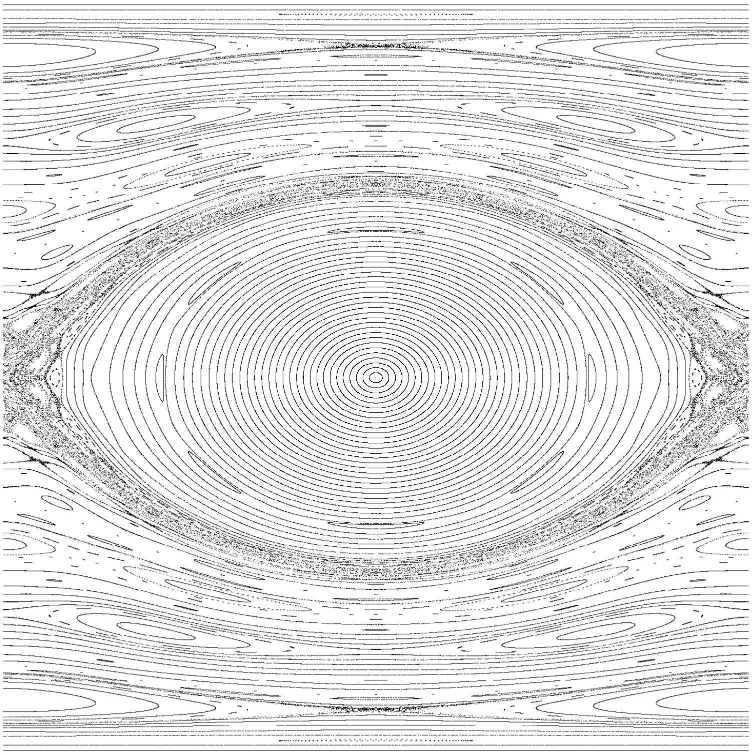


Analysis of empirical data on the tumbling of microrods in a shear flow

Master's Thesis in Complex Adaptive Systems

ANTON JOHANSSON

Department of Physics & Engineering Physics
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2012
Master's Thesis 2012

Thesis for the degree of
Master of Science in Complex Adaptive Systems

Analysis of empirical data on the tumbling of microrods in a shear flow

Author:
Anton Johansson

Supervisor:
Bernhard Mehlig

Department of Physics & Engineering Physics
Chalmers University of Technology
Gothenburg, Sweden 2012

Analysis of empirical data on the tumbling of microrods in a shear flow
Anton Johansson

Department of Physics & Engineering Physics
Chalmers University of Technology
SE-412 96 Gothenburg, Sweden

Chalmers Reproservice
Gothenburg, Sweden 2012

Front cover image:
The image on the front cover is a Poincaré map for a particle shaped like an ellipsoid when $n_x = 0$. The semi-principal axes are scaled as $10 : 1.15 : 1$. The x-axis is $\arcsin(\sin(\psi))$ and the y-axis is $\cos(\theta)$.
For further information, see section 2.5.

Analysis of empirical data on the tumbling of microrods in a shear flow

Anton Johansson
Department of Physics & Engineering Physics
Chalmers University of Technology
SE-412 96 Gothenburg, Sweden

Abstract

This thesis builds upon an existing experiment [1] investigating the tumbling of microrods in a shear flow. The orientational dynamics of microrods in simple shear flows and in turbulent flows is a subject of great importance. This is because the orientational dynamics of particles strongly affects the bulk properties of the suspension [2]. There is now a large number of studies theoretically investigating the tumbling of microrods in flows [2, 3, 4, 5, 6], but there comparatively little experimental data [1, 7, 8]. The laboratory setup [1] allows for recording of large amounts of data for these rods in the form of grayscale movies at high framerates. The primary aim of this thesis is to analyse this data and to compare the results to existing theory.

There are three observables in this experiment, the projected position of the rod in the channel, the projected length of the rod, and the projected orientation vector of the rod, all of these in the image plane of the recorded movies. The recorded movies are analysed using Matlab [9]. Each frame is analysed and the Canny edge-detection algorithm [10] is used to find the particle edges. The particle edges are subjected to an elliptic fit in order to approximate the position, orientation and length of the projected particle with the center, orientation, and length of the fitted ellipse. These properties are translated into the three dimensional components of the rod orientation vector $\boldsymbol{n}(t)$.

Computer simulations of the equations of motion in the form given in [2, 4] were performed. The results are compared to the trajectories observed in the experiment. There is qualitative agreement between the experimental and the theoretical results.

Keywords: Jeffery orbits, microfluidic channel, particle tracking, shear flow.

Acknowledgements

This thesis has been written under the supervision of Professor Bernhard Mehlig in the department of Physics at Gothenburg University. Thank you, Bernhard. I feel like I managed to do something useful.

Jonas Einarsson deserves a special thanks for guiding me and helping me find my mistakes and expressing my result in a human readable format, your guidance has been invaluable.

Yogeshwar Mishra under the supervision of Dag Hanstorp is the experimentalist who produced the movies I worked with. Without these movies there would have been no thesis.

Thanks also go to the rest of Soliden 3rd floor staff for the hospitality shown. Especially to my fellow master students, who made sure that I was never bored.

Mats Rostedt, for writing drivers to control our experimental setup and guiding me in writing proper Labview code.

Anders Martinsson, who has never feared a discussion involving mathematics, algorithms or philosophical issues.

Daniel Andersson, who is also worthy of a mention.

Anton Johansson, Gothenburg May 31, 2012

Contents

1

Introduction

The purpose of this thesis is to investigate the dynamics of particles in flows. This thesis builds upon a continuation of a previous experiment [1]. In this experiment, the authors use a microfluidic channel to study the orientational dynamics of microrods in a shear flow. The aim is to find how the orientation vector $\boldsymbol{n}$ varies as a function of time. The orientation vector describes how the particle is oriented in space, and $\boldsymbol{n}$ is a time dependent unit vector, see equation (1.1).

$$\boldsymbol{n}(t) = (n_x(t),n_y(t),n_z(t))^{\mathsf{T}}. \tag{1.1}$$

For $\boldsymbol{n} = (1,0,0)$ the particle is oriented along the x-axis in euclidean space.

The improvements implemented during this thesis are automating aspects of the data analysis and adding global positional tracking of the particles. Reversal of the flow was also implemented which enabled repeated laps and reuse of particles.

This report is divided into five main parts. This first introductory chapter describes the background of the subject and the reasons for why this subject is important. The first chapter also mentions the different challenges encountered during the course of this M.Sc. thesis, and describes the experiment itself. The theoretical tools required to understand the arguments put forward in this thesis are treated in chapter two. The third chapter delves deeper into the challenges and problems, and presents the solutions implemented. In the fourth chapter experimental results are presented for two different particles, all the available data at the time of writing this thesis. Chapter five deals with the interpretation of the data presented in chapter four, and how our data compare to our expectations, and discuss attempts to understand the data presented in chapter four.

All the analysis software developed during this thesis was implemented in Matlab [9]. The numerical simulations were also written in Matlab using the different built-in ordinary differential equation solvers. The reason for using Matlab is the it has built in libraries for reading video data, solving ordinary differential equations, and several different pre-implemented edge-detection algorithms, including the so-called Canny edge-detection algorithm.

1.1 Background

In 1922 George Barker Jeffery published a paper [3] called ”The motion of ellipsoidal particles immersed in a viscous fluid”. In this paper Jeffery expanded upon a previous paper [11], where Einstein had derived the how viscosity increases with the number of spherical particles in a fluid. Jeffery generalised the result to include particles of any ellipsoidal shape and derived the resulting equations of motion for ellipsoids in a shear flow implicitly, and solved them explicitly for ellipsoids with the condition that at least two axes of the three axes are of the same length.

In this thesis the three different semi-principal axes are of the lengths a_x in the x-direction, a_y in the y-direction, and a_z in the z-direction, their relations are illustrated in figure 1.1. This thesis focuses on tri-axial ellipsoidal particles, for which there is no known explicit solution to the equations of motion. The motion of these tri-axial ellipsoids is interesting because the motion itself is chaotic [4].

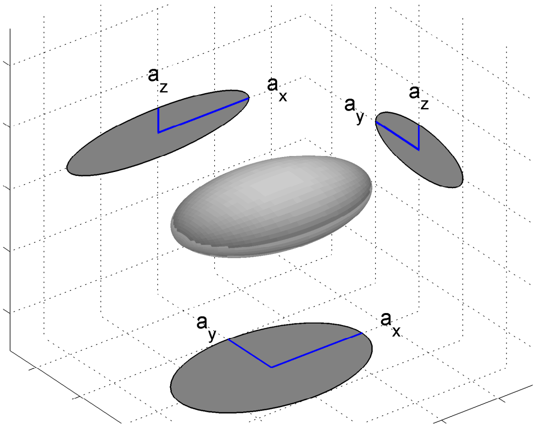


Figure 1.1: An ellipsoidal particle with the three different semi-principal axes of lengths $a_x = 8$, $a_y = 5$, and $a_z = 2$. The ellipse is projected onto each plane to show the relations between the different lengths. The orientation vector for this particle is $\boldsymbol{n} = (1,0,0)$.

1.2 Experimental setup

The channel [1] has the shape of a cuboid, with a length of 4.5 cm, a width of 2.5 mm, and a depth of 400 μm. The fluid in the channel is a mixture between water and glycerol [1]. The reason for mixing in glycerol into the solution is to make the fluid more viscous, and to lower the amount of air bubbles in the channel. The flow rates in this experiment are typically in the range of 5–11 μl/min. The flow is driven by a calibrated syringe pump.

The channel is placed into a motorised stage with stepper motors, above a stationary microscope and camera. A simplified image of the experimental setup can be seen in figure 1.2.

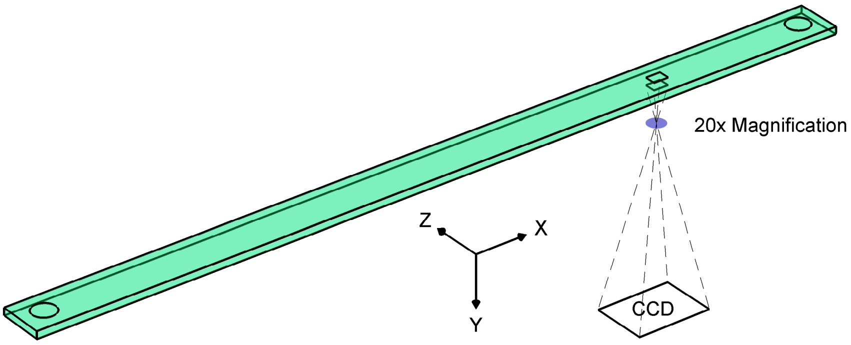


Figure 1.2: The experimental setup. The channel is attached to a movable stage, and is moved above the stationary camera to allow the viewing window to view different parts of the channel. The channel is 4.5 cm long, 2.5 mm wide, and 400 μm deep.

There are three observables; the projected position of the particle, the projected length of the particle, and the angle of the projection in the image plane.

The particles are so small, that they are assumed to have no inertia. This means that the particles are advected by the channel flow, and their speed at any instant is the same as the channel flow velocity at the position of the particle [12].

1.3 Runs and laps

Particles are reused by reversing the flow, and allowing the center of mass of the particles to advect both backwards and forwards in the channel. There are two ways of reversing the flow. The first one is to reverse the flow very gently far away form the inlet or outlet by gradually reducing the flow rate before the actual reversal, the intention is that the rod will not be disturbed from its current orbit. The second method is to let the rod enter either the inlet or outlet and then violently reversing the flow. The intention here is, by contrast, that the resulting disturbance will force the particle into another orbit.

The recorded movies are divided into runs and laps. A run starts when the rod exits the inlet, and stops when the rod enters the outlet. A lap corresponds to when the particle travel from one end of the channel to the other, and runs are composed of one or more laps.

1.4 The shear flow

The channel is 6.25 times as wide as it is deep, and the particle is placed close to the middle along the z-axis, see section 1.5, and close to the the bottom along the y-axis.

The reason for this is so that the shear in the broad direction, along the z-axis, of the channel is negligible compared to the shear in the y-coordinate. As a result the shear lies approximately only along the y-axis, see figure 1.3.

The analysed particles are an order of magnitude smaller than the depth of the channel, so they are small compared to the flow gradient, this allows for the shear strength to be estimated from a linearisation of the flow velocity.

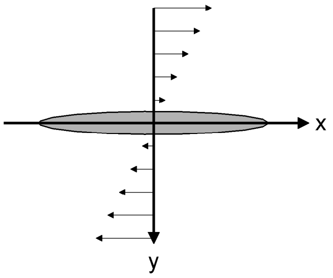


Figure 1.3: The shear flow.

The theoretical flow profile together with the most common particle position, 120 μm from the bottom, can be seen in figure 1.4. Section 2.1 summarises how this flow profile is computed.

