# Physics-Based Enrichment of Volumetric Velocity Measurements for Incompressible Flows

Azijli, Iliass

**DOI**
10.4233/uuid:fbe7a67d-334d-4c4e-8a7b-e37618644c81

**Publication date**
2016

**Document Version**
Final published version

**Citation (APA)**
Azijli, I. (2016). *Physics-Based Enrichment of Volumetric Velocity Measurements for Incompressible Flows*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:fbe7a67d-334d-4c4e-8a7b-e37618644c81

# PHYSICS-BASED ENRICHMENT OF VOLUMETRIC VELOCITY MEASUREMENTS FOR INCOMPRESSIBLE FLOWS

# PHYSICS-BASED ENRICHMENT OF VOLUMETRIC VELOCITY MEASUREMENTS FOR INCOMPRESSIBLE FLOWS

**Proefschrift**

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op dinsdag 7 juni 2016 om 10:00 uur

door

**Iliass AZIJLI**

ingenieur Luchtvaart- en Ruimtevaarttechniek
geboren te Amsterdam, Nederland.

Dit proefschrift is goedgekeurd door de

promotor: prof. dr. ir. drs. H. Bijl
copromotor: dr. R.P. Dwight

Samenstelling promotiecommissie:

| | |
|---|---|
| Rector Magnificus, | voorzitter |
| Prof. dr. ir. drs. H. Bijl, | TU Delft |
| Dr. R.P. Dwight, | TU Delft |

*Onafhankelijke leden:*

| | |
|---|---|
| Dr. R.G. Hanea, | Statoil, Noorwegen |
| Prof. dr. ir. A.W. Heemink, | TU Delft |
| Prof. E. Mémin, | INRIA, Frankrijk |
| Prof. dr. sc. techn. habil. J. Sesterhenn, | TU Berlin |
| Ir. B. Wieneke, | LaVision, Duitsland |
| Prof. dr. ir. P. Colonna, | TU Delft, reservelid |

An electronic version of this dissertation is available at
`http://repository.tudelft.nl/`.

*To all those who made this work possible, **thank you***
*you know who you are*

# CONTENTS

# SUMMARY

The goal of the present thesis is to enhance volumetric velocity data obtained from measurements of fluid flow in the incompressible regime. We focus on measurements obtained using tomographic particle image velocimetry (PIV) and three-dimensional particle tracking velocimetry (PTV). The usefulness of the measurement data can be limited by a number of factors:

- measurement noise;
- limited resolution (spatial or temporal);
- absence of estimates for the measurement uncertainty.

The current state-of-the-art algorithms that extract velocity fields from the measurements do not include physical models, raising the following question: can physics-based methods increase the usefulness of the measurement data by applying them as a postprocessing step to the extracted velocity fields? Four postprocessing methods are developed and investigated to:

- reduce the measurement noise of instantaneous tomographic PIV data;
- increase the accuracy of instantaneous 3D-PTV data that suffers from low spatial resolution;
- increase the accuracy of time-resolved 3D-PTV data that suffers from low spatial resolution;
- augment PIV-based pressure data with uncertainty estimates.

To reduce the measurement noise of instantaneous tomographic PIV data, we use *solenoidal Gaussian process regression* (SGPR). For incompressible flow, mass conservation dictates that the velocity field is divergence-free (solenoidal) and SGPR enforces analytically a divergence-free velocity field. SGPR is applied to synthetic and experimental test cases and compared with two other recently proposed solenoidal filters. For the synthetic test cases, SGPR is found to consistently return more accurate velocity, vorticity and pressure fields. From the experimental test cases, two important conclusions are drawn. Firstly, including an accurate model for the local measurement uncertainty further improves the accuracy of the velocity field filtered with SGPR. Secondly, all solenoidal filters result in an improved reconstruction of the pressure field, as verified with microphone measurements. Due to the sufficient spatial resolution of the PIV data, the parameters of SGPR are chosen *a priori*. The results obtained with SGPR are found to be relatively insensitive to the exact choice of these parameters, demonstrating the robustness of the filter.

To increase the accuracy of instantaneous 3D-PTV data that suffers from low spatial resolution, SGPR is again investigated, but with an important modification from the previous application: due to the sparseness of the data it becomes difficult to choose the parameters *a priori*. An efficient methodology is proposed that automatically finds the values for optimum interpolation based on the information provided by the measurement itself. Two approaches to enforce a solenoidal velocity field are investigated:

(i) a two-step approach, where initially, the velocity components are interpolated independently from one another onto a (fine) regular grid, followed by the application of a solenoidal filter, enforcing a divergence-free velocity field on a discrete level;

(ii) a one-step approach, where SGPR is used.

The interpolation methods are applied to synthetic and experimental data and show that enforcing the divergence-free constraint indeed results in consistently more accurate interpolations, where the one-step approach of SGPR is found to be superior to the two-step approach.

