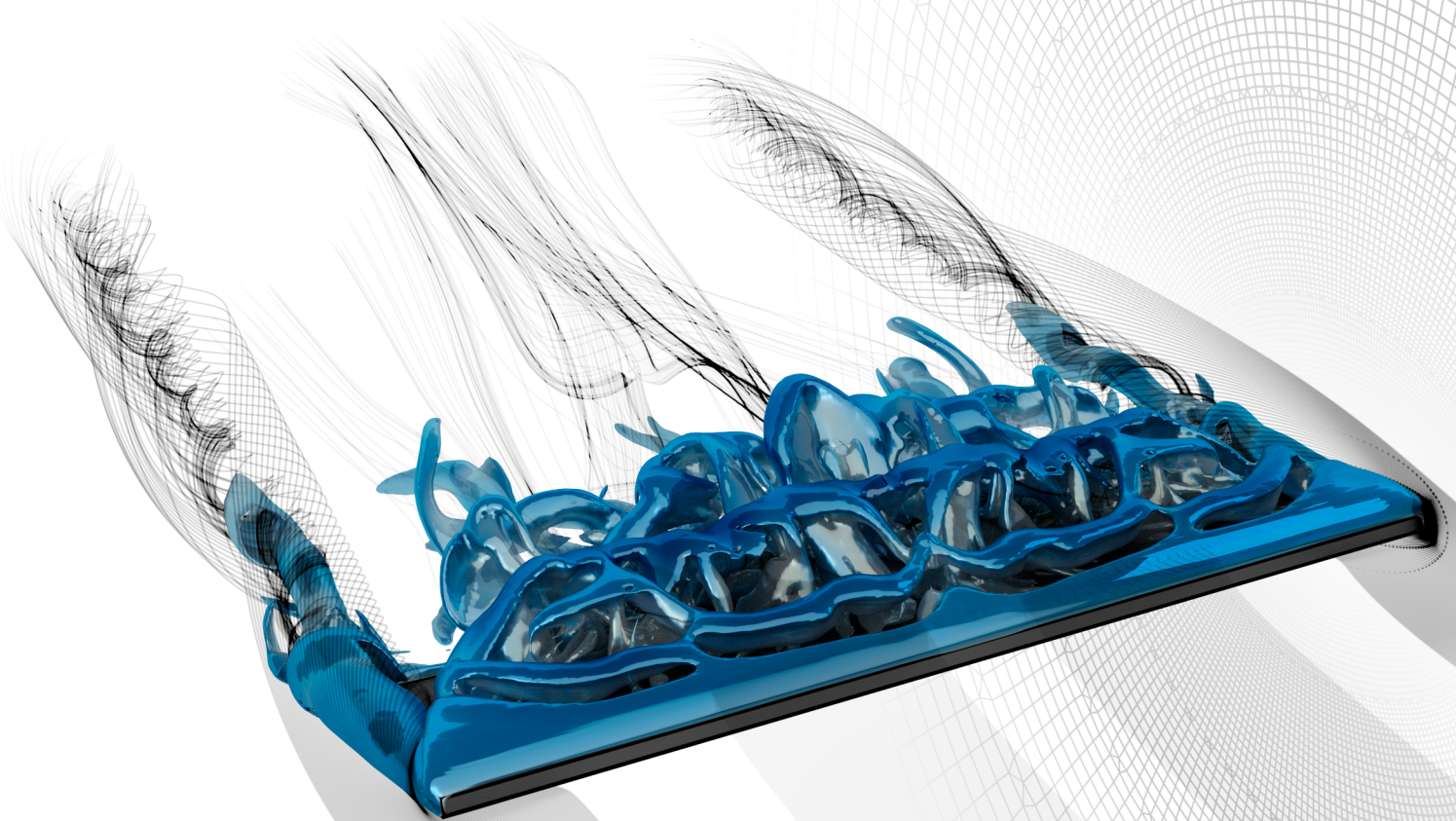
Modelling of Plates in Free Fall

Development of a Predictive Trajectory Model
by an Experimental and Numerical Approach

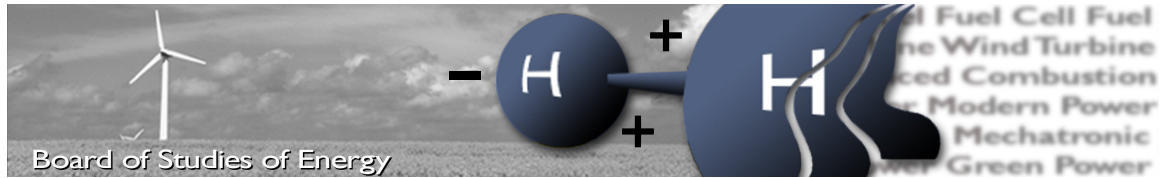
Anders Schou Simonsen

Master's Thesis Project, June 2015

Thermal Energy and Process Engineering



The front-page image is showing a plate in free fall with an isosurface of the Q-criterion, which is used for vortex detection, at 250 s^{-2} slightly coloured by the velocity magnitude. A mesh slice has been shown at the far end along with four sets of streamlines seeded upstream of the plate.



Title: Modelling of Plates in Free Fall
Semester: 4th M.Sc.
Project period: 02/02-2015 - 03/06-2015
ECTS: 30
Supervisor: Henrik Sørensen & Anna Lyhne Jensen
Group: TEPE4-1005

Anders Schou Simonsen

Copies: 4
Pages: 104 consisting of:
 10 pages front matter
 77 pages report
 17 figure and blank pages
Appendices: A - B
Supplements: Executive summary
 Enclosure A - D
 USB-stick

Synopsis:

Modelling of non-spherical particles is important in many applications. Currently, the motion of non-spherical particles is modelled using spherical assumptions, as the motion of such particles is well documented. However, using this approach, neither the orientation nor the derivatives with respect to time are taken into account, which is found to be crucial for accurate modelling.

This study extends the current framework by investigating plates in free fall using an experimental and numerical approach in order to develop an accurate predictive trajectory model. A CFD-model is developed, which is capable of predicting the trajectory of a given plate in a fast and accurate way. The model was verified by comparison with experimental trajectories. Using the CFD-model, numerous simulations were carried out in a parametric study, where various parameters were varied.

A predictive trajectory model was developed, which was primarily based on inviscid and irrotational flow theory. A panel method for predicting added mass and circulation coefficients was developed and coupled with the model. This allowed for accurate modelling of the plate trajectories for a wide range of geometries and densities.

The model was compared with an existing regime map from literature, where the motion was classified as a function of Re , which showed good resemblance. A new dimensionless number was proposed for general motion classification for freely falling objects, which yielded more clearly distinguishable regions.

By signing this document, the author confirms that the report does not include plagiarism.

PREFACE

This master's thesis project was carried out at the *Department of Energy Technology* at *Aalborg University*. The project title is *Modelling of Plates in Free Fall* and was conducted at the 4th semester of the M.Sc. in *Thermal Energy and Process Engineering* from February 2nd to June 3rd.

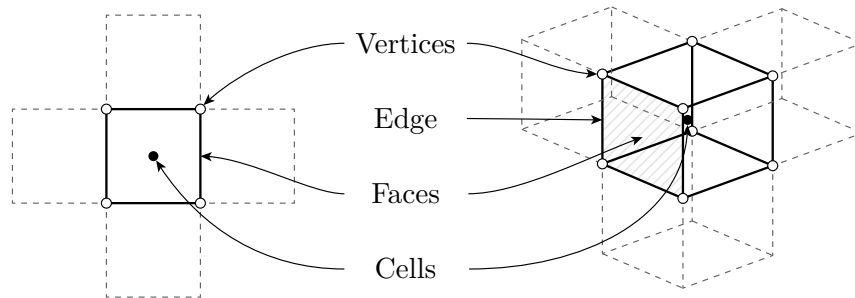
Readers Guide

An executive summary is attached in the front of the report. All figures, equations and tables are numbered according to their chronological placement in the chapters. Figure 3.2 is the second figure in chapter 3. References are listed based on the *Harvard method*, where sources are mentioned by the main author names and the year of publication. Additional information about the references can be found in the bibliography. Most electronic sources, such as articles and web-pages, can be found on the attached USB-stick. The content of this can be seen in *Appendix B - USB Content*.

Abbreviations will be presented the first time the term is written. All symbols and abbreviations, along with a short description of each, can be found in the *Nomenclature*. Matrices are denoted with boldface such as \mathbf{M} , whereas vectors are denoted with an arrow accent as \vec{u} . All units are written as non-italic, whereas variables are in italic. For instance, m is mass and m is meters. Some equations are including the term (1 m) , which is used to balance the units.

Position is denoted \vec{x} , velocity with \vec{v} and acceleration with \vec{a} . Orientation is denoted θ , angular velocity with $\dot{\theta}$ and angular acceleration with $\ddot{\theta}$. Dot products are denoted with \cdot .

Both 2D and 3D meshes will be employed in this study, where the mesh convention has been shown in the figure below for a 2D and a 3D mesh to the left and right respectively.



Acknowledgment

I would like to thank the supervisors Associate Professor Henrik Sørensen and Ph.D Student Anna Lyhne Jensen for providing help, supervision and feedback throughout the project.

NOMENCLATURE

Symbols

Symbol	Unit	Representation
A	m^2	Area
a	m/s^2	Acceleration
B	m	Plate depth
D	Various	Diameter or diffusivity constant
f	s^{-1}	Frequency
F	N	Force
g	m/s^2	Gravity
H	m	Plate height
i, N	—	Integer
I	$\text{m}^2 \cdot \text{kg}$	Moment of inertia
K	J	Kinetic energy
L	m	Plate length
\mathbf{L}	$\text{m}^2 \cdot \text{kg}$	Moment of inertia tensor
m	kg	Mass
\mathbf{M}	kg	Mass tensor
p	Pa or $\text{kg} \cdot \text{m/s}$	Pressure or translational momentum
r	m	Radius
n	—	Normal vector or integer
s	—	Standard deviation
S	Various	Scaling parameter, surface or strain rate
t	s	Time
\mathbf{T}	—	Transformation matrix
u	m/s	Velocity
\mathbb{V}	m^3	Volume
x, y	m	Position

Subscripts

Symbol	Representation	Symbol	Representation
$()_{\text{AM}}$	Added mass	$()_{\text{p}}$	Plate or pressure
$()_{\text{B}}$	Buoyancy	$()_{\text{R}}$	Rotational
$()_{\text{D}}$	Drag	$()_{\text{S}}$	Source panel
$()_{\text{f}}$	Fluid	$()_{\text{T}}$	Translational
$()_{\text{g}}$	Gravity	$()_{\text{V}}$	Vortex panel
$()_{\text{m}}$	Moving	$()_{\nu}$	Viscous

Greek symbols

Symbol	Unit	Representation
α	rad	Angle of attack
γ	m^2/s	Vortex strength
Γ	m^2/s	Circulation
ℓ	$\text{kg} \cdot \text{m}^2 \cdot \text{rad}/\text{s}$	Angular momentum
μ	$\text{Pa} \cdot \text{s}$	Dynamic viscosity
ρ	kg/m^3	Density
ϕ	Various	Velocity potential, scalar and mesh angle
φ	—	Relaxation parameter
σ	m^2/s	Source strength
τ	$\text{N} \cdot \text{m}$	Torque
θ	rad	Orientation
$\dot{\theta}$	rad/s	Angular velocity
$\ddot{\theta}$	rad/s ²	Angular acceleration

Commonly used dimensionless groups

Symbol	Definition	Representation
β	L/D	Aspect ratio
ρ^*	$\rho_{\text{p}}/\rho_{\text{f}}$	Density ratio
I^*	$32 I / \pi \cdot \rho_{\text{f}} \cdot L^4 \cdot (1 \text{ m})$	Dimensionless moment of inertia for plates
Re	$u \cdot L \cdot \rho_{\text{f}} / \mu$	Reynolds number
Si	$u \cdot f \cdot H \cdot \rho_{\text{f}} / g \cdot L \cdot \rho_{\text{p}}$	New group formed in this study

Abbreviations

Abbreviation	Representation
2D and 3D	Two and three dimensional respectively
6DOF	Six Degree of Freedom
CFD	Computational Fluid Dynamics
CFL-condition	Courant–Friedrichs–Lewy condition
DNS	Direct Numerical Simulations
NS-equations	Navier Stokes equations
PT-model	Predictive Trajectory model
SST-model	Shear Stress Transport model
UDF	User Defined Function

CONTENTS

Titlepage	I
Preface	V
Nomenclature	VII

Chapters:

1 Introduction	1
1.1 Existing framework	1
1.2 Trajectory characterization	2
2 Problem statement	5
2.1 Problem solving strategy	5
2.2 Report structure	7
3 Governing forces	9
3.1 Direct forces	9
3.2 Constructed forces	10
3.3 Motion	13
3.4 Partial conclusion	14
4 CFD-model	15
4.1 Domain	15
4.2 Governing equations	18
4.3 Turbulence modelling	19
4.4 Boundary conditions	20
4.5 Dynamic integration procedure	20
4.6 Meshing	24
4.7 Independence analyses	28
4.8 Partial conclusion	31
5 Experimental analysis	33
5.1 Setup and procedure	33
5.2 Data processing	34
5.3 Results	36
5.4 Discussion and sources of errors	42
5.5 Partial conclusion	43
6 Verification	45
6.1 CFD and experimental comparison	45
6.2 Partial conclusion	52

7	Parametric analysis	53
7.1	Overview	53
7.2	Partial conclusion	54
8	Predictive Trajectory model	55
8.1	Overview	55
8.2	Orientation vectors	56
8.3	Model development	57
8.4	Comparison with CFD-simulations	62
8.5	Modified PT-model	68
8.6	Dynamic investigation	70
8.7	Partial conclusion	73
9	Closure	75
9.1	Resume	75
9.2	Conclusion	77
10	Future work	79
	Bibliography	81
	<i>Appendices:</i>	
A	Panel Methods	83
B	USB Content	91

1 | INTRODUCTION

The motion of particles is important in many applications. Such applications include dust collectors, cyclone separators, fluidised beds and pulverised-coal combustors [Zastawny et al., 2011]. Other practical applications include trajectory prediction of wind borne debris like roof tiles [Kakimpa et al., 2012]. It is often assumed that the particles of interest are close to spherical despite the actual shape, which leads to inaccurate trajectories. The reason for doing so is that the motion of spheres has been investigated in numerous studies, which has led to accurate modelling of such. However, due to the symmetric nature of a sphere, neither the orientation nor the derivatives of this with respect to time is taken into account. As of such, inaccurate trajectories might be obtained when using the spherical assumption. This can be dealt with by expanding the existing framework to include non-spherical particles. This study deals with such, where plates of various sizes and densities have been investigated by an experimental and numerical approach in order to develop a *Predictive Trajectory model*, PT-model.

The PT-model can be used to predict more accurate trajectories for flat plates, but the principles and methods used will be developed in a general manner. As of such, the PT-model should be able to predict trajectories for other shapes as well, which might inspire new studies using some of the same methods. The motivation for investigating the motion of non-spherical particles is thus partly to expand the existing framework in the direction of predicting the motion for any given object.

1.1 Existing framework

Particles can be classified in geometric groups, where Figure 1.1 is showing examples of some basic shapes in both 2D and 3D with associated names.

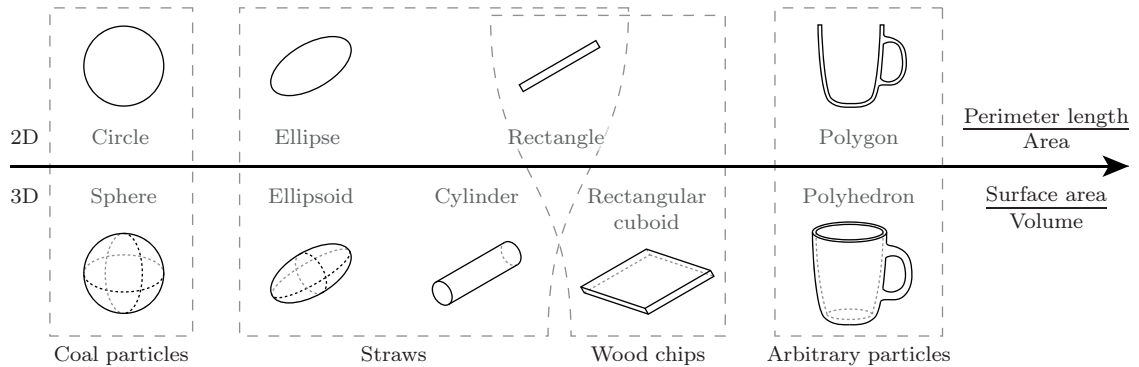


Figure 1.1: Different particles in 2D and 3D as a function of geometric ratios. The geometric groups can be related to various common shapes used in industrial applications, which have been shown in the bottom.

The different shapes in Figure 1.1 have been shown with respect to the perimeter to area ratio and surface area to volume ratio in 2D and 3D respectively. These parameters are often used to approximate the resemblance to spherical particles, where other ratios, such as the sphericity, are used as well [Widhalm, 2015]. The various geometries can be related to particles, which are commonly used in industry. These have been shown in the bottom of the figure, where the dashed lines are indicating the particle shapes related to the classification. For instance, straws are commonly assumed having an elliptical or rectangular shape in 2D, whereas an ellipsoidal or cylindrical shape is usually the assumed shape in 3D [Mandø et al., 2007].

The existing framework for modelling particles includes various shapes, where circles and spheres have been investigated in numerous studies, where the motion of such can be predicted with high accuracy. The forces on these objects do not depend on the orientation due to symmetry. Ellipses and ellipsoids have been investigated by reference [Rosendahl, 1998] by use of the superelliptic equation, where shapes from spheres to ellipses can be obtained by varying a superelliptic exponent. Other studies have been performed as well, where reference [Andersen et al., 2005b] investigated the motion of elliptical particles.

When the surface curvature changes from being continuous to discontinuous, objects such as rectangles, cylinders and rectangular cuboids are the most basic of such. The surfaces of these are not continuous due to the sharp edges, which can affect the aerodynamics. Motion prediction of such particles becomes increasingly more complex, as vortices are likely to be shed from the sharp edges, which will cause unsteady aerodynamic effects, though this can happen for smooth particles as well. The motion of rectangular plates in 2D has been investigated by reference [Jensen and Hærvig, 2014], where a PT-model was developed to extend the framework for specific plate dimensions. This model included new correlations for the centre of pressure location along with rotational lift and drag.

When the particle shapes become more irregular and tends towards the general classification for arbitrary shapes defined as polygons and polyhedrons in the 2D and 3D respectively, the motion becomes difficult to predict. Usually, experimental studies or resolved *Computational Fluid Dynamics*-, CFD-simulations, are required in order to obtain accurate trajectories. However, this approach is both time consuming and computational expensive to execute, so general approximations for such shapes is of high interest.

This study will extend the current framework for non-spherical particles by investigating rectangular plates with varying dimensions and densities. This will lead to a more general modelling framework, which can motivate further studies to include arbitrary shapes.

1.2 Trajectory characterization

Multiple models exist for predicting trajectories of falling objects, which have been developed by empirical measurements, CFD-models or a combination of both [Maxwell, 1854; Jensen and Hærvig, 2014; Andersen et al., 2005b; Jin and Xu, 2007]. One of the pioneers within falling objects was Maxwell in 1854, where the nature behind falling plates was investigated before classical aerodynamics was developed [Maxwell, 1854]. With time, computational power has allowed for in-depth analyses of the flow field by use of CFD-models, which has been utilized in several studies.

Reference [Andersen et al., 2005b] investigated trajectories of four different plates experimentally, where both periodic, chaotic and tumbling motions were observed. The trajectories were replicated by use of 2D CFD-simulations, which allowed for analyses of the forces acting on the plates over time. A quasi-steady PT-model was developed, which could predict the dynamics of the plates. This model was investigated by varying various parameters, where bifurcation between periodic and tumbling motion was found. The phenomena was investigated in another paper by the same authors [Andersen et al., 2005a], where a phase diagram of the motion was characterized by two dimensionless parameters. The bifurcations were in agreement with another phase diagram, which was constructed from experimental observations [Smith, 1970]. The dimensionless groups used for characterization have been shown in Equation 1.1.

$$Re = \frac{u L \rho_f}{\mu} \quad I^* = \frac{32 I}{\pi \rho_f L^4 (1 \text{ m})} = \frac{I}{\rho_f D^5} \quad (1.1)$$

Plates
Disks

Re is the *Reynolds number* and I^* is the dimensionless moment of inertia, which has been shown for both plates and disks as defined in the references [Smith, 1970; Field et al., 1997]. The moment of inertia for plates has been defined per unit depth in other studies, so the denominator is including (1 m) in order for the units to balance in this study. Reference [Field et al., 1997] characterized the motion of falling disks, where similar dimensionless groups were used as reference [Smith, 1970]. Both of the phase diagrams made by these references can be seen in Figure 1.2.

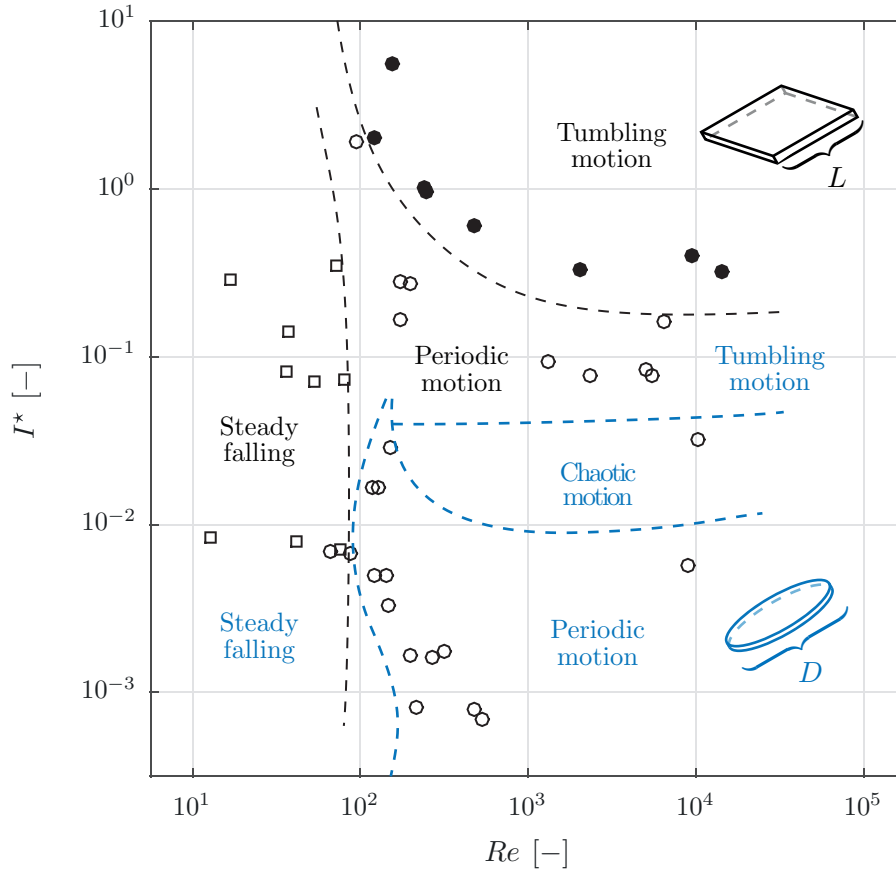


Figure 1.2: A regime map showing the motion of falling plates [Smith, 1970] and disks [Field et al., 1997] as a function of Re and I^* shown in black and blue respectively. The experimental observations for the falling plates have been shown with the markers, whereas the observations for the falling disks have been omitted for appearance.

As seen in Figure 1.2, the motion of freely falling objects can vary between steady falling, periodic motion and tumbling motion. Bifurcation between the different motion classifications can be seen as the dashed lines. The bifurcation tendencies for plates and disks are similar, though the lines for the disks are about a decade lower in terms of I^* . This might be due to the definition of I^* , which was shown in Equation 1.1. It should be noted that a chaotic region is apparent for falling disks, which is located in-between the periodic and tumbling regimes. A chaotic region for falling plates might exist as well, but has not been presented in the referred studies. The different motion classifications for falling plates have been shown in Figure 1.3.

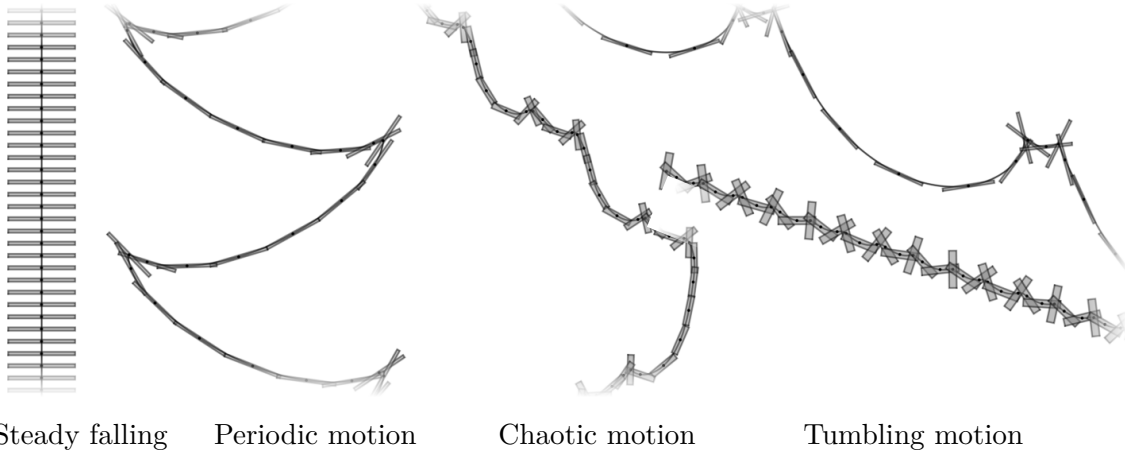


Figure 1.3: The steady falling, periodic, chaotic and tumbling motion classifications shown from left to right.

The classifications are easy to identify, though the steady falling classification might experience some periodic motion. This can be related to the unsteady nature of falling objects, where vortices will be shed causing periodic oscillations due to the varying surface pressure. These forces might be so small so the definition of steady falling will be a subject definition. However, in *Stokes flow* at $Re \ll 1$, viscous forces are dominant, so no vortices will be shed, thus causing linear motion of the plate. A definition of the steady falling motion is stated as:

"...the [plates would fall] steadily perpendicular to the flow without any sort of periodic motion and would move to a position perpendicular to the flow if dropped in some other position." - [Smith, 1970]

The dimensionless terms in Equation 1.1 are utilized in this study along with other dimensionless groups. Some of these are the aspect ratio of the plate, β , and the density ratio, ρ^* , which have been defined in Equation 1.2. Redefinitions of Re and I^* have been shown as well when expressed in terms of β and ρ^* . The moment of inertia of a rectangular plate has also been included, where m is the mass of the plate.

$$\begin{aligned}
 \beta &= \frac{L}{H} & Re &= \frac{u L \rho_f}{\mu} = \frac{u H \beta \rho_p}{\mu \rho^*} \\
 \rho^* &= \frac{\rho_p}{\rho_f} \\
 I &= \frac{m (L^2 + H^2)}{12} & I^* &= \frac{32 I}{\pi \rho_f L^4 (1 \text{ m})} = \frac{8 \rho^* (\beta^2 + 1)}{3 \pi \beta^3}
 \end{aligned} \tag{1.2}$$

The subscripts $()_p$ and $()_f$ denotes properties related to the plate and the fluid respectively. The definition of I^* in terms of β can be seen to have a third order dependency on β . Re is defined in terms of a velocity, which can be calculated in multiple ways. Some studies have used the apparent vertical terminal velocity, whereas other studies have calculated the terminal velocity by balancing out buoyancy, gravity and drag forces. However, the last definition relies on an accurate average drag coefficient, which might be difficult to predict. In this study, the terminal velocity is calculated as the average vertical velocity after periodic motion has been obtained. For tumbling motions, the terminal velocity is calculated as the average velocity magnitude as opposed to the average vertical velocity. It was found that this definition was more useful for tumbling plates.

Based on the effort on making a general model of freely falling objects it is clear that exact trajectories are only available by means of massive computational power and time. However, as modelling of fluid phenomena has an infinitely number of degrees of freedom, simplifications and assumptions are an important aspect when expanding the existing framework of such models.

2 | PROBLEM STATEMENT

Various methods can be used to predict trajectories of falling plates. One method is to model the phenomena using *Computational Fluid Dynamics*, CFD, where the flow field around the plate is resolved, which allows for evaluation of the forces acting on the plate. CFD-analyses typically requires large computational time, which limits the applicability of this method. As of such, multiple studies have proposed methods of approximating the trajectories for falling plates using simplified models, where this study will extend the framework within this field. In order to do so, a problem statement is formed to focus the study and limit the extent:

Can a predictive trajectory model be developed to accurately capture the dynamics of a wide range of plate geometries with varying densities?

In order to answer the question stated above, analyses of the various aspect of falling plates will be carried out. These analyses are divided into two main groups, which are a numerical and a modelling approach.

2.1 Problem solving strategy

The two main groups will be subdivided into smaller tasks. These tasks, along with a description of the analyses and solving strategies, will be stated in the following subsections.

2.1.1 Numerical approach

The numerical approach analyses falling plates by use of CFD-simulations, where the forces are found by resolving the flow field around the plates. This approach will investigate the following aspects:

- Identification of the governing forces for a falling plate in a resolved flow field.
- Development of a CFD-model to predict the trajectory for any given plate.
- Verification of the CFD-model by an experimental approach.
- Parametric study of falling plates for a wide range of plate sizes and densities.

The aspects listed above have been shown graphically in Figure 2.1.

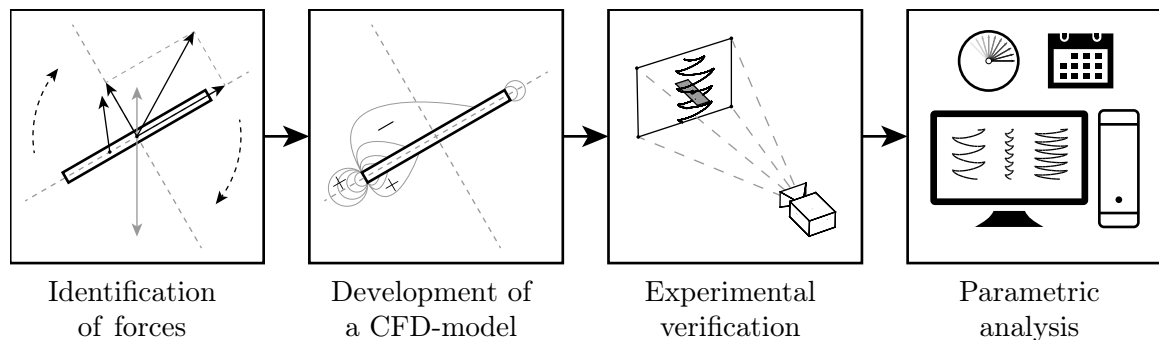


Figure 2.1: The different aspects of the numerical approach, where each box is related to each aspect in the above list.

The first aspect is to identify the governing forces for falling plates. This analysis will cover forces caused by the apparent flow field around the plate along with the conservative forces.

The second aspect will be covered by developing a CFD-model in *ANSYS Fluent*. This model will be used to predict trajectories of different plates by resolving the flow field. The model will utilize *User Defined Functions* coupled with the solver in *ANSYS Fluent*

to calculate the forces acting on the plate, after which a dynamic integration procedure will be applied to update the plate velocities. The focus of this model will be to make it computationally fast to execute, which allows for an extensive parametric analysis required to develop an accurate *Predictive Trajectory model*, PT-model.

The third aspect regarding verification of the CFD-model will be covered by conducting a set of experiments. These experiments serve to statistically validate the CFD-model for five different shaped plates with varying density. The experiments will be made by dropping the plates in water and recording the trajectories, after which the image sequences will be processed and compared with CFD-simulations of similar configurations.

The fourth aspect regarding a parametric study will be covered by utilizing the CFD-model to investigate the trajectories and forces for a wide range of plates. In total, 36 simulations will be carried out, where most will contribute to the development of the PT-model.

2.1.2 Modelling approach

The modelling approach uses the results obtained in the numerical approach to develop a PT-model. This approach will cover different aspects, which have been listed below:

- Identification of the constructed forces governing motion for a falling plate.
- Development of a PT-model using the results from the parametric analysis as guideline.
- Analysis of the PT-model.

The aspects listed above have been shown graphically in Figure 2.2.

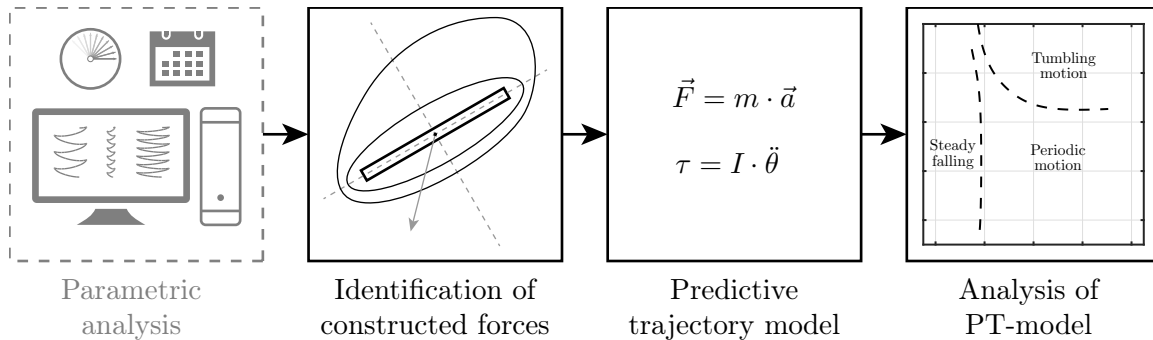


Figure 2.2: The different aspects of the modelling approach, where each box is related to each aspect in the above list. The parametric analysis is a continuation from Figure 2.1.

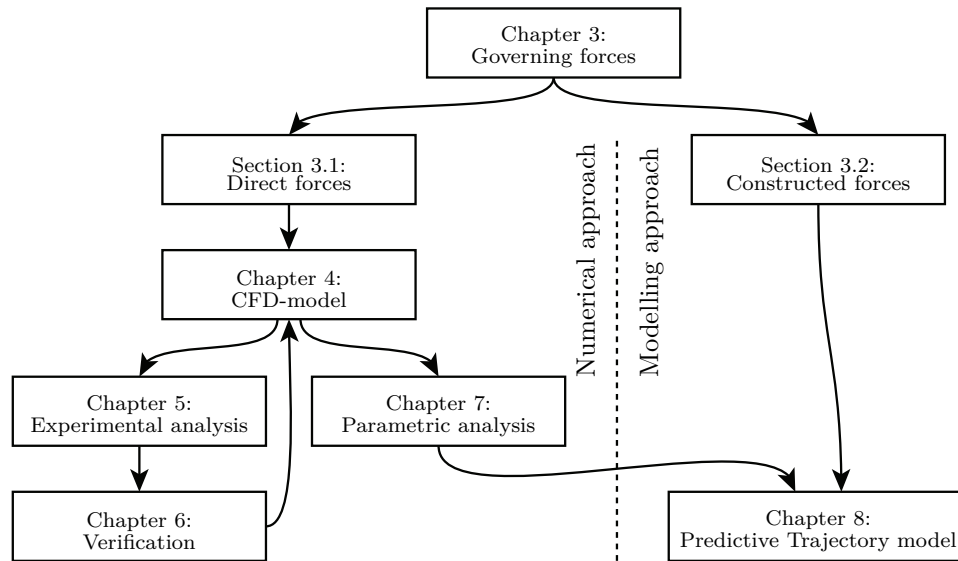
The first aspect it to identify the constructed forces governing falling plates. As the plate dynamics will be modelled, the apparent flow field around the plate is not known so the direct forces are not available. As of such, other approaches have to be applied in order to predict the dynamics. The constructed forces are analysed, which are including the aerodynamic, added mass, history and circulation forces.

The second aspect is to develop a PT-model using the results obtained from the parametric study. This PT-model will be formed by a set of coupled differential equations. Optimization routines will be utilized to tune various coefficients in the PT-model in order for it to match CFD-simulations of similar configurations.

The third aspect it to analyse the developed PT-model. This analysis will cover various regime maps and bifurcation of the trajectory motions, which, assuming the model is valid, can reveal interesting behaviours of plates in free fall.

2.2 Report structure

The report structure has been shown visually in Figure 2.3.



3 | GOVERNING FORCES

The chapter serves to cover the forces, which are acting on a plate in free fall. The description is separated into two main groups, which are direct and constructed forces. Direct forces are the ones caused by the apparent flow field, where various parameters can be integrated to yield force and torque contributions. The constructed forces are those, which are used when the flow field is not available. These covers aerodynamic, added mass, history and circulation forces, which will be utilized to develop the PT-model in *Chapter 8 - Predictive Trajectory model*.

3.1 Direct forces

The direct forces acting on a plate in free fall are shown in Equation 3.1.

$$\vec{F} = \vec{F}_p + \vec{F}_\nu + \vec{F}_g \quad (3.1)$$

\vec{F} is the total force, \vec{F}_p is the pressure force, \vec{F}_ν is the viscous force and \vec{F}_g is the gravitational force. These forces have been described in the following subsections.

3.1.1 Pressure forces

The pressure force can be divided into two parts, which are the static flow pressure and the hydrostatic pressure. These have been shown in Equation 3.2 as surface integrals.

$$\vec{F}_p = \underbrace{\int_S p d\vec{A}}_{\vec{F}_{p,S}} + \underbrace{\int_S (-\rho_f g y_{\text{Rel}}) d\vec{A}}_{\vec{F}_{p,H}} \quad (3.2)$$

p is the static pressure without the pressure increase due to hydrostatic effects, where the resulting surface integral is $\vec{F}_{p,S}$. The hydrostatic force is denoted $\vec{F}_{p,H}$ and \vec{A} is the normal area vector, which has a direction normal to the surface and a magnitude equal to the infinitesimal surface area. This formulation allows for direct implementation in *ANSYS Fluent*, where $d\vec{A}$ is the area normal vector of each face on the plate, though this will take a numerical value in the simulation and the surface integral will be performed as a sum instead.