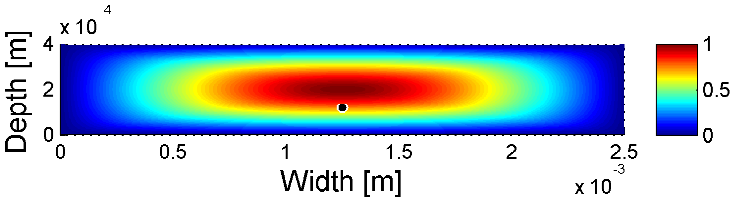


Figure 1.4: The theoretical flow profile of the channel. The flow at the borders is zero, due to the no-slip boundary condition [12]. The flow velocity is maximal in the middle of the channel. The color spectra goes from blue to red, where blue denotes zero flow velocity, and red maximum flow velocity. The black dot marks the particle position in the channel, at $y = 120\,\mu$m, $z = 1.25\,$mm.

1.5 Coordinate systems

In this thesis two different reference systems are used, the one used by Hinch and Leal [2], and by Mishra et. al. [1], where x is the flow direction, y is the shear direction, and so z becomes the vorticity, or the third direction, perpendicular to the other two. Jeffery and Yarin et. al. [4] instead use the convention that z' is the flow direction, y' the shear direction and x' is the vorticity. Both systems are right-handed and the

Table 1.1: Conversion table between the two different coordinate systems.

Hinch	Jeffery
x	z'
y	$-y'$
z	x'

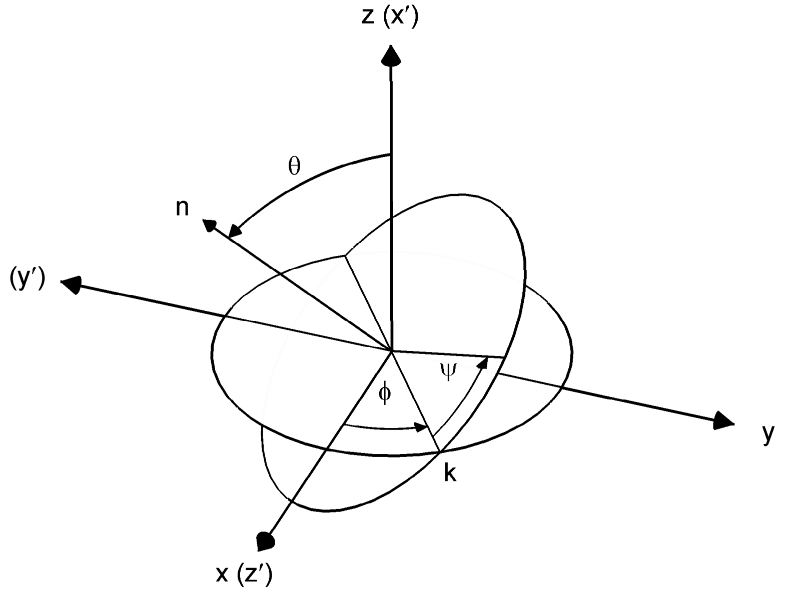


Figure 1.5: The three different coordinate systems used in this thesis. The Jeffery system is the primed system in parenthesis, the Hinch system is the non-primed letters and the Euler angles in Greek letters. The scheme of how to go from Euler angles to any of the other two systems can be found in section 2.2.

conversion between the two is given in table 1.1. The two different coordinate systems can be seen, together with the Euler angles used in section 2, in figure 1.5.

2

Theory

The aim of this chapter is to provide the theoretical tools required to understand how the ellipsoids move inside the shear flow.

2.1 Derivation of the channel flow profile

This section is a summary of [12, 13]. The Navier-Stokes equation for incompressible flow [12] is

$$\rho \left(\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} \right) = -\nabla p + \mu \nabla^2 \boldsymbol{u} + \boldsymbol{f} \tag{2.1}$$

where ρ is the fluid density, the left hand side effects of inertia, ∇p the pressure gradient, μ the viscosity, and $\boldsymbol{f}$ other acting forces. The flow is laminar so the effects of inertia are assumed to be small, and there are no other forces acting on the fluid, so equation (2.1) becomes

$$\rho \frac{\partial}{\partial t} \boldsymbol{u} = \mu \nabla^2 \boldsymbol{u} - \nabla p, \tag{2.2}$$

but $\boldsymbol{u}$ is time independent, because the flow is stationary, so equation (2.2) reduces to

$$\mu \nabla^2 \boldsymbol{u} = \nabla p. \tag{2.3}$$

However, the channel is very long compared to its width and depth, so it is reasonable to approximate the channel length as infinitely long. Then the flow is purely along the channel

$$\boldsymbol{u} = u(y,z) \hat{\boldsymbol{x}}. \tag{2.4}$$

The no-slip boundary condition for fluids sets the speed at each wall to zero. The pump push with a constant pressure, and the channel outlet is open. The pressure drop from end to end is constant so the difference in pressure is $\nabla p = f(y,z) = c$, where c is some constant. The end result of these simplifications are

$$\nabla^2 u(y,z) = f(y,z) \tag{2.5}$$

with the boundary conditions

$$u(0,z) = 0, \tag{2.6}$$

$$u(y,0) = 0, \tag{2.7}$$

$$u(h,z) = 0, \tag{2.8}$$

$$u(y,w) = 0, \tag{2.9}$$

here h is the hight of the channel, and w is the channel width. This is the Poisson equation in a rectangular channel. The following general solution to the Poisson equation comes directly from lecture notes by Professor Yuxi Zheng [13]. The solution begin by solving the eigenvalue problem to find a base in which to describe $f(y,z) = c$

$$\nabla^2 u(y,z) = -\lambda u(y,z). \tag{2.10}$$

One solution to this equation is

$$u(y,z) = \sin(\frac{n\pi y}{h})\sin(\frac{m\pi z}{w}), \tag{2.11}$$

where

$$\lambda = (\frac{n\pi}{h})^2 + (\frac{m\pi}{w})^2, \tag{2.12}$$

where n and m assume any value in $\mathbb{N}$. Using Fourier sine series expansion [13] it is possible to expand $f(y,z)$ into

$$f(y,z) = \sum_{n=1}^{\infty}\sum_{m=1}^{\infty} B_{nm}\sin(\frac{n\pi y}{h})\sin(\frac{m\pi z}{w}), \tag{2.13}$$

where

$$B_{nm} = \frac{4}{hw}\int\limits_0^h\int\limits_0^w f(y,z)\sin(\frac{n\pi y}{h})\sin(\frac{m\pi z}{w})dzdy. \tag{2.14}$$

The general solution [13], $u(y,z)$, is

$$u(y,z) = -\sum_{n=1}^{\infty}\sum_{m=1}^{\infty}\frac{B_{nm}}{(\frac{n\pi}{h})^2 + (\frac{m\pi}{w})^2}\sin(\frac{n\pi y}{h})\sin(\frac{m\pi z}{w}) \tag{2.15}$$

In this particular case B_{nm} can be simplified significantly

$$\begin{aligned} B_{nm} &= c\frac{4}{hw}\int\limits_0^h \sin(\frac{n\pi y}{h})dy\int\limits_0^w \sin(\frac{m\pi z}{w})dz && (2.16)\\ &= c\frac{4}{nm\pi^2}(-\cos(n\pi)+1)(-\cos(m\pi)+1) && (2.17)\end{aligned}$$

Thus

$$B_{nm} = \begin{cases} c\frac{16}{nm\pi^2} & \text{if } n \text{ and } m \text{ are odd,} \\ 0 & \text{otherwise.} \end{cases} \tag{2.18}$$

Let $n = 2k - 1$ and $m = 2l - 1$ and the final expression for $u(y,z)$ becomes

$$u(y,z) = -c\frac{16}{\pi^4}\sum_{k=1}^{\infty}\sum_{l=1}^{\infty}\frac{\sin(\frac{(2k-1)\pi y}{h})\sin(\frac{(2l-1)\pi z}{w})}{(2k-1)(2l-1)((\frac{2k-1}{h})^2+(\frac{2l-1}{w})^2)}, \tag{2.19}$$

which corresponds to equation (8) in [13]. In figure 2.1 the infinite sum from equation (2.19) is truncated after 300 terms.

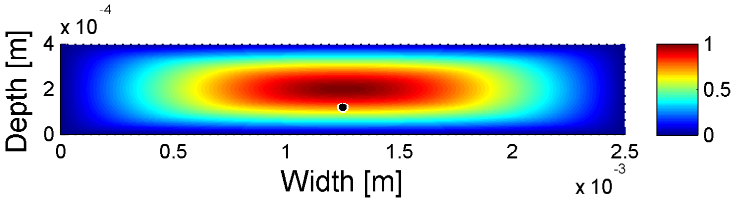


Figure 2.1: The theoretical relative flow profile of the channel. The flow at the borders is zero, due to the no-slip boundary condition. The flow in the middle of the channel is the maximum flow. The color spectra goes from blue to red, where blue denotes no flow, and red the maximum flow. The black dot marks the particle position in the channel, at $y = 120\,\mu\text{m}$, $z = 1.25\,\text{mm}$.

2.2 Jeffery's equations

Yarin et. al. [4] have derived a dimensionless formulation of Jeffery's original equations;

$$\frac{\mathrm{d}\phi}{\mathrm{d}t} = \frac{1}{2} + g_3\sin\psi - g_2\cos\psi, \tag{2.20}$$
$$\frac{\mathrm{d}\psi}{\mathrm{d}t} = g_1 + (g_2\cos\psi - g_3\sin\psi)\cos\theta, \tag{2.21}$$
$$\frac{\mathrm{d}\theta}{\mathrm{d}t} = (g_2\sin\psi + g_3\cos\psi)\sin\theta. \tag{2.22}$$

In these expressions g_1, g_2, and g_3 are functions defined as

$$g_1 = \frac{a_y^2 - a_z^2}{2(a_y^2 + a_z^2)}\left[-\frac{1}{2}(\cos^2\theta + 1)\sin 2\phi\sin 2\psi + \cos\theta\cos 2\phi\cos 2\psi\right], \tag{2.23}$$
$$g_2 = \frac{a_z^2 - a_x^2}{2(a_x^2 + a_z^2)}(-\cos\theta\sin 2\phi\sin\psi + \cos 2\phi\cos\psi), \tag{2.24}$$
$$g_3 = \frac{a_x^2 - a_y^2}{2(a_x^2 + a_y^2)}(\cos\theta\sin 2\phi\cos\psi + \cos 2\phi\sin\psi), \tag{2.25}$$

where a_x, a_y, and a_z are the axes lengths of the ellipsoid. The variables ϕ, ψ and θ are the Euler angles, with which the x'-axis can be rotated into the orientation axis of the particle. The rotation scheme to convert these angles into the Jeffery coordinate system is as follows;

1. Rotate z' ϕ around x' to obtain $\boldsymbol{k}$.
2. Rotate x' θ around $\boldsymbol{k}$ to obtain $\boldsymbol{n}$.
   The first and second step is sufficient to recreate the orientation vector $\boldsymbol{n}$. However, an ellipsoid has two more axes which provides another .
3. Rotate the ellipsoid ψ around $\boldsymbol{n}$ to orient the other axes of the particle, this rotation is an intrinsic rotation.

These equations form the basis for all numerical simulations. The orientation vector $\boldsymbol{n}$ can be expressed directly in the two different coordinate systems as

$$\boldsymbol{n}_{\text{Jeffery}} = (\cos\theta, \sin\theta\cos\phi, \sin\theta\sin\phi), \tag{2.26}$$

$$\boldsymbol{n}_{\text{Hinch}} = (\sin\theta\sin\phi, -\sin\theta\cos\phi, \cos\theta). \tag{2.27}$$

2.3 Jeffery orbits

As mentioned above, Jeffery [3] solved this system of equations explicitly for ellipsoids where $a_y = a_z$. These solutions are periodic with respect to time, and they are named Jeffery orbits [12]. A few concrete examples of Jeffery orbits can be seen in figure 2.2. It is convenient to represent these trajectories on component form of $\boldsymbol{n}$. In figure 2.3 the same initial conditions as in figure 2.2 are plotted in component form. The solution to Jeffery's equation [1] is expressed as

$$\frac{\mathrm{d}\boldsymbol{n}}{\mathrm{d}t} = \mathbf{B}\boldsymbol{n} - \boldsymbol{n}(\boldsymbol{n}^{\mathsf{T}}\mathbf{B}\boldsymbol{n}), \tag{2.28}$$

where

$$\mathbf{B} = \frac{1}{1+\lambda^2}(\lambda^2\mathbf{A} - \mathbf{A}^{\mathsf{T}}), \tag{2.29}$$

and $\lambda = a_x/a_z$ is the aspect ratio of the rod. The expression for $\mathbf{A}$ is

$$\mathbf{A} = \nabla\boldsymbol{u} = \begin{pmatrix} 0 & s & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \tag{2.30}$$

where s is the shear strength [1]. The solution to equation (2.29) is

$$\boldsymbol{n}(t) = \frac{1}{\sqrt{\tan^2\theta_0 + \cos^2\omega t + \lambda^2\sin^2\omega t}} \begin{pmatrix} \lambda\sin\omega t \\ \cos\omega t \\ \tan\theta_0 \end{pmatrix}, \quad \omega = \frac{s\lambda}{1+\lambda^2}, \tag{2.31}$$

where θ_0 is the orbit constant [1].