To increase the accuracy of time-resolved 3D-PTV data that suffers from low spatial resolution, we include additional physical knowledge in the form of the inviscid vorticity transport equation, discretized using the vortex-in-cell (VIC) method. The resulting spatio-temporal method is formulated as a minimization problem, where the objective function is the discrepancy between the measured velocity and the simulated velocity. The unknowns of the minimization problem are the initial velocity field and the time-varying velocity and vorticity boundary conditions. The minimization is efficiently solved by using a quasi-Newton method, supplied with exact discrete gradients obtained from a hand-coded adjoint code of the VIC method. Neglecting viscosity in the vorticity transport equation makes the VIC method time-reversible, avoiding the requirement from the adjoint to store the intermediate state. The spatio-temporal method is applied to an experimental test case and compared with SGPR, which interpolates the velocity fields at each time instant independently from one another. The spatio-temporal method results in clear improvements in the reconstructed velocity and vorticity fields.

Finally, to augment PIV-based pressure data with uncertainty estimates, we formulate a method based on the Bayesian framework. Briefly, using Bayes' rule, the posterior covariance matrix of the velocity field is obtained by combining the measurement uncertainty of the PIV velocity field with prior knowledge, like incompressibility of the velocity field. Once the posterior covariance matrix of the velocity is known, it is propagated through the discretized Poisson equation for pressure. Numerical assessment of this *a posteriori* uncertainty quantification method shows excellent agreement with Monte Carlo simulations. The method is applied to an experimental test case obtained using time-resolved tomographic PIV. The pressure reconstructed from the tomographic PIV data is compared to a microphone measurement conducted simultaneously at the wall to determine the actual error of the former. The comparison between true error and estimated uncertainty demonstrates the accuracy of the uncertainty estimates on the pressure. In addition, enforcing the divergence-free constraint is found to result in a significantly more accurate reconstructed pressure field. The estimated uncertainty confirms this result.

The developed postprocessing methods show that it is indeed possible to enhance volumetric velocity measurements of incompressible flows by incorporating physics-based numerical models.

# SAMENVATTING

Het doel van dit proefschrift is om volumetrische snelheidsdata te verbeteren, dat verkregen is met behulp van metingen van vloeistofstromingen in het niet-samendrukbare domein. We richten ons op metingen verkregen door middel van tomografische particle image velocimetry (PIV) en driedimensionale particle tracking velocimetry (PTV). Een aantal factoren kunnen de bruikbaarheid van de meetdata beperken:

- meetruis;
- beperkte resolutie (in de ruimte of tijd);
- afwezigheid van een schatting voor de meetonzekerheid.

De huidige geavanceerde algoritmes die snelheidsvelden van metingen afleiden, brengen fysische modellen niet in rekening, wat leidt tot de volgende vragen: kunnen op fysica gebaseerde methodes de nuttigheid van de meetdata verhogen door ze toe te passen als een nabewerkingstap op de verkregen snelheidsvelden? Vier nabewerkingmethodes zijn ontwikkeld en onderzocht om:

- de meetruis van instantane tomografische PIV data te verlagen;
- de nauwkeurigheid van instantane 3D-PTV data, met een lage ruimtelijke resolutie, te verhogen;
- de nauwkeurigheid van tijdsopgeloste 3D-PTV data, met een lage ruimtelijke resolutie, te verhogen;
- PIV afgeleide druk data te voorzien van onzekerheidsschattingen.

We gebruiken divergentievrije Gaussische proces regressie (SGPR) om de meetruis van instantane tomografische PIV data te verlagen. Voor een niet-samendrukbare stroming geldt, volgens de wet van behoud van massa, dat de snelheidsveld divergentievrij moet zijn en SGPR forceert analytisch een divergentievrije snelheidsveld. SGPR wordt toegepast op synthetische en experimentele testgevallen en vergeleken met twee andere recent voorgestelde divergentievrije filters. Voor de synthetische testgevallen volgt dat SGPR snelheids-, vorticiteits-, en drukvelden produceert die consistent nauwkeuriger zijn. Twee belangrijke conclusies volgen uit de experimentele testgevallen. Ten eerste, het meerekenen van een nauwkeurig model voor de lokale meetonzekerheid leidt tot verdere verbetering van de nauwkeurigheid voor de met SGPR gefilterde snelheidsdata. Ten tweede, alle divergentievrije filters leiden tot een verbeterde reconstructie van het drukveld. Dit is geverifieerd door middel van microfoonmetingen. Doordat de PIV data

voldoende ruimtelijke resolutie heeft, worden de parameters van SGPR van tevoren gekozen. De resultaten verkregen met SGPR blijken relatief ongevoelig te zijn voor de exact keus van deze parameters. Dit laat zien dat de filter robuust is.