The pressure forces are split in two terms in Equation 3.2, which will be utilized in the model in *Chapter 4 - CFD-model*. This allows for a more general identification of the forces acting on the plate. The hydrostatic pressure force will be implemented according to the last term in Equation 3.2, where y_{Rel} is the y-coordinate of the plate relative to its centre. The reason for formulating the hydrostatic forces in a relative reference frame is that this force will depend on the depth, at which the plate is located, due to increasing pressure. This is not desirable, as this will yield a new parameter, where the number of parameters are to be kept at a minimum in order to avoid increasing complexity.

3.1.2 Viscous forces

The viscous force has been shown in Equation 3.3.

$$\vec{F}_\nu = - \int_S \left(\mu \frac{\partial \vec{u}}{\partial \vec{n}} \right) dA \quad (3.3)$$

The viscous force is integrated along the surface of the plate, where the force is equal to the dynamic viscosity multiplied with the velocity gradient at the wall inside the viscous sublayer in the direction normal to the surface, \vec{n} .

3.1.3 Gravitational forces

The gravitational force has been shown in Equation 3.4.

$$\vec{F}_g = m \vec{g} \quad (3.4)$$

\vec{g} will be constant in this study of $[0, -9.81]$ m/s². Furthermore, it is assumed that the plate is made of an isotropic material, so the density is assumed constant over the plate, thus yielding that \vec{F}_g will act in the geometric centre of the plate, which will coincide with the centre of mass.

3.1.4 Torques

The pressure and viscous forces will typically not act in the centre of the plate, so torques are associated with the forces. These torques are shown in Equation 3.5, where τ is the total torque around the centre of the plate.

$$\tau = \tau_p + \tau_\nu = \tau_{p,S} + \tau_{p,H} + \tau_\nu \quad (3.5)$$

The subscripts in Equation 3.5 are related to the previous forces. There is no torque contribution from \vec{F}_g , as this force will act on the centre of mass, which is coinciding with the geometric centre. The torques in Equation 3.5 can be calculated by Equation 3.6.

$$\tau = \frac{1}{A} \int_S (\vec{r} \times \vec{F}) dA \quad (3.6)$$

\vec{r} is the vector going from the geometric centre to the point on the surface, at which the torque is integrated over. \vec{F} in this equation is representing any of the previously described forces, so the torque contribution from both the pressure and viscous forces are calculated. The reason for formulating the torque as an integral over the surface area is once again to relate it to the implementation in *ANSYS Fluent*, where the term dA is related to each face on the surface as previously described.

It should be noted that the hydrostatic torque is typically non-zero, but the magnitude is very small. However, this contribution is still implemented in *ANSYS Fluent* in *Chapter 4 - CFD-model*.

3.2 Constructed forces

The constructed forces span a wide range of forces, which are not available from direct measurements of the apparent flow field. The constructed forces used in this study are the aerodynamic, added mass, history and circulation forces, and will be used in the PT-model described in *Chapter 8 - Predictive Trajectory model*. These will be described in the following subsections.

3.2.1 Aerodynamic forces

The aerodynamic forces are commonly known when dealing with wing profiles, where drag and lift torque coefficients are used for characterization. When these dimensionless quantities are known, the forces acting on the airfoil can be calculated. Equation 3.7 is showing the aerodynamic drag force, where c_D is the coefficient of drag, q being the dynamic pressure and L the characteristic length. \vec{n}_D is the normal vector pointing in the direction of the drag, which is commonly the direction of the freestream velocity.

$$q = \frac{1}{2} \rho u^2 \quad (3.7)$$

$$\vec{F}_D = q L c_D \vec{n}_D \text{ (1 m)}$$

Only the coefficient of drag has been shown, but the lift force is calculated in similar manner.

3.2.2 Added mass forces

The added mass forces are caused by an additional inertia to an object due to the acceleration of the surrounding fluid. It is thus only acting when the plate is accelerating, as the surrounding fluid is accelerated along with it. The added mass forces can be generalized for a 2D object as shown in Equation 3.8 [Brennen, 1982].

$$\begin{aligned}\mathbf{M}_p &= \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & I \end{bmatrix} \\ \mathbf{M}_f &= \begin{bmatrix} m_{11} & m_{12} & m_{16} \\ m_{21} & m_{22} & m_{26} \\ m_{61} & m_{62} & m_{66} \end{bmatrix} = \frac{\mathbf{C}_{AM} \cdot \mathbf{M}_p}{\rho^*} \\ \mathbf{C}_{AM} &= \begin{bmatrix} C_{11} & C_{12} & C_{16} \\ C_{21} & C_{22} & C_{26} \\ C_{61} & C_{62} & C_{66} \end{bmatrix}\end{aligned}\tag{3.8}$$

\mathbf{M}_p is the *mass-inertia tensor* for the plate, whereas \mathbf{M}_f is the *added mass tensor* for the surrounding fluid. The latter matrix is constructed by multiplying the mass and moment of inertia of the displaced fluid with the dimensionless added mass coefficients, \mathbf{C}_{AM} . It should be noted that the displaced fluid can be calculated using the density ratio as $(L H (1 \text{ m}) \rho_f = m/\rho^*)$. As of such, the added mass matrix values can, for instance, be calculated as $m_{11} = C_{11} (L H (1 \text{ m})) \rho_f = C_{11} m/\rho^*$.

The added mass coefficient matrix, \mathbf{C}_{AM} , has size 3×3 in the general 2D case, whereas the size is 6×6 in 3D. For symmetric objects in 2D, \mathbf{C}_{AM} is diagonal, so the grey matrix elements in Equation 3.8 are zero for rectangular plates.

The added mass coefficients can be determined by multiple methods. One such method is by use of potential flow, which is utilized in this study. This has been described in *Appendix A - Panel Methods*, but it is recommended that the reader waits with this appendix till *Chapter 8 - Predictive Trajectory model*. When having calculated \mathbf{M}_f , the added mass forces can be calculated by Equation 3.9.

$$\vec{F}'_{AM} = \mathbf{M}_f \cdot \begin{bmatrix} a'_x \\ a'_y \\ \ddot{\theta} \end{bmatrix}\tag{3.9}$$

The superscript $()'$ is indicating variables in the local coordinate system, which is following the plate. The transformation to this coordinate system is described in details in *Section 8.2 - Orientation vectors*.

3.2.3 History forces

History force is an unsteady force, which is related to previous motion for an object, and is commonly known as *Basset force*. When an object is accelerated, the boundary layer will have a time delay in its build-up. This causes the shear stresses experienced by the object to change over time, as the velocity gradients near the object are changing [Brennen, 2005]. The principle has been shown in Figure 3.1.

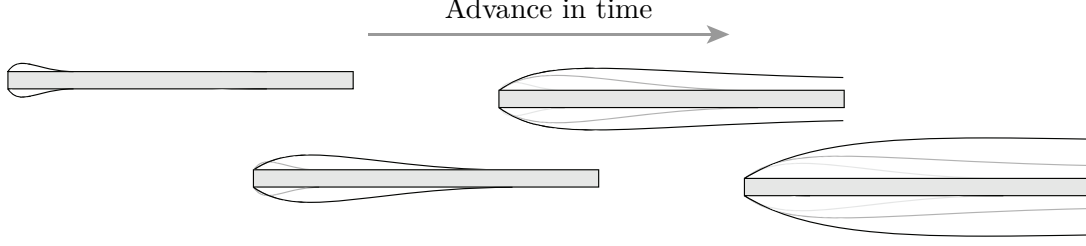


Figure 3.1: The boundary layer build-up as a function of time [Jensen and Hærvig, 2014].

As seen in Figure 3.1, the boundary layer will develop over time, where the thickness will increase along with the shear stresses. The development of the boundary layer can be modelled when assuming *Stokes flow*. However, an alternative approach will be utilized in this study when developing the PT-model, where a lagging velocity vector with a specified time constant will follow the actual plate velocity vector. This lagging velocity will act as an approximation of the surrounding fluid, which is following the plate. As of such, when the plate is turning and moving into its own wake, it will be subtracted from the plate velocity to yield a relative velocity seen by the plate. This will in turn approximate the boundary layer build-up, though not being an complete analogue to the *Basset force*. This method is described in details in *Chapter 8 - Predictive Trajectory model*.

3.2.4 Concept of circulation

The PT-model will utilize circulation to calculate the major external forces. Circulation, Γ , is defined in Equation 3.10.

$$\Gamma = \oint_C \vec{u} \cdot d\vec{l} \quad (3.10)$$

Equation 3.10 states that Γ is the closed curve integral of the dot product between the velocity vector and the infinitesimal curve length. This has been illustrated in Figure 3.2.

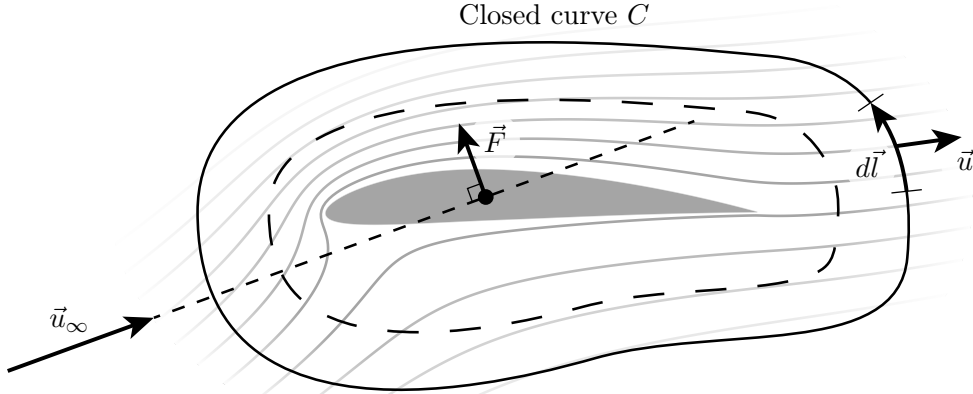


Figure 3.2: The concept of calculating the circulation around an object. The grey lines are streamlines.

As seen in Figure 3.2, an airfoil with grey associated streamlines has been shown. The closed curve shown as the solid black line enclosing the airfoil, C , can be integrated according to Equation 3.10 in order to obtain Γ . The circulation around an object is constant in an inviscid and irrotational flow field, so the dashed closed curve in Figure 3.2 has the same values of Γ as the solid closed curve integral. The circulation can also be found for a viscous flow field by performing a surface integral of the vorticity. This has been shown in Equation 3.11, where $(\nabla \times \vec{u})$ is the vorticity.

$$\Gamma = \iint_S (\nabla \times \vec{u}) \cdot d\vec{S} \quad (3.11)$$

$d\vec{S}$ is the infinitesimal surface normal vector whose magnitude is equal to the surface area. When Γ is known, the resulting circulation force, \vec{F}_Γ , acting on the airfoil can be calculated according to the *Kutta-Joukowski theorem* [Anderson, 1978] as shown in Equation 3.12.

$$\vec{F}_\Gamma = - \underbrace{\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}}_{90 \text{ deg rotation matrix}} \cdot \vec{u}_\infty \Gamma \rho_f \quad (3.12)$$

As shown in Equation 3.12 and Figure 3.2, the resulting force is perpendicular to the surrounding velocity vector, \vec{u}_∞ , rotated 90 deg in the counter clockwise direction. The streamlines in Figure 3.2 are leaving the airfoil at the trailing edge smoothly, which is known as the *Kutta-condition*, which can be stated as:

"A body with a sharp trailing edge in motion through a fluid creates about itself a circulation of sufficient strength to hold the rear stagnation point at the trailing edge of finite angle to make the flow along the trailing edge bisector angle smooth. For a body with a cusped trailing edge where the upper and lower surfaces meet tangentially, a smooth flow at the trailing edge requires equal velocities on both sides of the edge in the tangential direction." - [Kuethe and Chow, 1978]

It should thus be noted, that if the airfoil shown in Figure 3.2 had $\Gamma = 0$, the streamlines would separate on the upper half of the airfoil. This phenomena can be seen in Figure A.4 in *Appendix A - Panel Methods*, where panel methods are utilized to calculate the potential flow.

When dealing with objects in an inviscid and irrotational flow field such as the one shown in Figure 3.2, the resulting force in the direction of \vec{u}_∞ is zero, which is known as *d'Alembert's paradox* [Anderson, 1978]. As of such, the aerodynamic drag force previously described in Equation 3.7 is utilized in the PT-model to model drag forces.

In order to utilize circulation in the PT-model, an expression for Γ has to be made, when dealing with plates. This will be described in both *Chapter 8 - Predictive Trajectory model* and in *Appendix A - Panel Methods*, but once again it is recommended that the reader postpones this appendix.

3.3 Motion

When having calculated all forces acting on the plate, the accelerations of the plate can be calculated. This has been shown for the general 3D case in Equation 3.13 [ANSYS, Inc., 2013a]. The superscript $(\cdot)'$ indicates the local coordinate system, which is following the object.

$$\begin{aligned} \vec{a} &= \mathbf{M}^{-1} \cdot \vec{F} \\ \ddot{\vec{\theta}}' &= \mathbf{L}^{-1} \cdot \sum \left(\vec{\tau}' - \dot{\vec{\theta}}' \times (\mathbf{L} \cdot \dot{\vec{\theta}}') \right) \end{aligned} \quad (3.13)$$

The first equation is independent of the local coordinate system, so the local superscripts have been removed. \mathbf{M} is the mass matrix and \mathbf{L} is the moment of inertia tensor. As this study deals with planar motion, Equation 3.13 can be simplified to the general 2D case, which has been shown in Equation 3.14 to the left.

$$\underbrace{\begin{bmatrix} a_x \\ a_y \\ \ddot{\theta} \end{bmatrix}}_{\text{CFD-model}} = (\mathbf{M}_p)^{-1} \cdot \begin{bmatrix} F_x \\ F_y \\ \tau \end{bmatrix} \qquad \underbrace{\begin{bmatrix} a'_x \\ a'_y \\ \ddot{\theta} \end{bmatrix}}_{\text{PT-model}} = (\mathbf{M}_p + \mathbf{M}_f)^{-1} \cdot \begin{bmatrix} F'_x \\ F'_y \\ \tau \end{bmatrix} \quad (3.14)$$

In the CFD-model, the accelerations can be directly calculated by use of *Newton's Second Law*, where the mass and moment of inertia of the plate are known beforehand. As of such, the accelerations can be calculated without orienting the forces relative to the local coordinate system. In the PT-model, the added masses are taken into account, which can be seen to the right in Equation 3.14. In this case, the local coordinate system must be respected, which has been shown with the superscripts $()'$.

3.4 Partial conclusion

This chapter has described the direct and constructed forces. The direct forces are those, which can be calculated when the apparent flow field is known. Also, the conservative forces have been placed under this category, though they are independent of the flow field. The constructed forces are those, which are not available from the apparent flow field. The described ones are the aerodynamic, added mass, history and circulation forces. The constructed forces will be employed in the PT-model in *Chapter 8 - Predictive Trajectory model*.

As the governing forces have been described, a CFD-model can be made, which can calculate the trajectory of any given plate by resolving the flow field. This model has been described in the next chapter.

4 | CFD-MODEL

This chapter serves to describe the numerical model, which utilizes *Computational Fluid Dynamics*, CFD, where the governing differential equations of fluid motion are solved¹. The model was implemented in *ANSYS Fluent*, where a set of *User Defined Functions*, UDF's, were coupled with the program to calculate the forces acting on the plate and update the velocities. These UDF's can be found in *Enclosure A - ANSYS Fluent UDF's*.

The chapter will describe different approaches for modelling a plate in free fall by use of CFD. Also, the basic theory for a CFD-simulation will be described along with the chosen turbulence model. Furthermore, the dynamic integration procedure is described, which was implemented in the model by use of UDF's. Lastly, the meshing of the domain is described before the mesh and time-step independence are shown.

4.1 Domain

In order to model a falling plate, a domain must be defined, where the governing equations of fluid motion are to be solved. Multiple approaches can be used to model a falling plate, where such will be described in this section. Based on the pros and cons of the different approaches, one will be chosen for implementation.

4.1.1 Approaches to plates in free fall

In order to model a plate in free fall, the forces acting on it must be calculated. This implies that the flow field around the plate is resolved, as the calculated forces will otherwise be incorrect. The fluid domain must be of suitable size in order to resolve the flow field, yet small enough for fast computational time. Various approaches for modelling a plate in free fall has been employed in various studies [Jensen and Hærvig, 2014; Andersen et al., 2005b; Jin and Xu, 2007], where three different approaches have been shown in Figure 4.1. These have been shown in 2D, but they are all fully expandable to 3D. The 2D representation implied three degrees of freedom, which are the two spatial coordinates and the orientation of the plate.

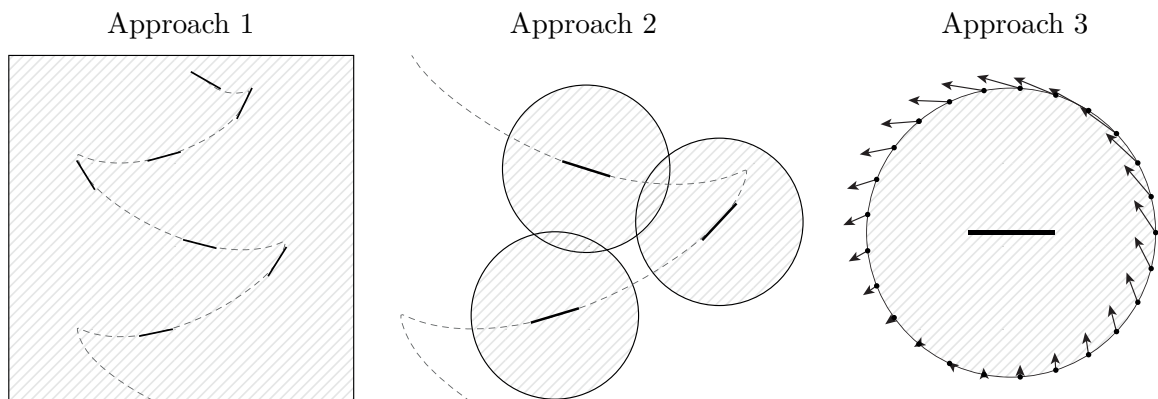


Figure 4.1: Three different approaches for modelling a falling plate.

Three different approaches for modelling a freely falling plate can be seen in Figure 4.1, where the shaded areas are the domains of the different approaches. Brief descriptions of these are shown in the following paragraphs.

¹The CFD-model can be found on the USB-stick under the path: `../ANSYS Fluent/`. An UDF-wrapper was coded in *MATLAB*, which can be found on this path as well, along with a guide for executing the model.

Approach 1

The first approach is an *Eulerian* approach, where the domain is a confined area, which is stationary over time. The falling plate is tracked over time, where it moves inside the domain. In this case, the domain can be the basin, in which the plate is dropped. This approach requires re-meshing, which implies that the mesh is updated at every iteration or time-step, where cells with poor quality are re-meshed in order to maintain a high mesh quality. As of such, additional calculations are required in order to perform such operations. Furthermore, as the mesh is constantly changing, each flow field scalar is interpolated from previous values before re-meshing, which can lead to interpolation errors. This method was implemented by reference [Jensen and Hærvig, 2014], where the required computational time was very large. An advantage of this approach is that wall effects will be included in the simulation, if the domain size is restricted to a size, where such effects are non-negligible. However, as the goal for this project is to analyse trajectories of falling plates in general manner, such effects are to be avoided.

Approach 2

The second approach is a *Lagrangian* approach, where the domain is defined around the plate, but moves over time with the plate. Instead of updating the mesh by use of re-meshing as in the first approach, this method moves the whole domain along with the mesh over time relative to a fixed coordinate system. Thus, this method does not require additional calculations to perform re-meshing. However, as the domain is restricted to a certain distance from the plate, some information about the flow might be lost. This problem can be dealt with by analysing the distance from the plate, at which the flow does not affect the forces on the plate significantly. As no walls are present in this approach, wall effects will not be affecting the trajectory, which leads to more general results, which is of interest in this study.

Approach 3

The third method is also a *Lagrangian* approach, where the domain is fixed around the plate. Instead of moving the mesh as in the second approach, only the boundary conditions are updated over time. Thus, this method neither requires re-meshing nor moving mesh methods, but the mathematics required to update the boundary conditions are more complex compared to the second approach. Furthermore, post-processing of the results are not as intuitive as the first two approaches, as the plate remains stationary. As with the second approach, wall effects are not present.

Approach for implementation

All three approaches should ideally yield the same trajectories, but the computational time required for simulation might differ a lot. Furthermore, as the implementation was done using UDF's in *ANSYS Fluent*, the method should be as intuitive as possible. Based on the pros and cons of the different approaches, the second one was chosen for implementation, as it did not require re-meshing procedures and was intuitive to implement and post process. Wall effects would not be present in this approach, which would lead to more general results.

As a parametric study was to be carried out, which is described in *Chapter 7 - Parametric analysis*, the required computational time per simulation should be kept at a minimum. Based on results from reference [Jensen and Hærvig, 2014], the first approach was found unfit for parametric studies due to the long computational time. Lastly, existing UDF's were available in *ANSYS Fluent* for dynamic meshes, which simplified the implementation significantly [ANSYS, Inc., 2013b].

4.1.2 Geometry

Based on the modelling approach, a suitable domain size had to be chosen. As previously stated, the size of the domain had to be of adequate size in order to ensure that the error of limiting the domain size was negligible. Results from reference [Jensen and Hærvig, 2014] were available, where the flow structure was analysed. The referred study used the first approach, where the domain was fixed and re-meshing was employed. This allowed for analysis of the distant flow field relative to the plate. These results indicated that most vortices had dissipated at about $r/L \approx 7.5$, where r is the radial distance from the plate and L is the plate length. This domain size also ensured that the plate was moving back into its own wake, which might affect its trajectory. This has been shown in Figure 4.2.

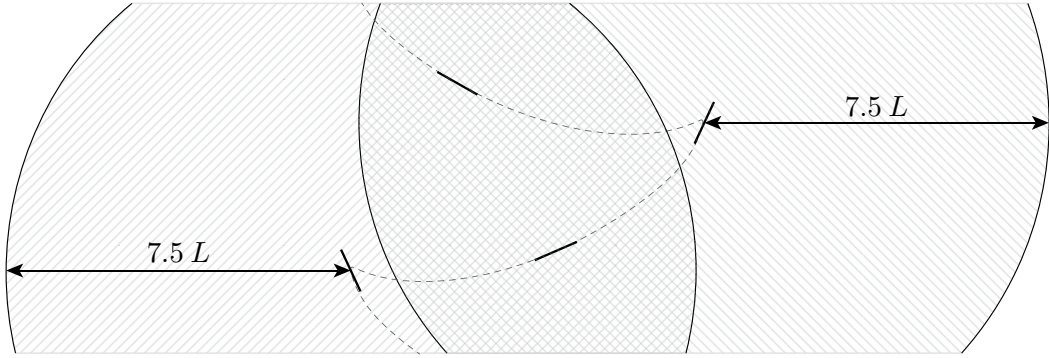


Figure 4.2: By ensuring a domain size in the order of $r/L \approx 7.5$, the wake from the plate will affect the plate as it is returning from a turning point. The plate length, trajectory and domain diameter are to scale and obtained from a simulation.

As seen in Figure 4.2, a domain size of $r/L \approx 7.5$ will ensure that the plate is affected by its own wake when it is falling. It can be seen that the domain size is almost capable of capturing the flow from the previous turning point. However, the trajectories will differ with plate dimensions and density, so the width of the trajectories between two turning points might differ.

In order to quantify the required domain size, the vorticity magnitude, $|\nabla \times \vec{u}|$, the turbulence kinetic energy, k , and the strain rate, S , have been investigated as a function of the radial distance from the plate. The reason for choosing these scalars was that the vorticity magnitude can be related to the vortex strength or the local rotation of the fluid, the turbulence kinetic energy is related to the turbulent fluctuation velocity and the strain rate is related to the deformation of a given control volume of the fluid [Bakker, 2006a]. If these scalars are small compared to the values near the plate, the flow can be seen as settled due to the low fluid motion. The results can be seen in Figure 4.3.

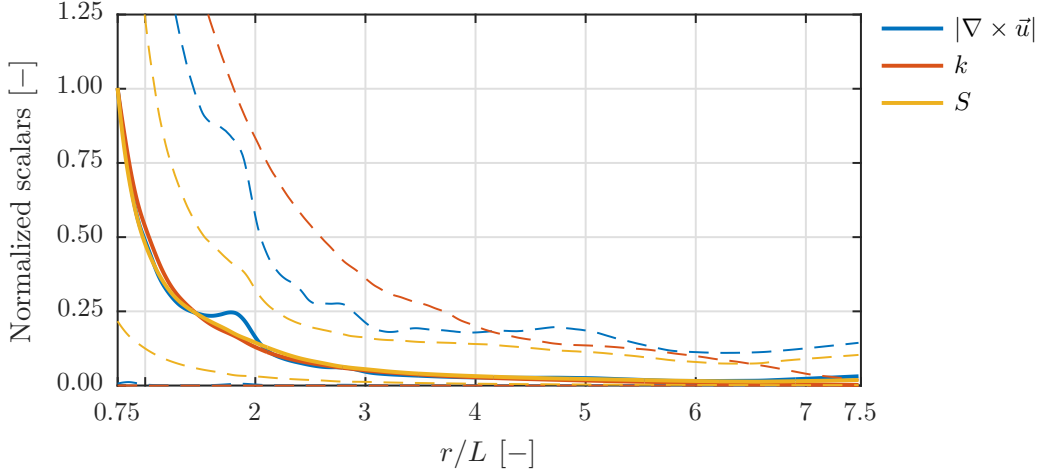


Figure 4.3: The vorticity magnitude, $|\nabla \times \vec{u}|$, the turbulence kinetic energy, k , and the strain rate, S , as a function of the radial distance from the plate. The curves are normalized to a mean value of 1 at $r/L = 0.75$. The solid lines indicate the mean values and the dashed lines indicate the 5 and 95 % percentiles of the values.

The curves in Figure 4.3 were constructed by exporting the flow variables at each time-step for a simulation spanning 1.2 s. The scalars were then interpolated as a function of the radial distance from the plate, after which the results were analysed. The solid curves are indicating the mean values, whereas the dashed lines are showing the 5 and 95 % percentiles of the values. Thus, 90 % of the values are within the two dashed lines for each scalar.

It can be seen that all scalars are decreasing rapidly near the plate, after which they settle near zero at $r/L \approx 4$. As of such, a domain with size $r/L = 7.5$ should be able to capture the important features of the flow near the plate.

4.2 Governing equations

This section serves to describe the basic theory behind a CFD-simulation, where multiple coupled partial differential equations are solved in order to model the fluid flow. The *Navier-Stokes equations*, NS-equations, are governing fluid flow and have been shown in Equation 4.1 without the energy-equation [Versteeg and Malalasekera, 2007].

$$\begin{aligned} \rho \left(\frac{\partial}{\partial t} + \nabla \cdot \vec{u} \right) &= 0 \\ \rho \left(\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla) \vec{u} \right) &= -\nabla p + \rho \vec{g} + \mu \nabla^2 \vec{u} \end{aligned} \quad (4.1)$$

The first equation of Equation 4.1 is the continuity equation, whereas the second are the momentum equations shown in a compact form. As the simulations were conducted in water, the density can be assumed constant, which simplifies the equations by dividing out ρ . The NS-equations are based on the general transport equation, which has been shown in Equation 4.2 for a given scalar, ϕ .

$$\frac{\partial(\rho \phi)}{\partial t} + \nabla \cdot (\rho \phi \vec{u}) = \nabla \cdot (D \nabla \phi) + S_\phi \quad (4.2)$$

The NS-equations can be derived by replacing ϕ with 1 for the continuity equation and \vec{u} for the momentum equations. As the chosen modelling approach employs a moving mesh, the general form of the transport equation is altered, as the velocities are relative to the moving mesh. This altered transport equation can be seen in Equation 4.3 [ANSYS, Inc., 2013a].

$$\frac{\partial(\rho \phi)}{\partial t} + \nabla \cdot (\rho \phi (\vec{u} - \vec{u}_m)) = \nabla \cdot (D \nabla \phi) + S_\phi \quad (4.3)$$

\vec{u}_m is the mesh velocity, where the relative velocity is apparent in Equation 4.3 as the term $(\vec{u} - \vec{u}_m)$.

4.3 Turbulence modelling

Turbulence modelling was utilized to approximate the turbulent phenomena in the flow. Doing so using the NS-equations alone is called a *Direct Numerical Simulation*, DNS, where the vortices of the flow are fully resolved till they dissipate as thermal energy at the *Kolmogorov scale* [Versteeg and Malalasekera, 2007]. However, DNS requires a very fine mesh and a very small time-step, which is not practical for general purposes and parametric studies [ANSYS, Inc., 2013c]. As of such, turbulence models are used to model the turbulent flow fluctuations by use of various scalars.

For flow phenomenas where both near wall and free stream flows are present, the *Shear Stress Transport turbulence model*, SST-model, is recommended [Casey and Wintergerste, 2000]. This model is a combination of the k - ε - and the k - ω -turbulence models. The k - ε -model is used in the free stream, whereas the k - ω -model is employed near the wall, as this is superior at predicting the flow structure when an adverse pressure gradient is present [ANSYS, Inc., 2013a].

The standard SST-model is a two equation turbulence model, where the extra scalars are the turbulent kinetic energy, k , and either ε or ω dependent on the y^+ value by use of a blending function [Autodesk, 2014]. However, instead of applying the standard SST k - ω -model, the Transition SST-model was chosen for implementation. The reason for using the Transition SST-model was to be able to correctly capture the flow forces acting on the plate, as these would be governing the motion of a falling plate. This model couples the standard SST-model with two additional scalars, which are the intermittency and the momentum-thickness Reynolds number. The intermittency, γ , is defined as the fraction of time during the transition phase where the flow is turbulent. Thus, a value of one indicates a fully turbulent region, whereas a value of zero indicates a fully laminar region. By introducing this scalar, the onset of transition to turbulent flow can be controlled. The momentum thickness Reynolds number is used to relate the free-stream turbulence intensity and the pressure gradient to the turbulent onset location [Suluksna and Juntasaro, 2008].

The Transition SST-model is quite stringent with respect to the near wall mesh, where the following guidelines are recommended [ANSYS, Inc., 2013a]:

- y^+ should be close to 1.
- The wall normal cell expansion ratio should be less than 1.1.
- A minimum of about 75 - 100 mesh vertices in the streamwise direction.

These guidelines are to be complied when meshing, which will be described in *Section 4.6 - Meshing*. y^+ is a dimensionless group, which can be seen in Equation 4.4 [ANSYS, Inc., 2013a].

$$y^+ = \frac{u^* y \rho}{\mu} \quad (4.4)$$

$$u^* = \sqrt{\frac{S_{\text{Wall}}}{\rho}}$$

S_{Wall} is the wall shear stress. The requirement on y^+ ensures that the transition onset location will be correctly captured, as increasing y^+ will move this location upstream.

Furthermore, it is also recommended that second order upwind discretization is used for the mean flow-, turbulence- and transition-equations, which is implemented in the model as well [ANSYS, Inc., 2013a].

4.4 Boundary conditions

The chosen approach of letting the domain and mesh move along with the plate required two boundary types. These were a no-slip wall for the plate and a suitable boundary type for the domain perimeter. These will be described in the following subsections.

Wall

The plate should impose a no-slip condition. This condition has been stated in Equation 4.5, where the shear stress is zero at the wall.

$$\mu \frac{\partial \vec{u}}{\partial \vec{n}} \Big|_{y=0} = 0 \quad (4.5)$$

A plate will in general have a non-zero roughness, which should be taken into account when modelling a wall. However, it was assumed that this effect was negligible and will not be investigated in this study. This also reduces the number of variables for the parametric study, though it is not assumed being of much importance.

Velocity inlet

The perimeter of the domain should allow fluid to both enter and exit depending on the direction, at which the mesh is moving. Multiple boundary types can be used for this condition [ANSYS, Inc., 2013a], where a velocity inlet was chosen for implementation in the model. When the flow exits the domain by a velocity inlet boundary type, *ANSYS Fluent* uses the upstream scalars to calculate the scalar flux exiting the domain [ANSYS, Inc., 2013c]. When flow enters the domain, the various scalar values must be specified which have been shown in Equation 4.6.

$$\begin{aligned} \vec{u} &= 0 \\ \gamma &= 0 \\ T_u &= 5 \% \\ \frac{\mu_t}{\mu} &= 10 \end{aligned} \quad (4.6)$$

The velocity was specified to zero, which is an absolute formulation. This is due to the stagnant fluid assumed being at the perimeter of the upstream flow. The intermittency, γ , is zero indicating a fully laminar flow, but the turbulence intensity, T_u , is defined to be 5 %. As no walls are present at the perimeter the intermittency will not be important in this region. The turbulent viscosity ratio is defined to be 10, but this value will not play any significant role at the upstream flow neither.

4.5 Dynamic integration procedure

As stated in *Chapter 3 - Governing forces*, the motion of a falling plate can be calculated by integrating the forces on the plate and updating the accelerations. This method can be carried out as either an explicit or implicit method. When choosing an explicit mesh update method, the velocity of the mesh is only updated once per time-step, namely at the beginning of it. Af of such, the plate will be moving with constant speed corresponding to the updated one at the beginning of the time-step. However, as the mesh velocity is

dependent on the flow field, implicit mesh update is recommended [ANSYS, Inc., 2013c] and will thus be implemented.

4.5.1 Implicit mesh update

This subsection will use examples of how the velocities and accelerations are updated in the CFD-model. All presented equations apply for both translational and rotational velocities and accelerations, but for simplicity, two common variables are used. These are u and a for velocity and acceleration respectively, where either u_x , u_y or $\dot{\theta}$ can be substituted for the velocity, and the same applies for acceleration.

Implicit mesh update requires velocity relaxation for convergence, which has been defined in Equation 4.7 [ANSYS, Inc., 2013c].

$$\begin{aligned} u_k &= u_k^* (\varphi) + u_{k-1} (1 - \varphi) \\ u_k^* &= u_{k-1} + a_k \cdot \Delta t \end{aligned} \quad (4.7)$$

k indicates the iteration number and u_k^* the undamped velocity, which is defined in the lower equation. The relaxation parameter, φ , is a value between 0 and 1. The reason for introducing this variable is to ensure a stable solution. A typical value is 0.1 - 0.2, so each iteration only adjusts to 10 - 20 % of the newly calculated undamped value, u_k^* . Equation 4.7 can also be interpreted as a discrete first order filter, where φ can be related to the time-constant.

It was found during implementation that the most time-consuming part of each time-step was to update the mesh position. The task was carried out at constant iteration intervals when using the in-built 6DOF-solver in *ANSYS Fluent*. It was found that this time-consuming task was only needed often in the beginning of each time-step and not so often at the end of a time-step. As of such, the UDF's were written to overwrite this method in *ANSYS Fluent* to only update the mesh at specified iterations. This method allowed for many mesh updates at the beginning of each time-step, where convergence was difficult to obtain. This method has been illustrated in Figure 4.4.

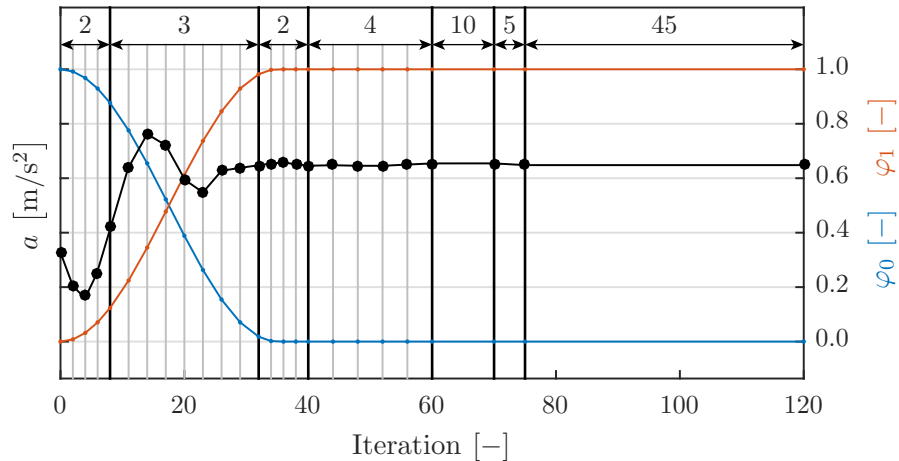


Figure 4.4: The method of updating the mesh at specified iterations during a time-step. φ_0 and φ_1 are blending functions.

As seen in Figure 4.4, the mesh is updated at specified iterations indicated by the vertical grey lines. A total of 120 iterations per time-step is used, where 25 mesh updates are executed. It can be seen that the spacing between each mesh update is smaller at the beginning compared to the end of a time-step. The differences between each update is shown in the top of Figure 4.4 as the numbers, where, for instance, the spacing is 2 iterations

in the beginning. The coloured lines in Figure 4.4 are indicating φ_0 and φ_1 in blue and red respectively and are related to the right y-axis. These blending functions are spanning between 0 and 1 and are used to guide the acceleration towards convergence. The function, which they are blending, are shown in Equation 4.8.

$$\begin{aligned} a_{k+1} &= \varphi_0 f_0 + \varphi_1 f_1 \\ 1 &= \varphi_0 + \varphi_1 \\ f_0 &= \text{Least squares weighted fit} \\ f_1 &= \text{Equation 4.10} \end{aligned} \tag{4.8}$$

The two blended functions, f_0 and f_1 , were used for more stable convergence. The fit function, f_0 , used the four most recent points, which were weighted by $[1.00, 1.00, 0.75, 0.35]$, where the leftmost value is the oldest value and the rightmost is the newly calculated acceleration. The reason for using an even number of points was the tendency for the acceleration to fluctuate slightly above and below the average value, where an even number will smooth this tendency out. Also, the oldest values were weighted more compared to the newest, as this was found to increase stability. The second function, f_1 , is defined later on in Equation 4.10.

After iteration 75 the mesh is not updated anymore, so the remaining 45 iterations are used to converge the flow field. The relative residual for a typical time-step has been shown in Figure 4.5, where the seven scalars are present.