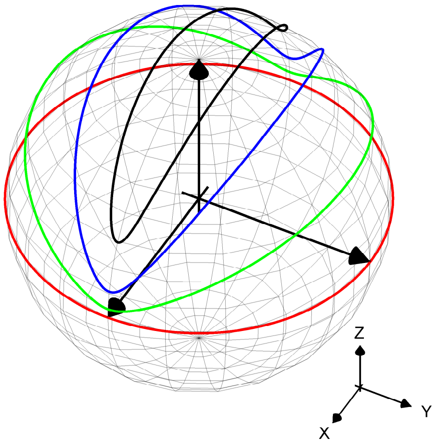


Figure 2.2: Four Jeffery orbits. The orientation vector follows this line for some different initial conditions, from top to bottom with the initial conditions; red: $(\phi,\psi,\theta) = (0{,}0{,}\pi/2)$, green: $(\phi,\psi,\theta) = (0{,}0{,}\pi/4)$, blue: $(\phi,\psi,\theta) = (0{,}0{,}\pi/6)$, and black: $(\phi,\psi,\theta) = (0{,}0{,}\pi/16)$.

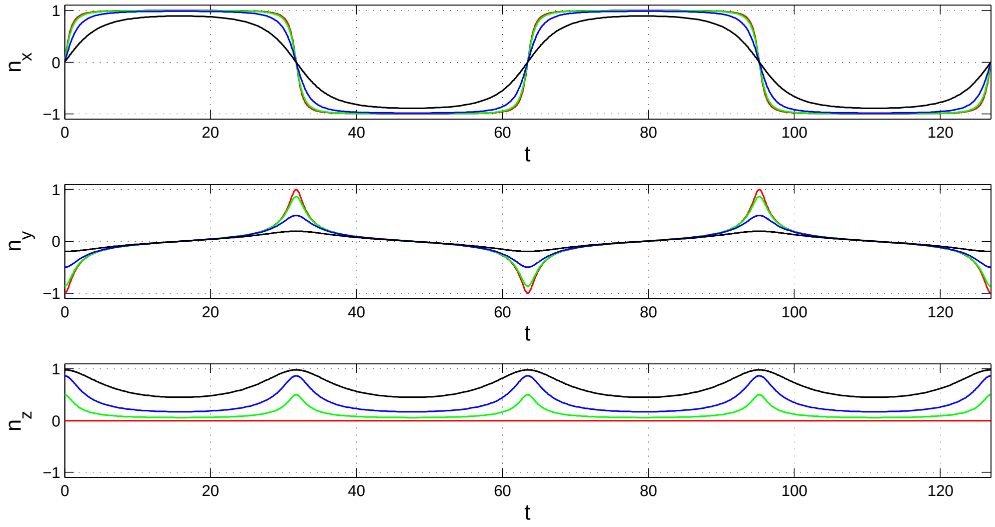


Figure 2.3: Four Jeffery orbits. These are the components of the orientation vector $\boldsymbol{n}$ with the initial conditions; red: $(\phi,\psi,\theta) = (0{,}0{,}\pi/2)$, green: $(\phi,\psi,\theta) = (0{,}0{,}\pi/4)$, blue: $(\phi,\psi,\theta) = (0{,}0{,}\pi/6)$, and black: $(\phi,\psi,\theta) = (0{,}0{,}\pi/16)$.

2.4 Poincaré map of orbits

Because an explicit solution exists for axisymmetric particles it is tempting to assume that any shape that approximates a symmetric ellipsoid behaves like a symmetric ellipsoid. However Hinch and Leal [2], and later Yarin et. al. [4], showed through simulations and analytical arguments that this is not the case.

One very important result from [2] is that the time derivative of ϕ always is non-zero and does not change sign. This is very useful, as it guarantees that it is possible to invert equation (2.20) to obtain

$$\frac{\mathrm{d}t}{\mathrm{d}\phi} = \frac{1}{\frac{1}{2} + g_3 \sin\psi - g_2 \cos\psi}. \tag{2.32}$$

Using this relation and the chain rule it is possible to rewrite equations (2.21) and (2.22) to change the integration variable from t to ϕ.

$$\frac{\mathrm{d}\psi}{\mathrm{d}\phi} = \frac{\mathrm{d}\psi}{\mathrm{d}t} \cdot \frac{\mathrm{d}t}{\mathrm{d}\phi} = \frac{g_1 + (g_2 \cos\psi - g_3 \sin\psi)\cos\theta}{\frac{1}{2} + g_3 \sin\psi - g_2 \cos\psi}, \tag{2.33}$$

$$\frac{\mathrm{d}\theta}{\mathrm{d}\phi} = \frac{\mathrm{d}\theta}{\mathrm{d}t} \cdot \frac{\mathrm{d}t}{\mathrm{d}\phi} = \frac{(g_2 \sin\psi + g_3 \cos\psi)\sin\theta}{\frac{1}{2} + g_3 \sin\psi - g_2 \cos\psi}. \tag{2.34}$$

These modified equations allow control over the ϕ variable, which is desirable as ϕ is driven by the shear flow.

From any given initial condition it is now possible to numerically integrate equations (2.33) and (2.34) to obtain how these varies as a function of ϕ. The values of ψ and θ do not automatically wrap so, since they are angles, it is meaningful to instead look at $n_z = \cos(\theta)$ and $\arcsin(\sin(\psi))$.

Once integrated the three Euler angles, ϕ, ψ, and θ, can be rearranged into a Poincaré map [14] by looking on how the values of the intrinsic rotation, $\arcsin(\sin(\psi(\phi)))$, and $n_z = \cos(\theta(\phi))$ vary for fixed spacings of ϕ. The definition of a Poincaré map requires that all points in the Poincaré map evaluate to another point in the map. This makes $\phi = n\pi$ a good choice, since any particle aligned with the vorticity will remain aligned. Thus, the plane $n_x = 0$ is the only choice for a Poincaré map that can describe the full ellipsoidal dynamics.

By running one such simulation for several different initial conditions a map is created of how the rod will behave for every possible configuration.

2.5 Poincaré maps

The image on the cover page is a Poincaré surface-of-section created for a simulation based of a method described in Yarin et. al. [4], for a particle with $a_x = 10$, $a_y = 1.15$ and $a_z = 1$. The cover image can be seen in figure 2.4. The relation between a_y and a_z define the particle asymmetry, ε, as

$$\varepsilon = \frac{a_y}{a_z} - 1, \tag{2.35}$$

and the surface-of-section in figure 2.4 has $\varepsilon = 0.15$. In figure 2.5 another nine Poincaré surfaces-of-section are presented.

The first figure, figure 2.5(a), is the surface-of-section for Jeffery orbits. The explicit solution for $\boldsymbol{n}$ in equation (2.31) has $n_x = 0$ when $t = m\pi/\omega$ and $m \in \mathbb{N}_0$, so equation (2.31) simplifies to

$$\boldsymbol{n}\left(\frac{m\pi}{\omega}\right) = \frac{1}{\sqrt{1+\tan^2\theta_0}}\begin{pmatrix} 0 \\ \cos m\pi \\ \tan\theta_0 \end{pmatrix}. \tag{2.36}$$

For every point in the surface-of-section the z-component only depends on the value of $\tan(\theta_0)$, which is constant. Therefore the value of n_z is independent of the intrinsic rotation ψ and form straight lines in the surface-of-section.

The Jeffery orbits are very sensitive to asymmetry, even for very small values of ε the Jeffery orbits begin to break down into circular trajectories. Figure 2.5(b) is a Poincaré map for $a_x = 10$, $a_y = 1.001$ and $a_z = 1$. The asymmetry, ε, is only $1/1000$ and the circular trajectories are already visible. For $\varepsilon = 1/100$ the circular trajectories are of considerable amplitude.

In figure 2.5(e), when the asymmetry has reached $1/2$ there are no longer any trajectories that resemble Jeffery orbits, except the trivial ones, where $\cos(\theta) = \pm 1$. The trivial orbits are those where the particle orientation vectors are parallel to the vorticity.

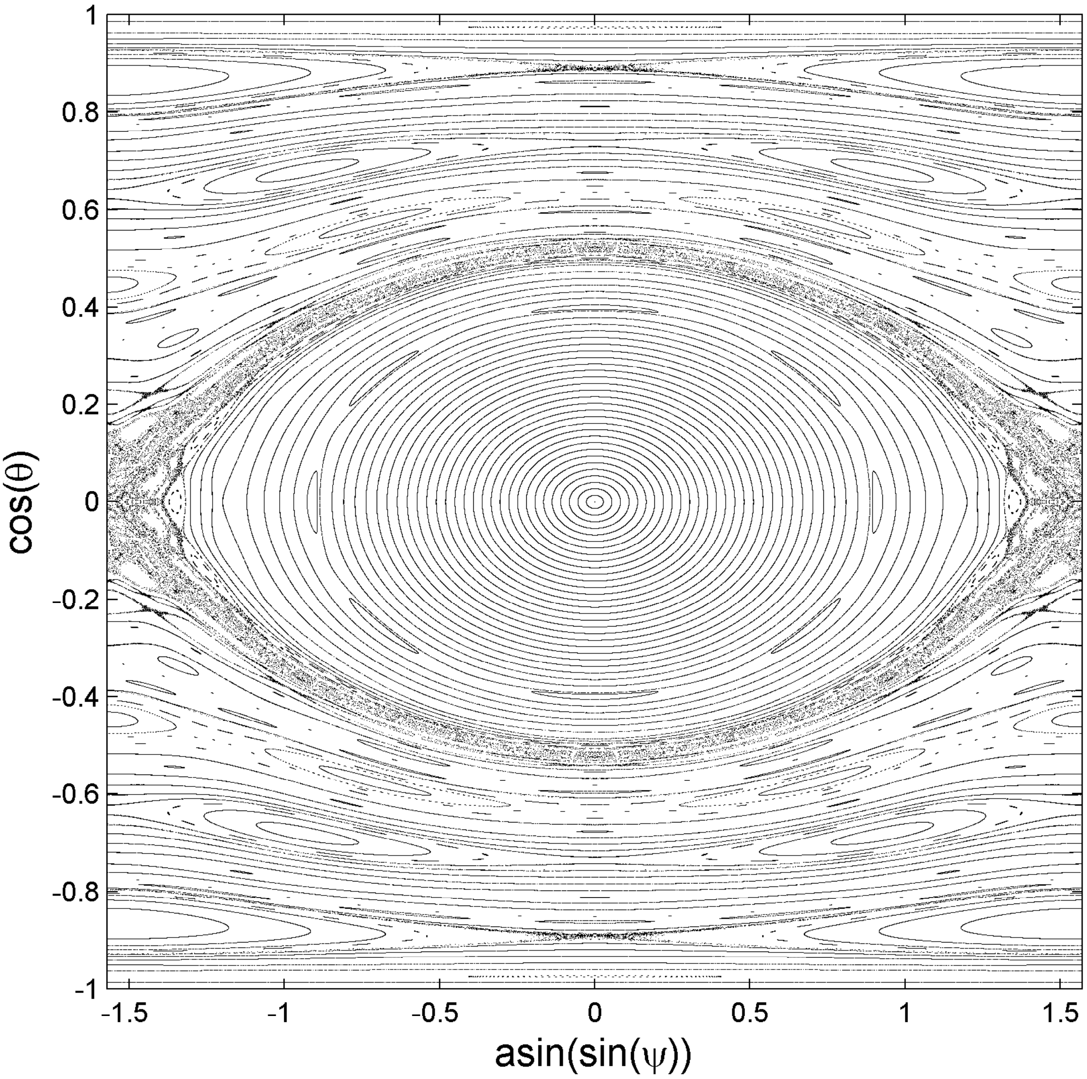
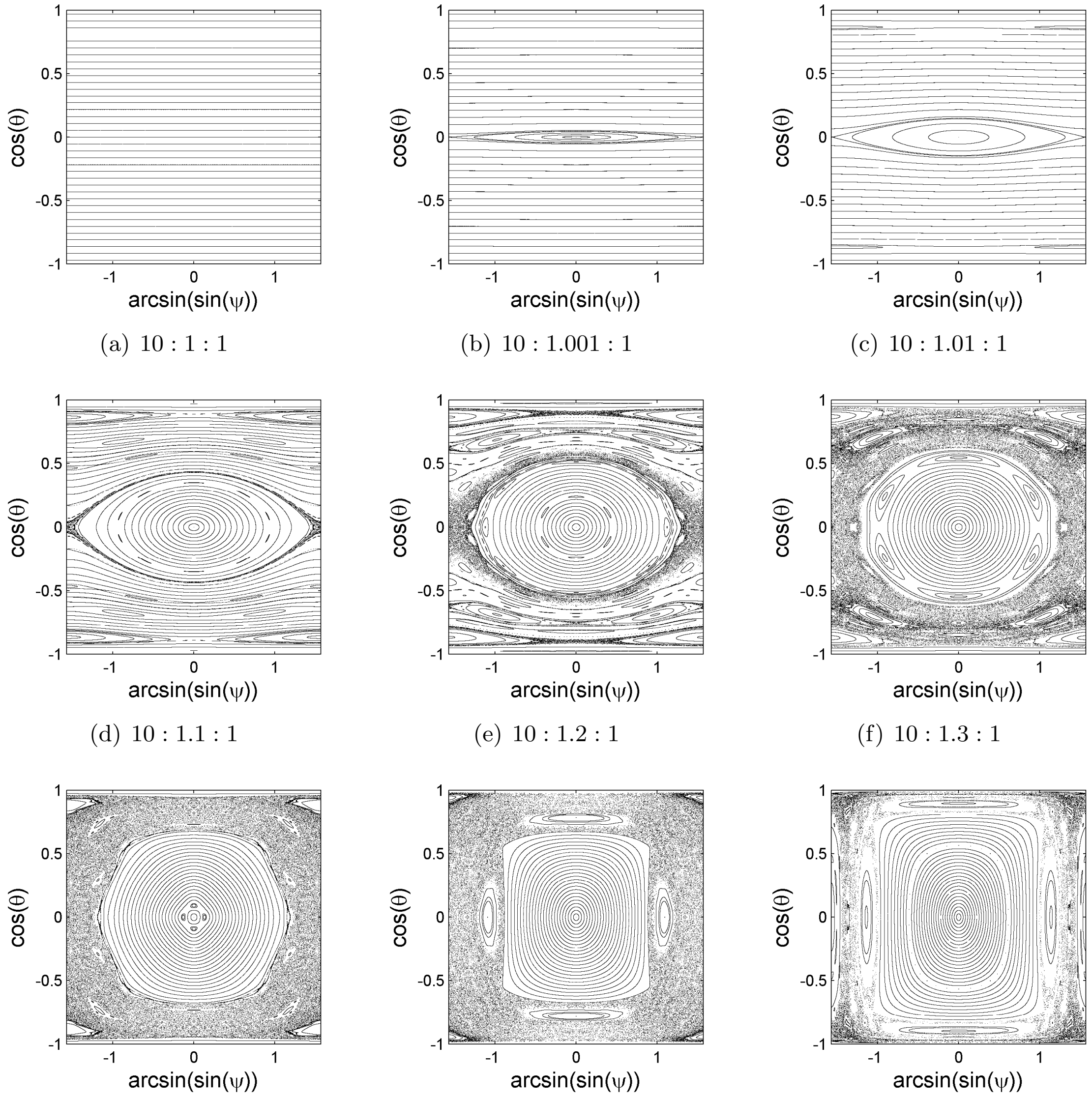