SGPR wordt weer onderzocht om de nauwkeurigheid van instantane 3D-PTV data, met een lage ruimtelijke resolutie, te verhogen. Echter, met een belangrijke modificatie vergeleken met de vorige toepassing: de lage ruimtelijke resolutie van de data maakt het moeilijk om de parameters van tevoren te kiezen. Een effici´ente methode is voorgesteld dat automatisch de waardes vindt voor een optimale interpolatie, gebaseerd op de informatie geleverd door de metingen zelf. Twee manieren zijn onderzocht om een divergentievrije snelheidsveld af te dwingen:

(i) een tweestappen plan, waarbij de snelheidscomponenten eerst onafhankelijk van elkaar worden ge´interpoleerd op een (fijn) rooster, gevolgd door toepassing van een divergentievrij filter, dat een divergentievrij snelheidsveld op een discreet niveau afdwingt;

(ii) toepassen van SGPR (één stap).

De interpolatiemethoden worden toegepast op synthetische en experimentele data en laten zien dat het afdwingen van een divergentievrij snelheidsveld inderdaad resulteert in consistent nauwkeurigere interpolaties, waarbij SGPR superieur blijkt te zijn aan de tweestappenplan.

Om de nauwkeurigheid van tijdsopgeloste 3D-PTV data, met een lage ruimtelijke resolutie, te verhogen, rekenen we aanvullende fysische kennis mee in de vorm van de wrijvingsloze vorticiteit-transport vergelijking, gediscretiseerd door middel van de vortex-in-cell (VIC) methode. De resulterende ruimtetijd methode wordt geformuleerd als een minimalisatie probleem, waarbij de kostfunctie de discrepantie tussen het gemeten snelheidsveld en gesimuleerde snelheidsveld is. De onbekende variabelen van het minimalisatie probleem zijn het initi´ele snelheidsveld en de tijdsvari´erende snelheids- en vorticiteitsrandvoorwaarden. De minimalisatie wordt effici´ent opgelost door middel van een quasi-Newton methode, dat geleverd wordt met exact discrete gradi´enten verkregen met behulp van een handgecodeerde geadjungeerde code van de VIC methode. Het verwaarlozen van de viscositeit in de vorticiteit-transport vergelijking maakt de VIC methode tijdsomkeerbaar. Dit vermijdt de behoefte van de geadjungeerde code om de tussenliggende condities op te slaan. De ruimtetijd methode wordt toegepast op een experimentele testgeval en vergeleken met SGPR, dat de snelheidsvelden voor elke tijdstap onafhankelijk van elkaar interpoleert. De ruimtetijd methode leidt tot duidelijke verbeteringen in de gereconstrueerde snelheids- en vorticiteitsvelden.

Tenslotte, om PIV afgeleide druk data te voorzien van onzekerheidsschattingen, formuleren we een methode gebaseerd op de Bayes kader. In het kort, met de regel van Bayes verkrijgen we de posterior covariantie matrix van het snelheidsveld door de meetonzekerheid van het PIV snelheidsveld te combineren met voorafgaande kennis, zoals niet-samendrukbaarheid van het snelheidsveld. Zodra de posterior covariantie matrix van de snelheid bekend is, wordt het door de gediscretiseerde Poisson vergelijking voor druk gevoerd. Numerieke beoordeling van deze *a posteriori* onzekerheidskwantificatie methode toont uitstekende overeenkomsten met Monte Carlo simulaties. De methode wordt toegepast op een experimenteel testgeval, dat verkregen is door middel van tijdsopgeloste tomografische PIV. De uit tomografische PIV gereconstrueerde druk wordt vergeleken met een microfoonmeting dat op hetzelfde moment is toegepast op de wand om de fout van de eerstgenoemde te bepalen. De vergelijking tussen de werkelijke fout en de geschatte onzekerheid demonstreert de nauwkeurigheid van de onzekerheidsschattingen voor de druk.

De ontwikkelde nabewerkingmethodes tonen aan dat het inderdaad mogelijk is om volumetrische snelheidsmetingen van niet-samendrukbare stromingen te verbeteren door op fysica gebaseerde numerieke methodes mee te rekenen.

# 1

# INTRODUCTION

*Volumetric velocimetry techniques allow the three velocity components of a fluid flow to be measured in a volume. In Section 1.1, we explain the workings of particle image velocimetry (PIV), since the experimental data used in the present thesis were obtained with this technique. A number of factors can limit the usefulness of the measurement data, namely measurement noise (Section 1.2), limited resolution (Section 1.3) and the absence of estimates for the measurement uncertainty (Section 1.4). In each of the mentioned sections, we outline the physic-based postprocessing methods we have developed and investigated in the present thesis to deal with the limitations, therefore increasing the usefulness of the measurement data. We emphasize that these methods are postprocessing methods, meaning that they take the velocity data extracted from the measurement techniques as the input. Therefore, these methods are applicable no matter which specific algorithm was used to extract velocity fields from the measurements.*