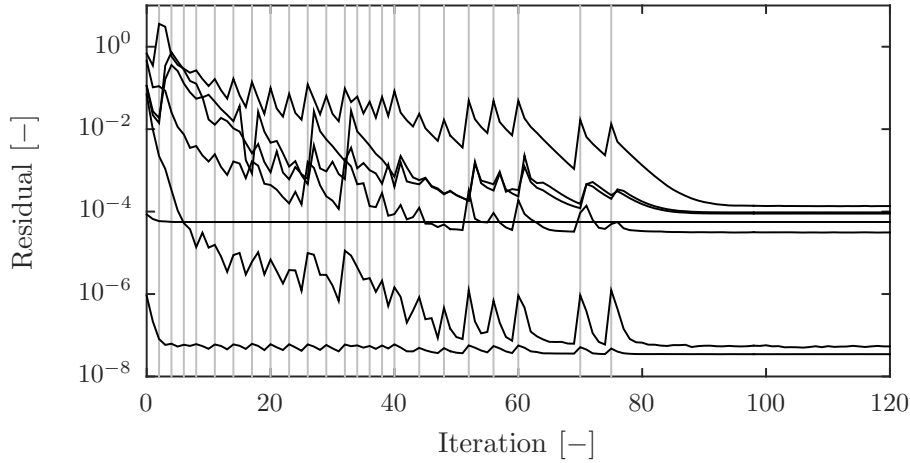


Figure 4.5: A typical relative residual plot for a single time-step. The vertical grey lines are indicating mesh updates.

As seen in Figure 4.5, the residual is increasing at each mesh update, which is indicated by the vertical grey lines. After iteration 75 the mesh is not updated anymore, and the residuals are converging towards constant converged values. The method of only updating the mesh at specified iterations allowed for fast simulations, which was found critical for parametric studies.

4.5.2 Integration scheme

A four-point integration scheme was employed to predict the acceleration in-between time-steps. The 6DOF-integration scheme in *ANSYS Fluent* is the *Adam-Moulton* linear multistep method [ANSYS, Inc., 2013a; Snyder et al., 2003], which was implemented in the UDF's as well, but it was modified for improved stability. The integration scheme was constructed by connecting the raw acceleration points, a_n , with a cubic spline curve, after

which this curve was integrated between the last two points. Assuming the spline curve constructed from four points is named $f(t, P_0, P_1, P_2, P_3)$, the scheme can be constructed by evaluating Equation 4.9, where n is the time-step number and $a_{n+0.5}$ is the average acceleration in-between t_n and t_{n+1} .

$$a_{n+0.5} = \frac{1}{t_{n+1} - t_n} \int_{t_n}^{t_{n+1}} f(t, a_{n-2}, a_{n-1}, a_n, a_{n+1}) dt = \frac{1}{24} [a_{n-2}, a_{n-1}, a_n, a_{n+1}] \cdot \begin{bmatrix} 1 \\ -5 \\ 19 \\ 9 \end{bmatrix} \quad (4.9)$$

However, Equation 4.9 can lead to numerical divergence, if there is significant noise. This was found to be the case, when the forces were integrated for small time-steps. As of such, the integration method was modified to introduce further damping. This was done by smoothing out the raw points by the midpoint rule, where these were connected by another spline curve. This new spline curve will typically be much smoother compared to the first method, which helped convergence. The new curve is integrated between n and $n+1$ once again in order to find an average acceleration, which is used in Equation 4.7. This integral has been shown in Equation 4.10, where the tilde accent, \tilde{a} , is referring to the new modified spline curve.

$$\begin{aligned} \tilde{a}_{n+0.5} &= \frac{1}{t_{n+1} - t_n} \int_{t_n}^{t_{n+1}} f\left(t, \frac{a_{n-3}+a_{n-2}}{2}, \frac{a_{n-2}+a_{n-1}}{2}, \frac{a_{n-1}+a_n}{2}, \frac{a_n+a_{n+1}}{2}\right) dt \\ &= \frac{1}{48} [a_{n-3}, a_{n-2}, a_{n-1}, a_n, a_{n+1}] \cdot \begin{bmatrix} -1 \\ 3 \\ -1 \\ 21 \\ 26 \end{bmatrix} \end{aligned} \quad (4.10)$$

The integration schemes shown in Equation 4.9 and Equation 4.10 have been visually shown in Figure 4.6, where the red and blue curves are the unmodified and modified spline curves respectively.

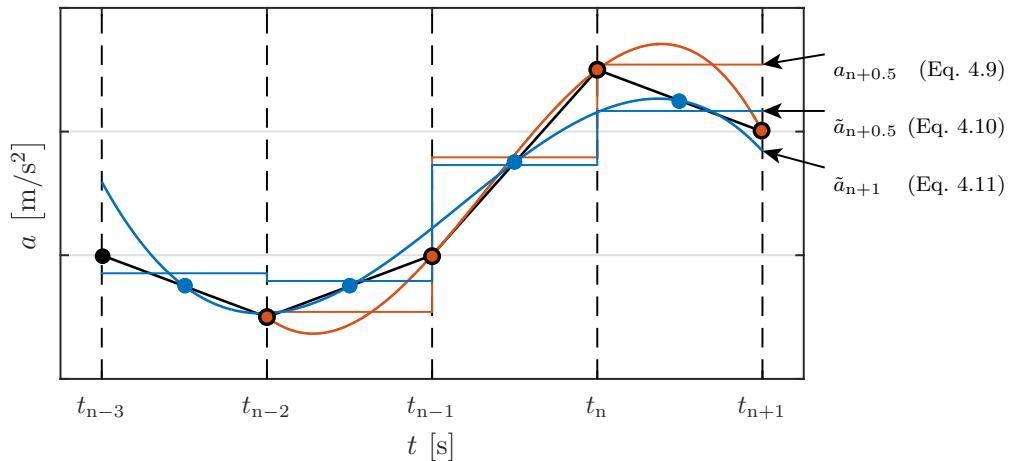


Figure 4.6: The four point integrations scheme used by *ANSYS Fluent* in red and the modified scheme in blue. The coloured dots are the points used to construct the corresponding spline curves.

The coloured curves in Figure 4.6 are showing the two different schemes described in Equation 4.9 and Equation 4.10. The coloured points are used to construct the spline

curves with the corresponding colours. It can be seen that the red dots are placed on top of the raw points in black, whereas the blue dots are placed in-between the raw points for each time-step. The blue curve, which is the modified four-point scheme, is not as aggressive to noise as the original scheme, as the values, which are used to construct the spline, are smoothed out by the midpoint-rule. Whether this scheme yields more realistic results can be debated, though the resulting trajectories are nearly identical, but the modified scheme proved more stable. It should be noted that both schemes are only applicable for constant time-steps in the form shown in Equation 4.9 and Equation 4.10.

In order to make the scheme consistent, the new point stored for further calculations, a_{n+1} , is evaluated as the value of the modified spline curve at $n + 1$. This has been shown in Equation 4.11.

$$\begin{aligned}\tilde{a}_{n+1} &= f\left(t_{n+1}, \frac{a_{n-3}+a_{n-2}}{2}, \frac{a_{n-2}+a_{n-1}}{2}, \frac{a_{n-1}+a_n}{2}, \frac{a_n+a_{n+1}}{2}\right) \\ &= \frac{1}{48} [a_{n-3}, a_{n-2}, a_{n-1}, a_n, a_{n+1}] \cdot \begin{bmatrix} -15 \\ 48 \\ -42 \\ 0 \\ 105 \end{bmatrix}\end{aligned}\tag{4.11}$$

An interesting feature of this point is the fact that a_n does not affect \tilde{a}_{n+1} . As the dynamic integration procedure has been defined, the mesh can be constructed, which is described in the next section.

4.6 Meshing

In order to solve the governing equations for fluid flow, a mesh consisting of a finite number of cells is to be constructed, on which the equations can be integrated. In order to obtain an accurate solution and comply with the meshing guidelines for the chosen turbulence method, certain considerations have to be taken into account when meshing. These are listed below:

- Turbulence model requirements:
 - y^+ should be close to 1.
 - The wall normal expansion ratio should be less than 1.1.
 - A minimum of about 75 - 100 mesh vertices in the streamwise direction.
- General mesh requirements:
 - The mesh should gradually expand.
 - The solution must be mesh independent.
 - The total cell count should be kept at a minimum in order to save computational time.

The conditions stated above were taken into consideration when meshing. In order to do so, all meshes were generated in *MATLAB* using a set of functions, which were written for this purpose². The overall mesh structure is described in the following subsection, but as a parametric study is to be conducted in *Chapter 7 - Parametric analysis*, the shown mesh in this chapter is only for a single plate geometry, namely 40 mm × 2 mm. As the mesh was generated in *MATLAB*, various parameters were used to adjust the plate geometry while maintaining a high mesh quality using the same overall mesh structure as described in this chapter.

²The meshing function can be found on the USB-stick under the path:
../MATLAB/Mesh Generator/Mesh_Generator.m.

4.6.1 2D mesh structure

As the plate is rectangular, symmetry can be used to simplify the meshing. As of such, only a quarter of the domain was meshed, after which it was flipped and merged twice. The following section thus only shows the top right part of the domain. A multi-block meshing approach was used, but triangular elements were used as well. These were utilized to refine the mesh such that the cell count could be kept at a minimum in order to reduce computational time. The overall mesh structure can be seen in Figure 4.7.

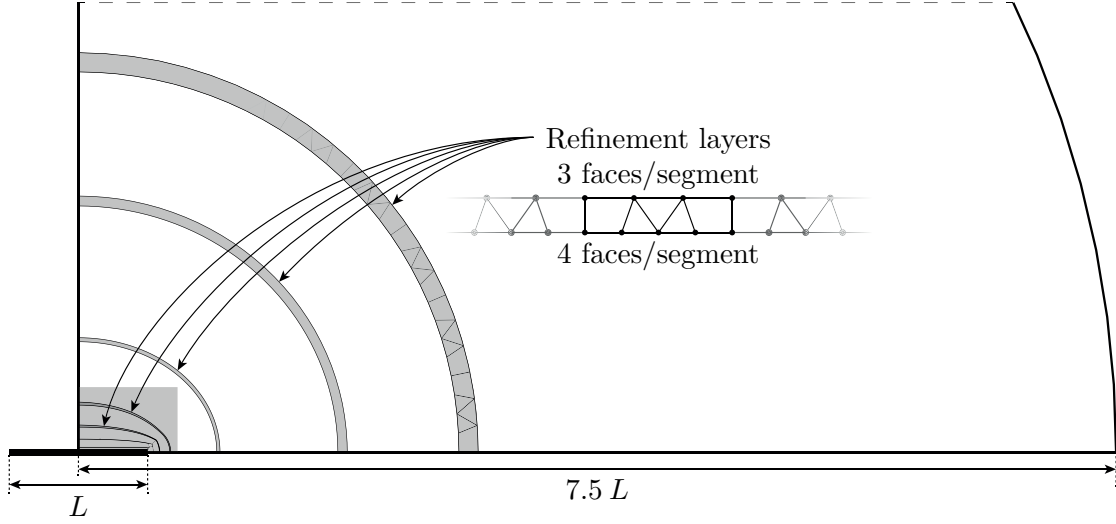


Figure 4.7: The overall mesh structure for a given plate with the different regions outlined. The grey shaded areas are the refinement regions, where four faces are reduced to three faces by means of a single layered unstructured mesh.

The overall mesh structure can be seen in Figure 4.7, where the outlines of the different regions have been shown. This also underlines the use of the multi-block approach, where square block are stretched and merged to fill the domain. The refinement regions have been shown as the shaded grey regions, where the number of faces are reduced by a 4:3-ratio by use of an unstructured mesh consisting of both quadrilateral and triangular cells. This was utilized to reduce the total cell count, as the outer regions do not need to be meshed as fine as the inner regions. In order to quantify the cell quality, the equiangular skewness, ES , and the orthogonal quality, Q , are used, which have been defined in Equation 4.12 [Bakker, 2006b; ANSYS, Inc., 2013c].

$$\begin{aligned}
 ES &= \max \left[\frac{\phi_{\text{Max}} - \phi_e}{\pi - \phi_e}, \frac{\phi_e - \phi_{\text{Min}}}{\phi_e} \right] \\
 Q &= \min \left[\left(\frac{\vec{A}_i \cdot \vec{f}_i}{|\vec{A}_i| \cdot |\vec{f}_i|} \right), \left(\frac{\vec{A}_i \cdot \vec{c}_i}{|\vec{A}_i| \cdot |\vec{c}_i|} \right) \right] \\
 OS &= 1 - Q
 \end{aligned} \tag{4.12}$$

ϕ_{Max} and ϕ_{Min} are the largest and smallest angles of the cell and ϕ_e is the equi-angle for the cell, which is 60 deg and 90 deg for a triangular and a quadrilateral cell respectively. As of such, an equi-angular cell has $ES = 0$, whereas a highly skewed cell has $ES = 1$. A mesh with good quality has $ES \leq 0.50$ [Bakker, 2006b]. The orthogonal quality, Q , relates the area normal vectors, \vec{A} , to the face normal vectors, \vec{f} , of face i and the same for the vectors spanning from two adjacent cell centres, \vec{c} . The minimum value is defined as the orthogonal quality, where a value of 1 is the best quality. In order to visualize the quality, a modified definition has been used defined as $1 - Q$, known as the *Orthoskew*, OS , where

a low value indicate a high quality mesh. The grey square near the plate in Figure 4.7 has been magnified in Figure 4.8, where both ES and OS have been shown.

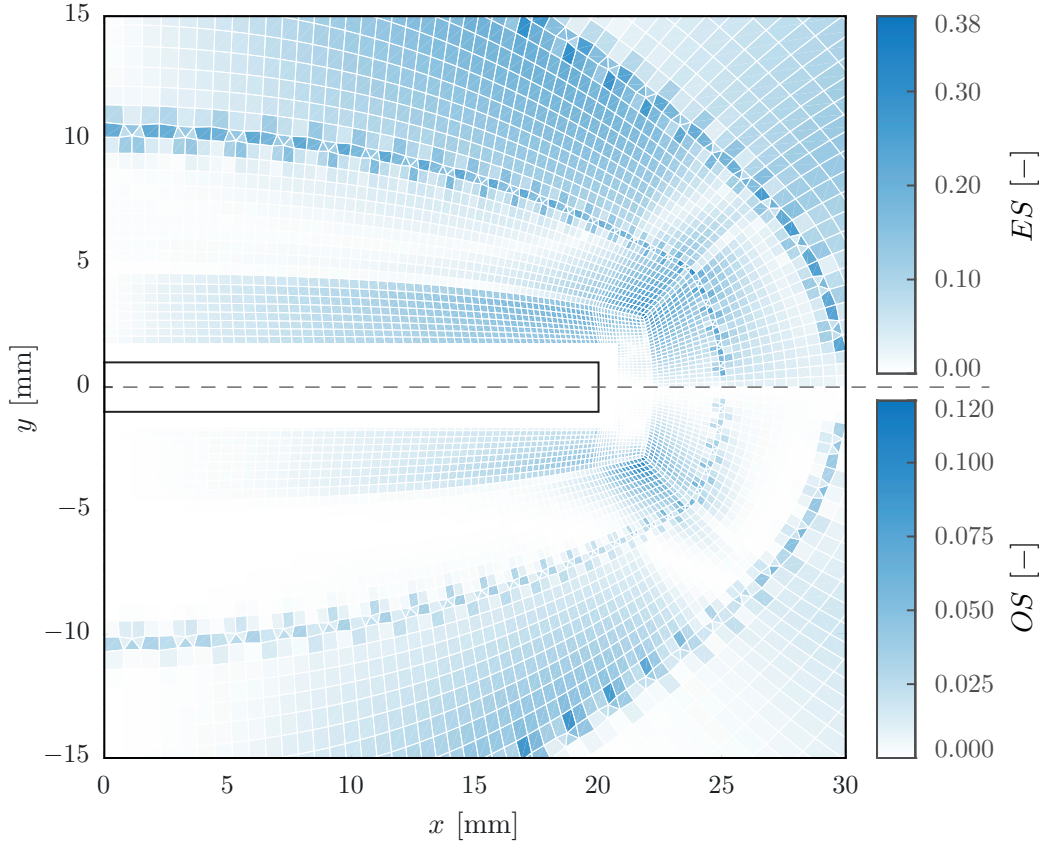


Figure 4.8: A magnified view of the grey square in Figure 4.7. The upper half is coloured by the equiangular skewness, whereas the lower half is coloured by the orthoskew.

A close up view of the mesh near the plate can be seen in Figure 4.8, where the different regions can be seen. The first couple of layers near the plate consist of square cells with $ES = 0$. The first cell centre is located at 0.04 mm, which results in y^+ -values of about 1. The inflation ratio is defined to be 1.1 in this region and the number of vertices along the long edge of the plate is 112 for the given plate of dimensions 40 mm \times 2 mm. These values are all adjusted to follow the mesh requirements for the Transition SST-turbulence model. Both ES and OS can be seen to be low indicating a high quality mesh.

As seen in Figure 4.8, the mesh is gradually expanding in the radial direction. This helps to keep the cell count low, which reduces the required computational time. Furthermore, multiple refinement layers are placed in the radial direction. A total of five refinement layers are placed between the plate and the outer perimeter, which reduces the number of faces to $(3/4)^5 = 24\%$. This reduces the cell count significantly without compromising the accuracy of the solution. A full view of the quarter mesh can be seen in Figure 4.9, where the size of cells at the perimeter can be seen to be quite large compared to the inner cells.

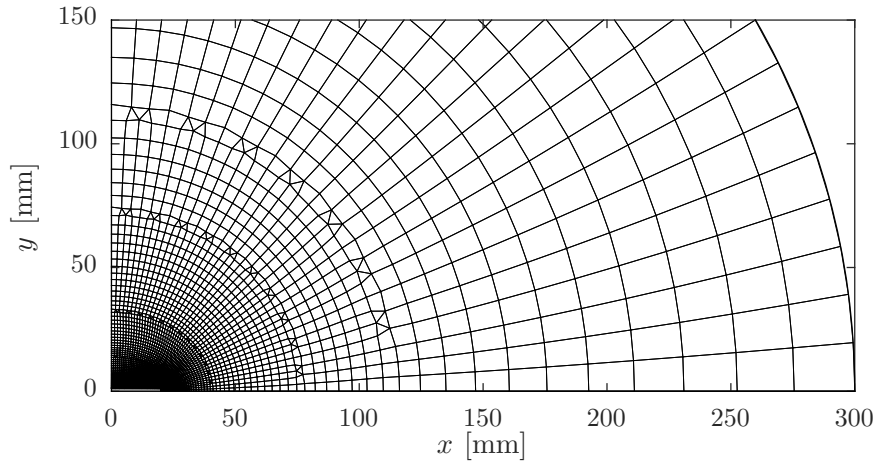


Figure 4.9: A quarter view of the mesh spanning from the origin to the perimeter.

As seen in Figure 4.9, the cells at the perimeter are quite large. However, it was found that the size of these cells did not affect the solution, as the velocity magnitudes within these cells were low, so the reason for including these cells was to have an accurate pressure far-field, which would affect the flow near the plate over time. Many parameters were defining the mesh, but the important ones are listed below:

- Aspect ratio on the long edge at $x = 0$, AR_0 .
- Aspect ratio on the long edge at $x = L/2$, $AR_{L/2}$.
- Aspect ratio on the short edge, AR_h .
- Number of refinement layers, n_{Refine} .

These parameters are the most dominant and will be varied during the mesh independence study in *Section 4.7 - Independence analyses*. As the 2D mesh has been defined, the 3D mesh, which will be used to investigate tip-effects, can be defined. This mesh will be based on the 2D mesh, so additional independence studies do not need to be carried out. The 3D mesh structure has been described in the next subsection.

4.6.2 3D mesh structure

3D domains have been constructed in order to investigate effects, which are not present in the 2D simulations. In order to simplify the process of mesh generation, the 2D meshes were extruded in the z -direction. The process was implemented in *MATLAB* as an extension on the scripts already written. The method has been illustrated in Figure 4.10.

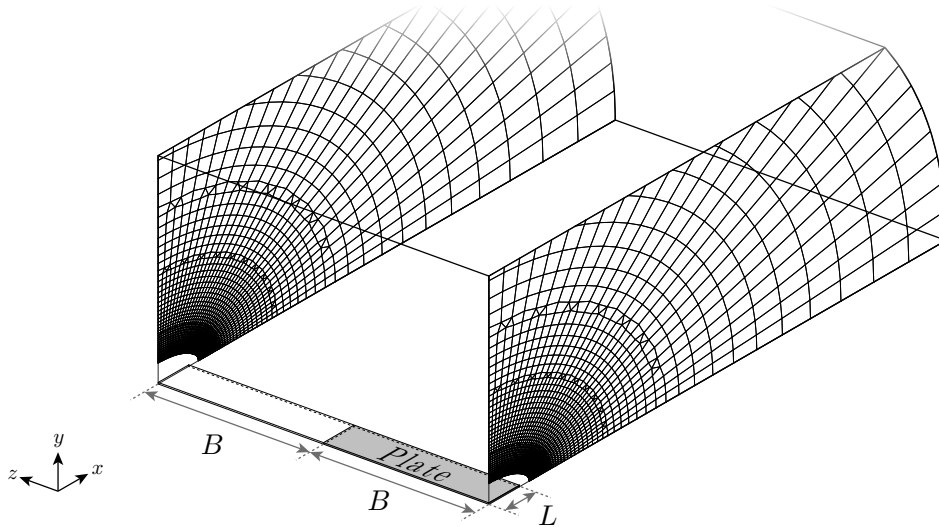


Figure 4.10: An illustration of the extrusion process for generating 3D meshes from existing 2D meshes.

As seen in Figure 4.10, the 2D mesh is extruded from $z = 0$ to $z = 2B$, where B is the length of the plate in the z -direction. Inflation in the z -direction was applied as well in order to keep the cell count down. A typical 3D mesh generated with this approach consisted of $\approx 3 \cdot 10^6$ cells. However, the meshes were not reflected along the x - y -plane, where a symmetry plane was employed in *ANSYS Fluent*, as this would halve the cell count compared to a full 3D mesh. The 2D UDF's were used for the 3D simulations with minor differences, so the plate was only allowed to move in the x - y -plane as with the 2D simulations.

4.6.3 Parallel considerations

The simulations were carried out using multiple computer cores in parallel in order to decrease the computational time. In order to reduce the communication in-between the cores, the number of shared faces should be kept at a minimum [Shang, 2014]. The partitioning of the mesh was assigned in *MATLAB*, where each core was handling a quarter of the symmetric mesh. Furthermore, as the dynamic modelling of the plate was implemented by use of UDF's, these had to be written with parallel considerations. As of such, statements like `#if !RP_HOST` and `#if !RP_NODE` appear in all UDF's, which have been shown in *Enclosure A - ANSYS Fluent UDF's*.

4.7 Independence analyses

As the CFD-model has been defined, mesh- and time-step-independence studies can be carried out. These will serve to ensure independent solutions, which will not change with increasing cell count or smaller time-steps. Both of these studies will be described in the following subsections.

4.7.1 Mesh independence analysis

The mesh independence analysis will be carried out by investigating the sensitivity of the trajectory for different cell counts. As the domain was meshed in *MATLAB*, the various defining parameters could easily be varied, after which simulations with the same conditions could be carried out. The tested meshes are shown in Table 4.1.

Table 4.1: The different meshes used in the mesh independence analysis.

Mesh	AR_0	$AR_{L/2}$	AR_h	n_{Refine}	ES_{Max}	OS_{Max}	n_{Cells}
1 (Very coarse)	10.0	5.0	3.0	4	0.50	0.27	$10.6 \cdot 10^3$
2 (Coarse)	10.0	4.0	2.0	4	0.38	0.16	$14.2 \cdot 10^3$
3 (Medium)	8.0	3.0	2.0	4	0.37	0.11	$16.5 \cdot 10^3$
4 (Fine)	7.0	2.0	1.5	5	0.38	0.12	$19.3 \cdot 10^3$
5 (Extra fine)	5.0	1.5	1.25	5	0.38	0.13	$25.8 \cdot 10^3$
6 (Super fine)	3.0	1.5	1.0	5	0.41	0.15	$37.7 \cdot 10^3$

As seen in Table 4.1, the cell count spans from 10.6 to $37.7 \cdot 10^3$ cells. The simulations were carried out at an initial inclination angle of $\theta = -30$ deg with a time-step of 1.0 ms. The simulations were stopped at 1.50 s, where the resulting trajectories can be seen in Figure 4.11.

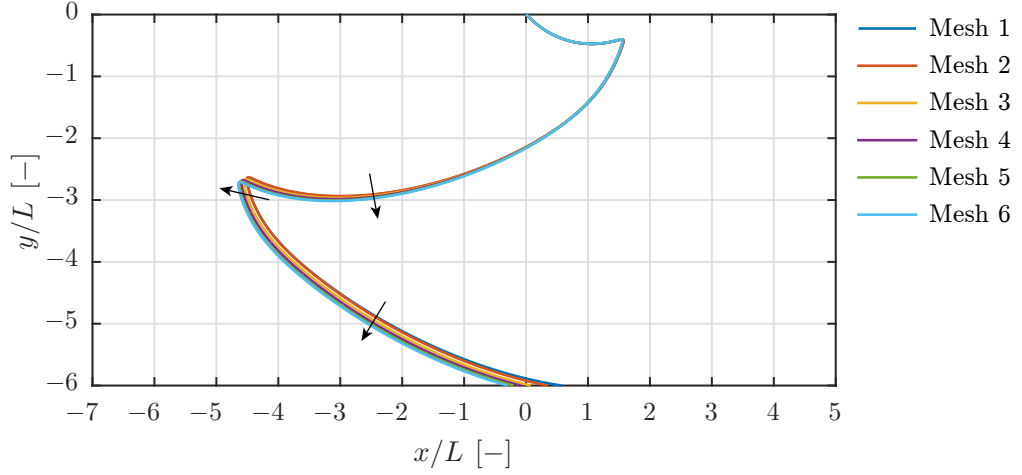


Figure 4.11: The trajectories for the different meshes shown in Table 4.1. The arrows are indicating the direction, at which the trajectory is altered when the cell count increases.

As seen in Figure 4.11, the trajectories obtained with the six different meshes in Table 4.1 are very similar. This indicates that the CFD-model is robust and not very sensitive to the total cell count. Based on the results from the mesh independence analysis, the fine mesh is chosen for further analysis. This mesh has $19.3 \cdot 10^3$ cells, but the parametric analysis will investigate plates with larger dimensions, which will increase the cell count.

Another study was carried out as well, where the domain size radius was varied between 5, 7.5 and 10 times the plate length, but no significant deviation in the trajectory could be seen. As of such, the domain size of $7.5 L$ is chosen for further studies. As a set of mesh independent parameters has been found, meshing of any plate size using these parameters is assumed mesh independent, which will be employed in the parametric analysis described in *Chapter 7 - Parametric analysis*.

4.7.2 Time-step independence analysis

The time-step independence analysis serves to investigate the sensitivity of the trajectory for different time-steps. Five different simulations were carried out, where the time-steps were $\Delta t = 0.5, 1.0, 2.5, 5.0$ and 10 ms. The resulting trajectories from the simulations can be seen in Figure 4.12.

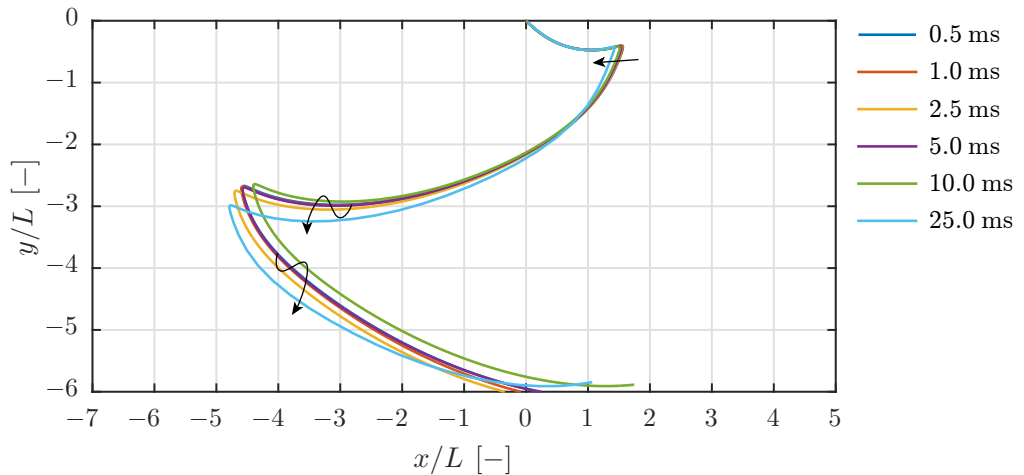


Figure 4.12: The trajectories for six different time-steps. The arrows are indicating the direction, at which the trajectory is altered when the time-step increases.

As seen in Figure 4.12, the obtained trajectories starts to differ from each other at time-steps larger than 2.5 ms. A general approach of finding a suitable time-step is to investigate the *Courant number*, C . For an incompressible fluid, this parameter relates the absolute mass flux in and out of the cell to its volume during a time-step and is defined in Equation 4.13 [ANSYS, Inc., 2013c].

$$C = \frac{\sum_{i=1}^{n_{\text{Faces}}} |\vec{u}_i \cdot \vec{A}_i|}{2 \mathcal{V}} \quad (4.13)$$

n_{Faces} is the number of faces for the cell, \vec{u}_i is the velocity vector on the face, \vec{A}_i is the area normal vector pointing outwards of the cell and \mathcal{V} is the volume of the cell. In order to ensure a stable and accurate solution, it is recommended to satisfy $C \leq 1$, which is known as the *Courant–Friedrichs–Lewy condition*, CFL-condition. However, *ANSYS Fluent* maintains stable at larger values, which was exploited in the model in order to reduce the required computational time. The value of C was monitored during the five time-step independence simulations, where the results can be seen in Figure 4.13.

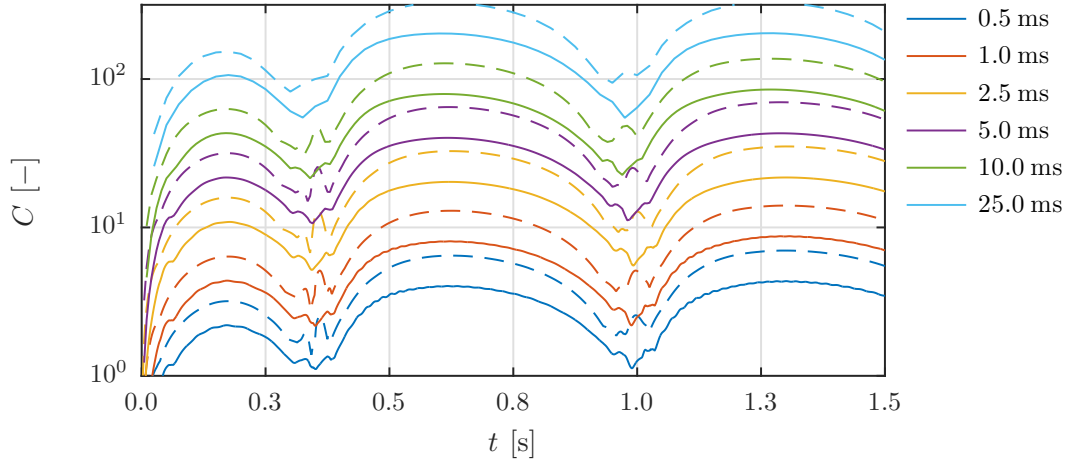


Figure 4.13: The values of C as a function of time. The dashed lines are the maximum cell values of the whole domain, whereas the solid ones are average filtered values.

The values of C can be seen in Figure 4.13 as a function of time for the five different time-step sizes. The dashed lines are the overall maximum values of C , whereas the solid ones are average values of the 25 largest cell values of C after the 50 largest values have been removed. This seems arbitrary, but a few cells will usually have large values of C unless the time-step is very small. The results in Figure 4.13 indicate that even though the values of C might be large, the trajectories of the plate do not change much as seen in Figure 4.12. Figure 4.14 is showing the distribution of C for an averaged simulation.

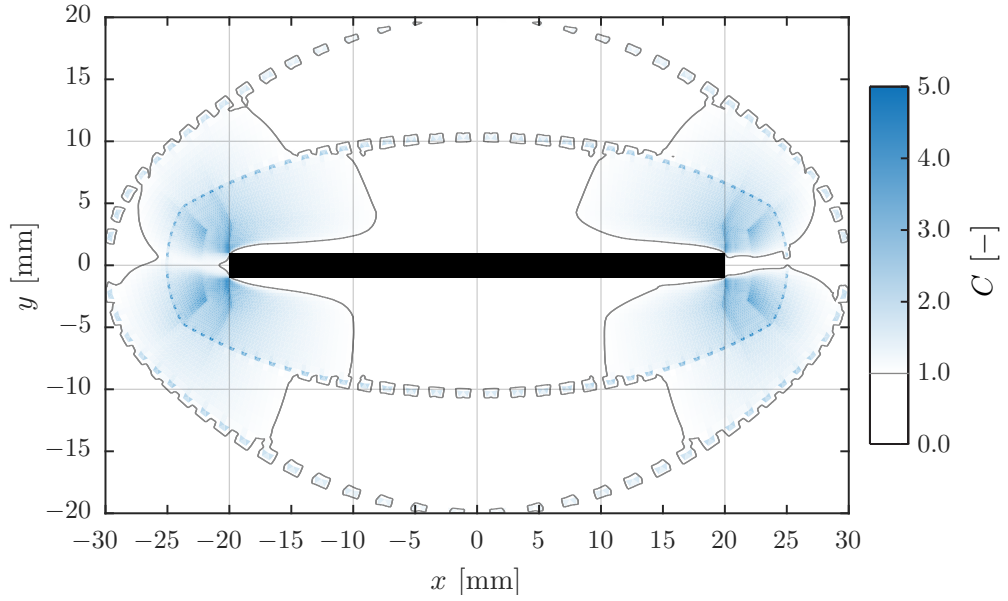


Figure 4.14: The averaged values of C for the mesh near the plate for a simulation spanning 1.2s with $\Delta t = 1.0\text{ms}$.

The averaged values of C for a 1.2 s simulation can be seen in Figure 4.14, where the thin black contour lines are indicating regions with $C = 1.0$. It can be seen that the CFL-condition is satisfied in most cells, but near the edges, where the cell sizes are small and the velocities are large, the condition is not satisfied.

As large values of C have been shown to not affect the trajectories significantly, a time-step of 1.0 ms was chosen for implementation. This time-step corresponds to an average filtered value of $C \approx 5$ as seen in Figure 4.13, which would reduce the required computational time while maintaining accurate solutions. Furthermore, it was found that large time-steps improved stability of the implicit mesh updates for both the in-built *ANSYS Fluent* 6DOF and the modified UDF-implemented algorithms.

4.8 Partial conclusion

This chapter described the CFD-model implemented in *ANSYS Fluent*. Different approaches for modelling a plate in free fall were investigated, where the pros and cons were discussed. An approach where the whole domain is moved over time was chosen for implementation, as this approach did not require re-meshing and could benefit from existing UDF's in *ANSYS Fluent*. The basic theory behind a CFD-simulation was covered along with the Transition SST-model used for turbulence modelling. A dynamic integration procedure was described, where a modified four-point *Adam-Moulton* linear multistep method was used for predicting the acceleration in-between two time-steps, which was implemented in the UDF's.

The meshing of the domain was also described, where the mesh requirements were implemented by use of a set of functions written in *MATLAB* for mesh generation. This allowed for precise control of the cell density throughout the domain, while keeping the total cell count low. Mesh and time-step independence studies were carried out in order to ensure independent results from the model. It was found that the trajectories were not very sensitive to the mesh density, so a fine mesh with $19.3 \cdot 10^3$ cells for a plate with dimensions $40\text{ mm} \times 2\text{ mm}$ was chosen for further studies, while a time-step of 1.0 ms was found sufficient for independent results.

As the CFD-model has been described and implemented in *ANSYS Fluent*, it must be verified by experiments in order to ensure confidence in the results. This will be described in the next chapter.

5 | EXPERIMENTAL ANALYSIS

In order to verify the CFD-model, a set of experiments were conducted. Trajectories for various plates were recorded, when dropped in a basin of water. This chapter will describe the experimental setup and procedure along with the processing of the results.

5.1 Setup and procedure

A sketch of the experimental setup can be seen in Figure 5.1, where both the water basin and the release mechanism have been shown.

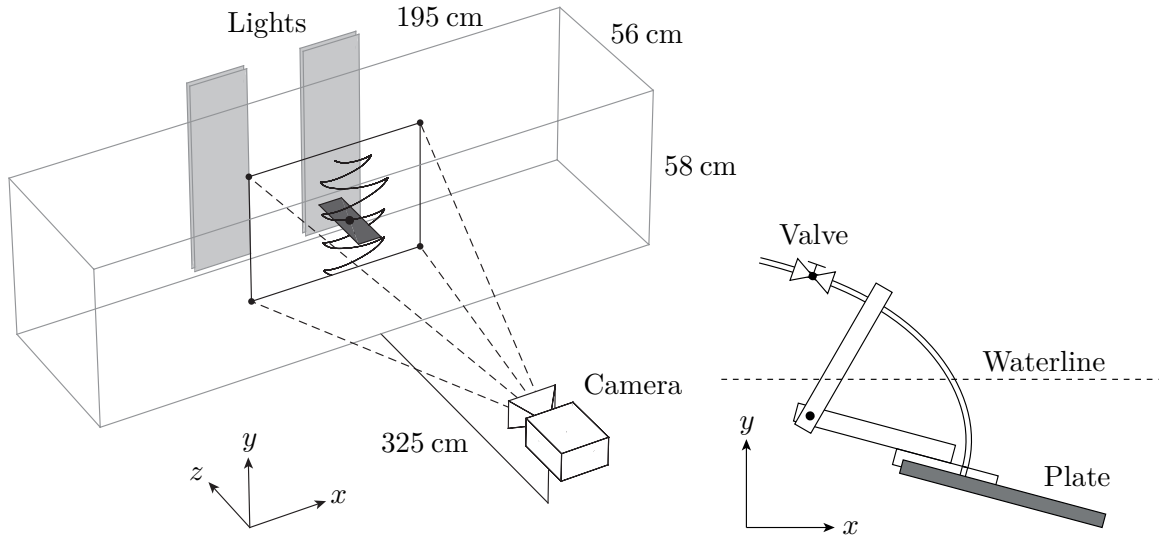


Figure 5.1: A sketch of the experimental setup and the release mechanism.