Figure 2.4: A Poincaré map for 10 : 1.15 : 1. The map is created by simulating equations (2.33) and (2.34) for different initial conditions. Compare this figure to the maps in figure 2.5.



(a) 10 : 1 : 1

(b) 10 : 1.001 : 1

(c) 10 : 1.01 : 1

(d) 10 : 1.1 : 1

(e) 10 : 1.2 : 1

(f) 10 : 1.3 : 1

(g) 10 : 1.5 : 1

(h) 10 : 2 : 1

(i) 10 : 3 : 1

Figure 2.5: Poincaré maps for nine different ellipsoidal particles. The dimensions are on the form $a_x : a_y : a_z$. The maps are created by simulating equations (2.33) and (2.34) for different initial conditions. Figure 2.5(a) is the Poincaré map for ellipsoidal particles with $a_y = a_z$. The particle dynamics depends sensitively upon the asymmetry ε. In figure 2.5(b) circular trajectories have begun to form, and in figure 2.5(d) small chaotic regions have begun to form around $\psi = \pm\frac{\pi}{2}$, $\cos(\theta) = 0$. When ε becomes large, as in figure 2.5(g), it is unlikely that any trajectory maintaining a constant value of n_z can be found, other than the trivial trajectories where the orientational vector of the particle is parallel to the vorticity.

3

Data extraction

The experiment described in section 1.2 produces raw data in the form of grayscale movies, in which the rod is seen from below as it travels along the channel. These movies are captured at a rate of 100 frames per second (fps). Each frame is represented by a matrix with 260×348 elements. One such element is referred to as a pixel, which is the smallest discrete unit of a frame or an image. These pixels carry 8 bits of intensity information, so each pixel consist of an integer value in the range of 0 to 255.

The length of video analysed for this M.Sc. thesis exceeds thirteen hours of raw footage. This corresponds to roughly five million frames, which is an unrealistic amount of data to be analysed manually. Therefore, automated tracking is a necessity.

A typical movie frame can be seen in figure 3.1, and for this frame the visual noise is low. In this context visual noise refers to the existence of few rod like objects in the immediate vicinity of the target rod. The edges of the target rod are sharp and well defined, and so they are easy to locate manually by visual inspection, as they are represented by large differences in the image contrast.

The aim of the data analysis is to recreate the orientation vector $\boldsymbol{n}$ from equation (1.1). In order to find the length and orientation of the projected rod in the image plane it is possible to approximate the projection by fitting an ellipse to to the edges of the projected rod. The long axis of the fitted ellipse then equals half of the projected rod length and the direction of the long axis shows how the projected rod is oriented in the image plane.

3.1 Identification of different laps

The positional data from the stage motors contains information on how the camera moves relative to the channel with respect to time. This means that it is possible to find the different laps by studying the motion of the stepper motors. The stepper motors move the channel so that the camera moves with the flow direction, the x-direction. An

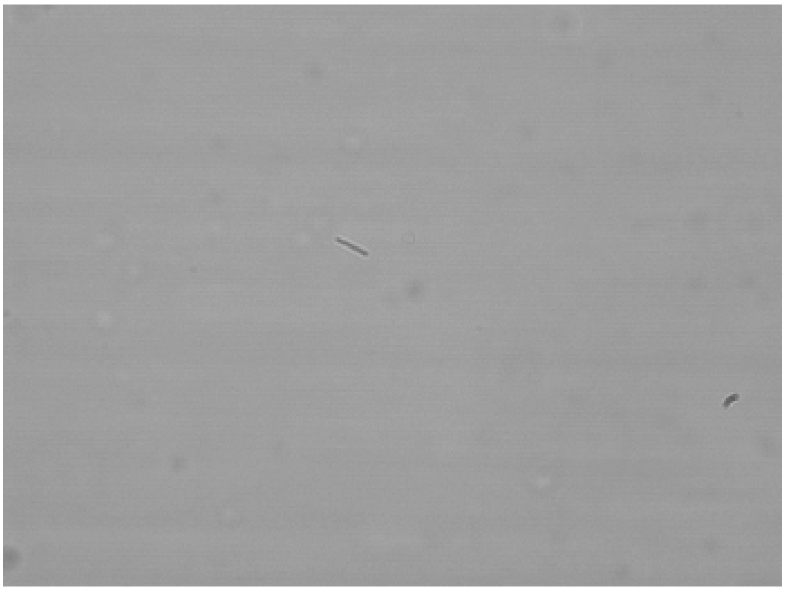


Figure 3.1: A typical movie frame.

example of the camera x-position, relative to the channel, as a function of time can be seen in figure 3.2. The positional information will be explained in more detail in section 3.7.

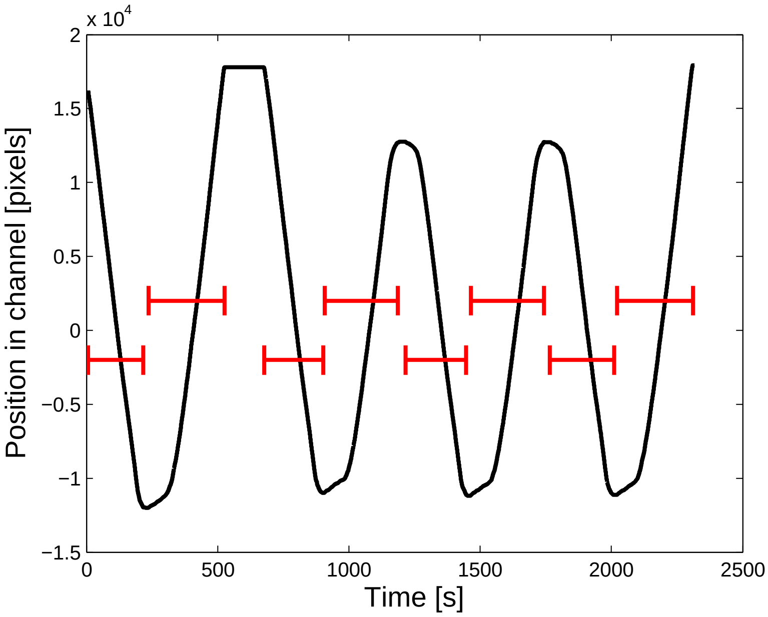


Figure 3.2: The camera x-position on the channel as a function of time for movie `M-Movie0067.avi`. The lines show the different detected laps and their flow direction.

3.2 Static noise reduction

Garbage, such as air bubbles or old rods, floating in the channel or stuck to the glass walls are not the only source of visual noise. There are also effects like improper lighting conditions and dirt on the camera lens. These types of disturbances are referred to as static noise, because they do not vary with time. These objects are often of high contrast, and so the risk that they interfere with the automated tracking is high.

The information hidden behind these artefacts is lost, but it is possible to lessen the impact the static noise have during the image analysis by replacing the artefacts with blind spots. This is achieved by subtracting the average frame from each frame, then adding the average frame color to each pixel in the frame.

$$f_k^* = f_k - \frac{1}{N} \sum_{n=1}^{N} f_n + \overline{f_k} \tag{3.1}$$

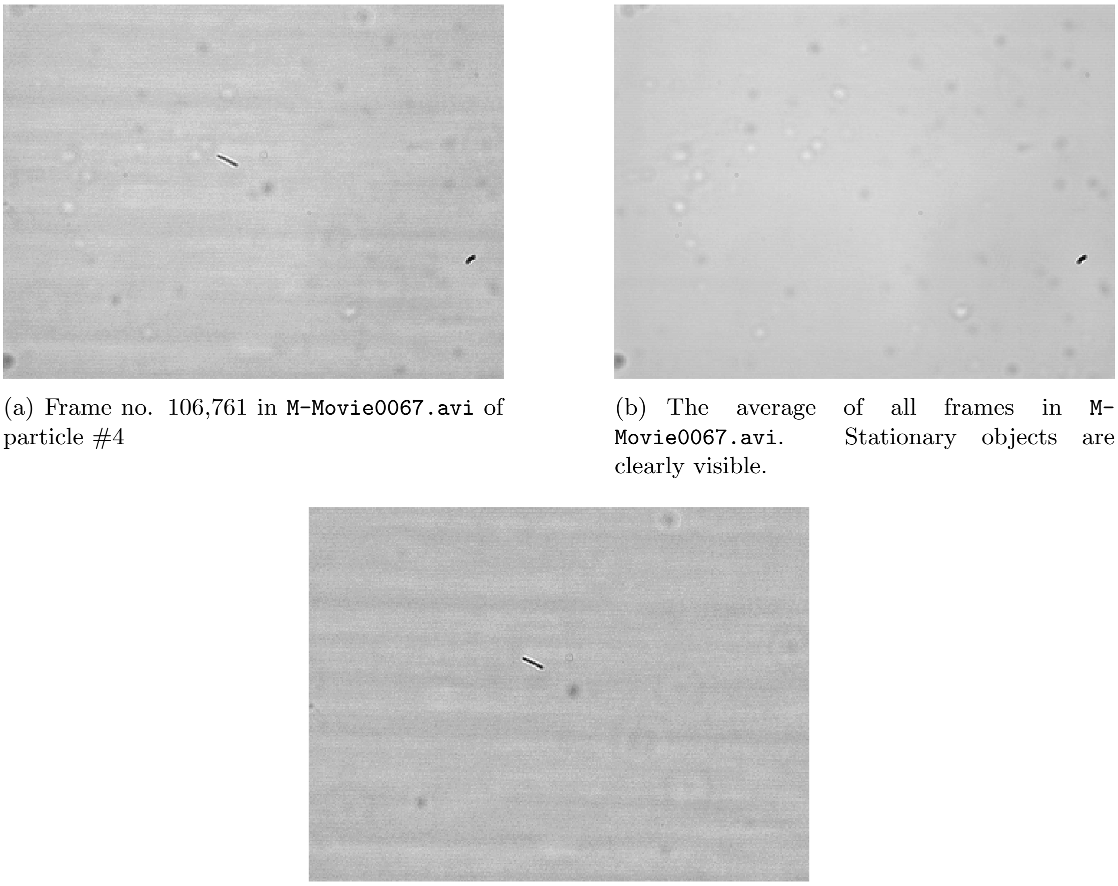
where f_k^* denotes the cleaned up frame, N is the total amount of frames, and $\overline{f_k}$ denotes the average pixel color for frame f_k. The effect of the static noise removal can be seen in figure 3.3.