## 1.1. VOLUMETRIC PIV

A number of velocimetry techniques, that allow the determination of the three velocity components in a volume of fluid, have recently become available. In particular: 3D particle tracking velocimetry (3D-PTV) (Maas et al, 1993), scanning particle image velocimetry (PIV) (Brücker, 1995), holographic PIV (Hinsch, 2002), magnetic resonance velocimetry (MRV) (Elkins et al, 2003), and most recently tomographic PIV (Elsinga et al, 2006). In the present thesis, we use data obtained from 3D-PTV and tomographic PIV and we will therefore explain these methods in more detail. For simplicity, we first explain PIV using the traditional and more widely used planar PIV, which returns two velocity components in a plane. Figure 1.1 illustrates the working principle. Briefly, PIV works as follows. The flow is seeded with tracer particles and the measurement domain is illuminated with a light source. Lasers are popular, since they can produce a high energy, pulsed, monochromatic light beam that can easily be shaped into a thin light sheet. This allows the imaging device to clearly capture the tracer particles in an approximately 2D illuminated region. With two images captured in rapid succession, one can derive the

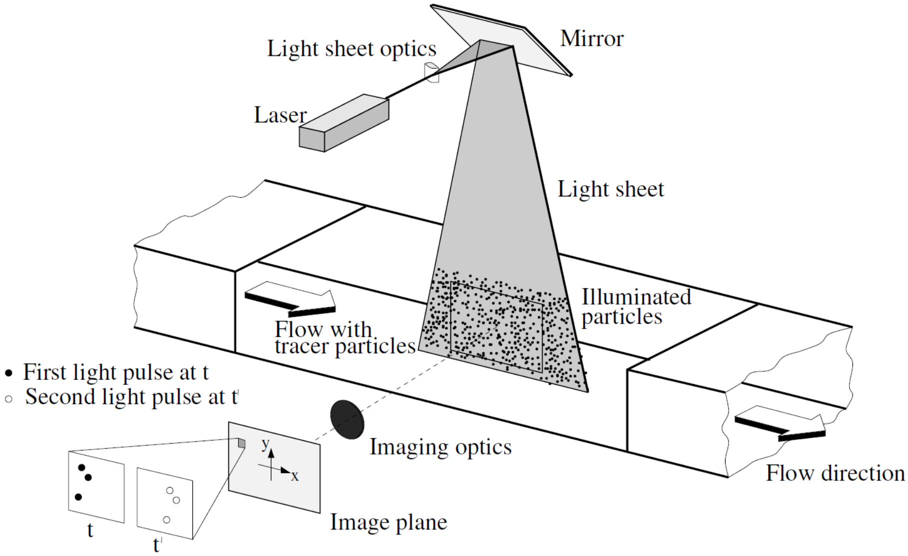
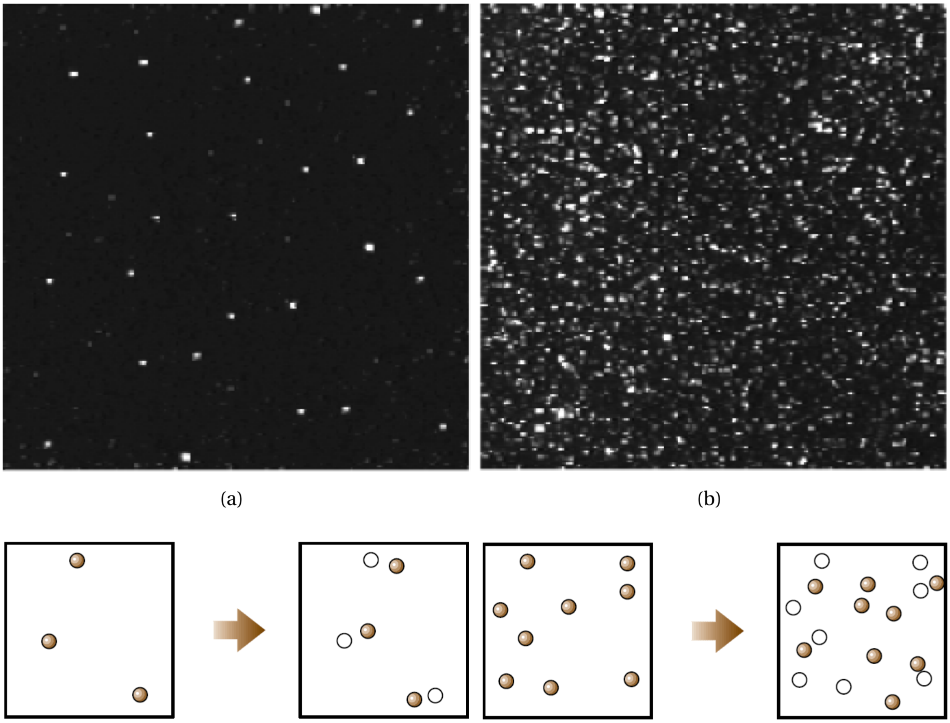


Figure 1.1: Working principle of planar PIV (from Raffel et al (1998)).

displacement of the tracer particles. From the known time separation of these images, the velocity of the tracer particles is obtained. Under the assumption that the tracer particles faithfully follow the fluid particles, the particle velocity is equal to the flow velocity.