The basin was ≈ 650 L and made of plexi glass. The depth of 58 cm allowed for a couple of turning points depending on the plate dimensions and density. The plates were mounted on the release mechanism, which can be seen in the right part of Figure 5.1. This mechanism was constructed by a small rod with a hole, where a vacuum tube was mounted on. The dimension of the release mechanism in the z -direction was 1.5 cm, which was only a fraction of the plate depth of 25 cm. In order to avoid interference from the release mechanism, the plates were released by turning the valve between the plate and the vacuum pump, after which the plate fell a few seconds after, when the pressure equalized. It was found that this procedure interfered the least compared to a sudden pressure equalization with the surrounding air.

The experiments were recorded with a camera located 325 cm away from the front of the basin, which allowed for a field of view of $\approx 55 \times 42$ cm. The position of the release mechanism was adjusted so the plates were falling in the x - y -plane. The plates had a large aspect ratio in the x - z -direction, which limited rotation around the x - and y -axes. The distance from the plate ends to the front and back of the basin was ≈ 15 cm, which limited wall effects.

In order to capture the trajectories accurately, a frame-rate of 60 fps was used with an exposure time of 5 ms for most plates. However, one plate was rotating significantly faster compared with the others, where a frame-rate of 95 fps was used with an exposure time of 1.25 ms. Due to the high frame-rates, additional lighting was required, which was provided by two large lamps on the backside of the basin. A white semi-transparent plate was placed

in-between the lamps and the basin in order to diffuse the light, which allowed for an even brightness throughout the image plane. Back-lighting was used in order to enhance the contrast difference between the plate and the background, as this allowed for easier post-processing of the image sequences. Furthermore, the plate edges closest to the camera would be very dim compared to the rest of the plate, which was utilized to obtain the orientation of the plate. The aperture used is listed below:

- Glass basin with dimensions $56 \times 58 \times 195$ cm with a volume of ≈ 650 L.
- Photon Focus MV1 digital video camera.
 - Navitar 25 mm, $f1.4$ lens.
 - Recording at 60/95 fps with an exposure time of 5/1.25 ms depending on the plate.
- Two OSRAM Deluxe EL 15 W main lamps.

Five different plates were tested, where the specifications of these have been listed in Table 5.1, where the masses were measured with a scale with an accuracy of ± 0.1 g. The densities and the moment of inertias were calculated based on the measured mass and the approximated volume.

Table 5.1: The five different plates with specifications. The frame-rate was 60 fps with a field of view of 597×459 mm for plate 1-4, whereas plate 5 was recorded at 95 fps with a field of view of 581×447 mm.

#	$L \times H \times B$ [mm×mm×mm]	ρ [kg/m ³]	m [g]	I [kg·mm ²]	$u_{y, \text{Terminal}}$ [cm/s]	n_{Turns} [–]	Motion classification
1	$40 \times 1 \times 250$	2700.0	27.0	3.60	6.31	16	Periodic
2	$40 \times 2 \times 250$	2725.0	54.5	7.28	16.25	5	Periodic
3	$80 \times 2 \times 250$	2692.5	107.7	57.48	8.85	8	Periodic
4	$80 \times 3 \times 250$	2665.0	159.9	85.40	16.25	4	Periodic
5	$20 \times 4 \times 250$	7770.0	155.4	5.39	32.95	4	Tumbling

As seen in Table 5.1, the plate dimensions span a wide range. Furthermore, both aluminium and steel was used, where the densities were ≈ 2700 and 7770 kg/m^3 respectively. The plates were chosen based on predicted trajectories by use of the model developed by reference [Jensen and Hærvig, 2014]. The motion classifications obtained from this model have been shown in Table 5.1, where the estimated number of turns, n_{Turns} , and the vertical terminal velocities, $u_{y, \text{Terminal}}$, have been shown as well. As stated in the last column of Table 5.1, both periodic and tumbling motion was expected.

Each plate was released and recorded ≈ 10 times in order to obtain statistical confident trajectories¹. The initial angle was constant for each plate, but it was not constant for all plates. Instead, the angle was adjusted to obtain a maximum number of turns, as the data processing would be based on the turns thus yielding more statistical confident trajectories.

5.2 Data processing

A total of 58 trajectories were recorded, where some were discarded due to skewed trajectories, which differed significantly from the x - y -plane. The recordings were made in the .tiff-format, which were imported into *MATLAB* for processed, which has been described in this section.

¹Animated sequences for all plates can be found on the USB-stick under the path: `../Experiments/Experiment_Animations.avi`.

5.2.1 Filtering

The trajectories and orientations of the plates were extracted from the recorded images by use of various filters. The basic procedure has been shown in Figure 5.2.

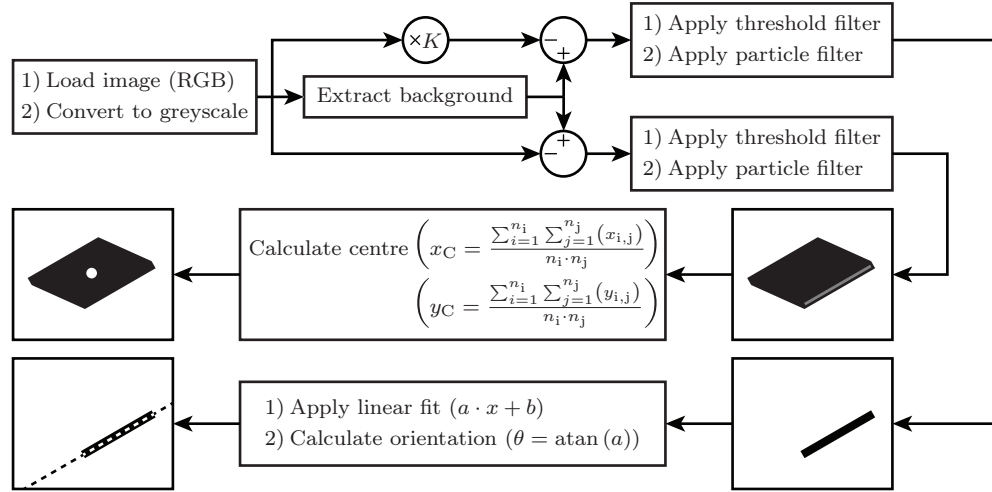


Figure 5.2: The filtering process shown as a block diagram. Various in-built functions in *MATLAB* were used to simplify the process.

The multiplication factor K was used to highlight the edges of the plates, which was tuned based on the dynamics of the images. An example of a processed trajectory can be seen in Figure 5.3, where additional information has been overlaid.

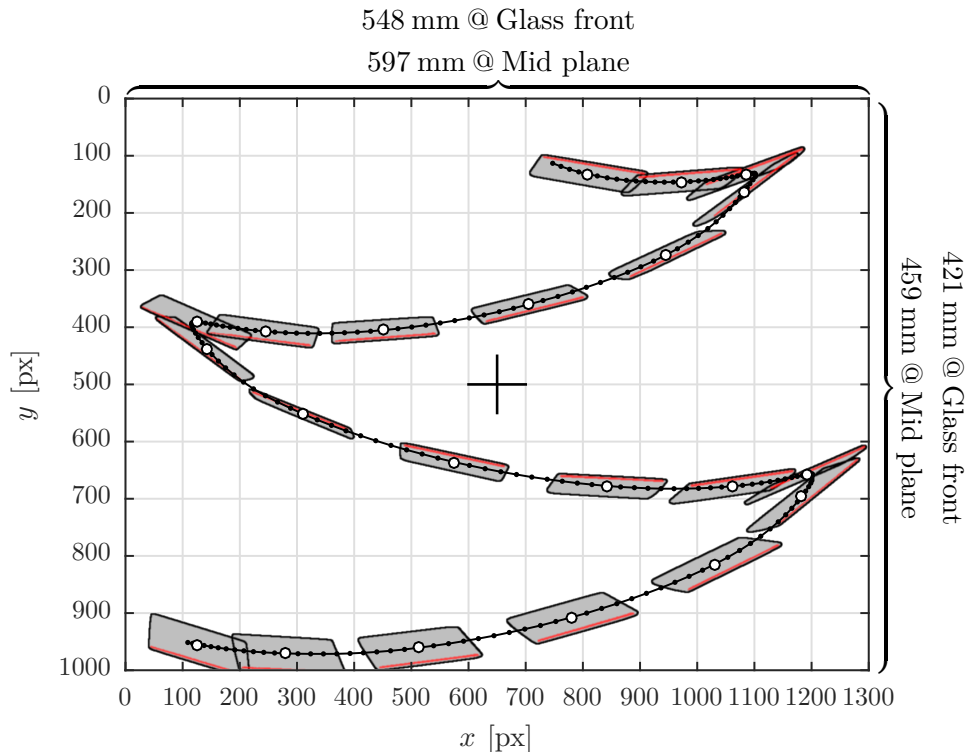


Figure 5.3: An example of a processed trajectory for the third plate with dimensions $80 \text{ mm} \times 2 \text{ mm} \times 250 \text{ mm}$. This white dots correspond to the mid plane, where the field of view was $597 \text{ mm} \times 459 \text{ mm}$.

Figure 5.3 is showing a single trajectory from the third plate with dimensions $80 \text{ mm} \times 2 \text{ mm} \times 250 \text{ mm}$ at different times. The small black dots are showing the location of the plate at all time-steps, whereas the white circles are only showing every tenth time-step.

The grey areas are showing the convex image after the particle filter used for plate detection, where these have been bounded by the black edge lines. The red lines are showing the front end of the plate closest to the camera, which was detected by another filter. This was possible as the basin was back lit, so the edges could be identified precisely as these were more dim compared to the plate itself.

As seen in Figure 5.3, the edge itself cannot be used to correctly predict the location of the plate, as the perspective and refractive warp of the image causes incorrect measurements. As of such, the centre of the plates were found by averaging all pixel values bounded by the plate, which has been shown by the white and black dots. The orientation of the plates could not be extracted from the bounded area, so the edges were used instead. The orientation angle with respect to the x -axis was found by applying a linear fit to the raw pixel values, where the gradient of this fit could be related to the plate orientation by trigonometric functions.

The centre of the plates were located in-between the front and back plane of the basin. This causes refractive warp, as the refractive index of water is larger compared with air. Furthermore, the camera was located 3.25 m from the front of the basin, so perspective effects was causing a slight warp as well. These effects might not alter the obtained trajectories significantly, but they have been back traced in order to obtain more accurate results. However, the mathematics of this procedure has not been included in this study. After the pixel values had been converted to world coordinates, the resulting trajectories could be statistically processed, which has been described in the next section.

5.3 Results

As seen in Table 5.1, five different plates were tested. These were chosen based on the PT-model proposed by [Jensen and Hærvig, 2014], where the motion was predicted to be both periodic and tumbling. This held true in the experiments, but due to the nature of the motion, two different processing methods were used on the results. This has been explained in the following subsections, where plates with periodic motion are processed first, after which the fifth plate with tumbling motion is processed.

5.3.1 Plate 1-4

Each of the plates were dropped in the basin about 10 times, where the trajectories and orientations were obtained by the methods previously described. All figures in this section are based on the third plate, but similar figures and results for all plates can be found in *Enclosure B - Experimental results*. The processed trajectories and orientations for the third plate with dimensions 80 mm \times 2 mm \times 250 mm have been shows in Figure 5.4.

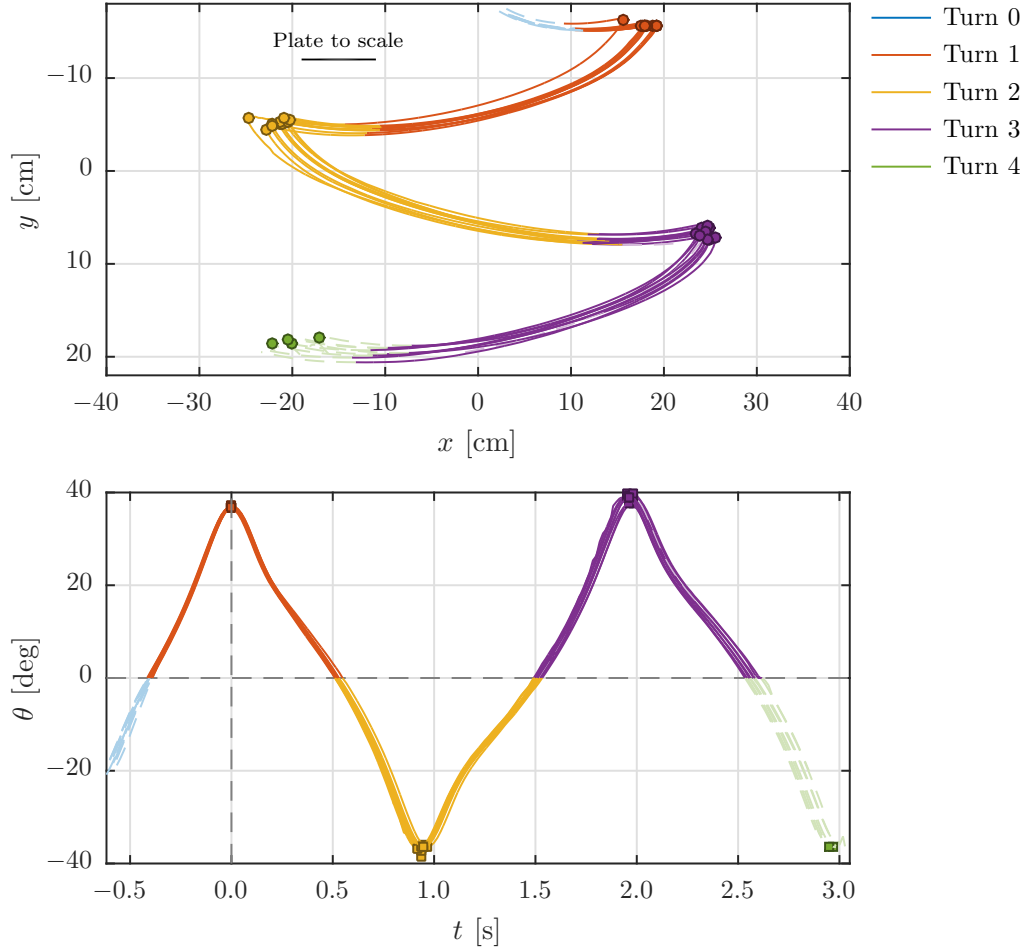


Figure 5.4: The processed trajectories and orientations for the third plate in Table 5.1. The colours are indicating the different turns.

Figure 5.4 is showing the processed trajectories and orientations in the upper and lower part respectively. Each turn is enclosed by an orientation angle of 0 deg, which can be seen in the lower part of Figure 5.4. The dots are indicating the local extrema in x , where the colours are indicating each turn. As seen in the figure, the green dots are included, though these are located in the dashed regions, as the raw data allowed an estimation of the local extrema at this depth despite the fact that the fourth turn was not complete.

The lower figure is showing the orientation angle as a function of local time, where the data sets have been adjusted in time to peak at $t = 0$ s indicated by the vertical dashed line. The orientation values have not been stretched nor adjusted. The squares in this figure are indicating the local extrema in θ .

The reason for splitting the data into turns is that not all trajectories were as close as the shown trajectories in Figure 5.4. As of such, the statistical calculations would yield a large confidence interval in the mean, which was caused by the sensitivity of the trajectories with respect to the initial conditions. It was found, especially for plates with small L -dimensions, that the trajectories would differ over time, as small disturbances would accumulate over time and alter the trajectories significantly.

Statistical processing

As the trajectories have been split into turns enclosed by orientation angles of 0 deg, the sub-trajectories can be overlaid and adjusted. This has been shown in Figure 5.5.

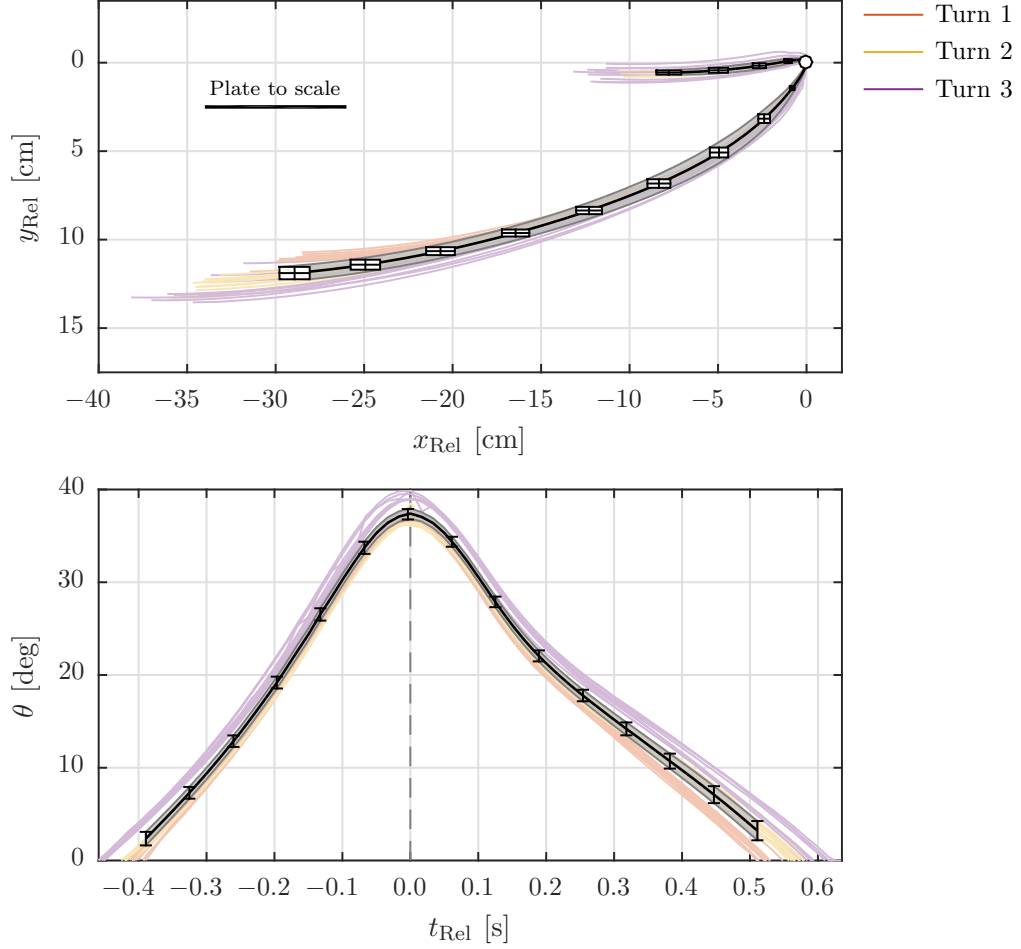


Figure 5.5: The processed trajectories and orientations for the third plate in Table 5.1. The colours are indicating the different turns.

Figure 5.5 is showing turn 1, 2 and 3 overlaid on each other. The turns have been adjusted to the local extrema in x , where every second turn has been flipped along the y -axis. The coincidence point at $(0,0)$ has been indicated by the white dot, and x_{Rel} and y_{Rel} are thus showing the local coordinate relative to this point. The different turns have been shown as the light coloured lines, where turn 0 and 4 have been omitted. This has been done as these turns were not complete, which is clear to see in Figure 5.4 as the dashed coloured lines. The lower part of Figure 5.5 is showing the orientation angle for each turn processed in the same way as the upper figure. These have been adjusted in time to peak at 0 s in relative time, t_{Rel} , and every second turn is flipped along the t -axis.

As all turns were processed to coincide at specified points, statistical methods could be applied to the data in order to estimate the mean trajectories and orientation angles. This was done by calculating the sample mean and the 95 % confidence interval. As the number of turns multiplied by the number of data sets was large, $n \geq 30$, the normal distribution was applied. These values have been shown in Equation 5.1 for a given variable, x .

$$\begin{aligned}
 \bar{x} - z_{\alpha/2} \frac{s}{\sqrt{n}} &< \bar{x} < \bar{x} + z_{\alpha/2} \frac{s}{\sqrt{n}} \\
 \bar{x} &= \frac{1}{n} \sum_{i=1}^n (x_i) \\
 s^2 &= \frac{1}{n} \sum_{i=1}^n (\bar{x} - x_i)^2
 \end{aligned} \tag{5.1}$$

s is the standard deviation and s^2 is the variance. The critical values, $z_{\alpha/2}$, where $\alpha/2 = 0.025$, were interpolated as a function of the number of degrees of freedom, n , for each data set, which, for instance, was 31 in Figure 5.5. The sample mean and the confidence interval will be presented in the format shown in Equation 5.2, where the mean is $\bar{x} = 1.0$ and the margin of error for \bar{x} is $E_{\bar{x}} = 0.1$. It should be noted that \bar{x} is an arbitrary variable, which only serves to define the representation of the results.

$$(\bar{x} = 1.0) \pm (0.1 = E_{\bar{x}}) \quad (5.2)$$

The 95 % confidence interval for the processed trajectory and orientation for the third plate can be seen as the shaded areas in Figure 5.5, where the mean is indicated by the solid black lines in the middle of the areas. The boxes in the upper part of this figure is indicating the confidence intervals in both directions, which is enclosed by the shaded area. In the lower figure, showing the orientation angle, the vertical black lines are indicating the confidence intervals, which are also enclosed by the shaded area. It can be seen that the confidence interval is not calculated to the limits of the data sets, as some trajectories were shorter in time compared to others.

Δ -results

The dots in Figure 5.4 indicating the local extrema in x for each turn were used to estimate the time period and the size of a single averaged turn, whereas the squares in the lower part of Figure 5.4 were used to estimate the magnitude of the orientation change. Both the dots and squares were used to compute the statistic means of the time per turn. This definition has been clarified in Figure 5.6.

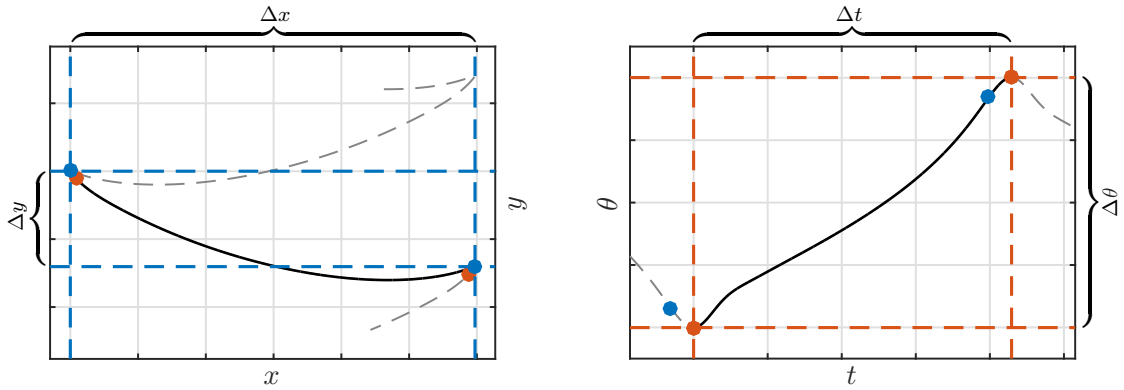


Figure 5.6: The method of calculating the Δ -values for a given trajectory. The blue dots relate to the extrema in the position, whereas the red ones are related to the extrema in the orientation.

As seen in Figure 5.6, the Δ -values are extracted between two turning points. It can also be seen that the extrema in the orientation are shifted in time relative to the extrema in the position. Once again, the means and the 95 % confidence intervals were computed according to Equation 5.1, where the results have been shown in Equation 5.3 for the third plate. These results which will be denoted as the Δ -results.

$$\begin{aligned} &(\Delta\bar{x} = 431.3 \text{ mm}) \pm (19.17 \text{ mm} = E_{\Delta\bar{x}}) \\ &(\Delta\bar{y} = 112.4 \text{ mm}) \pm (4.97 \text{ mm} = E_{\Delta\bar{y}}) \\ &(\Delta\bar{\theta} = 74.6 \text{ deg}) \pm (0.79 \text{ deg} = E_{\Delta\bar{\theta}}) \\ &(\Delta\bar{t} = 984 \text{ ms}) \pm (16.8 \text{ ms} = E_{\Delta\bar{t}}) \end{aligned} \quad (5.3)$$

As seen in Equation 5.3, the results are in general indicating small confidence interval, but sources of errors can cause disturbances, which can affect the results. These are discussed in *Section 5.4 - Discussion and sources of errors*.

5.3.2 Plate 5

Due to the tumbling motion of the fifth plate with dimensions $20 \text{ mm} \times 4 \text{ mm} \times 250 \text{ mm}$ and density 7770 kg/m^3 , another approach was used to process this data set. The filtering method previously described used for identifying the centre of the plate along with edge detection for orientation calculation was used for this plate as well. A filtered sample sequence can be seen in Figure 5.7.

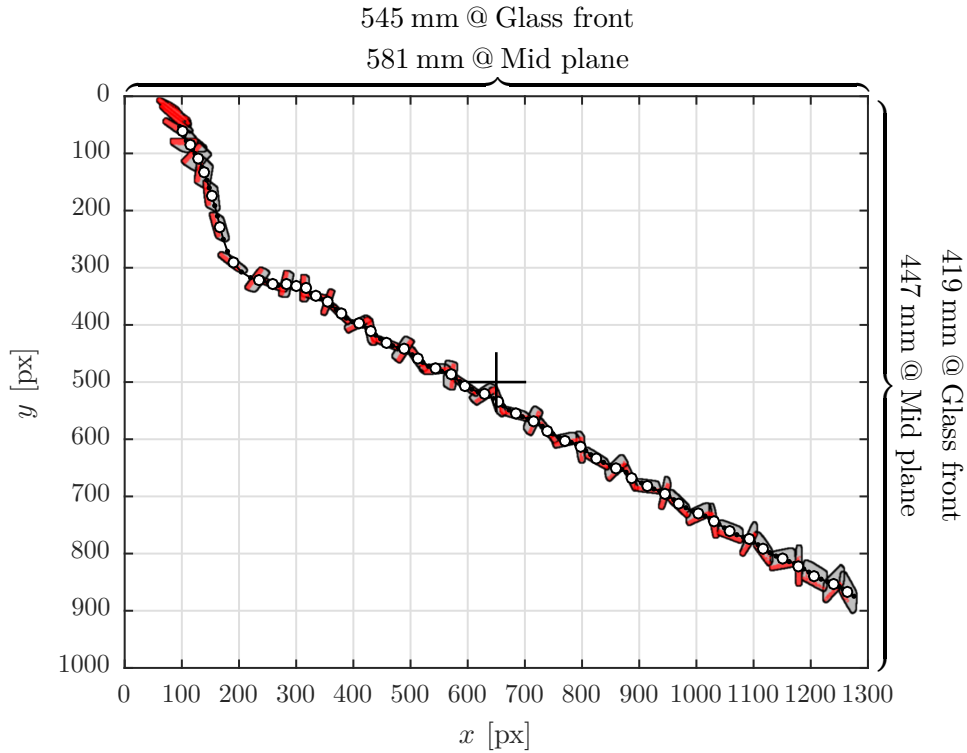


Figure 5.7: An example of a processed trajectory for the fifth plate with dimensions $20 \text{ mm} \times 4 \text{ mm} \times 250 \text{ mm}$. The white dots correspond to the mid plane, where the field of view was $581 \text{ mm} \times 447 \text{ mm}$.

As seen in Figure 5.7, the initial motion is arbitrary and chaotic, which was the case for all of the image sequences. Some of them exited the image plane before returning, but all of them entered a repeating tumbling motion, which is clear to see in the figure at $x \gtrsim 350 \text{ px}$. The processed data after the transformation from pixel values to world coordinates can be seen in Figure 5.8.

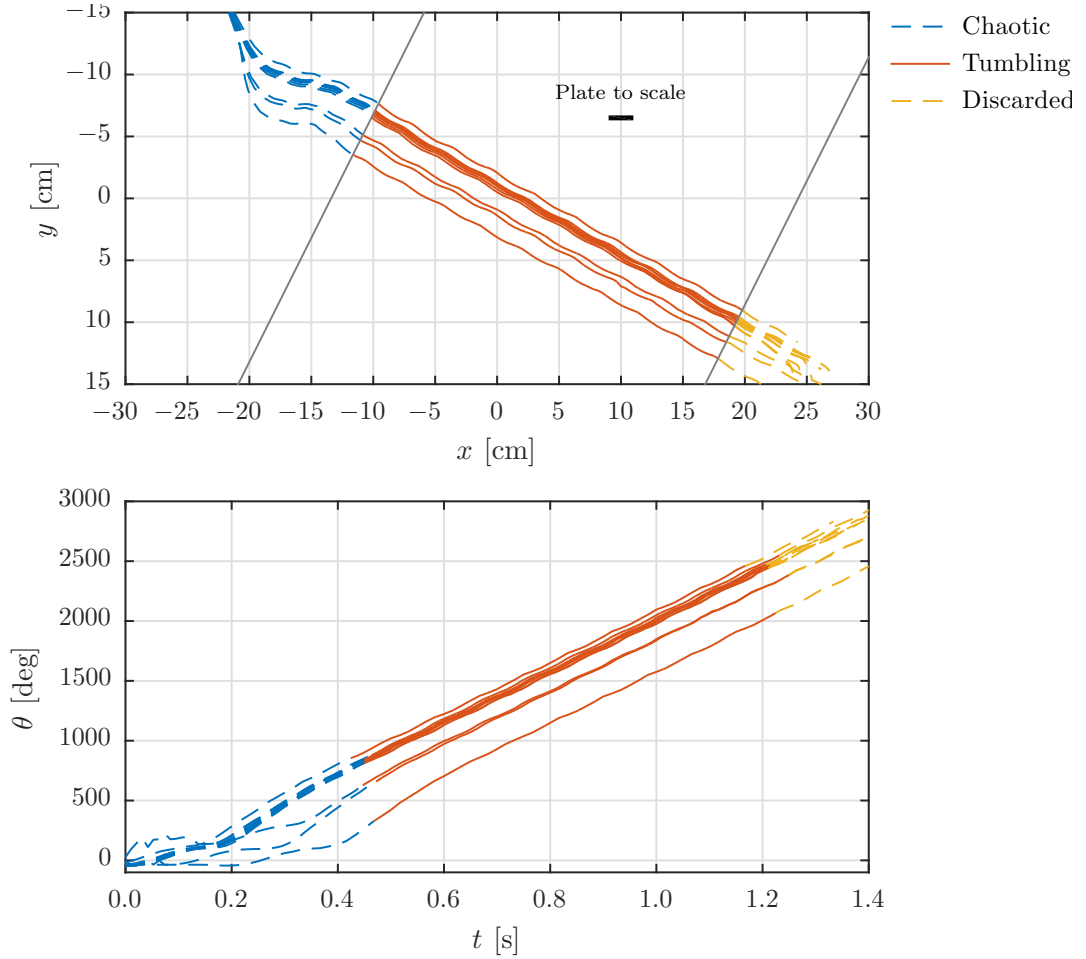


Figure 5.8: The processed trajectories and orientations for the fifth plate in Table 5.1. The colours are indicating the chaotic, tumbling and discarded regions.

Figure 5.8 is showing the trajectories and orientations for the fifth plate in the upper and lower part respectively. The upper figure is divided in three coloured regions, where the blue is the chaotic region, the red is the tumbling regions and the yellow region is the region close to the bottom of the basin. Due to the higher frame-rate of 95 fps and thus lower exposure time, the edges were not as easy to detect, as they were for the first four experiments. As of such, the yellow region was partly manually traced, which led to inaccurate measurements, so the yellow region was discarded for further processing. The lower part of Figure 5.8 is showing the orientation of the plate as a function of time, where the chaotic blue region is apparent.

In the tumbling region, the data sets are almost linear, so first order fits were applied in order to obtain the gradients. The gradients of interest were $\partial\bar{x}/\partial\bar{y}$, $\partial\bar{x}/\partial\bar{t}$, $\partial\bar{y}/\partial\bar{t}$ and $\partial\bar{\theta}/\partial\bar{t}$. $\partial\bar{x}/\partial\bar{y}$ was included as a variable of interest, though it can be derived directly from the other derivatives. The gradients were obtained for 11 sequences, after which the confidence intervals for the means were calculated. These results have been shown in Equation 5.4.

$$\begin{aligned}
 \left(\partial\bar{y}/\partial\bar{x} = 0.563\right) &\pm \left(0.013 = E_{\partial\bar{y}/\partial\bar{x}}\right) \\
 \left(\partial\bar{x}/\partial\bar{t} = 39.1 \text{ cm/s}\right) &\pm \left(0.63 \text{ cm/s} = E_{\partial\bar{x}/\partial\bar{t}}\right) \\
 \left(\partial\bar{y}/\partial\bar{t} = 22.0 \text{ cm/s}\right) &\pm \left(0.48 \text{ cm/s} = E_{\partial\bar{y}/\partial\bar{t}}\right) \\
 \left(\partial\bar{\theta}/\partial\bar{t} = 2159.4 \text{ deg/s}\right) &\pm \left(27.46 \text{ deg/s} = E_{\partial\bar{\theta}/\partial\bar{t}}\right)
 \end{aligned} \tag{5.4}$$

The motion itself between each turn could be analysed, but this was not done as it was almost linear as seen in Figure 5.8.

5.4 Discussion and sources of errors

This section serves to clarify some of the sources of errors, which might have affected the results.

Camera adjustments

The camera was located 3.25 m from the front of the basin, where it was aligned to face the basin perpendicular to its image centre. However, this adjustment was difficult to obtain, which could have affected the results. If this was not aligned, the image would have been warped, which would have affected the transformation map from pixel units to world units. However, this would have only slightly altered the trajectories.

The intrinsic matrix for the camera holds information about the pixel scaling, pixel aspect ratio, skewness and lens distortion [OpenCV, 2015]. It was assumed that the lens was of such a quality, that warping effects could be ignored. In order to compensate for such errors, the intrinsic matrix could have been found by use of various techniques. However, once again, such errors would have only slightly altered the results.

Plate inconsistencies

The plates were made by cutting sheet metal in the correct dimensions. However, during such a process, the plates might be bended, which would have altered the motion. Furthermore, the thickness of the sheet metal might not have been completely consistent, which would have altered the centre of gravity, but assuming an even inconsistency, the centre of gravity would have been placed at the midpoint of the plate.

The plates were weighted and the densities were calculated based on the dimensions of the plates. However, the volumes were not found experimentally, which might have differed from the calculated ones. This would not have changes the trajectories, but as the experiments serve to validate the CFD-model, such effects can alter the computed trajectories compared with the experimental ones.

Drifting

The plates were all 250 mm wide in the z -direction, which was chosen to minimize rotation about the x - and y -axes, yet avoiding wall effects. This proved to hold true for most plates, but the trajectories were still not completely in the x - y -plane. This effects can be traced back to the transformation maps, which would have yielded the correct position if the z -location was known. However, the z -coordinate was assumed being in the centre of the basin, so this might have caused incorrect measurements.

3D effects

The aspect ratios in the L - B -direction for all plates spanned between 3.125 and 12.5. As this ratio increases, 3D effects will not affect the trajectories as much, as a 2D simulation implicitly has an aspect ration of ∞ . However, the aspect ratio will affect the trajectories due to flow leakage at the tips, where the pressure difference on the upper and lower side of the plates will cause vortices to be formed, which will impact the forces and torques acting on the plate. This effect will be investigated in *Chapter 6 - Verification*, where 3DOF 3D CFD-simulations will be conducted.

5.5 Partial conclusion

This chapter covered the experimental analysis, where the trajectories and orientations of five plates were investigated. The plates varied in dimensions and densities, which resulted in both periodic and tumbling motion. Each plate was dropped multiple times and recorded, where each recording was processed in order to estimate the trajectory and orientation. This procedure was described, where two different filters were used to locate the centre of the plate and another for estimating the orientation of the plate. Transformation maps were utilized to relate pixel values to world coordinates, where both perspective and refractive warps were included.

The resulting trajectories were investigated for each plate, where these were split at each turn. This resulted in estimation of a mean path for each turn, where statistical methods were applied in order to find the 95 % confidence interval of the means. The statistic methods were also applied to the mean time, mean width and height and mean orientation difference for each turn, where these results were denoted the Δ -results.

The fifth plate experienced tumbling motion, where another processing approach was used. First order linear fits were used to calculate the gradients in both the horizontal motion, vertical motion and orientation as a function of time along with gradients between the vertical and horizontal motions. The confidence interval for the mean in these variables were found as well, which will be used to verify the CFD-model. This verification will be described in the next chapter.

6 | VERIFICATION

This chapter serves to verify the 2D CFD-model described in *Chapter 4 - CFD-model* by comparing the experimental results with simulations carried out under the same conditions as the experiments. The experimental results were processed with statistical methods, where the 95 % confidence intervals for the means were found for multiple variables. The mean trajectories were also found for each turn, which will be compared with the CFD-results.

The setup of the CFD-model will be described, after which the results from these simulations are compared with the experimental results. A discussion of this comparison will be made, which will lead to an investigation of tip effects. These effects will be investigated by carrying out 3DOF 3D CFD-simulations, where tip effects are included. A discussion of these results will follow, where the main differences between the 2D- and 3D-simulations will be investigated.

6.1 CFD and experimental comparison

This section will describe the setup of the 2D CFD-simulations, after which a comparison with the experimental results will be made.

6.1.1 CFD-setup

The 2D CFD-model described in *Chapter 4 - CFD-model* is used to estimate the trajectories and orientations of the five plates. As the experimental results are processed to yield mean results with 95 % confidence intervals, the results from the 2D CFD-simulations should be carried out to $t \rightarrow \infty$. This would yield periodic motion, where each turn would ideally be symmetric. As the phenomena is unsteady, the simulations will be simulated for 10 s in order to approximate $t \rightarrow \infty$.

The CFD-simulations were carried out in 2D, so the mass and moment of inertia should be converted to values per unit depth. This has been shown in Table 6.1 along with the cell count in the CFD-meshes.

Table 6.1: The five different plates with measured specifications along with the specifications per unit depth and the number of cells, n_{Cells} , in the CFD-model.

#	$L \times H \times B$ [mm]	ρ [kg/m ³]	m [g]	I [kg·mm ²]	m/B [g/m]	I/B [kg·mm ² /m]	n_{Cells} [cells]
1	$40 \times 1 \times 250$	2700.0	27.0	3.60	108.0	14.40	$18.27 \cdot 10^3$
2	$40 \times 2 \times 250$	2725.0	54.5	7.28	218.0	29.12	$19.32 \cdot 10^3$
3	$80 \times 2 \times 250$	2692.5	107.7	57.48	430.8	229.92	$34.74 \cdot 10^3$
4	$80 \times 3 \times 250$	2665.0	159.9	85.40	539.6	341.60	$36.62 \cdot 10^3$
5	$20 \times 4 \times 250$	7770.0	155.4	5.39	621.6	21.56	$19.28 \cdot 10^3$

As seen in Table 6.1, the values of m and I have been converted to values per unit depth shown as (m/B) and (I/B) . These values were used in the 2D CFD-simulations, where the cell counts spanned between 18.27 to $36.62 \cdot 10^3$ cells.