3.3 Canny edge-detection

Canny edge-detection, invented in 1986 by John Canny [10], is an edge-detection algorithm that emphasizes good detection and localization. Good detection means that edges should have a large probability of being found, and good localization means that edges they should appear close to the actual location of the edge. Canny was the first who also supplied the mathematical theory proving why the algorithm worked. This makes the Canny edge-detection algorithm a good choice since the representation of the rod is deformed as little as possible.

During a normal run the program is aware of roughly where the rod is, under the assumption that the rod has not moved too far since the last frame. A smaller window is then cut from the current frame using the old rod center as center and the Canny edge-detection algorithm is executed and will typically return a result similar to that in figure 3.4.

In general the rod is not in the vicinity of other objects so it is a simple matter to do an elliptic fitting, see section 3.4, to all edges found by the Canny edge-detection. However, sometimes the rod pass close to some other object, as seen in figure 3.4. When this occurs it is not obvious which edges are part of the rod, and which edges belong to the foreign object. If the rod has not moved far from its most recent position, it is possible to use the last ellipse center point as a reference point for a flood-fill. Flood filling is an algorithm where the closest edge pixel relative to the old center is selected, then only edge pixels that are directly adjacent to any other selected edge pixel are selected recursively. This process continues until no more unselected adjacent pixels remains. The result of the flood-fill algorithm can be seen in figure 3.5.



(a) Frame no. 106,761 in `M-Movie0067.avi` of particle #4

(b) The average of all frames in `M-Movie0067.avi`. Stationary objects are clearly visible.

(c) The resulting frame without static noise.

Figure 3.3: Removal of static noise.

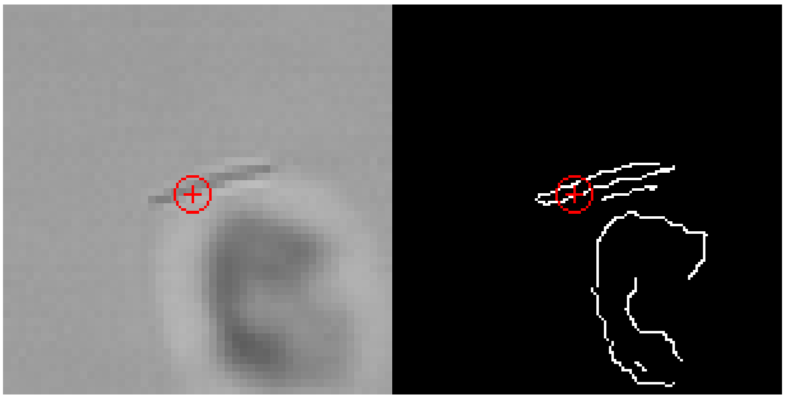


Figure 3.4: Canny edge-detection without flood-fill. The red circle marks the previous ellipse center point.

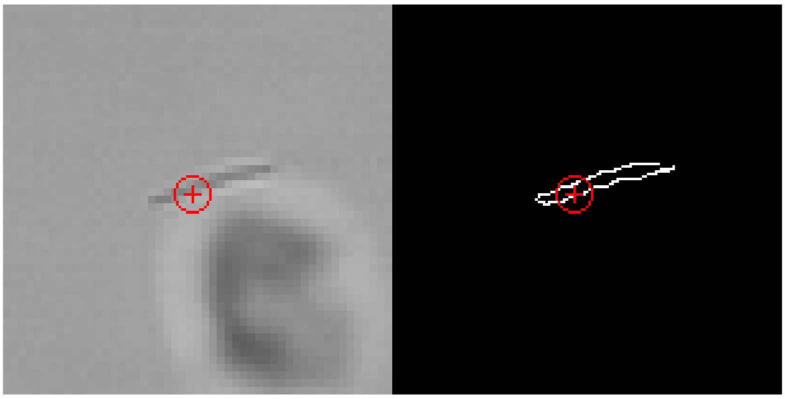


Figure 3.5: Canny edge-detection with flood-fill. The red circle marks the previous ellipse center point.

The edge-detection algorithm returns a list of coordinates of each pixel that are a part of an edge in the original image.

3.4 Elliptic fit

Once the edge-detection is complete an ellipse is fitted to the edge points. The fit is a least square method based on principal-component analysis, (PCA), which finds the direction in which the points are spread the most. For a set of points from an ellipse with zero mean, and an even distribution of points around the edge, the direction of maximum spread will coincide with the major axis of the ellipse.

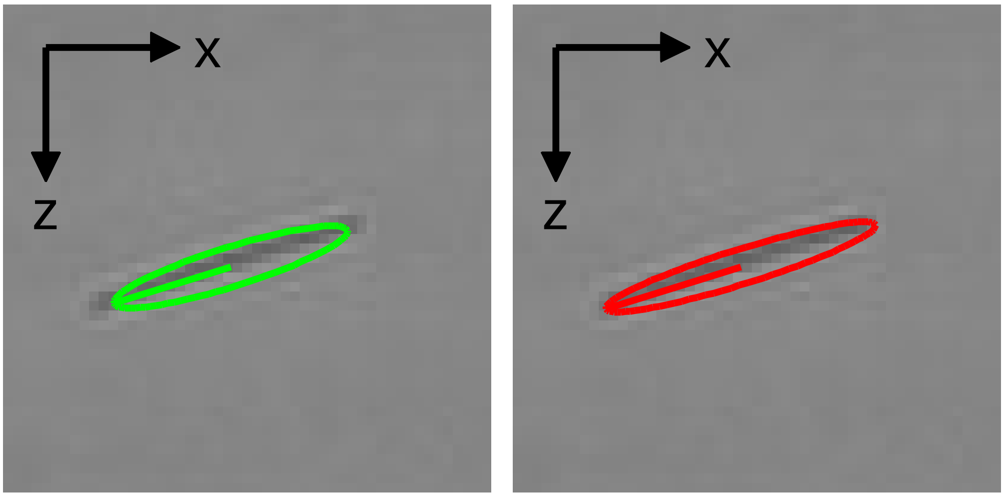
The simplest implementation of PCA is explained in [15]. Let $\boldsymbol{x}$ and $\boldsymbol{z}$ be vectors with the x and z values of all edge points in the fit. Create

$$\mathbf{A} = \begin{pmatrix} x_1 & z_1 \\ x_2 & z_2 \\ \vdots & \vdots \\ x_N & z_N \end{pmatrix} \tag{3.2}$$

and find the eigenvectors and the eigenvalues to

$$\mathbf{M} = \frac{1}{N-1}\mathbf{A}^\mathsf{T}\mathbf{A} \tag{3.3}$$

where $\mathbf{M}$ is a 2×2 matrix and N the number of edge points [15]. The eigenvector corresponding to the largest eigenvalue, λ_1, corresponds to the direction of the major axis, and the length of the axis is $\sqrt{2\lambda_1}$ [15]. Because the eigenvectors are perpendicular to one and another the second eigenvector,λ_2, corresponds to the minor axis with the length $\sqrt{2\lambda_2}$ [15]. The algorithm the tracking software uses is a more advanced version of



(a) PCA fit

(b) Elliptic fit

Figure 3.6: The PCA fit compared to the elliptic fit. The elliptic fit appears to follow the shape of the rod better than the regular PCA fit.

PCA [16] which is argued to be more stable than a simple principal component analysis. A comparison between the two methods can be seen in figure 3.6.

Once the angle is found, both the z-component and the x-component can be obtained, and from these the magnitude of the y-component can be computed.

For the tracking of the rod it does not matter whether the fitted ellipse is perceived as thicker or thinner than the actual rod, since the width of the rod is not used during the current tracking scheme, only its orientation, position, and length.

3.5 Approximation of the orientation vector

The orientation vector $\boldsymbol{n}$ is normalised, so in order to find the orientation from the observables in a movie, the particle length l_{rod} must be known. If l_{rod} is known the components of $\boldsymbol{n}$ can be determined as

$$n_x = \frac{x}{l_{\text{rod}}}, \tag{3.4}$$

$$n_z = \frac{z}{l_{\text{rod}}}, \tag{3.5}$$

$$|n_y| = \sqrt{1 - n_x^2 - n_z^2}. \tag{3.6}$$

Since the only available data is the projection of the rod it is not possible to determine the sign of n_y. However, the shear flow causes the n_y-component to change sign between every flip. For this reason an arbitrary initial sign has been assigned to n_y, so to make the data more comparable to the theoretical simulations.

3.6 Determining the rod length

The rod length l_{rod} is unknown at the start of the experiment. This is not a problem since Hinch et. al. [2] showed that any ellipsoid have a non-zero $\frac{\mathrm{d}\phi}{\mathrm{d}t}$. It is also shown in [2] that the particles spend more time aligned with the flow than perpendicular to it. This means that the rod is expected the have $n_y = 0$ at least some of the time. When $n_y = 0$ the orientation vector of the particle, $\boldsymbol{n}$, lies in the x-z-plane, the image plane. The alignment with the flow direction occurs once every flip, and each lap observed in this experiment contains 6, or more, flips. By plotting all of the measured projection lengths for a particle in a histogram it is possible to find the most likely particle length.

3.7 Particle position

In addition to a movie each experimental run is also accompanied by the positional information of the channel relative to the camera. This is achieved with a separate program which repeatedly queries the step motor controller for the current positional data and records the data together with a time stamp. This process continues during the recording of the entire movie.

By running both the camera video recording and the position recorder from the same computer the time stamp of their respective data is synchronised. The two different programs do not run at the same capture rate. For each frame, the position at the time of the frame is found by a linear interpolation between the closest positions before and after the frame in question.

When the tracking is completed, the camera position and the particle-within-frame position are combined into the particle position. Examples of paths particles have travelled can be seen in figure 4.2.

3.8 Dimensionless time

By design the pump rate is not constant during a large portion of a given lap. During the part of the lap where the pump is supposed to maintain a constant rate slow variations of the flow rate can still occur. The pump supplies a pressure to one side of the channel. When the flow rate changes, so does Δp from equation (2.3). If the change in flow rate is slow enough so that all transients can be neglected, the change in Δp translates linearly to the flow velocity u.

For particles that are small compared the the channel the shear, s, is approximated by $\frac{\partial u}{\partial y}(y,z)$ [1], which also scales linearly with Δp.

While it is not possible to observe the changes in the flow velocity directly, it is possible to observe the flow velocity indirectly, through the particle position. The distance the particle has travelled is

$$d(t) = \int\limits_0^t u(t')\mathrm{d}t'. \tag{3.7}$$

Equation (3.7) is useful because it scale with time in the same way as $\boldsymbol{n}(t)$. This means that it is possible to plot any time series versus $d(t)$ rather than t. The benefit is that when compared to $d(t)$ the time series will be less affected by flow changes.

4

Results

Two data sets are analysed, corresponding to two different particles. In the first data set contains a relatively large particle, and in the second data set there is a smaller particle, the particles can be seen in figure 4.1. The first data set contains the particle designated *particle* #1 and the second data set contains *particle* #4. Particle #1 is 23.3 pixels long and particle #4 is 17.3 pixels long. Histograms of the rod lengths can be seen in figure 4.3.

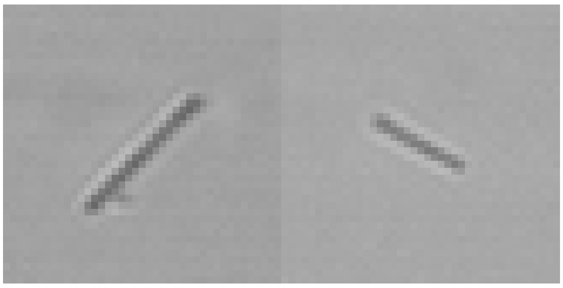
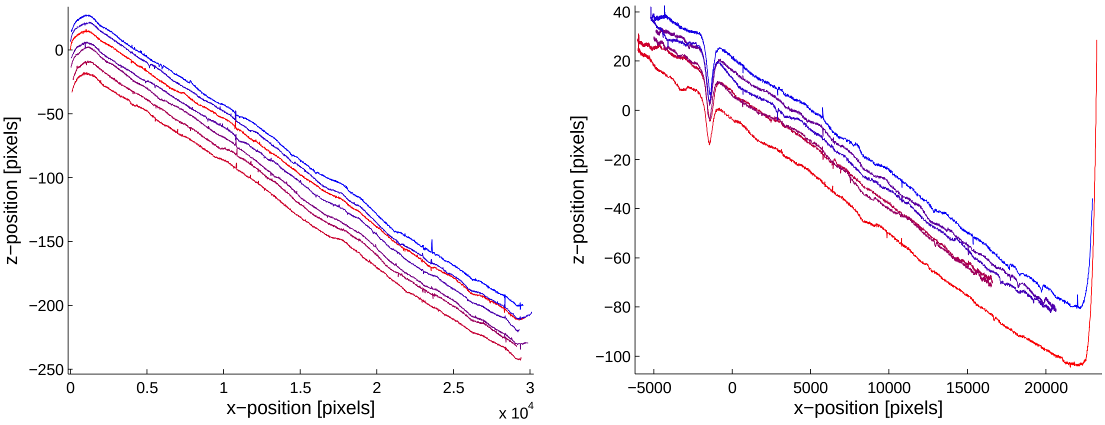


Figure 4.1: The two particles with the most data. The particle to the left is designated particle #1 and the particle to the right is designated particle #4. Particle #1 is almost twice as large as particle #4, and it carries a deformity. Particle #1 is 23 pixels long and particle #4 is 17 pixels long.