In the above discussion, we mentioned the idea of extracting the particle displacement from successive images. If the distance between distinct particles is much larger than the displacement, then it is easy to evaluate the displacement from the individual particles. This *low-image-density* mode of PIV is generally referred to as *particle tracking velocimetry* (PTV). Figure 1.2(a) shows an image representative for the application of PTV. Figure 1.2(c) shows a simple schematic where PTV can be applied. The left figure represent the first image, the right figure overlays the first image onto the second image. It is easy to see that the displacement of individual particles can be tracked unambiguously. The velocity vectors that PTV therefore returns are randomly scattered over the domain, corresponding to the particle locations. If the particle displacement is larger than the distance between distinct particles, it is no longer possible to unambiguously track individual particles between the successive images. In this *high-image-density* mode of PIV, the particle displacement is determined using a statistical approach, where so-called *interrogation windows* containing a number of particles are cross-correlated between the successive images. Figure 1.2(b) shows an image representative for the application of the cross-correlation based technique in PIV. Figure 1.2(d) shows a simple schematic where cross-correlation based PIV is applied. Again, the left figure represent the first image and the right figure overlays the first image onto the second image. Clearly, it is difficult to track individual particles unambiguously. Due to the construction of the interrogation windows, the velocity vectors returned by cross-correlation based PIV are on a regular grid. In the following, we will refer to the high-image-density mode



(a) (b)

(c) (d)

Figure 1.2: Low-image-density (PTV) (a), high-image-density (PIV) (b), schematic of PTV (c), schematic of PIV (d). (a) and (b) obtained from Raffel et al (1998), (c) and (d) adapted from Tropea et al (2007).

or cross-correlation based technique of PIV as simply *PIV* and use the term *PTV* for the low-image-density mode, despite the fact that the experimental setup of the two modes is identical. Historically though, PIV has an advantage over PTV since higher particle densities allow higher spatial resolutions to be obtained.

Going from planar PIV to volumetric PIV requires the use of multiple cameras at different viewing angles to measure the displacement of the particles. In addition, the illumination source is now bundled into a volume rather than a thin light sheet. Similar to the distinction we made for planar PIV between the cross-correlation based mode and the particle tracking mode, we make a distinction for volumetric PIV. The cross-correlation based mode will be referred to as *tomographic PIV* and the particle tracking mode will simply be referred to as *3D-PTV*. Tomographic PIV relies on a tomographic reconstruction of the particle distribution, followed by three-dimensional cross-correlation. 3D-PTV on the other hand triangulates the particle positions and aims to find matching pairs in the successive images.

The current state-of-the-art algorithms that extract the three-dimensional velocity fields from the particle images do not make use of the underlying governing equations. Instead, they completely rely on the measurements. This holds both for the state-of-the-art algorithms of tomographic PIV (Lynch, 2015; Novara, 2013) and 3D-PTV (Novara et al, 2015; Schanz et al, 2014). However, since the volumetric techniques measure the three velocity components in a volume, they enable the *full* velocity-gradient tensor,

$$\nabla \mathbf{u} = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} & \frac{\partial u}{\partial z} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} & \frac{\partial v}{\partial z} \\ \frac{\partial w}{\partial x} & \frac{\partial w}{\partial y} & \frac{\partial w}{\partial z} \end{bmatrix},$$

to be determined. Access to the full velocity-gradient tensor allows less ambiguous analysis of coherent structures, especially when the flow of interest is highly three-dimensional. This is very relevant for the fundamental study of flows and turbulence. Also, as pointed out earlier, it opens up the possibility to include the underlying governing equations when processing the measurement data. For an incompressible flow, conservation of mass yields:

$$\nabla \cdot \mathbf{u} = 0, \tag{1.1}$$

where $\mathbf{u}$ is the fluid velocity. Eq.1.1 states that the velocity field is divergence-free (solenoidal). Conservation of momentum can be expressed as:

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho \left(\mathbf{u} \cdot \nabla\right) \mathbf{u} = -\nabla p + \mu \nabla^2 \mathbf{u}, \tag{1.2}$$

where $\rho$ is the fluid density, $t$ is time, $p$ is the pressure and $\mu$ is the dynamic viscosity. Eq.1.2 are better known as the Navier-Stokes equations. The present thesis investigates the use of physics-based postprocessing methods aimed at increasing the usefulness of the measurement data. The usefulness of the measurement data can be limited by a number of factors:

- measurement noise;
- limited resolution (spatial or temporal);
- absence of estimates for the measurement uncertainty.

The post-processing methods take the extracted velocity fields as input. Therefore, they can be applied to tomographic PIV and 3D-PTV, independent of the specific algorithm used to obtain the velocity field. The following three sections explain in more detail the postprocessing methods we have developed and investigated in the present thesis to:

- reduce the measurement noise of instantaneous tomographic PIV data;
- increase the accuracy of 3D-PTV data that suffers from low spatial resolution;
- augment PIV-based pressure data with uncertainty estimates.

## 1.2. MEASUREMENT NOISE

Measurement noise can be reduced by applying an appropriate filter to the velocity data. The simplest approach works through a convolution of the data using an $(n \times n)$ kernel, with either equal weights or distance-based weighting, e.g. Gaussians. Due to their simplicity, these averaging techniques are widely used, also in the field of PIV (Raffel et al, 1998). A more sophisticated approach is to fit a polynomial trough the measurement points. For both approaches, the filtering can be done in space, time, or space and time simultaneously (Scarano and Poelma, 2009). These filtering techniques do not take into account physical knowledge in terms of the known underlying conservation laws.