As previously described, each simulation would be stopped after 10s, where periodic motion had been achieved. The extracted results used for the comparison with the experimental results were found from the last complete turn.

6.1.2 Trajectory comparison

A comparison between the CFD-simulations and the experiments can be seen in Figure 6.1 on the next page, where the rows are indicating the first four plates in ascend order from the top. The left column is showing the trajectories for a single turn, where the coordinates are relative to the extrema in x , which have been adjusted to $(0, 0)$ indicated by the white dots. The grey shaded regions are the 95 % confidence intervals, where the solid black line in the middle of the shaded regions are the means. The blue lines are the CFD-results for a single turn, which has been clipped at an orientation angle of 0deg as with the experimental results. As described in *Chapter 5 - Experimental analysis*, the confidence intervals could not be computed throughout the turns, as they were limited to the fastest turn. As of such, only a portion of the turn has been shown for the experimental results, where the corresponding CFD-trajectory has been indicated by the dark blue lines. The light blue lines are thus indicating the remaining of each turn. It should be noted that the transition from dark to light blue is based on the time-data from the experimental results.

The right column is showing the orientation angle as a function of relative time, t_{Rel} , where both the experimental and CFD-simulation results are adjusted to peak at $t_{\text{Rel}} = 0$ s. Once again, the experimental results were clipped due to inconsistency in duration of each turn from the experiments, which has been applied to the CFD-results as well indicated as the dark and light blue lines. It is more clear to see how the transition from dark to light is based on the time of the experimental results, as these figures are indicating time on the x-axis. The trajectories are deviating slightly from each other, where all CFD-results are yielding faster descend velocities as all CFD-results are below the experimental results at the end of each turn.

The orientation angles are deviating slightly from each other as well, where all CFD-results are yielding larger peak angles compared with the experimental results. Furthermore, at $t_{\text{Rel}} > 0$, the values of θ are not decreasing as rapidly for the CFD-results compared with the experimental values. At $t_{\text{Rel}} \approx 0.1$ to 0.2 s, the orientation angle starts to alter the descend gradient, which happens at the approximate same time for both the experimental and CFD-simulation results, but this deviation is difficult to quantify.

6.1.3 Δ -results comparison

The Δ -results are compared, where the definition of these were shown in Figure 5.6 in *Section 5.3 - Results* on page 39. The CFD-simulations were processed in a similar way to the experimental results. It was found that the obtained trajectories were almost identical and periodic for each turn, so no statistics was required to process these results further. This is apparent in Figure 6.2, where the complete 10 s trajectories for plate 1-4 have been shown.

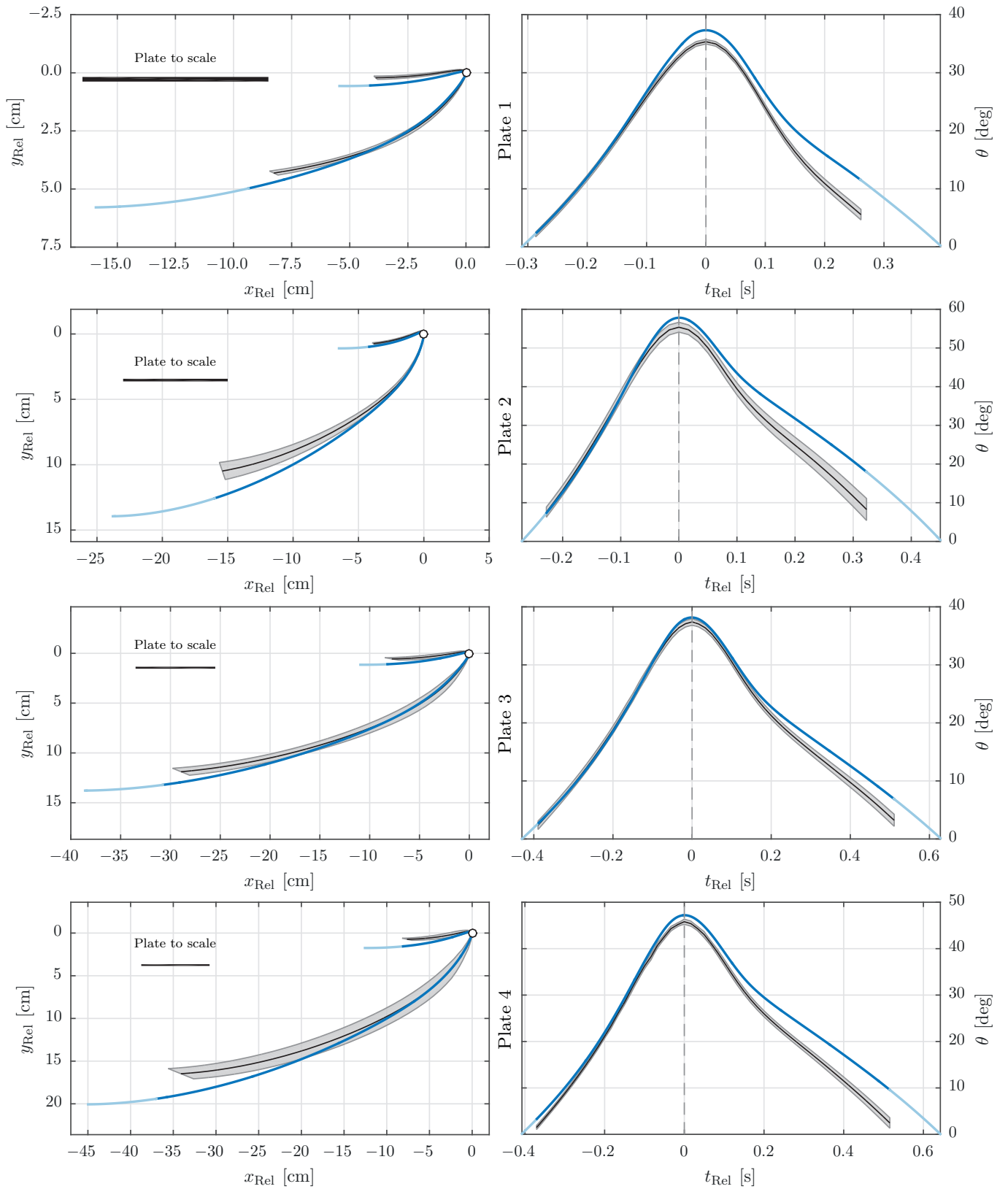


Figure 6.1: The results from the experiments and the CFD-simulations in grey and blue respectively. The grey shaded regions indicate the 95% confidence intervals of the means.

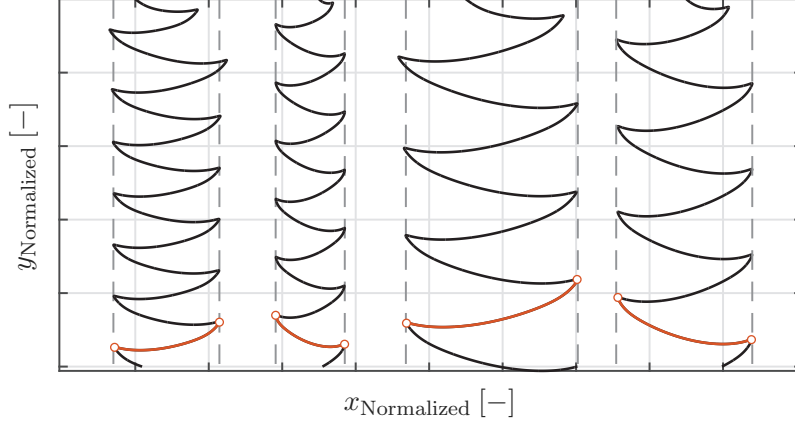


Figure 6.2: The trajectories for plate 1-4 in solid lines. The dashed lines are indicating the extrema in x , where periodic motion is apparent after the first couple of turns for all plates. The axes are normalized, so the trajectories are not to scale relative to each other.

As seen in Figure 6.2, periodic motion can be observed for all plates after the first couple of turns. As of such, the Δ -results could be obtained directly from the trajectories by using the last full period, which has been shown as the red lines. The Δ -results for the CFD-analyses, along with the experimental Δ -results, have been shown in Table 6.2 for all plates, where the deviation between the results relative to the means have been included as well. For instance, the deviation of Δx for plate 1 of 27.5 % has been calculated as $(214.0 - 162.3) / ((214.0 + 162.3) / 2)$. For clarification, the Δ -results are obtained by statistical processing an average turn yielding the mean values for a single period.

Table 6.2: The Δ -results from both the experiments and the CFD-simulations. The deviations are relative to the mean values and are between the 2D CFD-simulations and the experiments. The coloured cells are indicating whether the 2D or 3D simulations are closest to the experimental values, where green being closest.

	Unit	Plate 1	Plate 2	Plate 3	Plate 4	Average
$\Delta \bar{x}_{\text{Exp}}$	mm	162.3	251.1	431.3	470.3	—
Δx_{CFD}	mm	214.0	303.1	498.6	577.5	—
$\Delta x_{\text{CFD},3\text{D}}$	mm	216.4	—	435.8	—	—
Deviation	%	27.5	18.8	14.5	20.5	20.33
$\Delta \bar{y}_{\text{Exp}}$	mm	45.0	99.3	112.4	152.4	—
Δy_{CFD}	mm	51.9	127.2	127.0	183.7	—
$\Delta y_{\text{CFD},3\text{D}}$	mm	60.6	—	124.9	—	—
Deviation	%	14.2	24.6	12.2	18.6	17.40
$\Delta \bar{\theta}_{\text{Exp}}$	deg	70.7	112.7	74.6	92.2	—
$\Delta \theta_{\text{CFD}}$	deg	74.6	115.5	76.3	94.4	—
$\Delta \theta_{\text{CFD},3\text{D}}$	deg	77.0	—	77.0	—	—
Deviation	%	5.4	2.5	2.3	2.4	3.15
$\Delta \bar{t}_{\text{Exp}}$	ms	636	658	984	947	—
Δt_{CFD}	ms	705	720	1063	1053	—
$\Delta t_{\text{CFD},3\text{D}}$	ms	699	—	996	—	—
Deviation	%	10.3	9.0	7.7	10.6	9.40

Only two 3D simulations were carried out due to the large computational time caused by the cell count of $\approx 3 \cdot 10^6$ cells. As of such, some fields in the table are not available. As seen in Table 6.2, the deviations are all positive, yielding trajectories, which are spanning further to the sides while having larger deflection angles. Furthermore, the time between each turn is also larger for all CFD-results, which yields trajectories with lower oscillation frequencies. The largest deviations can be found for Δx , which has a mean deviation of 20.33 %, whereas the smallest deviations can be found for $\Delta \theta$, which has a mean of 3.15 %.

The reason for the deviations can be caused by the 2D approximation of the phenomena, where tip effects are not included. Such can affect the forces and torques acting on the plates, as the lower pressure on the upper surface of the plate compared with the lower will cause the flow to leak and form vortices. This will also reduce the pressure differences between the lower and upper surfaces, which will reduce both the total force and the torque. Tip effects are thus investigated further in order to determine the influence, which will be described in *Subsection 6.1.6 - Tip effects*. However, the coloured cells are not indicating overall improvements, where four of the eight colour-pairs are yielding better matches with the 2D simulations compared with the 3D ones.

The Δ -results from the fifth plate, which experienced tumbling motion, can be seen in Table 6.3.

Table 6.3: The results from both the experiments and the CFD-simulations for the fifth plate, which was experiencing tumbling motion. The deviations are relative to the mean values.

	Unit	Plate 5		Unit	Plate 5
$(\partial \bar{y} / \partial \bar{x})_{\text{Exp}}$	—	0.563	$(\partial \bar{y} / \partial \bar{t})_{\text{Exp}}$	cm/s	22.0
$(\partial \bar{y} / \partial \bar{x})_{\text{CFD}}$	—	0.490	$(\partial \bar{y} / \partial \bar{t})_{\text{CFD}}$	cm/s	21.1
Deviation	%	−13.8	Deviation	%	−4.2
$(\partial \bar{x} / \partial \bar{t})_{\text{Exp}}$	cm/s	39.1	$(\partial \bar{\theta} / \partial \bar{t})_{\text{Exp}}$	deg/s	2159.4
$(\partial \bar{x} / \partial \bar{t})_{\text{CFD}}$	cm/s	43.0	$(\partial \bar{\theta} / \partial \bar{t})_{\text{CFD}}$	deg/s	3075.9
Deviation	%	9.6	Deviation	%	35.0

As seen in Table 6.3, the deviation between the experimental and CFD-results are significant. Especially the values of $(\partial \bar{\theta} / \partial \bar{t})$ are deviating quite significantly with 35% difference relative to the mean. Once again, this can be caused by the 2D approximation of the phenomena, where the tip effects are causing the forces and torques on the plate to alter from the 2D approximation. As the plate was spinning fast at ≈ 2500 rad/s, the surrounding flow field was complex, which the CFD-model might not have been able to capture perfectly.

6.1.4 Statistical trajectory

Using the mean trajectories obtained from the statistical processing of the experimental results and the corresponding Δ -results, an approximation of the trajectory at $t \rightarrow \infty$ can be constructed. This can be done for the CFD-results as well, though this path will yield the exact same trajectory obtain from the simulations due to the large simulation time of 10 s, where periodic motion was obtained. The statistical trajectory for the third plate has been shown in Figure 6.3, where both the experimental and CFD-trajectories have been shown with labels showing the Δ -values.

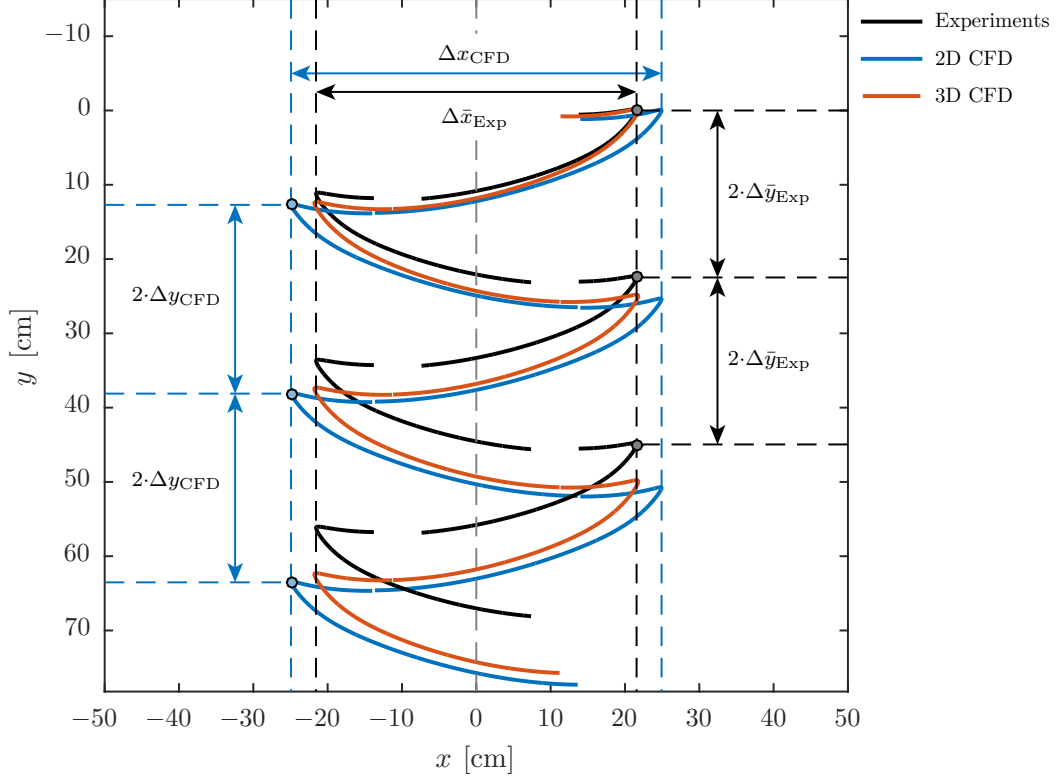


Figure 6.3: The statistical trajectory for the third plate. The black lines are related to the experimental results, the blue to the 2D CFD-results and the red to the 3D CFD-results.

The black curves in Figure 6.3 are related to the experimental results. All black segments are identical, though every second is flipped along the y-axis. The extrema are adjusted by the Δ -values, which represents the mean position for each turn. As of such, the figure is a representation of the periodic trajectory as $t \rightarrow \infty$. The missing regions in the experimental trajectory are once again caused by the clipping of the data due to varying time for each turn as shown in Figure 5.5 in *Section 5.3 - Results*.

The trajectory obtained from the 2D CFD-simulation span wider compared to the experimental trajectory as $\Delta x_{\text{CFD}} > \Delta \bar{x}_{\text{Exp}}$. The same can be seen in the vertical direction, where the CFD-trajectory is drifting compared to the experimental trajectory. However, the 3D CFD-simulation is yielding better comparison with the experimental trajectory, where $\Delta x_{\text{CFD},3\text{D}} \approx \Delta \bar{x}_{\text{Exp}}$, though the 3D CFD-simulation is still drifting in the vertical direction as $\Delta y_{\text{CFD},3\text{D}} > \Delta \bar{y}_{\text{Exp}}$. In general, the 3D simulations is a better approximation of the phenomena compared to the 2D simulation despite the small persistent deviations.

6.1.5 Discussion

The results from the experiments and the CFD-simulations were compared. The comparison between the trajectories yielded deviations, where the same deviating tendencies were observed for plate 1-4 as seen in Figure 6.1. Furthermore, the Δ -results were all larger in the CFD-simulations compared with the experimental results for the same plates. The Δ -results for the fifth plate were both larger and smaller, where the deviations spanned from -13.8 to 35% as seen in Table 6.3. The deviation in most values are probably caused by the 2D approximation, as tip effects will inevitably affect the forces and torques on the plate, especially when the aspect ratio between the x - and z -axes are in the lower ranges. As of such, tip effects will be investigated further in the next subsection.

6.1.6 Tip effects

The derivation between the trajectories of the experiments and the CFD-simulations can be caused by numerous things. One such thing is tip effects, which will affect the lift and drag as a function of the aspect ratio in the z -direction, which will be denoted AR . This can be found to alter the lift and drag coefficients with the approximations in Equation 6.1 [Anderson, 2001].

$$c_{L,3D} = \frac{\partial c_{L,2D}}{\partial \alpha} \left(\frac{AR}{AR + 2} \right) \alpha \quad (6.1)$$

$$c_{D,i} = \frac{c_L^2}{\pi AR e}$$

In Equation 6.1, $c_{L,3D}$ is the reduced lift coefficients as a function of the 2D lift coefficient, $c_{L,2D}$, the aspect ratio, AR , and the angle of attack, α . $c_{D,i}$ is the induced drag coefficients, which is an additional drag term, which is only present in 3D. Here, e is the *planform efficiency factor*, which is related to the circulation distribution, Γ . Equation 6.1 has been derived with the *lifting-line theory* [Anderson, 2001], where a finite wing is typically discretized into a number of span wise panels, where Γ is found for each. The change in Γ in the span wise direction will cause a trailing vortex, which will cause ascending and descending fluid on each side of it. This will in turn cause the relative angle of attack of the neighbouring panels to increase or decrease, thus altering Γ on the other panels. This has been illustrated in Figure 6.4.

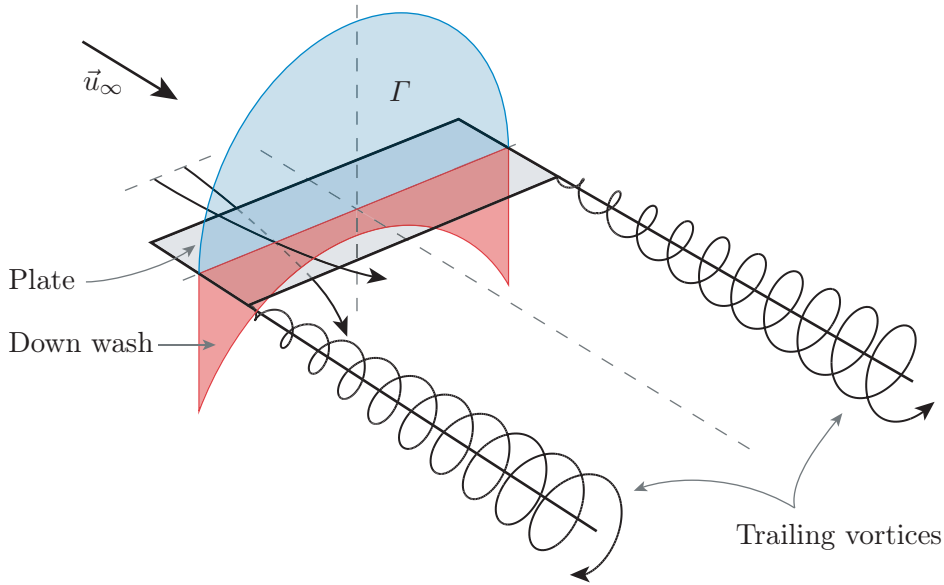


Figure 6.4: A visualization of the lifting-line theory, where the trailing vortices are causing a down wash along the span of the wing, which in turn reduces the circulation, Γ , on the wing.

Figure 6.4 indicated a flat plate with a non zero angle of attack. The lower pressure on the upper surface compared to the lower one causes trailing vortices to be formed at the tips. These will cause the vertical velocity seen by the wing to alter along the span of the wing. This change in vertical velocity will change the relative angle of attack along the wing, which will alter the circulation, Γ . For a square plate, the circulation distribution will be elliptical, which is taken into account in Equation 6.1 as $e = 1$.

The aspect ratio in the z -direction for the plates used in the experiments vary between $250/20 = 12.5$ and $250/80 = 3.125$. Substituting these values into Equation 6.1, the reduction 3D lift coefficient, $c_{L,3D}$, reduces to 86.2 and 61.0% for the two mentioned limiting

values of AR . This would alter the trajectories significantly, which is assumed to be the major source of error between the experiments and the 2D simulations.

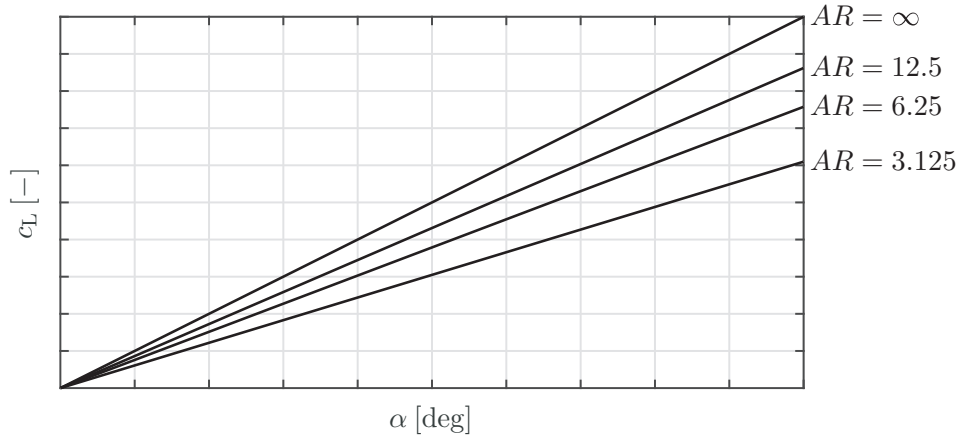


Figure 6.5: The reduction in c_L caused by the finite span of the plates. The values of AR relates to the plates in the experiments.

As seen in Figure 6.5, the finite span of the plates in the experiments will reduce the values of c_L significantly. $AR = 12.5$ corresponds to plate 5, $AR = 6.25$ to plate 3-4, and $AR = 3.125$ to plate 1-2. Along with the reduction in c_L , the induced drag term, $c_{D,i}$, will be present as well. As of such, the difference between 2D and 3D in this case will be decreased lift and increased drag. However, many parameters are affecting the actual trajectory, so a general tendency is difficult to identify.

6.2 Partial conclusion

This chapter served to verify the CFD-model described in *Chapter 4 - CFD-model*. Five simulations were carried out, which corresponded to the experiments. The simulations were all 10 s long, where a repeating pattern was observed after the first couple of turns. The last complete turn from each of the simulations were compared with the statistically averaged turns from the experiments. The results indicate an overall disagreement, where the CFD-simulations had wider trajectories. This could be observed in Table 6.2, where the average deviation for Δx was 20.3 %. The same tendency was observed for Δy , where the average deviation was 17.4 %.

Two 3D simulations were carried out, where tip effects were present. These simulations did match the experimental results better. Based on these findings, it was assumed, that the primary source of error between the experiments and the 2D CFD-model was caused by 3D effects. Tip effects were investigated with correlations derived from the *lifting-line theory*, which revealed up to 39 % reduction in c_L caused by the tip effects along with the introduction of induced drag.

Despite the deviation from the experiments and the 2D CFD-simulations, the model is assumed verified in the case of aspect ratios in the x - z -plane going towards ∞ . As of such, it can be used with confidence in a parametric analysis, where different parameters will be varied and simulated. This is described in the next chapter.

7 | PARAMETRIC ANALYSIS

As the CFD-model had been verified, a parametric analysis could be carried out. The purpose of this analysis was to investigate the trajectories for a wide range of plate geometries and densities. A total of 36 simulations were carried out, which the computationally fast CFD-model allowed for¹. The results were used to fit the PT-model in *Chapter 8 - Predictive Trajectory model*.

7.1 Overview

In order to capture a wide range of trajectories, 36 simulations were carried out. These were distributed over two parameters, which were the aspect ratio of the plate, $\beta (L, H)$ and the dimensionless moment of inertia, $I^* (\rho_f, \rho_p, L, H)$. These two dimensionless groups were varied according to Figure 7.1.

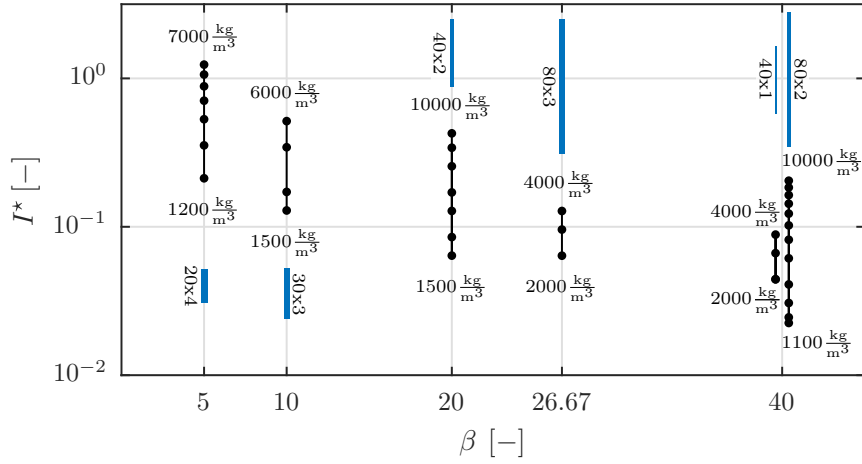


Figure 7.1: The 36 CFD-simulations as a function of β and I^* . The plate densities at the top and bottom of each column are indicating the extreme values, and the blue rectangles are indicating the plates to scale with each other along with specified dimensions.

Five distinctive values of β are present in Figure 7.1, but six different geometries were simulated. Two geometries are present at $\beta = 40$, where plates with dimensions 40 mm \times 1 mm and 80 mm \times 2 mm were used. The motion of each plate was analysed, where Re was calculated according to the definition given in *Section 1.2 - Trajectory characterization*, where the characteristic velocity was calculated as the apparent vertical velocity for periodic motion, whereas the average velocity magnitude was used for tumbling motion. The results can be seen in Figure 7.2.

¹Animations for various CFD-simulations can be found on the USB-stick under the path: `../ANSYS Fluent/CFD_Animations.avi`.

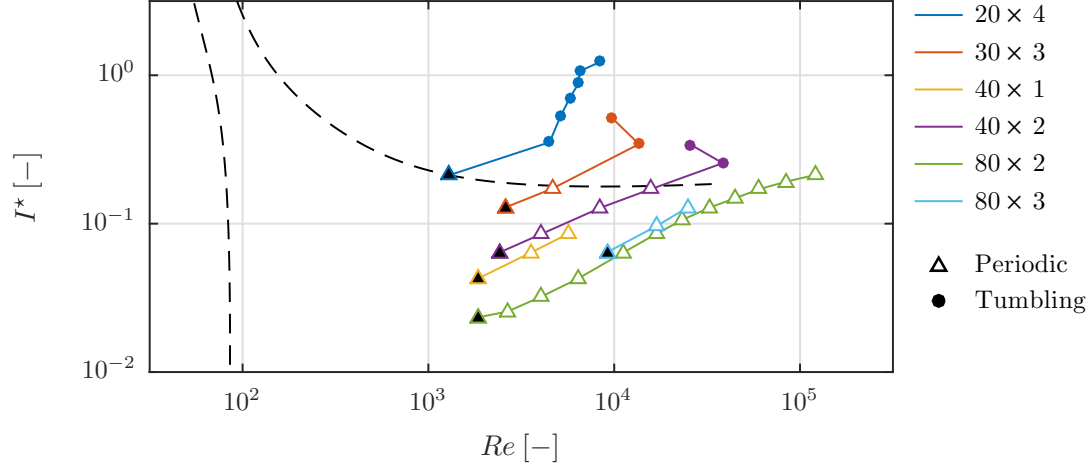


Figure 7.2: The 36 different CFD-simulations carried out in the parametric analysis. Each unique geometry has its own colour, and the simulation with the lowest density is indicated with the black filled markers. The legend is indicating the dimensions of the plates in units of mm.

It can be seen that the results from the CFD-model matches the bifurcation lines from reference [Smith, 1970], which were shown in Figure 1.2 in *Chapter 1 - Introduction* as well. However, though the density was varied extensively from 1100 to 10000 kg/m³ and the aspect ratio from 5 to 40, no trajectories yielded steady falling plates. It can be seen that all trajectories were located at $Re > 10^3$, which might suggest that the kinematic viscosity of the fluid should be increased in order to obtain steady falling trajectories.

The 36 simulations were each simulated for 5 s, which took ≈ 7 hours for each simulation running in parallel on a dedicated computer for this purpose with a quad core Intel Core i7-4770 CPU @ 3.40 GHz. In total ≈ 15 days of simulation time was used to compute the trajectories. In order to automatize the parametric study, *ANSYS Fluent Journal Files* were used, which allowed for automatic continuation to the next simulation without any user input. The total cell count for the simulations spanned between 18.3 and $36.6 \cdot 10^3$ cells.

7.2 Partial conclusion

This chapter has described the parametric analysis carried out with the developed CFD-model in *Chapter 4 - CFD-model*. Both periodic and tumbling motions were found, which were matching the bifurcation lines observed in other references. This implicitly verifies the CFD-model, though no steady falling trajectories were obtained. The 36 CFD-simulations will be used in the next chapter to tune the PT-model to be developed.

8 | PREDICTIVE TRAJECTORY MODEL

The *Predictive Trajectory model*, PT-model, will serve to predict the trajectory for a given plate defined by L , H and ρ_p in a fluid with properties ρ_f and μ . The PT-model will be formed as a set of coupled differential equations, which can be solved orders of magnitude faster compared to a full CFD-simulation.

This chapter will define a set of orientation vectors, which will be used in the PT-model. A PT-model partly based on inviscid and irrotational flow will be developed, which is investigated in *Section 8.4 - Comparison with CFD-simulations* in order to find its weaknesses. These weaknesses will be further investigated, where additional terms are included in the PT-model in order to obtain more accurate trajectory predictions relative to the CFD-simulations. Lastly, the dynamics of the model will be investigated in *Section 8.6 - Dynamic investigation*, where different regime maps will be formed based on the motion of the plates in free fall.

The purpose of the PT-model was to develop a model capable of predicting trajectories for various plates. Furthermore, it was developed in a general manner, where the methods used can be applied to arbitrary shapes. This leads to a more general framework, which allows for further studies within the proposed framework.

8.1 Overview

The PT-model was partly based on inviscid and irrotational flow theory, which has been applied by other studies as well [Andersen et al., 2005b]. By assuming these flow conditions, the motion of a rigid body is governed by *Kirchhoff's Equations* [Lamb, 1895]. These equations take added mass into account, which was found crucial for accurate modelling of the phenomena. External forces were combined with *Kirchhoff's Equations*, where three main groups were identified. These have been shown in Figure 8.1, which serves as an overview of the PT-model.

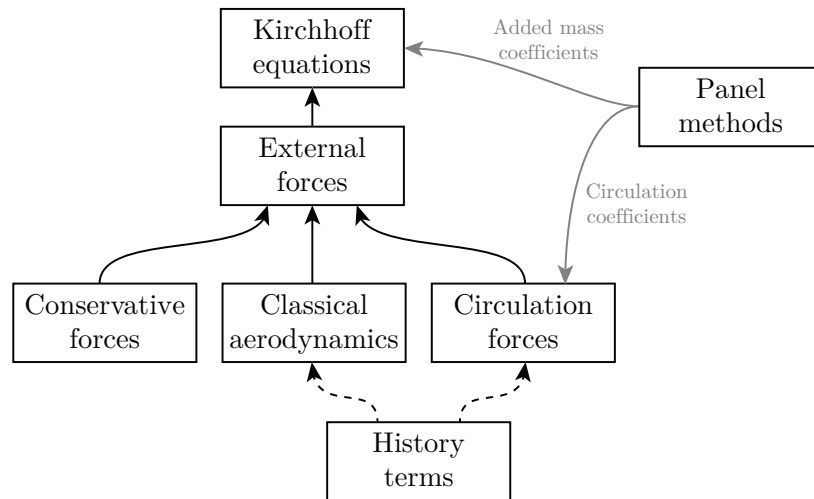


Figure 8.1: An overview of the PT-model and the interactions between the various forces and applied methods. The history terms are included as a modification to the model, thus the dashed connections.

As seen in Figure 8.1, an overview of the PT-model is given. The model was based on *Kirchhoff's Equations*, where external forces were incorporated. These were divided into

conservative, aerodynamic and circulation forces, which were described in *Subsection 8.3.2 - External forces*. Panel methods were applied to calculate added mass and circulation coefficients, which were only functions of the geometry. History forces were also investigated, but these would be implemented as modifications, which will be described in *Section 8.5 - Modified PT-model*.

As the model was primarily based on inviscid and irrotational flow theory, the PT-model was not expected to follow the trajectories obtained from the CFD-simulations perfectly. As of such, an optimization tool was developed, which could tune coefficients in the PT-model to match the CFD-simulations better.

8.2 Orientation vectors

Before the PT-model could be formed, a set of orientation vectors were required, so the calculated forces were acting in the right direction. Equation 8.1 is showing these vectors as angles, where the normal vectors can be formed by $\vec{n}_- = [\cos(\theta_-), \sin(\theta_-)]$, where the blank fields indicate any of the angle subscripts in Equation 8.1.

(Translational angle)	$\varphi = \text{atan2}(u_y, u_x)$	
(Angle of attack)	$\alpha = \text{asin}((\vec{n}_u \times \vec{n}_\theta)_z) $	
(Lift)	$\theta_L = \varphi + \frac{\pi}{2} \cdot \text{sign}((\vec{n}_u \times \vec{n}_\theta)_z \cdot (\vec{n}_u \cdot \vec{n}_\theta))$	
(Drag)	$\theta_D = \varphi + \pi$	(8.1)
(Top)	$\theta_{\text{Top}} = \theta + \frac{\pi}{2} \cdot \text{sign}((\vec{n}_u \times \vec{n}_\theta)_z)$	
(Trailing edge)	$\theta_{\text{Te}} = \theta + \pi \cdot H(\vec{n}_u \cdot \vec{n}_\theta)$	
(Leading edge)	$\theta_{\text{Le}} = \theta + \pi \cdot H(-\vec{n}_u \cdot \vec{n}_\theta)$	

All angles in Equation 8.1 are relative to the x-axis, and H is the *Heaviside step function*. φ is the angle, at which the plate is moving independent of its own orientation. The operator atan2 is the *four-quadrant inverse tangent*, which is a modified version of atan that detects angles from $-\pi$ to π , and is common in many programming languages [MathWorks, 2015]. α is the angle of attack, which is always positive and between 0 and 90 deg. \vec{n}_θ is the normal vector along the plate, and \vec{n}_u is the normal vector in the direction of movement. The remaining angles are used to form various relative vectors, which are all defined in the global coordinate system. Three examples are shown in Figure 8.2, where the associated vectors have been overlaid.

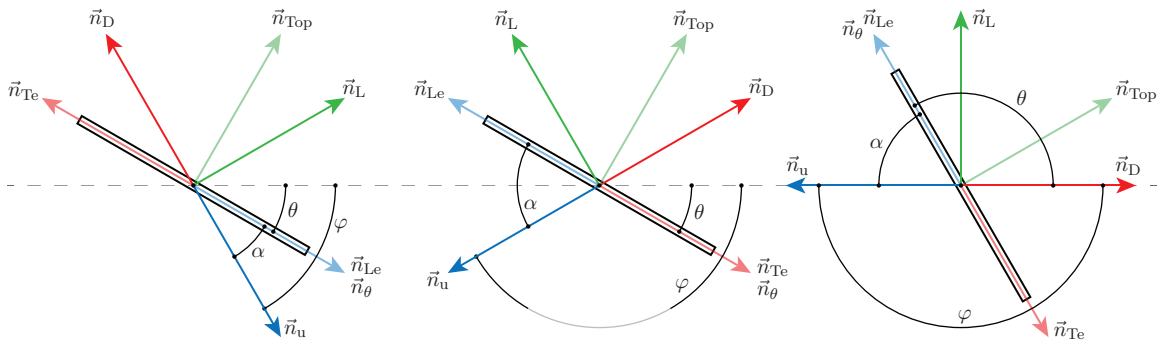


Figure 8.2: Four examples with various plate orientations and translational directions. The vectors are related to Equation 8.1.