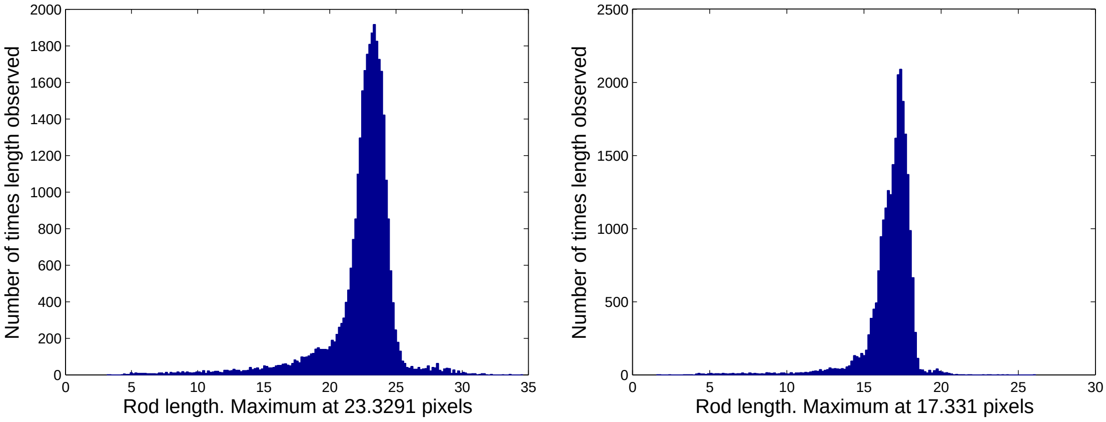
For the first particle, particle #1, the experimental procedure included a reset of the relative position between each lap, and as a result the individual laps have an unknown offset towards each other. This discrepancy in the laboratory protocol was changed before the rest of the data for the other particles were gathered. The first particle also have a deformity, see figure 4.1, which did not seem to have any visual effect on the trajectories. Some typical flow trajectories for the two primary data sets can be seen in figure 4.2. The difference in scale between the x-axis and the z-axis is roughly two orders of magnitude, this is because the camera and the channel is almost perfectly aligned, so the flow is almost purely along the x-axis, and the angle is in the order of 0.01 radians. The bends that occur on ends of some of the runs are due to the particle entering or exiting one of the outlets, this flow can result in a relatively large movement in z.



(a) Particle #1

(b) Particle #4

Figure 4.2: These are some example flow trajectories for both data sets. Because of the discrepancy in the operator procedures for the first data set for particle #1 the trajectories themselves are not comparable by position to each other, only by relative shape. For particle #4 all coordinates share a point of reference so they are comparable. The bump in the later data set is probably an air bubble that has formed somewhere in the channel which distorts the flow. The difference in scale between the x-axis and the z-axis are almost two orders of magnitude. The downward slope, with an angle in the order of 0.01 radians, is caused by a small misalignment between the channel and the camera.



(a) Particle #1

(b) Particle #4

Figure 4.3: The length distributions of the two rods, as measured by the tracking software.

The data can be divided into two classes, those trajectories that are similar to Jeffery orbits, and those who are not. Both these types are found in both of the data sets.

4.1 Data set 1

Orbits that look like Jeffery orbits appear to be common, and some examples of these orbits can be seen in figures 4.4, and 4.5. Amongst the rarer laps, those which do not look like Jeffery orbits, are laps like the ones in figures 4.6 and 4.7.

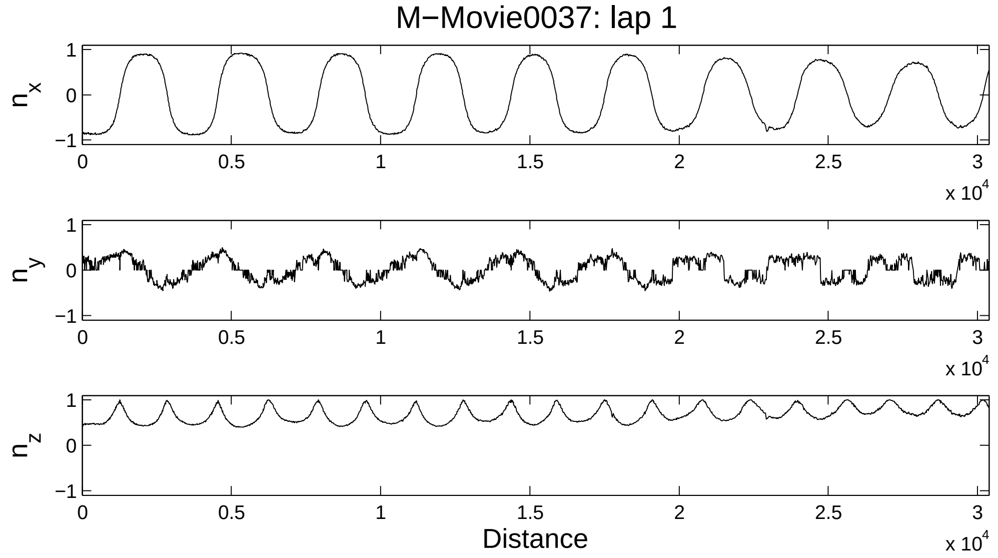


Figure 4.4: Data for particle 1, movie 37, lap 1.

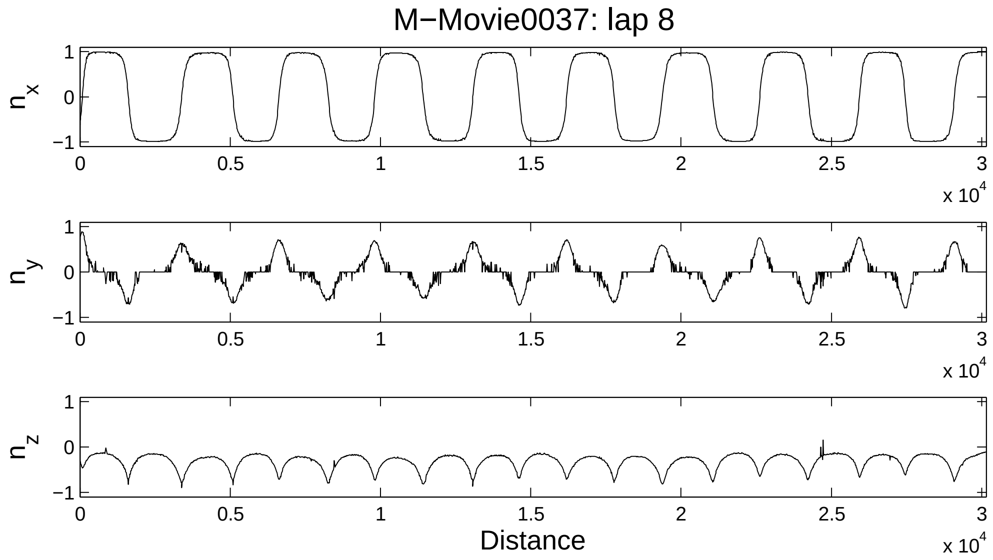


Figure 4.5: Data for particle 1, movie 37, lap 8.

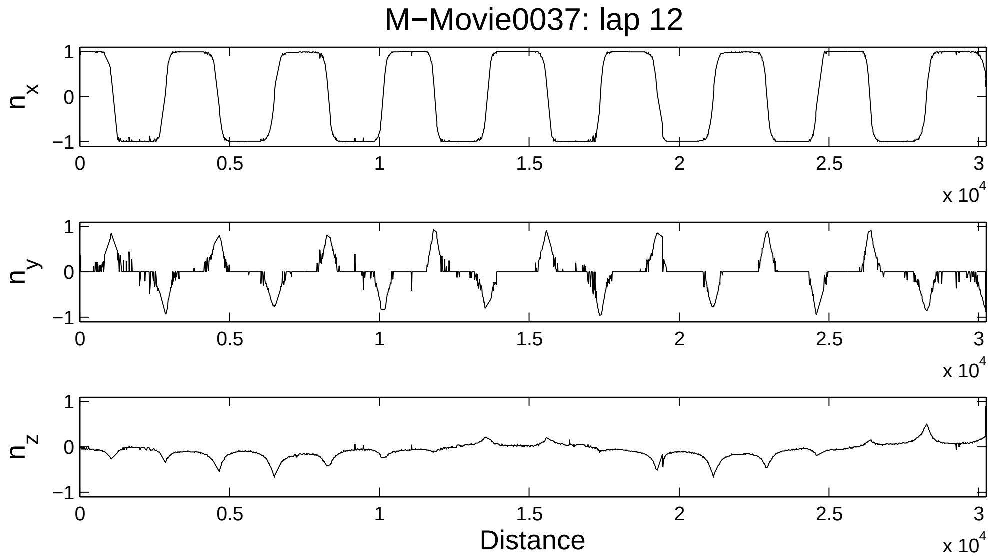


Figure 4.6: Data for particle 1, movie 37, lap 12.

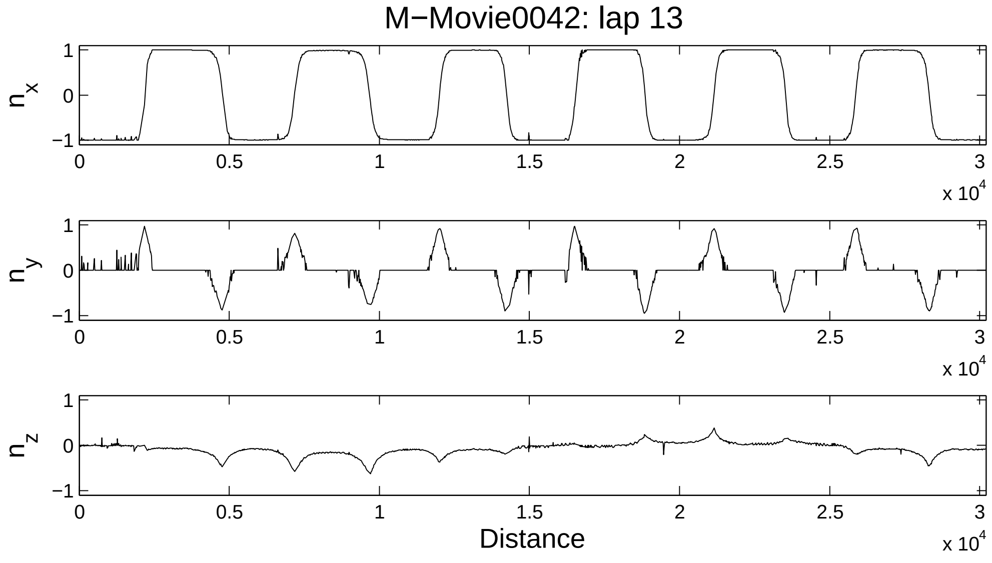


Figure 4.7: Data for particle 1, movie 42, lap 13.

4.2 Data set 2

As in the first data set, there are trajectories that appear to be Jeffery orbits. Figures 4.8 and 4.9 are examples of Jeffery orbits. Trajectories that are different from the Jeffery orbits, see figure 2.3, are shown in figures 4.10 and 4.11.

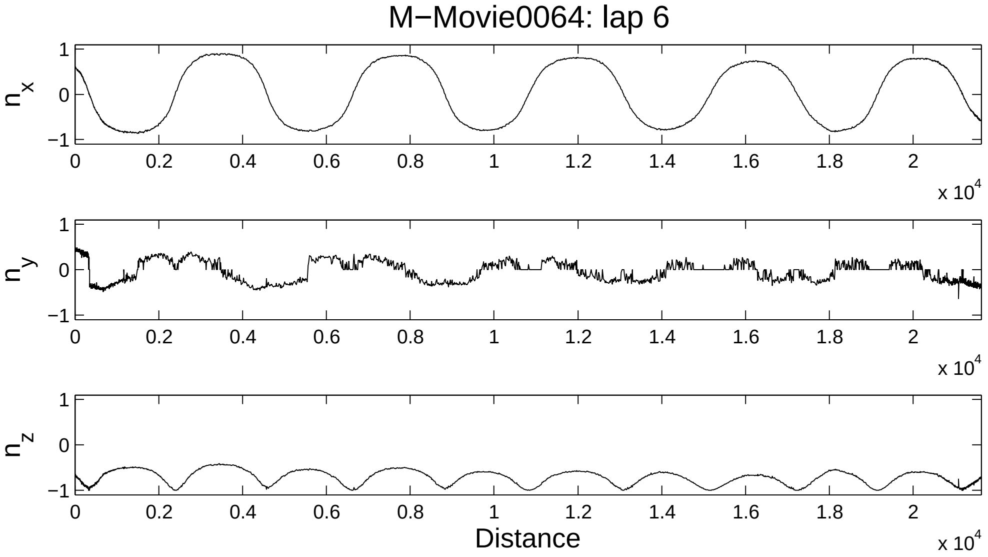


Figure 4.8: Data for particle 4, movie 64, lap 6.