An alternative is known as *proper orthogonal decomposition* (POD) ((Raiola et al, 2015) and references therein). As the name suggests, the basic idea of the method is to decompose a dataset in such a way that the maximum amount of energy is captured by the smallest amount of orthogonal modes. With the assumption that the physical phenomena are contained in the high energy modes and the measurement noise in the low energy modes, the method aims to reduce the noise by only retaining the most energetic modes.

The above methods do not require volumetric measurements and do not restrict the flow to the subsonic regime. More recently proposed methods that do attempt to specifically enforce conservation laws in principle require incompressible flow and volumetric measurements. For example, de Silva et al (2013) and Schiavazzi et al (2014) proposed the use of solenoidal filters, which enforce mass conservation. Despite their restricted use, the advantage of these methods is that they do not require the user to specify tuning parameters, like kernel size or the number of modes to retain. In addition, application of these solenoidal filters to both synthetic and experimental test cases revealed that they result in a lower spatial attenuation, which is of course desirable since we do not wish to lose the true physics in the process of reducing the measurement noise. Furthermore, they do not require time-resolved data. Solenoidal filters can still be used if the measurements are only available in a plane, provided that the out-of-plane velocity gradient is negligible. Chapter 3 focuses on solenoidal filters and specifically we derive a version of Gaussian process regression that also includes measurement uncertainty explicitly. We call it *solenoidal Gaussian process regression* (SGPR).

Before proceeding, it is worth mentioning a recently proposed spatio-temporal filter that also includes the vorticity transport equation (Schneiders et al, 2015b). The idea of the method is to combine a short-sequence in time-resolved tomographic PIV measurements with multiple time-marching simulations, in which these simulations were started at the various snapshots of the measurements and integrated (forwards and backwards) towards the instantaneous time of interest. Like the solenoidal filters, a desirable property is its reduced spatial attenuation. However, contrary to the solenoidal filters, it requires time-resolved data.

## 1.3. LIMITED RESOLUTION

In the present thesis, we focus on limited *spatial* resolution instead of temporal resolution. The idea of increasing temporal resolution using underlying physical models has been investigated by Scarano and Moore (2012), who used a linearized advection model, and later by Schneiders et al (2014), who improved upon this idea by using the nonlinear momentum equation instead.

Increasing the spatial resolution of the data is mostly a concern in the case of data sets obtained with PTV, where the seeding concentration is lower than in PIV. In addition, there is a trend to move tomographic PIV to large scale applications, enabled by the use of helium-filled soap bubbles (HFSB) as seeding material (Scarano et al, 2015). The major problem with this type of tracer is obtaining sufficient seeding concentration. The low seeding density makes the use of particle tracking a preferred option and results in sparse data sets, where the velocity vectors are non-uniformly scattered over the domain of interest, corresponding to the location of individual tracer particles. This is contrary to tomographic PIV, which returns velocity vectors on a regular grid, allowing easy flow field visualization and obtaining derived quantities such as vorticity, vortex identification metrics or pressure. Therefore, there is a need to use scattered data interpolation methods, that are able to produce accurate fields on a regular grid, even for sparse data sets. Scattered data interpolation methods like radial basis function interpolation can be used for this (Casa and Krueger, 2013). A number of methods used Kriging (Gunes et al, 2006; Inggs and Lord, 1996; Lee et al, 2008). These methods only exploit the spatial information in the measurements, thereby being applicable to instantaneous measurements. Also, they do not include physical constraints. Chapter 4 uses SGPR to interpolate sparse 3D-PTV data, thereby enforcing the divergence-free constraint.

For time-resolved measurements, the possibility to exploit the temporal information becomes an option. For PIV data with spatial gaps, due to the absence of particles or limited optical access for illumination or imaging, Sciacchitano et al (2012a) used a finite-volume method to solve the unsteady incompressible Navier-Stokes equations in these gaps. The method essentially exploits the temporal information from the measurements and combines it with a physics-based model of the flow evolution. However, the method relies on fully specified time-resolved boundary conditions from the measurements since they are used as input to the solver. Chapter 5 investigates a method where the boundary conditions are part of the unknowns that need to be determined. The Navier-Stokes equations are used in their velocity-vorticity form, thereby removing the pressure term, which is not measured with PIV (Chorin et al, 1990):

$$\frac{\partial \boldsymbol{\omega}}{\partial t} + (\mathbf{u} \cdot \nabla)\boldsymbol{\omega} = (\boldsymbol{\omega} \cdot \nabla)\mathbf{u} + \nu \Delta \boldsymbol{\omega}, \tag{1.3}$$

where $\boldsymbol{\omega}$ is the vorticity and $\nu$ is the kinematic viscosity.