Two inputs are required to construct all vectors in Figure 8.2, which are the plate orientation, θ , and the translational direction, $\varphi = \text{atan2}(u_y, u_x)$. \vec{n}_L , \vec{n}_D and \vec{n}_{Top} are always on the opposite side of the plate relative to \vec{n}_u . \vec{n}_{Le} is always in the direction of \vec{n}_u along the plate, where \vec{n}_{Te} is in the opposite direction. It can be seen that \vec{n}_θ is always coinciding with either \vec{n}_{Le} or \vec{n}_{Te} .

All vectors in Figure 8.2 are based on the global coordinate system. However, the local coordinate system, which is following the plate, is advantageous to use when developing the model. The transformations matrices from global to local coordinates and vice versa have been shown in Equation 8.2, where the superscript $()'$ is indicating local coordinates. The example is showing the transformation of the plate velocity, \vec{u} .

$$\underbrace{\begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix}}_{\mathbf{T}_{G \rightarrow L}} \cdot \begin{bmatrix} u_x \\ u_y \end{bmatrix} = \begin{bmatrix} u'_x \\ u'_y \end{bmatrix} \quad (\text{Global to local})$$

$$\underbrace{\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}}_{\mathbf{T}_{L \rightarrow G}} \cdot \begin{bmatrix} u'_x \\ u'_y \end{bmatrix} = \begin{bmatrix} u_x \\ u_y \end{bmatrix} \quad (\text{Local to global}) \quad (8.2)$$

As the set of orientation vectors along with the transformation matrices, \mathbf{T} , has been defined, the PT-model can be developed, which is described in the next section.

8.3 Model development

The core of the PT-model was based on *Kirchhoff's Equations*. These equations would govern the motion of the plate, where added mass was taken into account. External forces would be coupled with these equations as shown in Figure 8.1. *Kirchhoff's Equations* are used to model the motion of a rigid object in an inviscid and irrotational fluid based on a momentum approach. These equations have been shown in Equation 8.3 for the general 3D case.

$$\begin{bmatrix} \vec{p} \\ \vec{\ell} \end{bmatrix} = \mathbf{M} \cdot \begin{bmatrix} \vec{u}' \\ \vec{\theta}' \end{bmatrix}$$

$$\frac{\partial}{\partial t} \begin{bmatrix} \vec{p} \\ \vec{\ell} \end{bmatrix} = \begin{bmatrix} \vec{p} \times \vec{\theta}' \\ \vec{\ell} \times \vec{\theta}' + \vec{p} \times \vec{u}' \end{bmatrix} + \begin{bmatrix} \vec{F}' \\ \vec{\tau}' \end{bmatrix}_{\text{Ext}} \quad (8.3)$$

$$\mathbf{M} = \mathbf{M}_p + \mathbf{M}_f$$

\vec{p} and $\vec{\ell}$ are the translational and rotational momenta respectively. The subscript $()_{\text{Ext}}$ is indicating external forces. \mathbf{M} is the mass matrix, where \mathbf{M}_p is the *mass-inertia tensor* and \mathbf{M}_f is the *added mass tensor* as defined in Equation 3.8 in *Subsection 3.2.2 - Added mass forces* on page 11, though this definition was in 2D. When evaluating Equation 8.3 in 2D, where u_z , $\dot{\theta}_x$ and $\dot{\theta}_y$ are all zero, and assuming a diagonal added mass tensor due to the symmetric geometry of a plate, the equations simplify to Equation 8.4.

$$\begin{aligned}
\begin{bmatrix} p_x \\ p_y \\ \ell_z \end{bmatrix} &= \mathbf{M} \cdot \begin{bmatrix} u'_x \\ u'_y \\ \dot{\theta} \end{bmatrix} \\
\frac{\partial}{\partial t} \begin{bmatrix} p_x \\ p_y \\ \ell_z \end{bmatrix} &= \begin{bmatrix} \dot{\theta} p_y \\ -\dot{\theta} p_x \\ p_x u'_y - p_y u'_x \end{bmatrix} + \begin{bmatrix} F'_x \\ F'_y \\ \tau \end{bmatrix}_{\text{Ext}} \\
\mathbf{M} = \mathbf{M}_p + \mathbf{M}_f &= \underbrace{\begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & I \end{bmatrix}}_{\mathbf{M}_p} + \underbrace{\begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & 0 \\ 0 & 0 & m_{66} \end{bmatrix}}_{\mathbf{M}_f}
\end{aligned} \tag{8.4}$$

Notice that $\dot{\theta}$ does not have the superscript $(\cdot)'$ in the 2D case, as this variable is independent of the coordinate system being used. Recall that m and I are the mass and moment of inertia of the plate, whereas m_{11} and m_{22} are the added masses and m_{66} the added moment of inertia, which were caused by the surrounding fluid. By combining and simplifying the equations in Equation 8.4, the inviscid and irrotational flow model could be expressed as three coupled differential equations for a symmetric plate in 2D. This has been shown in Equation 8.5.

$$\begin{aligned}
(m + m_{11})a'_x &= (m + m_{22})\dot{\theta} u'_y + F'_{x,\text{Ext}} \\
(m + m_{22})a'_y &= -(m + m_{11})\dot{\theta} u'_x + F'_{y,\text{Ext}} \\
(I + m_{66})\ddot{\theta} &= (m_{11} - m_{22})u'_x u'_y + \tau_{\text{Ext}}
\end{aligned} \tag{8.5}$$

The equations in Equation 8.5 are simply a rewritten form of Equation 8.4. The left hand side is intuitive, where, for instance, the horizontal acceleration is multiplied with the mass of the plate and the added mass along the long axis of the plate, m_{11} . The right hand side consists of the external forces in the local coordinate system along with the force and torque contributions caused by *Kirchhoff's Equations*. The latter terms do not seem intuitive, as they are using local velocity products and arbitrary masses. As of such, these force and torque contributions were verified numerically using panel methods in order to gain confidence in the expressions. This was done by varying the local velocities, u'_x , u'_y and $\dot{\theta}$, and calculating the corresponding forces and torque. The results indicated that the expressions hold true, though the numerical results have not been included in this study.

As the coupled differential equations governing the dynamics of a rigid body in an inviscid and irrotational fluid has been defined, the added masses and external forces can be defined. These have been described in the following subsections.

8.3.1 Added mass

The added masses in Equation 8.5 can be calculated using Equation 8.6.

$$\mathbf{M}_f = \frac{\mathbf{C}_{\text{AM}} \cdot \mathbf{M}_p}{\rho^*} \quad \mathbf{C}_{\text{AM}} = \begin{bmatrix} C_{11} & 0 & 0 \\ 0 & C_{22} & 0 \\ 0 & 0 & C_{66} \end{bmatrix} \tag{8.6}$$

The dimensionless density ratio, $\rho^* = \rho_p/\rho_f$, has been used for simplicity. The added mass coefficient matrix, \mathbf{C}_{AM} , is diagonal due to the symmetry of a plate, where the three coefficients can be calculated using panel methods. This procedure has been described in

Appendix A - Panel Methods, where the dependencies of the coefficients on β has been shown in Figure 8.3¹.

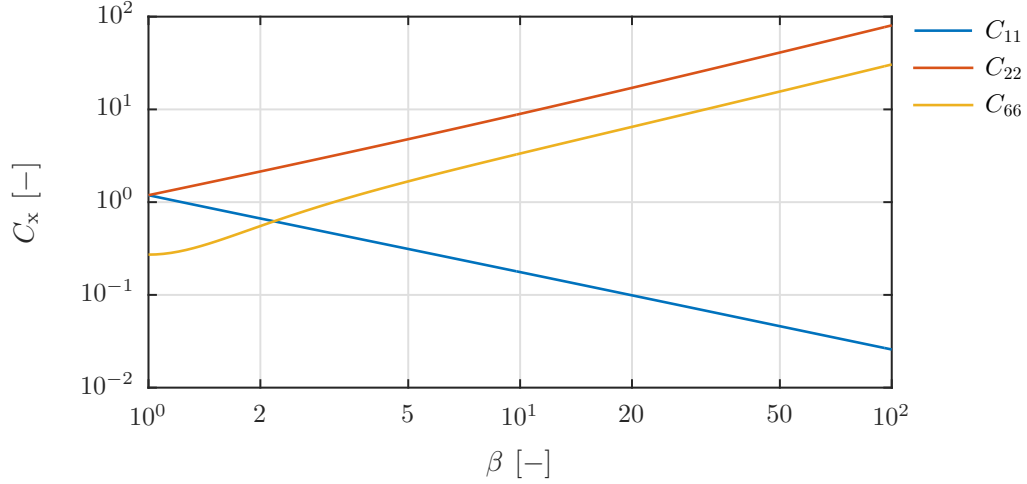


Figure 8.3: The dependency of the added mass coefficients, C_{11} , C_{22} and C_{66} , as a function of the aspect ratio of the plate, $\beta = L/H$.

The added mass coefficients seen in Figure 8.3 are almost linear as a function of β in a log-log plot. The two translational added mass coefficients, C_{11} and C_{22} , coincide at $\beta = 1$, which is expected, as both x- and y-symmetry exists at this point. When β increases, C_{11} decreases mainly due to the decrease in height of the plate, and C_{22} increase mainly due to the decrease in volume within the plate. The added moment of inertia coefficient, C_{66} , can be seen to increase exponentially non-linearly at small values of β , but after $\beta \gtrsim 5$ the tendency is almost linear in a log-log plot.

8.3.2 External forces

As the differential equations governing the plate dynamics has been defined along with the added mass coefficients, the external forces can be defined. These have been shown in their expanded forms in Equation 8.7.

$$\begin{aligned}\vec{F}'_{\text{Ext}} &= \vec{F}'_g + \vec{F}'_B + \vec{F}'_D + \vec{F}'_I \\ \tau_{\text{Ext}} &= \tau_g + \tau_B + \tau_D\end{aligned}\quad (8.7)$$

As seen in Equation 8.7, multiple terms are contributing to the external forces and torque. The subscripts $()_g$, $()_B$, $()_D$ and $()_I$ are gravity, buoyancy, drag and circulation respectively. These will each be described in the following paragraphs.

Conservative forces

Gravity and buoyancy are the conservative forces, which have been shown in Equation 8.8.

$$\begin{aligned}\vec{F}'_g &= \begin{bmatrix} \sin(\theta) & 0 \\ 0 & \cos(\theta) \end{bmatrix} \cdot (\vec{g} m) & \vec{F}'_B &= \begin{bmatrix} \sin(\theta) & 0 \\ 0 & \cos(\theta) \end{bmatrix} \cdot \left(-\vec{g} \frac{m}{\rho^*}\right) \\ \tau_g &= (\vec{n}_\theta L_{CG}) \times (\vec{g} m) & \tau_B &\approx 0\end{aligned}\quad (8.8)$$

The buoyancy force is formulated using the density ratio, ρ^* , for simplicity. The trigonometric functions are used to transform the global vector into the local vectors along the plate. The

¹The calculations of the added mass coefficients can be seen on the USB-stick under the path: `../MATLAB/Panel Methods/Panel_Methods.m`

torque contribution due to gravity, τ_g , is zero when the centre of gravity location, L_{CG} , is zero, which is the case throughout this study. The torque contribution due to buoyancy is negligible for a symmetric plate, and was thus not included in the model.

Drag forces

A consequence of the inviscid flow approach is *d'Alembert's paradox*, which states that an object in an inviscid flow does not experience any drag force. As of such, the drag must be modelled. This was done using a classical aerodynamic approach, where the drag was quadratic in the translational velocity. This has been shown in Equation 8.9.

$$\vec{F}'_D = \mathbf{T}_{G \rightarrow L} \cdot \left(\vec{n}_D \left(\frac{1}{2} \rho |\vec{u}|^2 L (1 \text{ m}) c_D(\alpha, \beta, Re) \right) \right) \quad (8.9)$$

As seen in Equation 8.9, the classical quadratic drag equation is multiplied to the normal vector pointing in the direction of the drag force, \vec{n}_D , after which it is multiplied to the transformation matrix from global to local coordinates, $\mathbf{T}_{G \rightarrow L}$.

The drag coefficient, $c_D(\alpha, \beta, Re)$, was found as a function of three parameters. The function was found by performing steady-state CFD-simulations for six different values of β in a fully turbulent flow at $Re = 10^6$. These simulations were carried out at $\alpha = 0 \text{ deg}$, where the parallel drag coefficients were found. Furthermore, another set of simulations were carried out, where β was fixed at 40 and Re was varied between 10^{-1} to 10^6 in 76 steps, where both the parallel and perpendicular drag coefficients were found. Two expressions were fitted to these results, where each of them consisted of three continuous function. Lastly, a set of simulations were carried out at $\beta = 40$ and $Re = 10^6$, where α was varied from 0 to 90deg with 5deg increments. These were fitted with a trigonometric function. All the functions were blended and implemented in the PT-model, where the final expression can be seen in *Enclosure D - Coefficient of drag*.

The drag torque, τ_D , in Equation 8.7 is caused by the resistance of the plate to rotate due to the displaced fluid. This term was modelled using the tangential velocity along the plate, where both the translational and rotational velocities were included. This approach was different from references [Andersen et al., 2005b; Jensen and Hærvig, 2014], where only the rotational velocity was used. The tangential velocity along the plate, $u_{\text{Tan}}(x')$, has been shown in Equation 8.10, where the local infinitesimal tangential drag force has been shown as well.

$$\begin{aligned} u_{\text{Tan}}(x') &= u'_y + \dot{\theta} x' \\ dF_{\text{Tan}} &= \frac{1}{2} \rho \left(|u_{\text{Tan}}(x')| u_{\text{Tan}}(x') \right) c_{D,\text{Tan}} dA \end{aligned} \quad (8.10)$$

$c_{D,\text{Tan}}$ is the tangential drag coefficient. The tangential forces along the plate can be integrated to yield the resistance torque on the plate. This has been shown in Equation 8.11.

$$\tau_D = \int_{-L/2}^{L/2} x' dF_{\text{Tan}} \approx \sum_{i=1}^N (\Delta F_{\text{Tan},i} \cdot x'_i) \cdot \frac{L}{N} \quad (8.11)$$

The integral in Equation 8.11 was evaluated in the PT-model by dividing the plate into 25 sections and performing numerical integration, which allowed for fast execution with good accuracy. The discrete integral has been shown in Equation 8.11 as well.

Circulation forces

The remaining term in Equation 8.7 is \vec{F}'_F , which is related to the circulation, Γ , around the plate in inviscid and irrotational flow. Γ must be modelled, where multiple approaches

has been used in literature. Reference [Mahadevan et al., 1998] used a constant values of Γ , whereas reference [Tanabe and Kaneko, 1994] used the classical Kutta-Joukowski model. The latter approach was used by [Andersen et al., 2005b] as well, but with small modifications. Equation 8.12 shows Γ , which consists of a translational and a rotational term, Γ_T and Γ_R respectively.

$$\Gamma = - \underbrace{\left(\frac{L}{2}\right) C_T |\vec{u}| S_\alpha S_{\text{Sign}}}_{\Gamma_T} + \underbrace{\left(\frac{L}{2}\right)^2 C_R \dot{\theta}}_{\Gamma_R}$$

$$S_{\alpha,1} = \sin(\alpha)$$

$$S_{\alpha,2} = \frac{\sin(2\alpha)}{2}$$

$$S_{\text{Sign}} = \text{sign}\left((\vec{n}_\theta \times \vec{n}_u)_z\right)$$
(8.12)

The variable S_α is a scaling parameter on Γ_T as a function of α . The classical Kutta-Joukowski model yields $S_\alpha = S_{\alpha,1}$, which matches the results obtained by the panel methods in Appendix A. However, this method has a discontinuity, when the separation point switches side at $\alpha = 90$ deg. Reference [Andersen et al., 2005b] took stall into account by using $S_\alpha = S_{\alpha,2}$, which also removed the discontinuity. Furthermore, this approach allows for steady falling plates, where $\alpha = 90$ deg. It should be noted that $(\partial S_{\alpha,1}/\partial\alpha) = (\partial S_{\alpha,2}/\partial\alpha)$ at $\alpha = 0$. The variable S_{Sign} is simply a term required for Γ_T to have the correct sign due to the convention of α used in this study.

By use of the panel methods, numerous simulations were carried out in order to investigate, whether Equation 8.12 was a complete description of Γ under inviscid and irrotational flow conditions. The reason for doing so was to eliminate any concerns about another unknown term, which might depend on quadratic velocities or velocity products. It was found that Equation 8.12 completely described Γ , so confidence could be gained in the expression. However, the calculations for this verification has not been included in this report.

The translational and rotational circulation coefficients in Equation 8.12, C_T and C_R respectively, could be found by use of panel methods as well, and are only dependent on the plate geometry. The method of doing so has been described in *Appendix A - Panel Methods*, where the results can be seen in Figure 8.4².

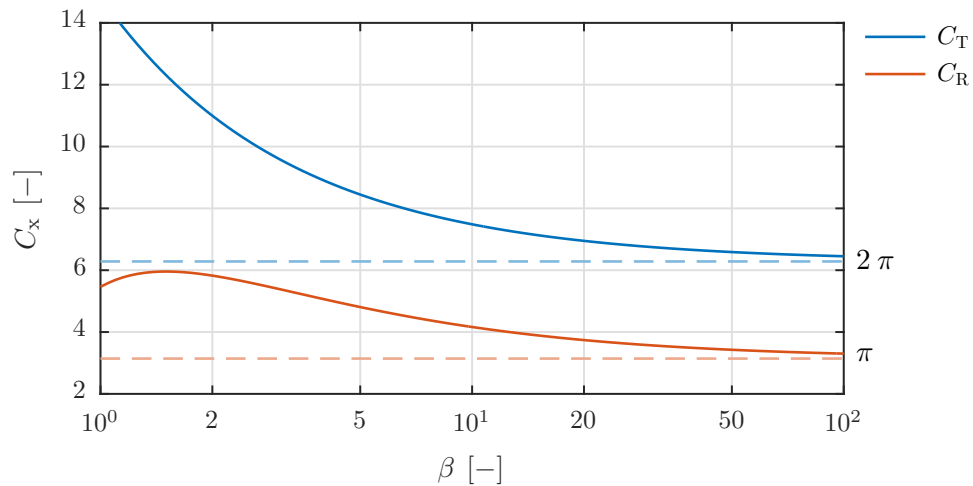


Figure 8.4: The dependency of the circulation coefficient, C_T and C_R , on the aspect ratio of the plate, $\beta = L/H$.

²The calculations of the circulation coefficients can be seen on the USB-stick under the path: `../MATLAB/Panel Methods/Panel_Methods.m`

As seen in Figure 8.4, the circulation coefficients vary significantly dependent on β . However, as $\beta \rightarrow \infty$ the values of C_T and C_R converge towards 2π and π respectively. At $\beta \rightarrow \infty$, the plate has no thickness, so the results can be compared with the *thin airfoil theory*. This theory states that at low values of α , the lift coefficient slope is 2π [Flandro et al., 2012], which is in agreement with the results obtained from the panel methods. However, the value of C_R cannot be verified using the *thin airfoil theory*, and no references could be found to verify the circulation for a rotating thin airfoil.

When Γ has been calculated, the local force contribution, \vec{F}'_l , can be calculated according to the *Kutta-Joukowski theorem*. This has been shown in Equation 8.13.

$$\vec{F}'_l = \begin{bmatrix} -u'_y \\ u'_x \end{bmatrix} \rho_f \Gamma \quad (8.13)$$

The local velocities are used in Equation 8.13, which is a reformulation of Equation 3.12 in *Subsection 3.2.4 - Concept of circulation* on page 13.

8.3.3 Numerical implementation

The PT-model was developed in *MATLAB*, where it was coded in such a way that it could be executed by the in-built ODE-solvers. The PT-model in this format has been shown in *Enclosure C - Predictive Trajectory model*³. The in-built solvers in *MATLAB* did not allow for fast execution of the model. As of such, a predictor-corrector integration method was implemented, as this method proved more stable with less function evaluations per integration step. A *PECECE*-scheme was used [Hairer et al., 1992], where *P*, *E* and *C* stands for *Predict*, *Evaluate* and *Correct* respectively. By using this scheme along with a time-step of 0.50 ms, a fast, accurate and stable integration procedure was developed. Furthermore, as a minimal integration procedure had been programmed, the PT-model could be compiled along with this procedure using the *MATLAB Coder*, which allowed for much faster execution. Combining this with parallel utilization, a 5 s simulation using the PT-model could be executed in ≈ 30 ms. This proved to be critical for fast parameter tuning.

As the PT-model has been developed, the accuracy of this can be evaluated by comparison with CFD-simulations. This has been described in the next section.

8.4 Comparison with CFD-simulations

The validity of the developed PT-model was investigated by evaluating it with the same properties and initial conditions as the CFD-simulations carried out in *Chapter 7 - Parametric analysis*.

When executing the PT-model with the calculated circulation coefficients using the panel method, the trajectories were found to differ significantly from the CFD-simulations. As of such, the values of the circulation coefficients, C_T and C_R , and the value of $c_{D,Tan}$ were tuned to match the CFD-simulations. These three variables were chosen, as they were affecting the trajectories significantly. In order to tune the coefficients automatically, an objective function had to be defined, which quantified the error between the trajectories obtained from the PT-model and the CFD-simulations. This objective function took multiple parameters into account, which have been listed in the table below:

³An interactive version of the PT-model can be found on the USB-stick under the path: `../MATLAB/Predictive Trajectory Model/Predictive_Trajectory_Model.m`

- Integrated squared distance error.
- Integrated squared velocity error.
- Integrated squared acceleration error.
- Squared Δ -results as defined in *Section 5.3 - Results*.

The method of including all the above mentioned parameters yielded an objective function, which could be adjusted to weight differently in order to investigate different aspect of the trajectories⁴.

After having implemented the objective function, an optimization environment was coded, where different plates could be tuned to minimize the objective function, which would yield trajectories matching the CFD-simulations the best. A plate with dimensions of 80 mm \times 2 mm with densities of 3000 and 6000 kg/m³ was tuned to the corresponding CFD-simulations, where the results can be seen in Figure 8.5 on the next page. The position and orientation, along with the first and second derivatives of these variables, have been shown. The arrows are indicating in which direction the curves are moving with respect to time.

As seen in Figure 8.5, the obtained trajectories using the PT-model with tuned coefficients were found to match the CFD-simulations with good accuracy. It can be seen that the values of θ , $\dot{\theta}$ and $\ddot{\theta}$ are matching very well, which can be seen in the figures labelled 4, 5 and 6 respectively. It was found that the primary variables influencing these were the resistance torque drag coefficient, $c_{D,Tan}$, and the rotational circulation coefficient, C_R .

The trajectories were found to match pretty well, but not as good as the orientation. This may not be apparent to see from the positions in the figures labelled 1, but when investigating the derivative in the figures labelled 2 and 3, the deviations become apparent. Especially the values of $\partial^2 x / \partial t^2$ and $\partial^2 y / \partial t^2$ are deviating significantly in some regions. These regions are at the turning points, where the separation point switches side.

As previously described, the circulation coefficients and the resistance torque drag coefficient had to be tuned in order to obtain the trajectories shown in red in Figure 8.5. In order to quantify the relative error between the tuned values of C_T and C_R and the values obtained from the panel methods, two scaling parameters were introduced, λ_T and λ_R , which are defined in Equation 8.14.

$$\lambda_T = \frac{C_{T,Tuned}}{C_{T,Panel}(\beta)} \quad \lambda_R = \frac{C_{R,Tuned}}{C_{R,Panel}(\beta)} \quad (8.14)$$

As seen in Equation 8.14, values of $\lambda > 1$ indicate that the tuned coefficients are larger compared with the calculated ones using the panel methods. The tuned coefficients corresponding to the trajectories shown in Figure 8.5 can be seen in Table 8.1, where the values in parentheses are indicating the values of λ_T and λ_R . For instance, $8.53 = 6.65 \cdot 1.28$, which is shown for C_T .

Table 8.1: The results from the optimization routines for a plate with dimensions 80mm \times 2mm, when tuned to match CFD-trajectories for $\rho_p = 3000$ and 6000kg/m³.

	C_T [–]	C_R [–]	$c_{D,Tan}$ [–]
Panel methods	6.65 (1.00)	3.48 (1.00)	–
Tuned	8.53 (1.28)	6.51 (1.87)	0.518

⁴The actual implementation of the objective function has been described in a note located on the USB-stick under the path ../MATLAB/Predictive Trajectory Model/Objective function.pdf. However, this is only for the interested reader.

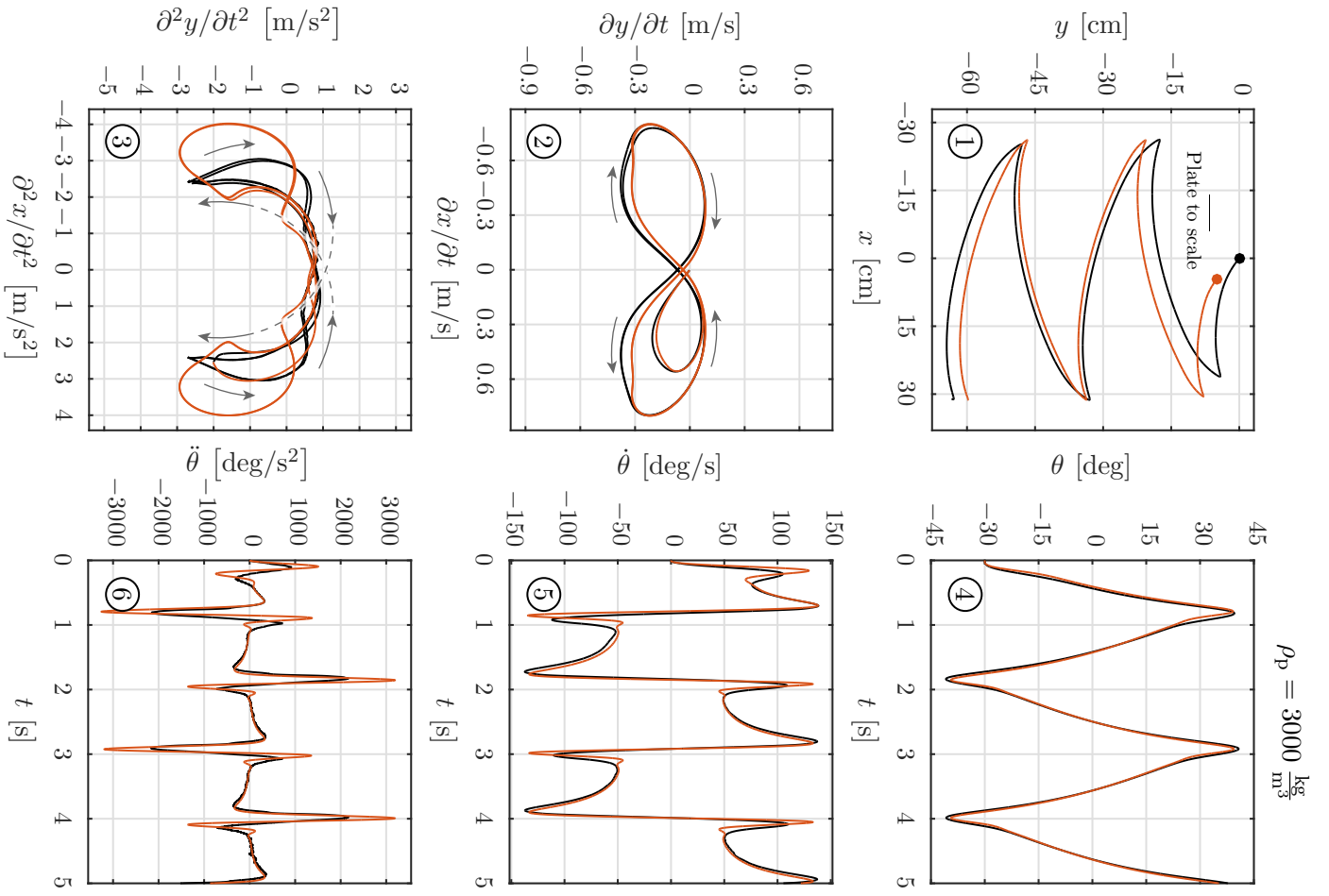
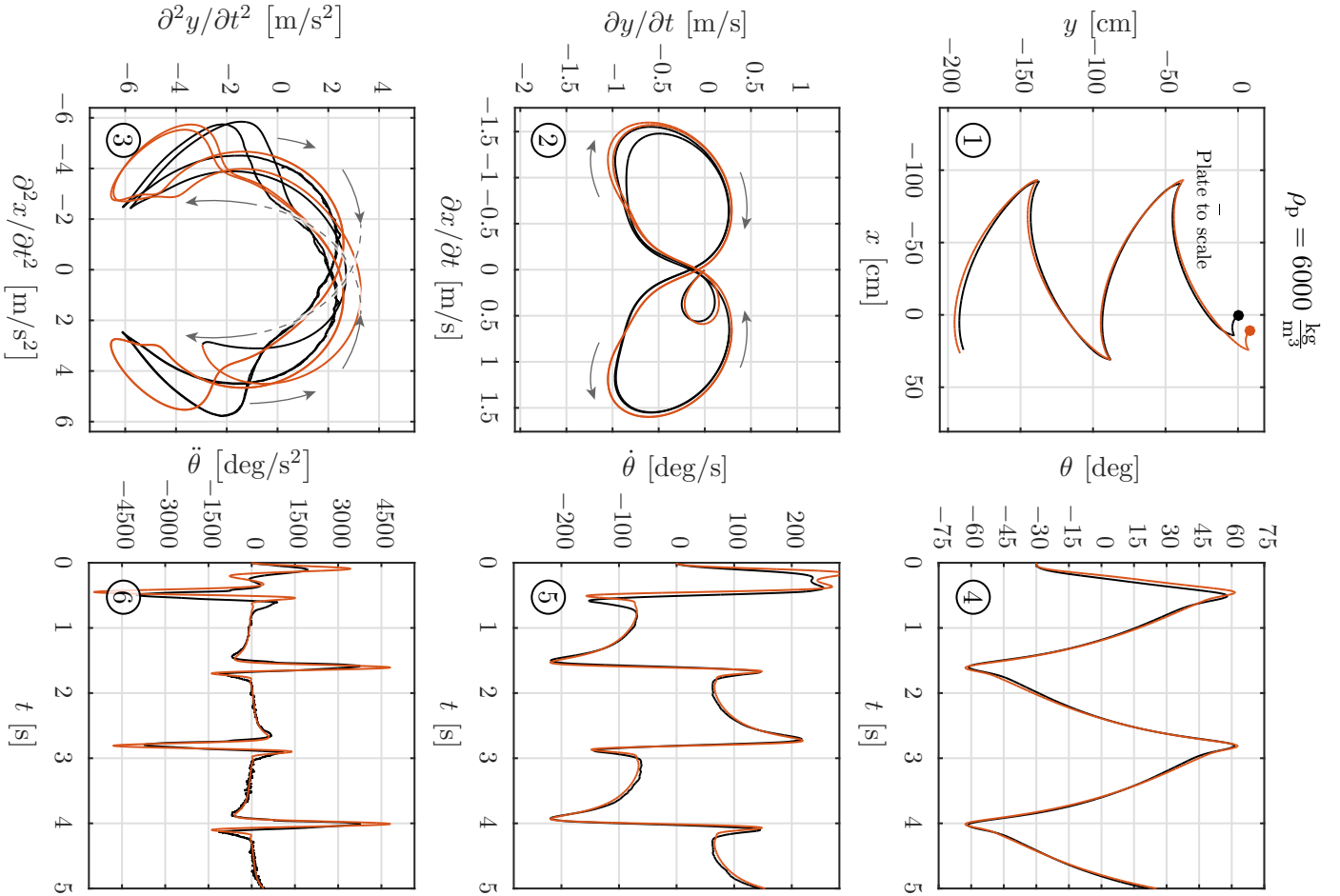


Figure 8.5: The obtained trajectories for a plate with dimensions $80 \text{ mm} \times 2 \text{ mm}$ with densities of 3000 and 6000 kg/m^3 below and above this caption respectively. The black curves are from the CFD-simulations and the red from the PT-model. Notice that the figures labelled (1) are adjusted, as the initial period is not matching well.



As seen in Table 8.1, the required multiplication factors, λ , of the circulations coefficients to obtained the best match with the CFD-trajectories were 1.28 and 1.87 for C_T and C_R respectively. The reason for λ_T and λ_R begin non-unity can be caused by many things, but by investigating the circulation as a function of time, an indication of the weaknesses of the inviscid PT-model can be found. This has been shown in Figure 8.6.

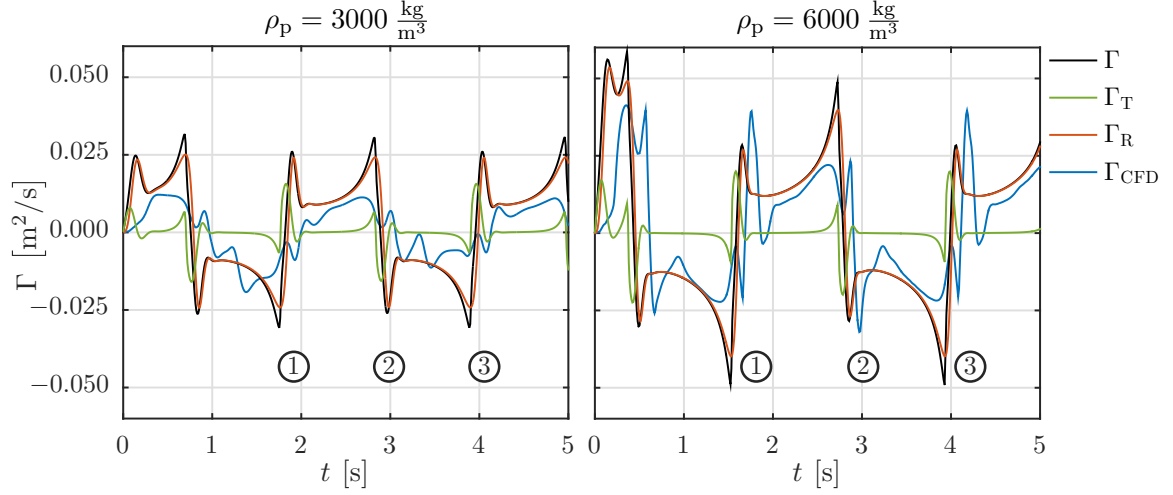


Figure 8.6: The values of the translational and rotational circulation contributions, Γ_T and Γ_R respectively, and the sum of these, $\Gamma = \Gamma_T + \Gamma_R$, from the simulations in Figure 8.5. The blue line is the integrated circulation from the CFD-simulations, Γ_{CFD} .

As seen in Figure 8.6, the rotational circulation, Γ_R , is dominant compared with the translational circulation, Γ_T . This was found by references [Andersen et al., 2005b; Pesavento and Wang, 2004] as well, where the latter concluded that $\Gamma_R \approx 10 \cdot \Gamma_T$ for a falling ellipse. The circulation from the CFD-simulations, Γ_{CFD} , was found by applying the closed surface integral of the vorticity as stated in Equation 3.11 in *Subsection 3.2.4 - Concept of circulation*. The surface integral was performed on a surface bounded by a radius of $2.5L$ relative to the plate centre. The results can be seen as the blue curve in Figure 8.6, which is showing similar tendencies as the PT-model. However, a general tendency can be seen, where the plate with the larger density experiences significantly larger values of Γ_{CFD} compared with the plate with the lower density at the labelled points, (1) - (3). This phenomena was not captured by the PT-model, which might indicate a missing circulation term. Notice the circulation being about twice as large for the plate with the larger density compared with the plate with the lower density.

Table 8.1 is showing the tuned values of C_T and C_R for a single plate geometry with two different densities. The results could be seen to match the CFD-simulations well, but the multiplication factors were not unity. This was investigated for a wide range of geometries and densities, where the multiplication factors, λ_T and λ_R , can be seen in Table 8.2 along with the tuned values of $c_{D,\text{Tan}}$. The first row is indicating the plate dimensions in the following format: $L \times H$ (β), and the tuned coefficients are formatted as: $(\lambda_T \mid \lambda_R \mid c_{D,\text{Tan}})$. The letter in each cell is indicating the plate motion, which is either *P* or *T* indicating *Periodic* and *Tumbling* respectively.

Table 8.2: The optimized multiplication factors on C_T and C_R , λ_T and λ_R respectively, along with the tuned values of $c_{D,Tan}$ for different plate sizes and densities. The format for each cell is $(\lambda_T \mid \lambda_R \mid c_{D,Tan})$.

ρ_p [kg/m ³]	ρ^* [–]	$30 \times 3 (\beta = 10)$	$40 \times 2 (\beta = 20)$	$80 \times 2 (\beta = 40)$
1500	1.5	1.60 \mid 1.50 \mid 2.35 (<i>P</i>)	1.61 \mid 1.59 \mid 1.37 (<i>P</i>)	–
2000	2.0	1.33 \mid 1.59 \mid 2.22 (<i>P</i>)	1.51 \mid 1.68 \mid 0.86 (<i>P</i>)	1.58 \mid 1.70 \mid 1.10 (<i>P</i>)
4000	4.0	1.61 \mid 1.48 \mid 3.48 (<i>T</i>)	1.10 \mid 1.90 \mid 0.10 (<i>P</i>)	1.55 \mid 1.81 \mid 0.68 (<i>P</i>)
6000	6.0	2.20 \mid 1.70 \mid 2.50 (<i>T</i>)	1.23 \mid 2.00 \mid 0.51 (<i>T</i>)	1.47 \mid 1.89 \mid 0.61 (<i>P</i>)
8000	8.0	–	1.80 \mid 2.00 \mid 3.09 (<i>T</i>)	1.11 \mid 2.00 \mid 0.20 (<i>P</i>)
Mixed	–	2.20 \mid 1.70 \mid 2.50	2.54 \mid 1.90 \mid 1.59	1.36 \mid 1.95 \mid 0.56

Each field in Table 8.2 is showing the optimized values of λ_T , λ_R and $c_{D,Tan}$. All fields with specified densities were optimized to a single CFD-simulation, whereas the row labelled *Mixed* took all densities for the current plate geometry into account. It can be seen that all three tuned variables are varying, though some tendencies can be detected. Equation 8.15 is showing the general tendency of the tuned parameters in Table 8.2.

$$\begin{aligned}
\lambda_T &\approx \max \left(\min \left(1.69 - 7.50 \cdot 10^{-3} \beta, 1.65 \right), 1.35 \right) \\
\lambda_R &\approx \max \left(\min \left(1.40 + 5.22 \cdot 10^{-3} \beta + 54.8 \cdot 10^{-3} \rho^*, 2.1 \right), 1.5 \right) \\
c_{D,Tan} &\approx \max \left(\min \left(2.85 - 59.8 \cdot 10^{-3} \beta, 2.25 \right), 0.45 \right)
\end{aligned} \tag{8.15}$$

As seen in Equation 8.15, the variables are only first order fitted, as higher order fits were not found suitable for the given data set. Also, the fits have been clamped to maximum and minimum values. These fits will be used in *Section 8.6 - Dynamic investigation*, where the dynamics of the PT-model will be analysed.