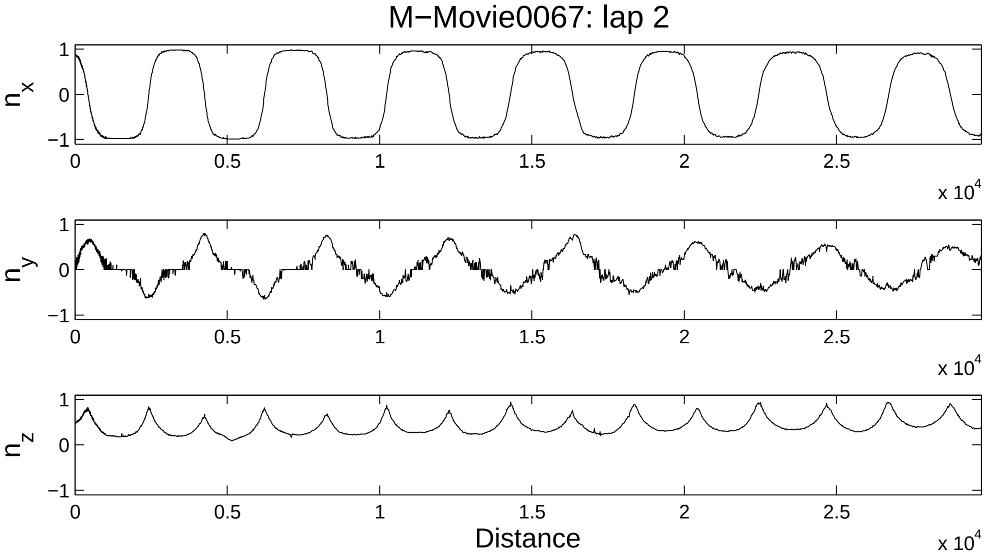


Figure 4.9: Data for particle 4, movie 67, lap 2.

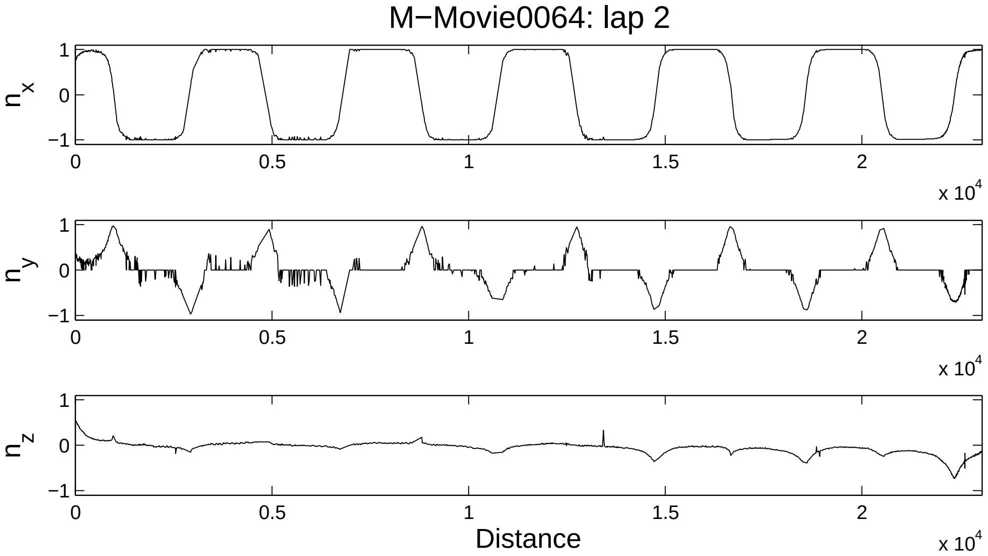


Figure 4.10: Data for particle 4, movie 64, lap 2.

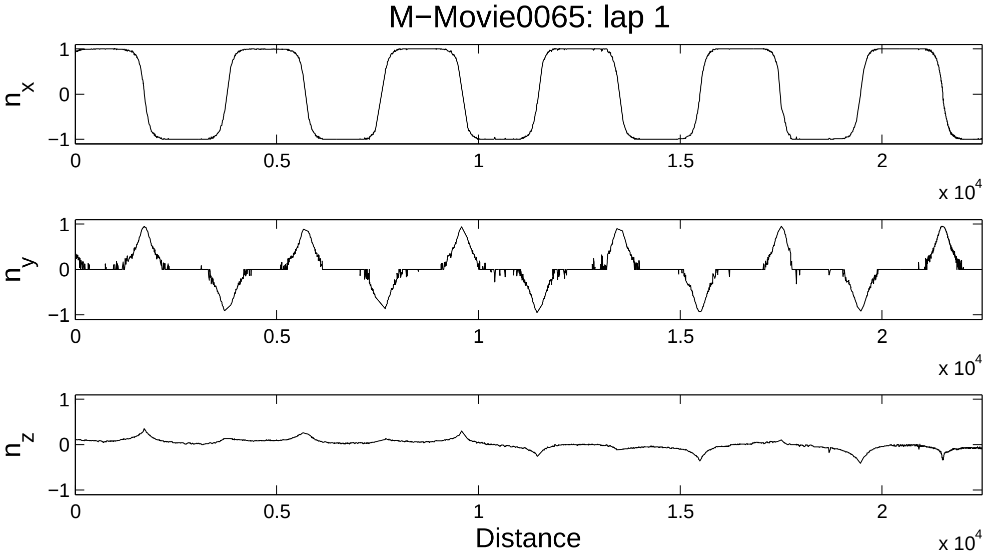


Figure 4.11: Data for particle 4, movie 65, lap 1.

4.3 Comparison to simulations

Both particles have shown three types of trajectories that bear a resemblance not only to each other but also to simulated trajectories.

The first type is trajectories that look similar to the Jeffery orbits from figure 2.3, these have the largest magnitude of n_z in the surface-of-section and n_z does not change sign between the different flips. The values of n_z are almost constant. An example of where these orbits lie in the surface-of-section can be found in figure 4.12. This simulated trajectory in figure 4.12 resembles figures 4.4, 4.8, and the second half of figure 4.9.

The second type is trajectories that still are similar to the Jeffery orbits, where the values of n_z varies more than for the first type, but not so much that n_z changes sign. The position of these types of trajectories is shown in figure 4.13. Some examples of these orbits are figure 4.5 and the first half of figure 4.9.

The third trajectory type is circular trajectories, where n_z does change both sign and magnitude, an example of this behaviour can be seen in figure 4.14. Trajectories like the one from figure 4.14 can be seen in figures 4.6, 4.7, 4.10, and 4.11.

There is also a forth type of trajectory, the trajectories which move within the chaotic region. On short time scales they follow close-by orbits, so it is not possible to distinguish chaotic orbits from stable orbits for the short laps that are presented in this thesis.

Since there are orbits that are similar to Jeffery orbits present in both data sets, other than the trivial orbits, where $n_z = 1$, it is plausible that the particle asymmetries are relatively small, compare to the Poincaré map in figure 2.5(e).

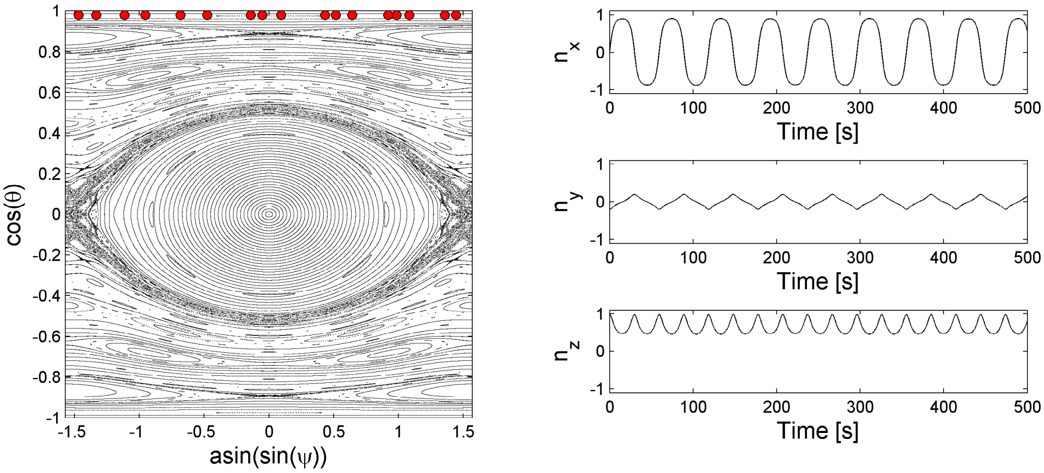


Figure 4.12: A Poincaré map for an ellipsoid $a_x = 10$, $a_y = 1.15$, and $a_z = 1$. This simulated trajectory bear resemblance to figures 4.4, 4.8, and the second half of figure 4.9.

4.4 Runs and stitches

In figure 4.15 a stitch of several laps has been made for particle #1 and in figure 4.16 the same for particle #4. The first particle seems to stay in a stable orbit, while the second particle has a prominent drift towards high absolute values of n_z. It is unclear what the differences between the two particles are. The particles show a similar dynamics for single laps, but are very different when viewed over several laps.

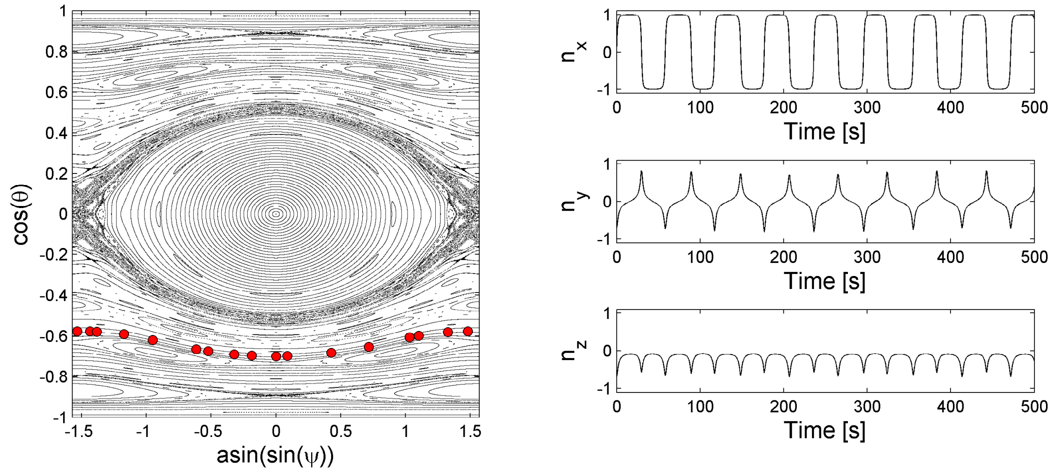


Figure 4.13: A Poincaré map for an ellipsoid $a_x = 10$, $a_y = 1.15$, and $a_z = 1$. This simulated trajectory resembles the ones seein in figure 4.5 and the first half of figure 4.9.

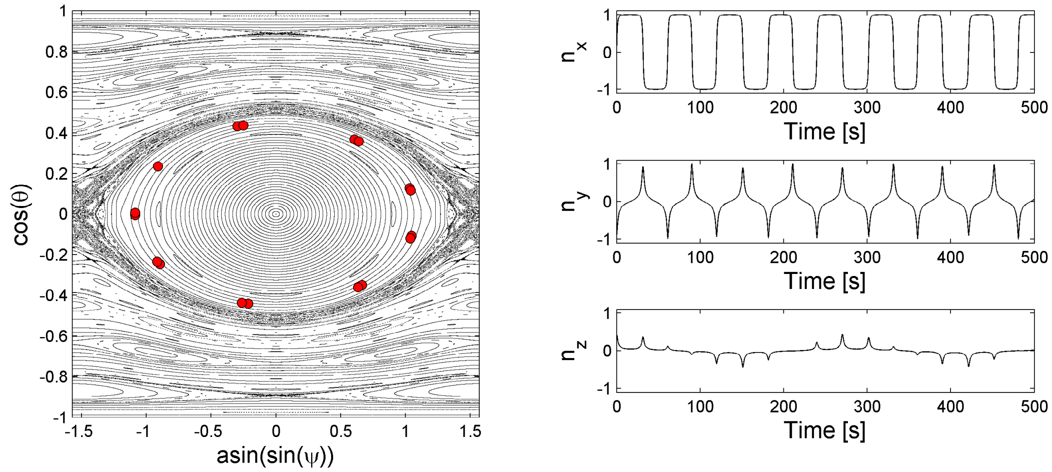


Figure 4.14: A Poincaré map for an ellipsoid $a_x = 10$, $a_y = 1.15$, and $a_z = 1$. This simulated trajectory bear resemblance to figures 4.6, 4.7, 4.10, and 4.11.