## 1.4. MEASUREMENT UNCERTAINTY

One can argue that a measurement is useless if no estimate of the measurement uncertainty is supplied with it. For a long time, the PIV community has relied on *a priori* estimates for the measurement uncertainty, with the so-called *empirical universal constant of 0.1 pixels* being a popular rule of thumb (Sciacchitano et al, 2013). The *a posteriori* uncertainty quantification of PIV fields has only become possible recently (Charonko and Vlachos, 2013; Sciacchitano et al, 2013; Timmins et al, 2012; Wieneke and Prevost, 2014). In the present thesis, we will not focus on developing an additional *a posteriori* uncertainty quantification for the velocity field. Instead, we will focus on augmenting the PIV-based pressure fields with uncertainty estimates. With the velocity field available, the pressure can be obtained by making use of the Navier-Stokes equations (van Oudheusden, 2013). Though extremely useful, not much attention has been given to a rigorous quantification of the uncertainties associated with PIV-based pressure fields, which is understandable considering that *a posteriori* estimates for the velocity field have only become available recently. Therefore, Chapter 6 focuses on quantifying the uncertainty of PIV-based pressure data.

# 2

# DATA ASSIMILATION

*This chapter explains data assimilation in detail, since the present thesis focuses on the application of physics-based postprocessing methods to measurement data. Specifically, we rely on the statistical perspective of data assimilation based on the Bayesian framework (Section 2.2), which provides a mathematically and statistically consistent formulation. The physics-based postprocessing methods that will be explained in subsequent chapters follow naturally from the Bayesian framework. In Section 2.3, we discuss data assimilation methods that only assimilate spatial fields, while in Section 2.4, we discuss data assimilation methods that assimilate spatio-temporal fields.*

## 2.1. INTRODUCTION

The idea of combining measurements with physical models is commonly referred to as *data assimilation*. Depending on the application and the people using it, there are a number of competing definitions. An appropriate definition for the present work is from Wikle and Berliner (2007):

> *Data assimilation is an approach for fusing data (observations) with prior knowledge (e.g. mathematical representations of physical laws; model output) to obtain an estimate of the distribution of the true state of a process.*

So, to perform data assimilation, one needs the following components:

(i) a physical model (usually numerical);

(ii) experimental observations;

(iii) a (statistical) model relating physical model and observations, including an error model for observations, and inadequacy of the physical model;

(iv) a means to fuse the two sources of information.

Notice that the definition refers to the *distribution* of the true state of the process, meaning that it refers to a statistical formulation of data assimilation rather than a deterministic one. Specifically, we will use the Bayesian perspective, which is described in more detail in Section 2.2. The Bayesian perspective provides a mathematically and statistically consistent formulation of the state estimation problem, where the prior knowledge and observations are both expressed in terms of probability density functions (pdf's). Indeed, the Bayesian approach is an inherently scientific approach since it is analogous to the scientific method: one has a prior belief about a certain process, collects data and then updates that belief given this new data.

Though general and powerful, the formulation that follows from Section 2.2 is complex and generally requires assumptions and simplifications to be made to allow for its practical application. Sections 2.3 and 2.4 will list a number of practical data assimilation methods, state the assumptions they are based on and point out the equivalences between them. Section 2.3 will focus on methods that only assimilate spatial fields, so the temporal dimension is not included. Section 2.4 will consider spatio-temporal fields. This subdivision reflects the type of volumetric velocity measurements we consider in the present work and the type of physical knowledge we include.

## 2.2. A BAYESIAN PERSPECTIVE

The definition of data assimilation mentioned the *state* of a process[1]. In this chapter, we will simply designate it $\boldsymbol{\phi}(\mathbf{x}, t) : \mathbb{R}^3 \times \mathbb{R}_+ \to \mathbb{R}^{n_\phi}$, i.e. it is a vector containing $n_\phi$ variables, where each variable is a function of space $\mathbf{x}$ and time $t$. In the present work, it will primarily be the velocity field, but at this stage it is not important to mention what it entails. The introductory section mentioned the prerequisites for data assimilation, namely statistical models for the *prior, measurements* and finally a means to fuse these two sources of information, i.e. obtain the *posterior.*

### 2.2.1. THE PRIOR KNOWLEDGE

The state is defined on a spatial domain $\Omega$ with boundary $\Gamma$. The underlying physics of fluids can be expressed generally as:

$$\frac{\partial \boldsymbol{\phi}(\mathbf{x}, t)}{\partial t} = \boldsymbol{\mathcal{M}}\left(\boldsymbol{\phi}(\mathbf{x}, t)\right) + \mathbf{q}(\mathbf{x}, t), \tag{2.1}$$

$$\boldsymbol{\phi}(\mathbf{x}, t_0) = \boldsymbol{\phi}_0(\mathbf{x}) = \boldsymbol{\Phi}_0(\mathbf{x}) + \mathbf{a}(\mathbf{x}), \tag{2.2}$$

$$\boldsymbol{\phi}(\mathbf{x}, t)|_\Gamma = \boldsymbol{\phi}_b\left(\boldsymbol{\xi}, t\right) = \boldsymbol{\Phi}_b\left(\boldsymbol{\xi}, t\right) + \mathbf{b}\left(\boldsymbol{\xi}, t\right). \tag{2.3}$$

Eq.2.1 expresses the time evolution of the state, where $\boldsymbol{\mathcal{M}}$ is a nonlinear operator. The initial condition is represented by Eq.2.2 and the boundary condition by Eq.2.3, where the coordinate $\boldsymbol{\xi}$ runs over the boundary $\Gamma$. Unknown error terms representing the error in the model, initial and boundary conditions are represented by the random processes **q**, **a** and **b**, respectively.