The average values for λ_T and λ_R were ≈ 1.39 and ≈ 1.77 respectively, which were found by evaluating the expressions in Equation 8.15 at $\beta = 40$ and $\rho^* = 3$.

8.4.1 Validity of PT-model

The reason for most values of λ in Table 8.2 being larger than unity was assumed caused by effects at the turning points. Vortices were shed and the boundary layer was being disturbed by the wake field. Such effects were not taken into account in the PT-model. An example of the flow field at a turning point can be seen in Figure 8.7 for a plate with dimensions $40 \text{ mm} \times 2 \text{ mm}$ and $\rho_p = 3000 \text{ kg/m}^3$.

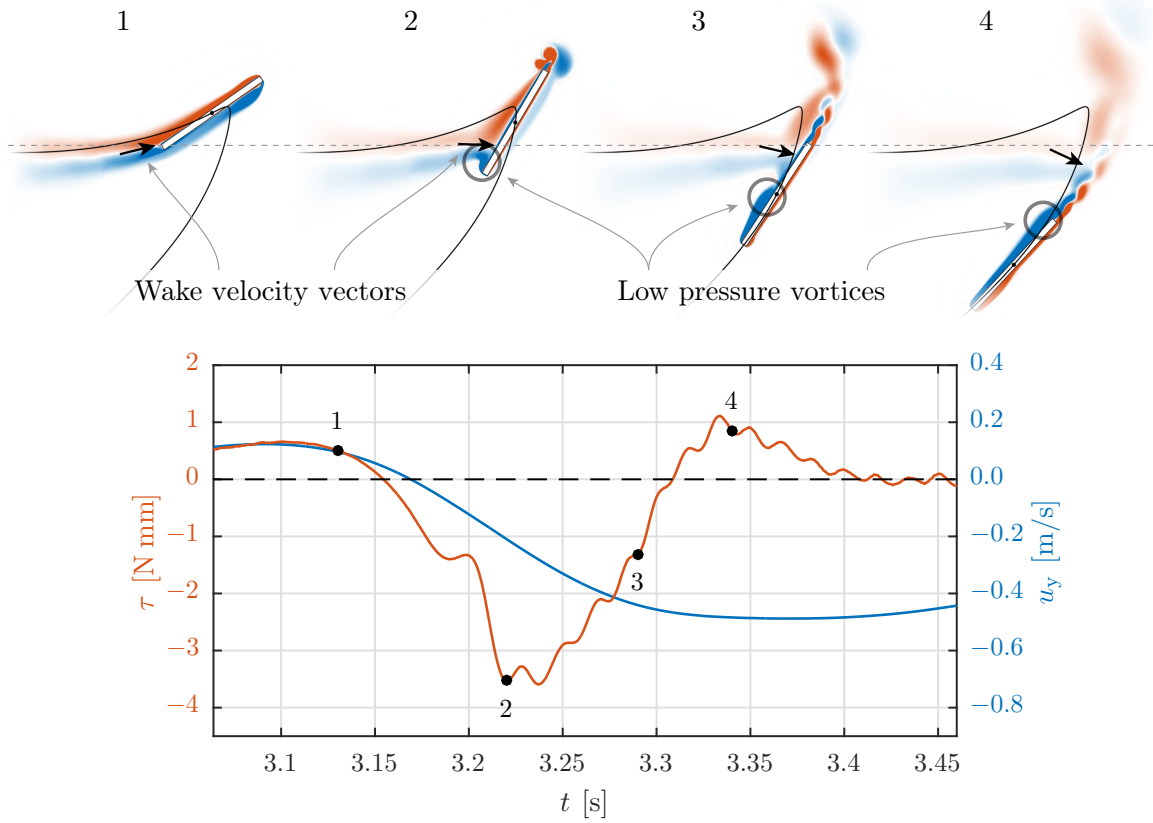


Figure 8.7: The torque and vertical velocity at four instances during a turning point. The upper images are indicating the vorticity, $(\nabla \times \vec{u})_z$, where blue, white and red indicate -75 , 0 and 75 s^{-1} respectively. The black curve is the trajectory of the plate.

As seen in Figure 8.7, vortices were constantly shed in the wake of the plate. This was especially apparent at low velocities, which can be seen at instance 3 and 4. The vortices were affecting the torque as well, which is seen as the red line in the lower figure. At instance 1, the wake was following the trailing edge of the plate, but as the plate slowed down, the wake collided with the plate. This caused torque, which is apparent in the lower figure, when $u_y > 0$ and $\tau < 0$. By investigating the wake velocity centre, which was located at $(\nabla \times \vec{u})_z = 0$, the y-location almost remains constant in the global coordinate system over time. This has been indicated by the horizontal dashed lines in the upper figure of Figure 8.7.

When the vertical velocity switched sign, $u_y < 0$, a low pressure vortex started to built-up at the leading edge, which was now going downwards. This has been shown at instance 2, 3 and 4 as the shaded circles. The vortex centre location causes a significant torque contribution, which was apparent in the lower figure. When the torque went from negative to positive, the vortex centre was at the middle of the plate, which was indicating, that it had a significant contribution to the overall torque.

The above mentioned effects were not taken into account in the PT-model, but despite of this, the results in Figure 8.5 are matching well. The mentioned effects are clear to see in the figures labelled 6 in Figure 8.5 indicating $\dot{\theta}$, where large peaks are present. These were caused by the vortices and the wake, where the change in sign is present as well in the figure. Based on the obtained results, it was concluded, that the expression for the circulation shown in Equation 8.12 was compensating for the effects at the turning points. This was probably the cause for the non-unity values of λ_T and λ_R . However, the results shown in Figure 8.5 were from two periodic falling plates, where the effects in the

tumbling region were much more complex and more difficult to identify. This could also be identified in the optimization routine, where the tuned values of λ did converge, though the trajectories were not as well aligned with the CFD-simulations compared with the periodic ones.

Based on the analysis of the PT-model, it could be concluded that the model was capable of accurately predicting the trajectories for periodic falling plates. However, tuning of the circulation coefficients was needed in order to compensate for the effects at the turning points. When the plates were tumbling, the PT-model was able to predict the trajectories for most geometries and densities, though some cases could not be replicated with the PT-model. This was assumed caused by the highly complex flow phenomena for tumbling plates, which the model did not take into account and was unable to compensate for. As of such, it was assumed that additional terms were required to model these effects. However, a general expression, which makes physical sense, was difficult to obtain.

8.5 Modified PT-model

As mentioned in *Chapter 3 - Governing forces*, history forces can play a significant role in the dynamics of a rigid body. As of such, an alternative method of imposing the history force was introduced. This method assumes that the wake had a velocity, which was tracing the plate velocity with a given time constant. This has been shown in Equation 8.16. Notice that τ is denoting time constants and not torques in Equation 8.16.

$$\begin{aligned}
\vec{u}_{\text{Ref}} &= \vec{u} \cdot S_u \\
\frac{\partial \vec{u}_W}{\partial t} &= \frac{\vec{u}_{\text{Ref}} - \vec{u}_W}{\tau_W} \\
\vec{u}_{\text{Rel}} &= \vec{u} - \vec{u}_W \cdot \Phi \\
\Phi_{\text{Ref}} &= \min \left(\left(1 - \frac{\vec{n}_W \cdot \vec{n}_u + 1}{2} \right) \cdot S_\Phi, 1 \right) \\
\frac{\partial \Phi}{\partial t} &= \frac{\Phi_{\text{Ref}} - \Phi}{\tau_\Phi}
\end{aligned} \tag{8.16}$$

As seen in Equation 8.16, the reference velocity for the wake, \vec{u}_{Ref} , is a fraction of the plate velocity, \vec{u} , where S_u is a scalar between 0 and 1. The second equation states the derivative of \vec{u}_W with respect to time, where a time constant, τ_W , is present. This equation can be seen as a simple P-controller. When \vec{u}_W was known, the relative velocity seen by the plate could be calculated, \vec{u}_{Rel} , which is shown in the third equation. Basically, the wake velocity was subtracted from the plate velocity by use of a blending function, Φ . This function ensured that the wake velocity was only affecting the relative velocity, when the plate was moving into its own wake. The blending function is shown in the fourth equation, where S_Φ is a scalar. Furthermore, the blending function was implemented as a differential equation, which dampened the effects of this term, which was found advantageous. This was implemented with a simple time constant, τ_Φ , as well. The tendency of Φ can be seen in Figure 8.8.

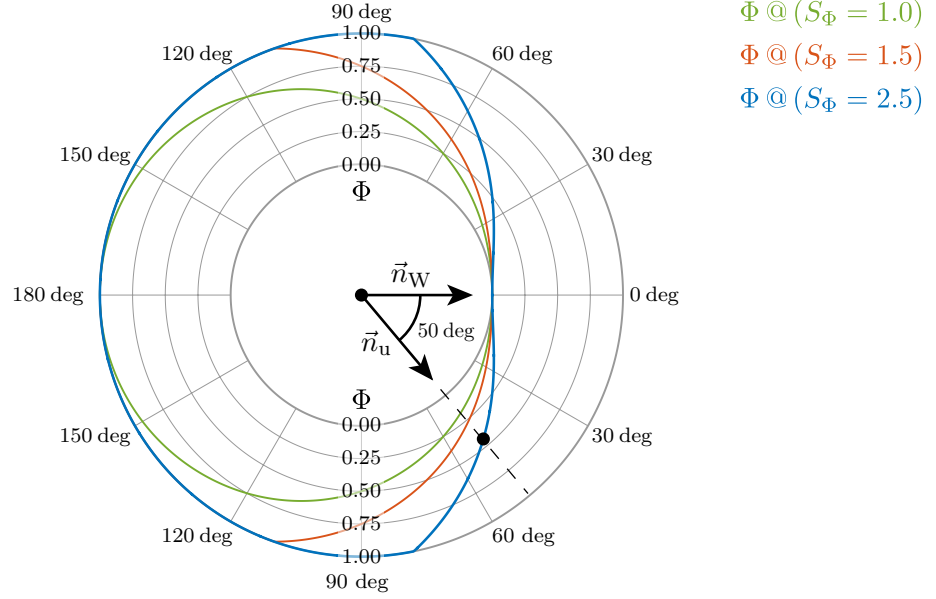


Figure 8.8: The method of blending the wake velocity influence as a function of the relative angle between the two normal vectors obtained from \vec{u}_W and \vec{u} , \vec{n}_W and \vec{n}_u respectively.

As seen in Figure 8.8, the blending function, Φ , can be tuned by varying S_Φ . An example has been given, where the angle between the two normal vectors is 50 deg and $S_\Phi = 2.5$, which results in $\Phi = 0.45$. As of such, only 45 % of \vec{u}_W will be subtracted from \vec{u} in the given example. Furthermore, when the two velocity vectors are aligned, $S_\Phi = 0$, which makes physical sense, since the wake is lagging behind the plate.

It was found that by implementing the modifications shown in Equation 8.16, the PT-model could be improved for specific cases. This was found by including S_u , τ_W , S_Φ and τ_Φ in the optimization routines and tuning these constants along with the circulations coefficients, C_T and C_R , and the tangential drag coefficient, $c_{D,Tan}$. The results can be seen in Table 8.3.

Table 8.3: Tuned coefficients for a plate with dimensions 80 mm \times 2 mm and mixed densities of 2000, 4000, 6000 and 8000 kg/m³. The rows are indicating the unmodified and the modified PT-model.

	λ_T	λ_R	$c_{D,Tan}$	τ_W	S_u	τ_Φ	S_Φ
Unmodified	1.36	1.95	0.56	—	—	—	—
Modified	1.05	1.85	1.02	0.204 s	0.19	0.054 s	2.95

Notice that the values of λ_T , λ_R and $c_{D,Tan}$ for the unmodified PT-model are the same as in Table 8.2. As seen in Table 8.3, the modifications shown in Equation 8.16 altered the optimal values of the tuning coefficients in λ_T , λ_R and $c_{D,Tan}$. The two latter did not change significantly, but λ_T did. This value is almost unity, which is yielding $C_{T,Tuned} \approx C_{T,Panel}$ ($\beta = 40$). As of such, the modifications improved the PT-model in the sense, that the translational circulation coefficient could be used directly from the panel methods without tuning. However, four new coefficients were introduced, which could not be determined without tuning. Nonetheless, the method of developing new terms has been proposed, which might inspire further studies with analogies to the actual phenomena, where a lagging wake component is introduced.

Numerous other modifications were implemented as well, where some were improving the resemblance with the CFD-simulations in other regimes. However, these have not been shown in the report, but the results suggests, that the set of equations in the unmodified

PT-model are not completely capable of modelling the trajectories throughout all motion regimes.

As the PT-model has been developed and tuned to match the CFD-simulations as a function of ρ^* and β , a dynamic investigation of it can be carried out. This has been described in the next section. It should be noted that the dynamic investigation is based on the unmodified PT-model.

8.6 Dynamic investigation

As a PT-model capable of predicting the trajectories of a given plate had been developed, the dynamics of the model could be investigated. The same dimensionless groups as discussed in *Chapter 1 - Introduction* were used, which were Re and I^* . The motion classification as a function of these groups were explored by varying ρ_p and β from $10^1 - 10^4 \text{ kg/m}^3$ and $5 - 100$ with 250 and 100 steps respectively, which were spaced in a logarithmic manner. In total, 25000 simulations were carried out, which took $\approx 15 \text{ min}$ to compute due to the fast computational time of the PT-model. The multiplication factors on C_T and C_R , λ_T and λ_R respectively, along with the tangential drag coefficient, $c_{D,Tan}$, were calculated using the fitted expressions in Equation 8.15. The results can be seen in Figure 8.9, where different motion regimes have been outlined.

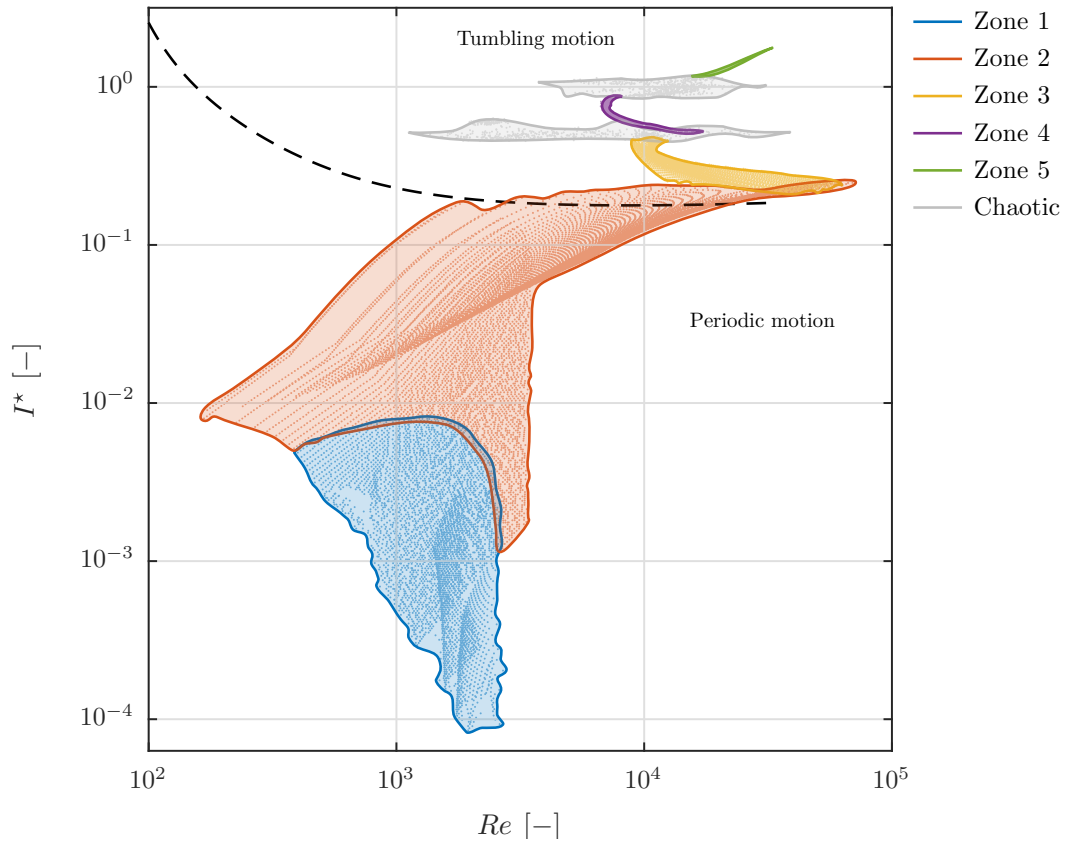


Figure 8.9: The motion regime obtained from the modified PT-model. Each dot is a simulation, where the different zones are related to Figure 8.10.

As seen in Figure 8.9, multiple zones exist with characteristic motions. Zone 1 and 2 are periodic, whereas the others are tumbling. The dashed line is the same as shown in Figure 1.2 in *Chapter 1 - Introduction*, which is the bifurcation between periodic and tumbling motion obtained from reference [Smith, 1970]. It can be seen that the PT-model is capable of capturing the bifurcation between periodic and tumbling motion, although

zone 2 is slightly extending into the tumbling regime. The grey zones in Figure 8.10 were chaotic and irregular, where not characteristic motion was apparent. The motion zones have been shown in Figure 8.10.

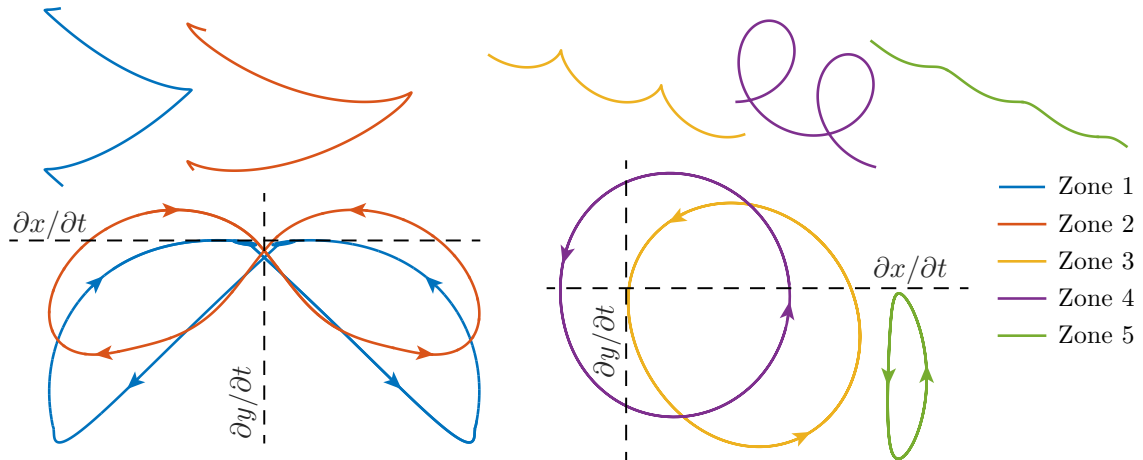


Figure 8.10: The various characteristic motion regimes from the PT-model. The upper part is indicating the trajectory and the lower part the horizontal and vertical velocities. Both coordinate systems are increasing towards the upper right corner.

The five characteristic motion classifications have been shown in Figure 8.10, where the upper part is showing part of the trajectory, and the lower part is the horizontal and vertical velocities. It should be noted that not all motions in Figure 8.10 were found in the CFD-simulations, though they might exist. Also, the PT-model was able to model the steady falling motion, though the viscosity of the fluid had to be increased compared with the used value for water of $\mu = 1.003 \cdot 10^{-3}$ Pa s. The zones have been defined as shown in the list below:

- Zone 1:
 - Periodic motion.
 - Constant vertical velocity sign, thus no elevation at turning points.
- Zone 2:
 - Periodic motion.
 - Alternating vertical velocity sign, thus elevation at turning points.
- Zone 3:
 - Tumbling motion.
 - Constant horizontal velocity sign and alternating vertical velocity sign.
 - Cusps in trajectory.
- Zone 4:
 - Tumbling motion.
 - Alternating horizontal velocity sign.
- Zone 5:
 - Tumbling motion.
 - Constant horizontal velocity signs (The vertical velocity sign might alter).
 - No cusps in trajectory.

By classifying the motions using the velocity signs, clear distinction can be made when analysing the trajectories. The regime map shown in Figure 8.9 has Re along the x-axis, but this value will change, if, for instance, either L , ρ_f or \vec{g} is changed. For instance, it follows that by doubling L , the velocity will approximately double as well, which will quadruple Re . As of such, Re is not independent of changes in these variables. As of such,

the mismatch in Figure 8.9 between the bifurcation lines and the PT-model might be, by part, caused by the physical plate size.

This study serves to make a general classification of falling objects, so a new dimensionless group is proposed. This group is independent in changes in the above mentioned variables, as ratios are used instead. The definition of the new dimensionless group can be seen in Equation 8.17, which has been denoted Si .

$$Si = \frac{u f H \rho_f}{g L \rho_p} = \frac{u f}{g \beta \rho^*} \quad (8.17)$$

Si is the ratio between the average acceleration and gravity magnitude, $(u f)/g$, divided by $\rho^* \beta$. f is the frequency of oscillations, and is defined as $f = \Delta t^{-1}$, where Δt is the same as obtained in Section 5.3 for periodic motion, whereas it is defined as $\Delta t = (2\pi)/\dot{\theta}$ for tumbling motions. The characteristic velocity, u , defined in the same manner as stated in Section 1.2 - *Trajectory characterization*. By remapping the x-axis in Figure 8.9 from Re to Si , a new regime map is obtained. This has been shown in Figure 8.11.

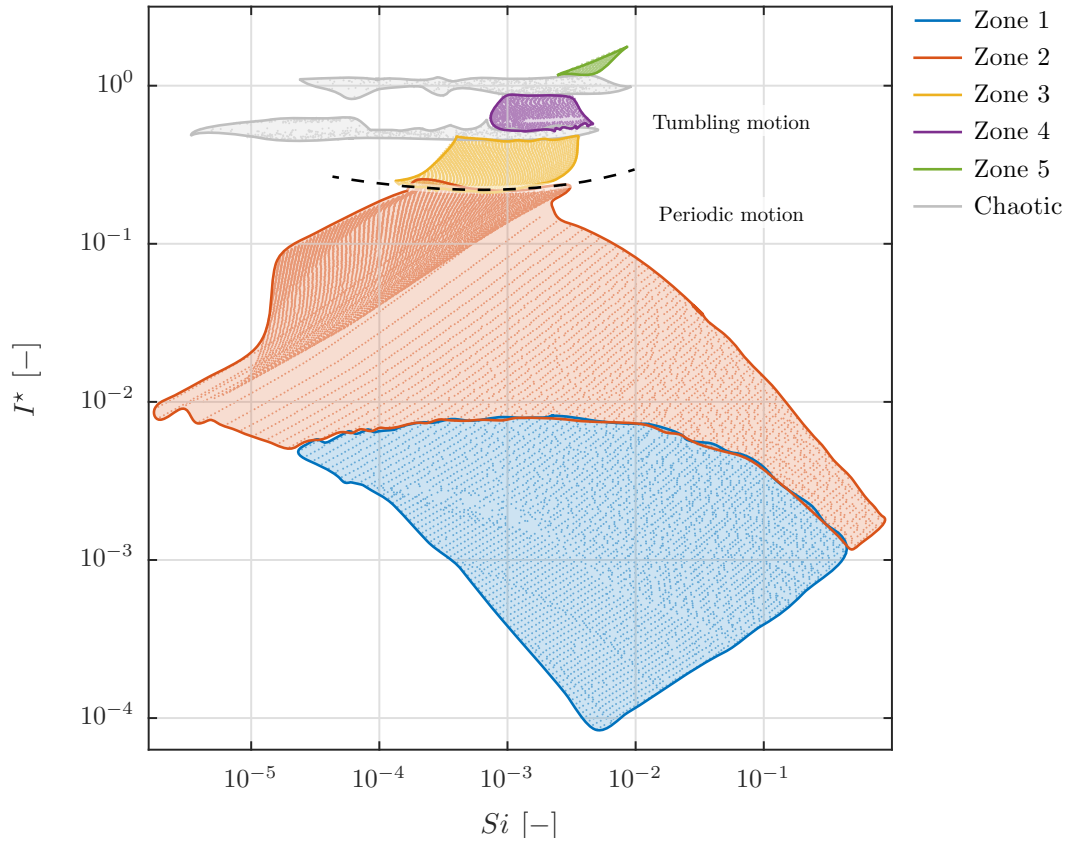


Figure 8.11: The motion regime obtained from the modified PT-model, when shown as a function of Si and I^* . Each data point is assigned to the same zone as in Figure 8.9.

As seen in Figure 8.11, the distinctions between the different zones are more clearly defined. Furthermore, no overlapping is present between the different motion zones as it was in Figure 8.9.

8.7 Partial conclusion

A *Predictive-Trajectory model*, PT-model, was developed to predict the trajectory for a given plate defined by L , H and ρ_p in a fluid with properties ρ_f and μ . The PT-model was based on *Kirchhoff's equations*, which were combined with external forces. The resulting coupled differential equations relied on various constants, where some of these were calculated using panel methods. Three added mass coefficients, C_{11} , C_{22} and C_{66} , and two circulation coefficients, C_T and C_R , were obtained as function of the aspect ratio of the plate, β , which were verified by comparison with other studies along with the *thin airfoil theory*. However, it was found that the circulation coefficients and the resistance torque drag coefficient had to be tuned in order to obtain matching trajectories with the CFD-simulations. As of such, an optimization tool was developed, which could tune the coefficients based on an objective function taking various aspects into account.

The optimization routine yielded that the circulation coefficients had to be tuned to ≈ 1.39 and ≈ 1.77 for λ_T and λ_R respectively, in order to obtain trajectories matching CFD-simulations under the same conditions. By surface integrating the circulation for two CFD-simulations with varying densities, a tendency was found. The simulation with the larger plate density experienced significantly larger circulation when exiting a turning point compared to the simulation with the lower plate density. The flow field was investigated at a turning point, where it was found that low pressure vortices were created at the leading edge, which caused significant torque. Furthermore, the wake field was colliding with the plate as it slowed down for the turning point, which also affected the torque.

A modified version of the PT-model was proposed, which took the wake field into account. This was done by assigning a velocity vector to the wake field, which was lagging the plate velocity by a given time constant. A blending function was introduced, so the wake velocity vector was only affecting the plate, when it was entering its own wake. It was found that these modifications could be tuned to relieve the translational circulation coefficient, so it did not need tuning. However, the findings was based on a single plate geometry, and several new parameters were introduced in the process, which had to be tuned. However, the process of introducing new terms was proposed, which might inspire similar studies.

The PT-model was investigated by varying various parameters and analysing the trajectories. It was found that the PT-model was capable of capturing the bifurcation between periodic and tumbling motion as found by reference [Smith, 1970], though some inconsistency existed. The regime map from the mentioned reference was based on Re , but it was found that this dimensionless group was not independent of several parameters. As of such, a new dimensionless group, Si , was proposed, which was independent of the most dominant parameters regarding the phenomena. A new regime map was constructed by remapping the data points obtained with the PT-model, where the new map revealed clear distinction between the different motion classifications. In total, five characteristic motions were found in the PT-model, though not all of them were found using the CFD-model.

9 | CLOSURE

9.1 Resume

This study has investigated the nature behind freely falling plates by an experimental and numerical approach. A *Computational Fluid Dynamics*, CFD-, model was developed in order to obtain trajectories for various plate geometries and densities, when the flow field around the plate was resolved. By doing so, effects such as vortices and boundary layers built-up are included and affecting the trajectories. The CFD-model was developed in *ANSYS Fluent*, where the flow domain was defined as a circle around the plate, where a diameter of $15 L$, where L was the length of the plate of interest, was found adequate to capture the main fluid effects while keeping the cell count down. The domain was meshed in *MATLAB*, where a set of functions were written for this purpose. By doing so, an automatic meshing routine could be formed, which allowed for fast mesh generation and manually parallel partitioning of the cells. The *MATLAB* meshing routine was rewritten for 3D mesh generation as well, which was done to investigate 3D effects such as tip vortices.

In order to investigate a wide range of plate geometries, the CFD-model had to be computationally fast to execute. The default *6 degree of freedom*, 6DOF, solver in *ANSYS Fluent* was found to be slow and inconsistent, so a new 6DOF-solver was written in C and coupled with *ANSYS Fluent* by use of *User Defined Functions*, UDF's. The major advantage of doing so was the possibility to manually assign the iteration numbers, at which the mesh should be updated in position and velocity. A speed up of ≈ 10 was gained along with improved stability compared to the default 6DOF-solver. For modelling turbulence, the Transition SST-model was used, which is capable of predicting wall shear stresses accurately, which is critical for freely falling plates. The complete CFD-model was capable of simulating 5 s of free fall in ≈ 7 hours, which was significantly faster compared with other studies.

The CFD-model was verified by carrying out a set of experiments, where five different plate geometries were tested. The trajectories were recorded using a high-speed camera, after which they were statistically processed. The results were compared with CFD-simulations of similar plate geometries and densities, where the results yielded some deviations. As of such, 3D CFD-simulations were carried out in order to investigate 3D effects such as tip effects. These results matched the experimental trajectories better compared with the 2D results, so it was concluded that the deviations in the 2D model was being caused by missing 3D effects. As a result, it was concluded that the 2D model was verified for large aspect ratios in the direction normal to the trajectory plane.

As the CFD-model was verified, an extensive parametric study was carried out, which was possible due to the fast execution time of the CFD-model. In total, 36 simulation were carried out, where the density of six different plate geometries was varied. The results showed that the CFD-model was capable of predicting whether the motion was periodic or tumbling, when compared with experimental results from other references. Furthermore, the resolved flow fields indicated the complex flow structure governing plates in free fall. Especially tumbling plates experiences complex vortex structures, whereas the flow field for periodic motion was more predictable.

A *Predictive Trajectory model*, PT-model, was developed based on *Kirchhoff's Equations*, which can predict the motion of a rigid body in an inviscid and irrotational flow. The model, when using this approach, consisted of three coupled differential equations describing the position and orientation as a function of time. The forces on the plate were primarily

modelled using the circulation, Γ , around the plate, which had a translational and a rotational contribution to it. Two coefficients were needed to complete the expression for Γ , which were found using panel methods. Such methods are used to solve for the potential flow around a given object by super positioning basic solutions to *Laplace's equation*, while imposing zero normal velocity on the body. By imposing the *Kutta-Joukowski condition*, the circulation around a given plate could be found, which allowed for numerical values for the circulation coefficients. The panel methods were also used to determine the added mass coefficients. By including the coefficients obtained from the panel methods in the PT-model, the differential equations could be solved to yield the motion of any given plate. However, it was found that the circulation coefficients and a drag coefficient related to the resistance torque had to be tuned in order to obtain similar trajectories to the CFD-simulations carried out in the parametric analysis. As of such, an optimization tool was developed in *MATLAB*, which could tune given coefficient to yield the best match with the CFD-simulations. The optimization routine used an objective function taking both position and orientation into account along with the first and second derivatives of these variables with respect to time.

The results from the optimization routine yielded accurate trajectories when compared to the CFD-simulations. However, the resulting translational and rotational circulation coefficient, C_T and C_R , were tuned to ≈ 1.39 and ≈ 1.77 times the values obtained from the panel methods respectively, which indicated that the PT-model was missing some terms in order to completely capture the complex flow phenomena governing the freely falling plates. As of such, the model was modified to include effects from the wake of the plate. This was done by implementing a wake velocity vector, which was lagging behind the plate velocity vector by use of a simple time constant and a proportional controller. A blending function was introduced, which ensured that the wake did only affect the plate velocity, when it was entering its own wake at the turning points. When implemented in the model and tuned to match the CFD-simulations, the new multipliers on C_T was 1.05. This indicated that by including these modifications to the PT-model, the value of the translational circulation coefficient, C_T , obtained from the panel methods could be used directly without tuning, as the optimal multiplication coefficients of 1.05 was almost unity. However, when including the extra terms, new coefficients were introduced to the model, which had to be tuned, as these could not be obtained from the panel methods. However, the process of imposing nature-like phenomena like the wake field was introduced, which might inspire new studies within this area.

As an accurate PT-model had been developed, the dynamics of it could be analysed. This was done by varying various parameters in the model and analysing the obtained trajectories. The results showed good resemblance with other references, which verified the model in some extent. Five characteristic motion regimes were identified, which used the signs of the horizontal and vertical velocities for identification. By classifying the trajectories in this manner, various regimes could be identified by use of the *Reynolds number*, Re , and the dimensionless moment of inertial, I^* . However, the motion regimes were found to be overlapping, when plotted using these groups. Furthermore, it was found that the use of Re for motion classification had some drawbacks, as this group was not independent of several important parameters governing the phenomena. As of such, a new dimensionless group was proposed for classifying freely falling objects, which was denoted, Si , and proved to clearly separate the motion regimes, when plotted against I^* . Furthermore, the definition of Si was made to be independent of gravity, fluid density and plate length, which made it a more suitable group for classification. The definition of Si can be seen after the conclusion, where u and f are the characteristic velocity and frequency of oscillation respectively, g the gravity magnitude, β the aspect ratio of the plate and ρ^* the density ratio between the plate and the fluid.

9.2 Conclusion

This study has investigated freely falling plates by an experimental and numerical approach. A CFD-model was developed in *ANSYS Fluent*, where the fluid domain was moving and rotating along with the plate as it was falling, thus eliminating the use of re-meshing. A 6DOF-solver was programmed, which used a modified four-point *Adam-Moulton* linear multistep method for integrating the accelerations, which was coupled with the solver in *ANSYS Fluent* by use of *User Defined Functions*. This method proved to reduce the computational time by a factor of ≈ 10 compared with the default solver.

The CFD-model was verified by comparison with experiments for five different plates, where the results indicated $\approx 10\%$ deviation. By carrying out 3D CFD-simulations it was found that the deviations in the 2D CFD-model were caused by 3D effects such as tip vortices. As of such, the 2D CFD-model was concluded verified for large aspect ratios in the direction normal to the trajectory plane. A parametric study was carried out with the CFD-model, where 36 simulations were carried out, which the computationally fast CFD-model allowed for. The plate motions were found to be periodic and tumbling, where the bifurcation between these motions was in agreement with other studies.

A *Predictive Trajectory model*, PT-model, was developed, which could predict the motion for a given plate with specified density in a fluid defined with a density and a viscosity. This model was partly based on inviscid and irrotational flow, where panel methods were utilized to obtain added mass and circulation coefficients by use of potential flow theory. The circulation was modelled by use of two terms, each having a coefficient. The terms were a translational and a rotational, where the coefficients were denoted C_T and C_R . By evaluating the PT-model for similar cases as the CFD-simulations carried out in the parametric study, the validity of the model was evaluated. It was found that the circulation coefficients should be tuned to ≈ 1.39 and ≈ 1.77 times the values obtained from the panel methods for C_T and C_R respectively. It was concluded that the non-unity values were caused by the terms compensating for missing circulation terms, which were needed to correctly model the turning points of the trajectories. After the circulation coefficients were tuned, the PT-model was found to accurately predict the trajectories for a wide range of plates with different densities.

The dynamics of the PT-model was investigated by varying different parameters and analysing the motion. The results yielded five distinctive motion classification, two periodic and three tumbling. These were plotted as a function of Re and I^* , where the latter being the dimensionless moment of inertia. However, some motion regimes were found to be overlapping, and in combination with Re not being independent of important parameters governing freely falling objects, a new dimensionless group was proposed. This group was denoted Si and was independent of the most important parameters. When plotting the motion classifications as a function of Si instead of Re , clear bifurcation between the motions was found. As of such, it was concluded that Si is a more suitable dimensionless group for characterizing the motion of freely falling objects.

$$Si = \frac{u f}{g \beta \rho^*}$$

10 | FUTURE WORK

This chapter discusses suggestions for further work.

Verify CFD- and PT-model in more motion regimes

This study conducted experiments with five plates, which experienced periodic and tumbling motion. However, the results from the PT-model suggests, that five characteristic motion regimes are present, or six if steady falling is included. As of such, more experiments should be carried out in different fluids with varying densities along with more plates. Also, experiments should be conducted for similar plates, which only varied in the z -direction, which could reveal the influence of the tip effects.

Implementation of dynamic integration procedure

The default 6DOF-solver in *ANSYS Fluent* proved to be slow and inconsistent in this study. As of such, a new integration procedure was coded, which proved more stable and much faster. This procedure should be expanded for more general applications and incorporated in *ANSYS Fluent*. Especially the method of specifying the iterations, at which the mesh should be updated, which was found to be the crucial part of the procedure.

Resistance torque drag coefficient calculations

The PT-model relied on various coefficients, where the resistance torque drag coefficient, $c_{D,Tan}$, was one of them. This coefficients should be derived in a more general manner, so it can be eliminated in the optimization procedure, thus yielding a more general PT-model.

Implementation of PT-model in CFD-software

As an accurate PT-model had been developed, it should be implemented in CFD-software packages. By doing so, accurate and fast trajectories could be obtained, where the resulting flow field could be calculated without the need for solving the motions of the object. Furthermore, general correlations could be found for a stream of rectangular shaped particles, which could be obtained by inclusion of interaction between the rectangles. This could prove useful for, for instance, straw-fired boilers.

Panel method implementation in PT-model

The PT-model was modified to be affected by its own wake. This method proved successful for some cases, but a more general applicable expression should be developed. One promising implementation could be to incorporate the panel methods in the PT-model. By doing so, the plate could be programmed to shed vortices continuously from the trailing edge, as it was seen doing in the CFD-simulations. These vortices should be assigned with a vortex strength and a dissipation rate. At each time-step, the influence of all the vortices on the plate could be found, which would alter the right hand side of the vortex-source panel method linear matrix system, which could be solved to yield the overall required circulation to maintain the *Kutta-condition*. This would yield history forces, where the plate is affected by its previous trajectory, even in cases where it is experiencing chaotic motion. This method has been illustrated in Figure 10.1.