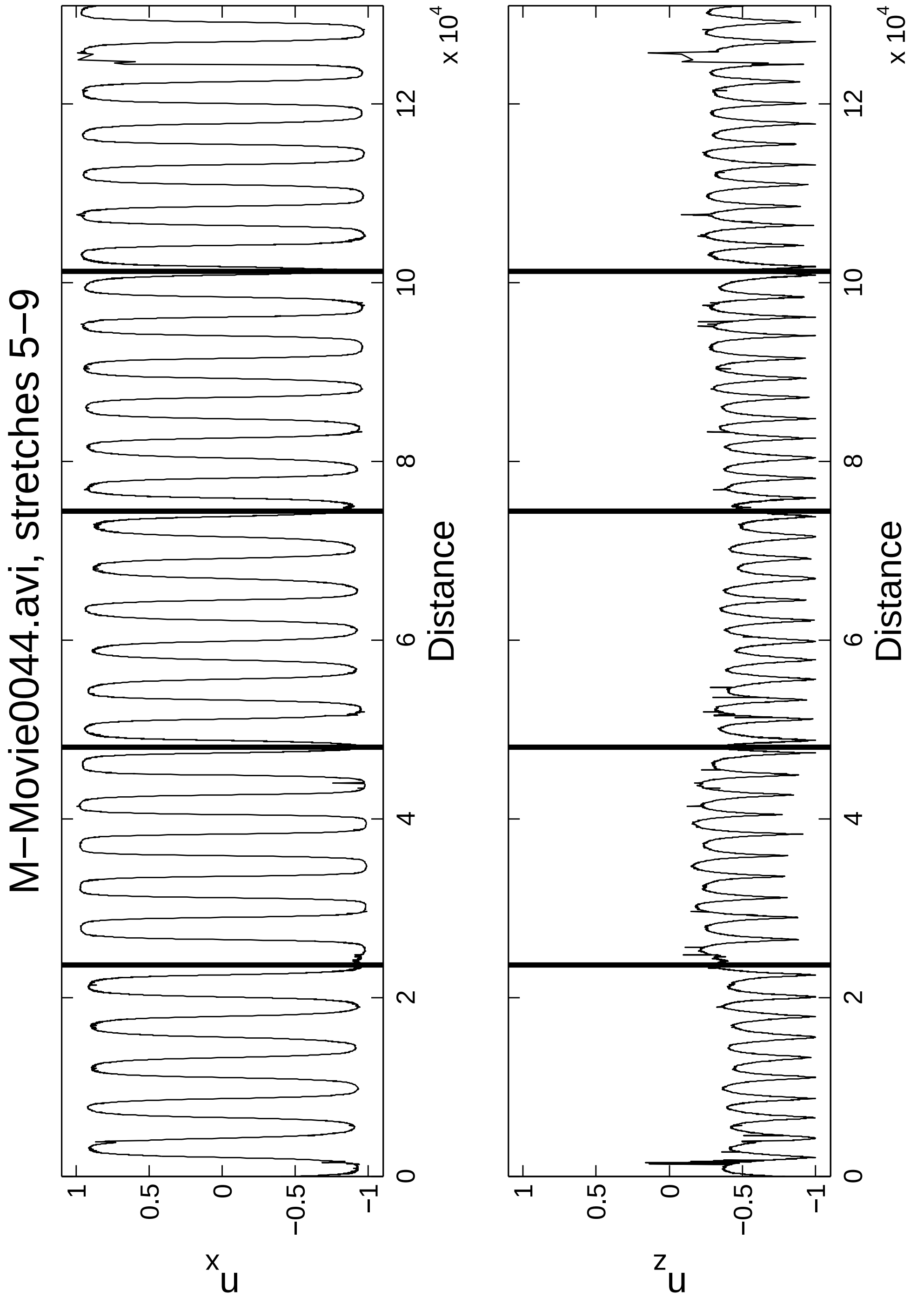


Figure 4.15: Several stitched laps for particle #1 shows a stable trajectory. The laps are reversed at each line, and the ideal situation would be a mirror image on both sides of a stitch.

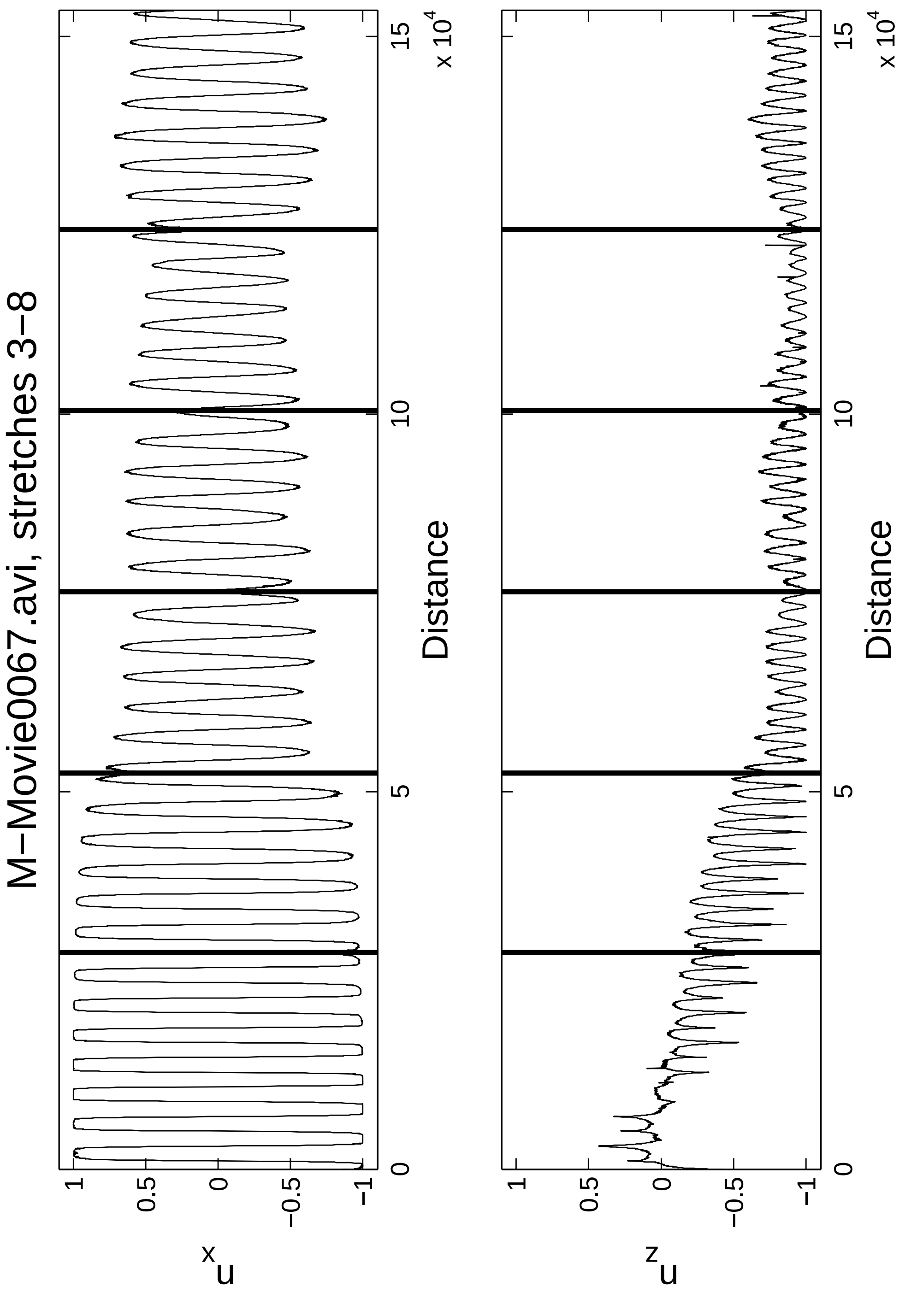


Figure 4.16: Several stitched laps for particle #4 shows an unstable trajectory that drifts towards high values of n_z. The laps are reversed at each line, and the ideal situation would be a mirror image on both sides of a stitch.

5

Discussion

Mishra et. al. [1] pose the question of why the behaviour of the trajectories in experimental data differs from the trajectories predicted by Jeffery [3]. They give two possible causes, the effects of particle asymmetry and the influence of thermal noise, however in [1] there is not enough data to draw a conclusion regarding which effect that is dominant, and under what conditions. In order to come to a conclusion much longer time series are needed, which would require either better flow control, or a longer channel. The trajectories analysed in this thesis was many times longer than the trajectories analysed by Mishra et. al., and the total number of laps analysed was also much grater. In this thesis it was shown that a there is a qualitative agreement between the trajectories from the experiment, and simulated trajectories for asymmetric particles.

If it turns out to be too hard to reliably create longer time series it might be possible to analyse the data in other ways. One way would be to look at scatter plots for the value of n_z at times when $n_x = 0$. It is possible that scatter plots for asymmetry and thermal noise behave differently. The benefit would be that it is not as important to have long laps, longer than what can be achieved with the current channel, but it would be enough with many recorded laps.

It is important that the rod is long, relative to its girth. If not, when $|n_y|$ becomes large the projection of the orientation vector will not reach the edge of the particle shadow, see figure 5.1. Without a more sophisticated method of fitting the projection which uses other available information, such as the current trajectory, it is not possible to recover the orientation vector $\boldsymbol{n}$.

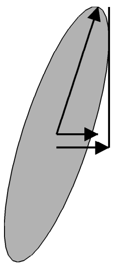


Figure 5.1: The projection of $\boldsymbol{n}$ does not reach the border of the particle shadow, this means that its length cannot be retrieved through normal ellipse fitting.

5.1 Camera tracking

The engine of the tracking software was written before the recording of the particle position was implemented and so the tracking software relies purely on image analysis and object recognition to find the particle. The camera used to record the movies is manually controlled by an operator. This means that the relative particle position cannot always be reliably estimated between frames.

If instead the global particle position was used the predicted position will be less random. If the problem is more predictable it might be useful not only to include information from only one previous frame, but from several frames. One way to accomplish this would be to implement a Kalman filter, which is a linear quadratic estimation algorithm, to estimate the current state of the particle. Another possibility, given that it is possible to fetch images from the camera directly, is to use a Kalman filter during the actual recording to automate the control of the camera itself.

Bibliography

[1] Y. N. Mishra, J. Einarsson, O. A. John, P. Andersson, B. Mehlig, D. Hanstorp, A microfluidic device for the study of the orientational dynamics of microrods, in: Proceedings of SPIE - The International Society for Optical Engineering, Vol. 8251, 2012.

[2] E. Hinch, L. Leal, Rotation of small non-axisymmetric particles in a simple shear flow, Journal of Fluid Mechanics 92 (03) (1979) 591–607.

[3] G. Jeffery, The motion of ellipsoidal particles immersed in a viscous fluid, Proceedings of the Royal Society of London. Series A 102 (715) (1922) 161–179.

[4] A. L. Yarin, O. Gottlieb, I. V. Roisman, Chaotic rotation of triaxial ellipsoids in simple shear flow, Journal of Fluid Mechanics 340 (1997) 83–100.

[5] M. Wilkinson, H. R. Kennard, A model for alignment between microscopic rods and vorticity, submitted, arXiv:1205.6968v1 (2012).

[6] D. Vincenzi, Orientation of non-spherical particles in an axisymmetric random flow, submitted, arXiv:1206.0945v1 (2012).

[7] T. Kaya, H. Koser, Characterization of Hydrodynamic Surface Interactions of Escherichia coli Cell Bodies in Shear Flow, Phys. Rev. Lett. 103 (13) (2009) 1–4.

[8] S. Parsa, J. S. Guasto, M. Kishore, N. T. Ouellette, J. P. Gollub, G. A. Voth, Rotation and alignment of rods in two-dimensional chaotic flow, Physics of Fluids 23 (4).

[9] MATLAB, version 7.13.564 (R2011b), The MathWorks Inc., Natick, Massachusetts, 2011.

[10] J. Canny, A computational approach to edge detection, Pattern Analysis and Machine Intelligence, IEEE Transactions on PAMI-8 (6) (1986) 679 –698.

[11] A. Einstein, Eine neue Bestimmung der Moleküldimensionen, Annalen der Physik 324 (2) (1906) 289–306.

[12] J. Einarsson, Low Reynolds Number Particle Dynamics, Master's thesis, Chalmers University of Technology, Gothenburg, Sweden (2011).

[13] Y. Zheng, Poisson equation in a rectangle, University Lecture (2010).
URL `http://www.math.psu.edu/yzheng/m597k/m597kLVI8.pdf`

[14] A. J. Lichtenbert, M. A. Lieberman, Regular and Stochastic Motion, Springer, 1983.

[15] S. N. R. Wijewickrema, A. P. Papliński, Principal component analysis for the approximation of an image as an ellipse, in: 13th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision 2005, WSCG'2005 - In Co-operation with EUROGRAPHICS, Full Papers, 2005, pp. 69–70.

[16] R. Halíř, J. Flusser, Numerically stable direct least squares fitting of ellipses, in: Proceedings of the 6th International Conference in Central Europe on Computer Graphics and Visualization, 1998, pp. 125–132.