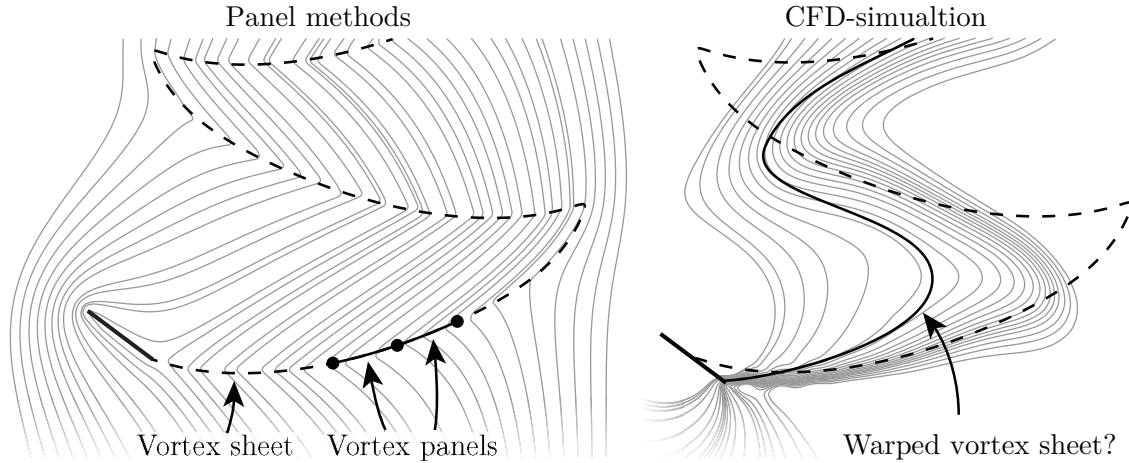


Figure 10.1: An example of implementing panel methods into the PT-model to the left. The grey lines are streamlines and the dashed are the trajectories.

To the left in Figure 10.1, the existing source-vortex panel method with enforced *Kutta-condition* has been shown along with a vortex sheet trailing after the plate. This sheet is constructed from a number of vortex panels, which are inducing velocity components on the plate panels. As of such, the circulation on the plate is altered in order to compensate for the increased velocities. The right image is made from a CFD-simulation, where streamlines have been overlaid. The same streamline tendency can be observed, where they are twisting upstream. The image for the panel method has been constructed by using the trajectory as the vortex sheet, but if implemented, the vortex sheet should not remain fixed in space. Instead, it should be affected by itself and the plate panels, which would warp its shape. One possibility is that the warped vortex sheet would take a shape similar to the suggested one in the CFD-simulation indicated by the solid line. However, as a linear system of equations is to be solved at every time-step, the computational time would increase significantly, and the tuning of the dissipation rate and the vortex strengths on the sheet itself would be time-consuming. As of such, it was not investigated in this study. However, it might prove to be the missing link for modelling accurate trajectories throughout all motion regimes.

BIBLIOGRAPHY

- Andersen, A., Pesavento, U., Wang, Z. J., 2005a. Analysis of transitions between fluttering, tumbling and steady descent of falling cards. *Journal of Fluid Mechanics*, Vol. 541, pp. 91-104.
- Andersen, A., Pesavento, U., Wang, Z. J., 2005b. Unsteady aerodynamics of fluttering and tumbling plates. *Journal of Fluid Mechanics*, vol. 541, pp. 65-90.
- Anderson, J. D., 1978. *Introduction to flight*. 3rd edition. McGraw-Hill.
- Anderson, J. D., 2001. *Fundamentals of Aerodynamics*. 3rd edition. McGraw-Hill.
- ANSYS, Inc., 2013a. *ANSYS Fluent Theory Guide*. Release 15.0. ANSYS, Inc.
- ANSYS, Inc., 2013b. *ANSYS Fluent UDF Manual*. Release 15.0. ANSYS, Inc.
- ANSYS, Inc., 2013c. *ANSYS Fluent User's Guide*. Release 15.0. ANSYS, Inc.
- Autodesk, 2014. Sst k-omega turbulence models.
<http://help.autodesk.com/cloudhelp/2014/ITA/SimCFD/files/GUID-0F5C4828-9F91-46B6-A16A-2578D72DCFCC.htm>, accessed 23-02-2015.
- Bakker, A., 2006a. Flow fields in applied computational fluid dynamics.
<http://www.bakker.org/dartmouth06/engs150/02-post.pdf>, accessed 02-03-2015.
- Bakker, A., 2006b. Meshing in applied computational fluid dynamics.
<http://www.bakker.org/dartmouth06/engs150/07-mesh.pdf>, accessed 23-02-2015.
- Brennen, C. E., 1982. A review of added mass and fluid inertial forces. *Naval Civil Engineering Laboratory*.
- Brennen, C. E., 2005. *Fundamentals of Multiphase Flows*. Cambridge University Press.
- Casey, M., Wintergerste, T., 2000. *Best Practice Guidelines*. Version 1.0. ERCOFTAC.
- Field, S. B., Klaus, M., Moore, M. G., Nori, F., 1997. Chaotic dynamics of falling disks. *Nature*, Vol. 388, pp. 252-254.
- Flandro, G. A., McMahon, H. M., Roach, R. L., 2012. *Basic Aerodynamics*. Cambridge University Press.
- Hairer, E., Nørsett, S. P., Wanner, G., 1992. *Solving Ordinary Differential Equations I*. 2nd edition. Springer.
- Jensen, A. L., Hærvig, J., 2014. Extending the existing modelling framework for non-spherical particles to include flat plates in free fall.
- Jin, C., Xu, K., 2007. Numerical study of the unsteady aerodynamics of freely falling plates. *Communications in Computational Physics*, Vol. 3, No. 4, pp. 834-951.
- Kakimpa, B., Hargreaves, D. M., Owen, J. S., 2012. An investigation of plate-type windborne debris flight using coupled cfd-rbd models. part ii: Free and constrained flight. *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 111, pp. 104-116.
- Katz, J., Plotkin, A., 1991. *Low-Speed aerodynamics*. International Edition. McGraw-Hill, Inc.
- Kuethe, A. M., Chow, C. Y., 1978. *Foundations of Aerodynamics*. 5th edition. Wiley.

- Ladson, C. L., Brooks, C. W., Hill, A. S., 1996. Computer program to obtain ordinates for naca airfoils. <http://www.pdas.com/refs/tm4741.pdf>, accessed 24-04-2015.
- Lamb, H., 1895. Hydrodynamics. Cambridge University Press.
- Mahadevan, L., Ryu, W. S., Samuel, A. D. T., 1998. Tumbling cards. *Physics Fluids*, Vol. 11, No. 1.
- Mandø, M., Yin, C., Sørensen, H., Rosendahl, L., 2007. On the modelling of motion of non-spherical particles in two-phase flow. 6th International Conference on Multiphase Flow.
- Marshall, J. S., 2001. Inviscid Incompressible Flow. Wiley.
- MathWorks, 2015. Matlab 2015a documentation. <http://www.mathworks.com/help/>, accessed 30-04-2015.
- Maxwell, J. C., 1854. On a particular case of the descent of a heavy body in a resisting medium. *Cambridge and Dublin Mathematical Journal*, Vol. 9, pp. 115-118.
- Moran, J., 1984. An introduction to theoretical and computational aerodynamics. Wiley.
- Munson, B. R., Okiishi, T. H., Huebsch, W. W., Rothmeyer, A. P., 2010. Fluid Mechanics. 7th edition. Wiley.
- OpenCV, 2015. Camera calibration and 3d reconstruction. http://docs.opencv.org/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html, accessed 18-05-2015.
- Pesavento, U., Wang, Z. J., 2004. Falling paper: Navier-stokes solution, model of fluid forces, and center of mass elevation. *Physical Review Letters*, Vol. 93, No. 14.
- Rosendahl, L. A., 1998. Extending the Modelling Framework for Gas-Particle Systems. Aalborg University.
- Shang, Z., 2014. Impact of mesh partitioning methods in cfd for large scale parallel computing. *Computers & Fluids*, Vol. 103, pp. 1-5.
- Smith, E. H., 1970. Autorotating wings: An experimental investigation.
- Snyder, D. O., Koutsavdis, E. K., Anttonen, J. S. R., 2003. Transonic store separation using unstructured cfd with dynamic meshing. 33rd AIAA Fluid Dynamics Conference and Exhibit.
- Suluksna, K., Juntasaro, E., 2008. Assessment of intermittency transport equations for modeling transition in boundary layers subjected to freestream turbulence. *International Journal of Heat and Fluid Flow*, Vol. 29, pp. 48-61.
- Tanabe, Y., Kaneko, K., 1994. Behavior of a falling paper. *Physical Review Letters*, Vol. 73, No. 10, pp. 1372-1377.
- Versteeg, H. K., Malalasekera, W., 2007. An Introduction to Computational Fluid Dynamics. No. 978-0-13-127498-3 in 2nd edition. Pearson.
- Widhalm, M., 2015. Lagrangian trajectory simulation of rotating regular shaped ice particles. SAE International.
- Zastawny, M., Mallouppas, G., Zhao, F., Wachem, B., 2011. Derivation of drag and lift force and torque coefficients for non-spherical particles in flows. *International Journal of Multiphase Flow*, Vol. 39, pp. 227-239.

A | PANEL METHODS

Panel methods are a way to approximate the forces on a given object in a potential flow field, which is steady, inviscid and irrotational. This section will cover potential flow and the properties of such. The solution to the potential flow will be calculated using two different panel methods.

The first panel method will cover the *source panel method*, which will be used to calculate added mass coefficients.

The second panel method will cover the *vortex panel method*. This method will in combination with the *source panel method* be used to calculate circulation coefficient around an object as a function of the relative velocity and the rotational speed of the object.

A.1 Potential flow

Potential flow is defined as being inviscid, incompressible, irrotational and steady [Munson et al., 2010], which has been shown in Equation A.1.

$$\begin{aligned} \mu &= 0 & (\text{Inviscid}) & \qquad \qquad \nabla \times \vec{u} = 0 & (\text{Irrotational}) \\ \nabla \cdot \vec{u} &= 0 & (\text{Incompressible}) & \qquad \qquad \frac{\partial \vec{u}}{\partial t} = 0 & (\text{Steady}) \end{aligned} \tag{A.1}$$

In general, potential flow will yield the flow structure for *Stokes flow* where $Re \ll 1$. This is especially useful for estimating added mass, as the kinetic energy of the flow field relative to the kinetic energy of the object of interest are yielding the dimensionless added mass coefficient. As of such, the objective for this chapter is to describe the method of calculating these coefficients. Furthermore, potential flow can be used to calculate the circulation, Γ , for an object by imposing the *Kutta-Joukowski condition*.

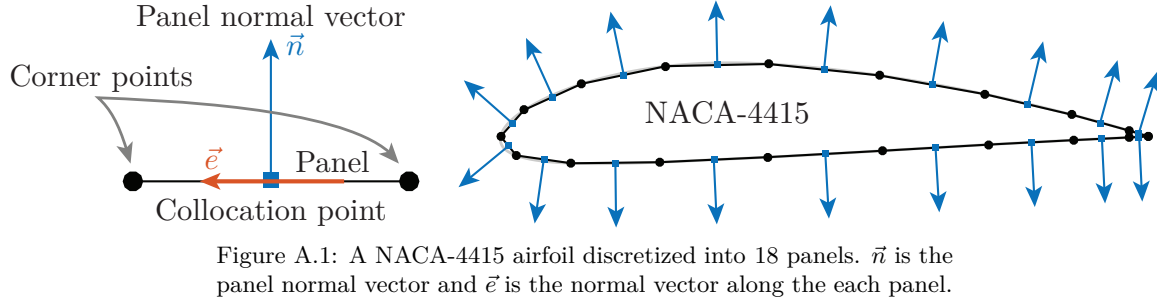
By assuming that Equation A.1 holds true about the flow field, a new scalar can be introduced called the *velocity potential*, ϕ . This scalar is related to the velocity by the spatial gradients, which has been shown in Equation A.2 [Munson et al., 2010].

$$\begin{aligned} \nabla \phi &= \vec{u} \\ \Downarrow (\text{Equation A.1}) \\ \nabla^2 \phi &= 0 \end{aligned} \tag{A.2}$$

The second equation in Equation A.2 is *Laplace's equation*, which can be solved by numerous methods. One such method is by use of panel methods, where, instead of meshing a domain on which to solve the differential equation, the object of interest is discretized into a number of panels, where different basic solutions to *Laplace's equation* can be placed. As *Laplace's equation* is a linear partial differential equation, the contribution of each panel can simply be added by super positioning, after which the final solution can be computed.

A.2 Panel methods

In this section, a NACA-4415 airfoil [Ladson et al., 1996] will be used as an example. Figure A.1 shows an airfoil, which has been discretized into 18 panels.



As seen in Figure A.1, the surface of the airfoil has been divided into 18 corner points shown as black dots, where these are connected into panels indicated by the black lines. For each panel, a collocation point is placed at the midpoint shown as blue squares, where the associated normal vector points outwards of the airfoil has been shown as well as blue vectors. The normal vectors going in the direction of each panel, \vec{e} , has only been shown to the left.

Each panel will be associated with basic solutions to *Laplace's equation*. These solutions will be made from continuous sources and vortices, where the strengths of these are solved to impose zero normal velocity at the collocation points, as no fluid is allowed to enter or exit the airfoil itself. As the basic solutions satisfies *Laplace's equation*, they can be super positioned with each other to form the solution to the potential flow.

The basic solutions to be associated to the panels can be of various types. Each type has some certain properties, where source and vortex panels are used in this study. These have been shown with streamlines and velocity potentials in Figure A.2.

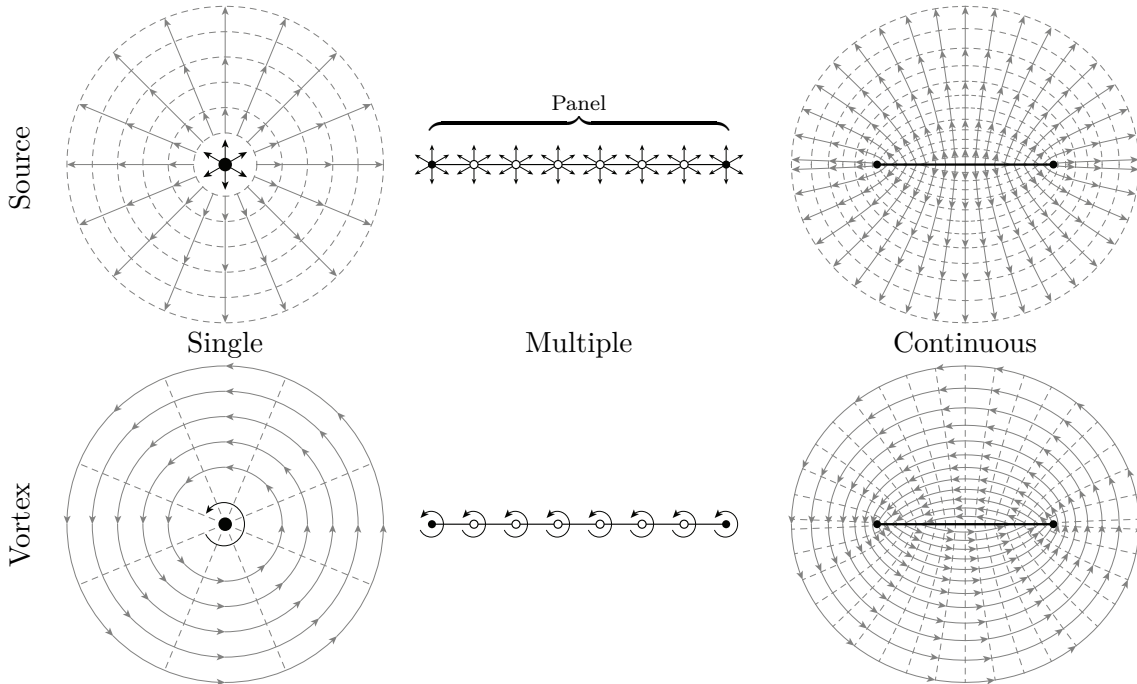


Figure A.2: Characteristics of a source and a vortex shown as a single, as multiple along a panel, and as continuous along a panel. The vectors are indicating streamlines and the dashed lines are indicating iso-lines of the velocity potential, ϕ , which are increasing in the direction of the streamlines.

Figure A.2 is showing sources and vortices at the top and bottom row respectively. The columns are showing a single basic solutions, multiple placed along a single panel, and continuous basic solutions placed along a panel, respectively. The first and third columns labelled *Single* and *Continuous* are showing streamlines with the solid vectors and velocity potential with the dashed lines. The velocity potential is increasing in the direction of

the streamlines. It should be noted that the velocity potential is perpendicular to the streamlines, thus all intersections are having right angles with each other. As of such, an analytical formulation of the streamlines is available, which is called the *stream function*. However, this will not be of interest in this chapter, so it will not be explained in details.

A source can be interpreted as a single point, at which fluid is discharged from. As of such, the streamlines of a source are pointing in the radial direction relative to the source itself. A vortex can be interpreted as a rotational point, where no fluid is discharged from. Instead, the surrounding fluid is rotating around the point, so the streamlines are pointing in the tangential direction relative to the vortex itself. The equations governing these two types of basic solutions are shown in Equation A.3, where the subscripts $()_S$ and $()_V$ are indicating a source and a vortex respectively. The subscripts $()_0$ are indicating the locations of the basic solutions.

$$\begin{aligned} \phi_S &= \left(\frac{\sigma}{2\pi} \right) \ln(r) & \phi_V &= - \left(\frac{\gamma}{2\pi} \right) \text{atan2} \left(\frac{y - y_0}{x - x_0} \right) \\ u_{x,S} &= \frac{\partial \phi_S}{\partial x} = \left(\frac{\sigma}{2\pi} \right) \frac{x - x_0}{r^2} & u_{x,V} &= \frac{\partial \phi_V}{\partial x} = \left(\frac{\gamma}{2\pi} \right) \frac{y - y_0}{r^2} \\ u_{y,S} &= \frac{\partial \phi_S}{\partial y} = \left(\frac{\sigma}{2\pi} \right) \frac{y - y_0}{r^2} & u_{y,V} &= \frac{\partial \phi_V}{\partial y} = \left(\frac{-\gamma}{2\pi} \right) \frac{x - x_0}{r^2} \end{aligned} \quad (\text{A.3})$$

x and y are any given point within the flow field. As the equations for a single source and vortex are known, the continuous forms can be derived by Equation A.4, which apply for the third column of Figure A.2. The capital letters, U and Φ , are indicating values for a continuous panel, in contrast to single points, where u and ϕ was used.

$$\begin{aligned} \Phi_S &= \left(\frac{\sigma}{2\pi} \right) \int_{x_1}^{x_2} \phi_S dx = \left(\frac{\sigma}{2\pi} \right) ((x - x_1) \ln(r_1^2) - (x - x_2) \ln(r_2^2) + 2y(\alpha_2 - \alpha_1)) \\ U_{x,S} &= \int_{x_1}^{x_2} u_{x,S} dx = \left(\frac{\sigma}{4\pi} \right) \ln \left(\frac{r_1^2}{r_2^2} \right) \\ U_{y,S} &= \int_{x_1}^{x_2} u_{y,S} dx = \left(\frac{\sigma}{2\pi} \right) (\alpha_2 - \alpha_1) \\ U_{x,V} &= \int_{x_1}^{x_2} u_{x,V} dx = \left(\frac{\gamma}{2\pi} \right) (\alpha_2 - \alpha_1) \\ U_{y,V} &= \int_{x_1}^{x_2} u_{y,V} dx = \left(\frac{-\gamma}{4\pi} \right) \ln \left(\frac{r_1^2}{r_2^2} \right) \\ \alpha_n &= \text{atan2} \left(\frac{y}{x - x_n} \right), \quad n = 1, 2 \end{aligned} \quad (\text{A.4})$$

Notice that α is not to be confused with the angle of attack in the main report. The orthogonal properties of the two basic solution can be seen as $U_{x,S} = -U_{y,V}$ and $U_{y,S} = U_{x,V}$, though σ and γ differ. The velocity potential for a vortex panel has not been included, as this will not be used. It should be noted that Equation A.4 only holds true, when the panel is oriented horizontally, which has been shown in Figure A.3.

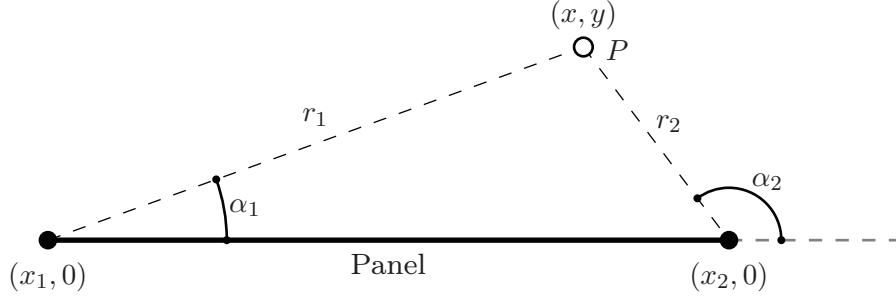


Figure A.3: An example of the local coordinates for a single panel, which is oriented horizontally. This simplifies the equations in Equation A.4 and is used in the actual implementation as well.

As the velocities for both a single source and vortex panel are known, the strengths, σ and γ for a source and vortex panel respectively, can be calculated to yield the potential flow field for a given object. This process has been described in the following subsections.

A.2.1 Source panel method

The source panel method is used to calculate the dimensionless added mass coefficients. As of such, the *Kutta-condition* should not be enforced on the object. This will yield the potential flow around the object when $\Gamma = 0$. The only enforced conditions should thus be, that no fluid is entering or exiting the object. This conditions can be imposed by calculating the influence of all panels on each collocation point. This has been illustrated for the i^{th} collocation point in Equation A.5.

$$\underbrace{u_{\text{Normal},i}}_{b_i} = \vec{n}_i \cdot \vec{u}_\infty = - \sum_{j=1}^{j=N} \underbrace{\left(\vec{n}_i \cdot \vec{U}_S(\sigma_j, x_i - x_{j,1}, y_i - y_{j,1}, x_i - x_{j,2}, y_i - y_{j,2}) \right)}_{A_{i,j}|_{\sigma_j=1}} \quad (\text{A.5})$$

The five parameters going into the function $\vec{U}_{S,j}$ are the strength of the j^{th} source panel, σ_j , and the four relative coordinates at the end between the corner points of the j^{th} panel and the i^{th} collocation point. In order for Equation A.4 to hold true, the coordinate differences are rotated, so the panel is oriented horizontally. As of such, the resulting vector of $\vec{U}_{S,j}$ should thus be transformed back to the global coordinate system before the dot product is taken with respect to the normal vector of the i^{th} collocation point, \vec{n}_i .

By applying Equation A.5 to all collocation points, a matrix can be assembled, where the influence of all panels on all collocation points is present. By isolating the source panel strengths, a full matrix system can be assembled, which has been shown in Equation A.6.

$$\underbrace{\begin{bmatrix} A_{1,1} & A_{1,2} & \cdots & A_{1,N} \\ A_{2,1} & A_{2,2} & \cdots & A_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ A_{N,1} & A_{N,2} & \cdots & A_{N,N} \end{bmatrix}}_{\mathbf{A}_{\text{Source,Normal}}} \cdot \underbrace{\begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_N \end{bmatrix}}_{\vec{\sigma}} = \underbrace{\begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{bmatrix}}_{\vec{b}_{\text{Normal}}} \quad (\text{A.6})$$

For instance, the coefficient $A_{2,5}$ is the influence of the 5^{th} panel on the 2^{nd} collocation point, where b_2 is the normal velocity caused by the surrounding velocity field. By solving the system of linear equations in Equation A.6, the required source strengths to enforce that no fluid is entering or exiting the object can be found. This has been shown for the NACA-4415 airfoil in Figure A.4 along with a plate with $\beta = 3$ for two different flow directions.

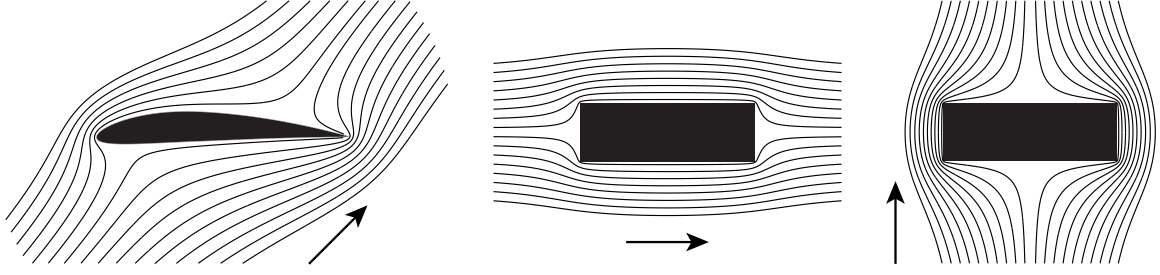


Figure A.4: Examples of the potential flow field obtained by superpositioning the basic solutions from all panels along the surfaces of the objects. The vectors are indicating the relative fluid direction.

The objects seen in Figure A.4 are made from panels, which are located along the surfaces according to Figure A.1. Each panel is made from a continuous source panel, where the strengths of these, σ , are solved according to Equation A.6. The *Kutta-condition* is not enforced, which is clear to see in the streamlines for the airfoil in Figure A.4, where the flow is not following the trailing edge of the airfoil. As this condition is not enforced, the results can be used to calculate the added mass coefficients. This is done by integrating the kinetic energy within the flow field and relating it to the kinetic energy of the object itself, if it had the same density as the surrounding fluid. This can be obtained by evaluating Equation A.7 [Marshall, 2001].

$$\begin{aligned}
 K_f &= -\frac{1}{2} \rho_f \int_S \Phi \frac{\partial \Phi}{\partial n} dA = -\frac{1}{2} \rho_f \sum_{j=1}^N \left((\vec{n}_j \cdot \vec{u}_\infty) \Phi_j \Delta L_j \right) \\
 K_{\text{Obj}} &= \frac{1}{2} \rho_f A (1 \text{ m}) |\vec{u}_\infty|^2 \\
 C_{\text{AM}} &= \frac{K_f}{K_{\text{Obj}}}
 \end{aligned} \tag{A.7}$$

ΔL_j is the length of the j^{th} panel. By evaluating Equation A.7, the added mass coefficient, C_{AM} can be found for different flow directions. For instance, C_{11} is obtained by evaluating Equation A.7 when the flow direction is horizontal as shown in the middle of Figure A.4. In order to verify the obtained added mass coefficients, a plate with aspect ratio 20 is evaluated, where reference [Jensen and Hærvig, 2014] calculated the added mass coefficient by use of CFD. The obtained results are shown in Equation A.8.

$$\mathbf{C}_{\text{AM}} = \underbrace{\begin{bmatrix} 0.0973 & 0 & 0 \\ 0 & 17.09 & 0 \\ 0 & 0 & 6.49 \end{bmatrix}}_{\text{Source panel method}} \quad \mathbf{C}_{\text{AM}} = \underbrace{\begin{bmatrix} 0.0731 & 0 & 0 \\ 0 & 17.3 & 0 \\ 0 & 0 & 7.03 \end{bmatrix}}_{[\text{Jensen and Hærvig, 2014}]} \tag{A.8}$$

As seen in Equation A.8, the added mass coefficients are very similar. Some differences can be seen, but the main difference is, that the panel method will converge to finite values as the number of panels is increased. As of such, the source panel method is assumed more accurate, as the CFD-approach depends on many parameters such as mesh quality, time-step size, speed and many others. Furthermore, the panel method allows for fast calculation of the added mass coefficients for any given geometry, which will improve the PT-model. A converged result will only take a few seconds to calculate for any geometry using the source panel method.

A.2.2 Vortex panel method

The vortex panel method can be used to calculate the circulation for lifting objects. This can be ensured by specifying a point on the object, at which separation should occur. As of such, the *Kutta-condition* can be enforced by specifying the separation point at the trailing edge. However, in order to do so for all angles of attack, circulation should be added to the solution of *Laplace's equation*. A source panel has $\Gamma = 0$, so no circulation is present in any of the objects in Figure A.4. However, by substituting the source panels with vortex panels, circulation can be introduced, and the *Kutta-condition* can be enforced by including this constraint in the system of linear equations.

The vortex panel method differs from the source panel method by imposing zero tangential velocity just below the surface of the object. As of such, the collocation points will be placed just under each panel inside the object. The reason for doing so is the self-induced effect of the vortex panel, where the velocity just over and under the panel have opposite signs. The zero tangential velocity at the collocation points has been shown in Equation A.9.

$$\underbrace{u_{\text{Tan},i} = \vec{e}_i \cdot \vec{u}_\infty}_{b_i} = - \sum_{j=1}^{j=N} \underbrace{\left(\vec{e}_i \cdot \vec{U}_V(\gamma_j, x_i - x_{j,1}, y_i - y_{j,1}, x_i - x_{j,2}, y_i - y_{j,2}) \right)}_{A_{i,j}|_{\gamma_j=1}} \quad (\text{A.9})$$

Here, \vec{e} is the tangential normal vector going from $(x, y)_1$ towards $(x, y)_2$ along the panel, where the subscripts indicate the corner points of the panel. Once again, the function \vec{U}_V is first transformed to the local coordinate system of the panel, where it is horizontally oriented, after which it is transformed back to the global coordinate system, before the influence coefficient, $A_{i,j}$, has been found.

In order to enforce the *Kutta-condition*, the circulation around the trailing edge should be zero. This can be specified by including the constraint that $\gamma_1 + \gamma_N = 0$ [Katz and Plotkin, 1991], where 1 and N are the two panels next to the specified separation point. These can be seen in Figure A.1 as the two panels nearest to the trailing edge on the upper and lower surface. The constraint is included in the system of linear equations, which has been shown in Equation A.10.

$$\underbrace{\begin{bmatrix} A_{1,1} & A_{1,2} & \cdots & A_{1,N-1} & A_{1,N} \\ A_{2,1} & A_{2,2} & \cdots & A_{1,N-1} & A_{2,N} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ A_{N,1} & A_{N,2} & \cdots & A_{1,N-1} & A_{N,N} \\ 1 & 0 & \cdots & 0 & 1 \end{bmatrix}}_{\mathbf{A}_{\text{Vortex,Tan}}} \cdot \underbrace{\begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \gamma_N \end{bmatrix}}_{\vec{\gamma}} = \underbrace{\begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \\ 0 \end{bmatrix}}_{\vec{b}_{\text{Tan}}} \quad (\text{A.10})$$

As seen in Equation A.10, there is $N + 1$ equations, but only N unknowns, so the system is thus overdetermined. Reference [Katz and Plotkin, 1991] suggest deleting an equation, as this will only have a minor effect on the resulting values of γ , if there are enough panels. Another approach was suggested by reference [Moran, 1984], where the overdetermined system of linear equations was solved using the *least-squares method*. However, it was found that the latter method yielded penetrating flow at large rotational velocities, so the first method was implemented in the *MATLAB* code. Two example results have been shown in Figure A.5, where the *Kutta-condition* is enforced.

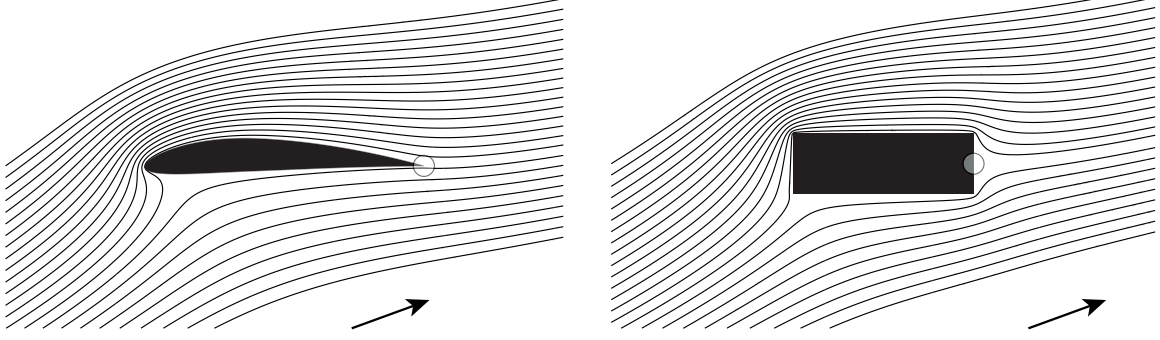


Figure A.5: Two examples of potential flows around the airfoil shown in Figure A.1 and a plate with $\beta = 3$. The surrounding flow is inclined with 20 deg, and the white transparent circles indicate the location, at which the *Kutta-condition* is enforced.

As seen in Figure A.5, the *Kutta-condition* has been enforced by $\gamma_1 + \gamma_N = 0$, where panel 1 and N are located at the white transparent circles. However, the vortex panel method does not ensure zero normal flow velocity at the plate, though it seems so from Figure A.5. In this specific example, it seems to hold true, but for cases with $|\vec{u}| = 0$ and $\dot{\theta} \neq 0$, the zero normal flow velocity cannot be sustained by vortex panels alone. As of such, the source panel method is used in combination with the vortex panel method.

A.2.3 Vortex-source panel method

The source-vortex panel method is combining both previously described methods in order to impose zero normal velocity and zero tangential velocity just below the surface of the plate. Similar to the previous described methods, matrices can be formed yielding the normal velocity component caused by the vortex panels and the tangential velocity components caused by the source panels. These will not be described in details, but instead of multiplying with $\vec{n}_{\text{Normal},i}$ and $\vec{n}_{\text{Tan},i}$ in Equation A.5 and Equation A.9 respectively, the opposite is done. This will yield two new influence matrices, which can be combined with the existing to yield both zero normal and tangential velocity just below the surface. The matrices are combined as stated in Equation A.11.

$$\begin{bmatrix} \mathbf{A}_{\text{Source,Normal}} & \mathbf{A}_{\text{Vortex,Normal}} \\ \mathbf{A}_{\text{Source,Tan}} & \mathbf{A}_{\text{Vortex,Tan}} \end{bmatrix} \cdot \begin{bmatrix} \vec{\sigma} \\ \vec{\gamma} \end{bmatrix} = \begin{bmatrix} \vec{b}_{\text{Normal}} \\ \vec{b}_{\text{Tan}} \\ 0 \end{bmatrix} \quad (\text{A.11})$$

It should be noted that the last row of $\mathbf{A}_{\text{Source,Tan}}$ is zero, whereas the last row of $\mathbf{A}_{\text{Vortex,Normal}}$ is defined as shown in Equation A.10, where the *Kutta-condition* is imposed by $\gamma_1 + \gamma_N = 0$. The linear system of equations can be solved to yield $\vec{\sigma}$ and $\vec{\gamma}$, after which Γ can be found. As vortex panels have been used, the circulation around the objects is non-zero. The circulation around the j^{th} panel can be found by $\Gamma_j = \gamma_j \cdot \Delta L_j$, and the total circulation can be found by summing the circulation from all the panels, as the super positioning principle applies. For instance, the total circulation around the two objects in Figure A.5 are -1.366 and $-1.632 \text{ m}^2/\text{s}$ for the airfoil and plate respectively.

As mentioned in *Chapter 8 - Predictive Trajectory model*, the circulation will be modelled according to inviscid and irrotational flow theory, where the circulation can be seen in Equation A.12. This equation is formulated using the classical Kutta-Joukowski model, where the translational circulations is a function of $\sin(\alpha)$. Notice that α is the angle of attack and not the panel angles as shown in Figure A.3.

$$\Gamma = - \underbrace{\left(\frac{L}{2}\right) C_T |\vec{u}| \sin(\alpha)}_{\Gamma_T} + \underbrace{\left(\frac{L}{2}\right)^2 C_R \dot{\theta}}_{\Gamma_R} \quad (\text{A.12})$$

The unknowns in Equation A.12 are the translational and rotational circulation coefficients, C_T and C_L . These can be determined using the source-vortex panel method, where the procedure is to calculate the circulation for a given object in two different cases, where α and $\dot{\theta}$ are varied. The results can be formulated into a linear system of equations as seen in Equation A.13 to the left.

$$\begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \cdot \begin{bmatrix} C_T \\ C_R \end{bmatrix} = \begin{bmatrix} \Gamma_1 \\ \Gamma_2 \end{bmatrix} \quad \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \\ \vdots & \vdots \\ A_{N,1} & A_{N,2} \end{bmatrix} \cdot \begin{bmatrix} C_T \\ C_R \end{bmatrix} = \begin{bmatrix} \Gamma_1 \\ \Gamma_2 \\ \vdots \\ \Gamma_N \end{bmatrix} \quad (\text{A.13})$$

$$A_{i,1} = -\left(\frac{L}{2}\right) |\vec{u}_i| \sin(\alpha_i)$$

$$A_{i,2} = \left(\frac{L}{2}\right)^2 \dot{\theta}_i$$

The linear system of equations to the left in Equation A.13 is formed with the minimum required number of cases, which is two. Γ_1 and Γ_2 are the circulation calculated for the two different test cases, where α and $\dot{\theta}$ are varied. The linear system of equations can then be solved to obtain values for C_T and C_R for a given geometry. However, as the panel method is still an approximation of the potential flow, more test cases can improve the accuracy. This has been shown to the right in Equation A.13, where N test cases has been used. This system is overdetermined, so the *least-squares method* is used to solve for C_T and C_R . In practice, 9 cases were used for determining the coefficients.

B | USB CONTENT

The content list of the USB-stick can be seen below.

- ANSYS Fluent:
 - Case files for a simulation with a plate of dimension $40 \text{ mm} \times 2 \text{ mm}$.
 - UDF-wrapper used to generate the ANSYS Fluent UDF's.
 - A guide on how to execute a simulation, Guide.pdf.
 - A video of various CFD-simulations, CFD_Animations.avi.
- Experiments:
 - An image-sequence for plate 2.
 - A video of the experimental results, Experiment_Animations.avi.
- MATLAB
 - The mesh generator used to generate the 2D meshes.
 - The panel method used to calculate the added mass and circulation coefficients.
 - An interactive version of the PT-model.
- References:
 - Electronic references from the report in the format used in the bibliography.
- PDF-files:
 - Report - Modelling of Plates in Free Fall.pdf
 - Enclosures - Modelling of Plates in Free Fall.pdf
 - Executive summary - Modelling of Plates in Free Fall.pdf