---

[1] In the present work, we only consider the state estimation problem. For the more general combined parameter and state estimation problem, we refer to the book by Evensen (2009).

### 2.2.2. THE MEASUREMENT MODEL

We express the measurements as:

$$\mathbf{d} = \mathcal{H}\left(\boldsymbol{\phi}\left(\mathbf{x}, t\right)\right) + \boldsymbol{\epsilon}, \tag{2.4}$$

where $\mathbf{d} \in \mathbb{R}^m$, i.e. $m$ measurements and $\boldsymbol{\epsilon}$ represents the measurement error. Typically, $\boldsymbol{\epsilon} \sim \mathcal{N}\left(0, R\right)$. With $\mathbf{q}$,$\mathbf{a}$,$\mathbf{b} = 0$, Eqs.2.1-2.3 represent a deterministic system that is well-posed. Therefore, adding the measurements from Eq.2.4 leads to an overdetermined system. Such a system could be resolved with e.g. a least-squares approach but that would be an *arbitrary* choice. Modeling the uncertainties in each equation using $\mathbf{q}$,$\mathbf{a}$,$\mathbf{b}$ and $\boldsymbol{\epsilon}$ allows us to find an informed, statistical resolution.

### 2.2.3. THE POSTERIOR

Since $\mathbf{q}$,$\mathbf{a}$,$\mathbf{b}$ are random processes, $\boldsymbol{\phi}$, $\boldsymbol{\phi}_0$, $\boldsymbol{\phi}_b$ become random processes and $\mathbf{d}$ is a multivariate random variable. They will therefore be represented using a probability density function $\rho\left(\cdot\right)$. Ultimately, we are interested in the following distribution:

$$\rho\left(\boldsymbol{\phi}, \boldsymbol{\phi}_0, \boldsymbol{\phi}_b \middle| \mathbf{d}\right),$$

which we regard as the solution of the system (2.1-2.4). It represents the distribution of the complete state (including initial and boundary conditions), *given* the measurements. This distribution can be calculated by making use of Bayes' rule. If for simplicity we neglect $\boldsymbol{\phi}_0$ and $\boldsymbol{\phi}_b$ for now, we have:

$$\rho\left(\boldsymbol{\phi}|\mathbf{d}\right) = \frac{\rho\left(\boldsymbol{\phi}\right)\rho\left(\mathbf{d}|\boldsymbol{\phi}\right)}{\rho\left(\mathbf{d}\right)}, \tag{2.5}$$

from the definition of conditional probability (Bayes, 1763). Naming the various terms:

(i) $\rho\left(\boldsymbol{\phi}\right)$ is the *prior*, since the measurements have not been incorporated yet;

(ii) $\rho\left(\mathbf{d}|\boldsymbol{\phi}\right)$ is the *likelihood*, the probability of observing the data for a given state;

(iii) $\rho\left(\mathbf{d}\right)$ is the *evidence*, the marginal probability of observing the data, independent of the state and therefore a constant for the purposes of investigating $\rho\left(\boldsymbol{\phi}|\mathbf{d}\right)$;

(iv) $\rho\left(\boldsymbol{\phi}|\mathbf{d}\right)$ is the posterior and ultimately the distribution we are interested in.

Neglecting the denominator in Eq.2.5, Bayes' rule reduces to:

$$\rho\left(\boldsymbol{\phi}|\mathbf{d}\right) \propto \rho\left(\boldsymbol{\phi}\right)\rho\left(\mathbf{d}|\boldsymbol{\phi}\right). \tag{2.6}$$

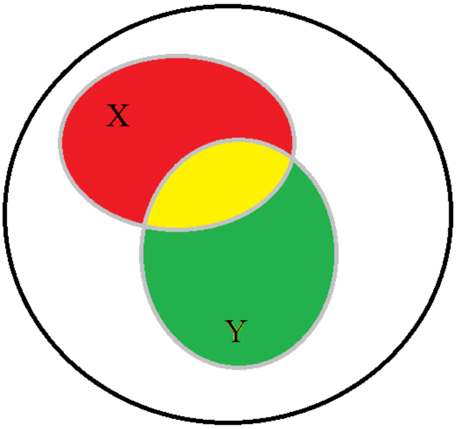


Figure 2.1: Venn diagram of possible outcomes for $X$ and $Y$.

SIDE NOTE: A GRAPHICAL DERIVATION TO BAYES' RULE

Consider two events $X$ and $Y$. Figure 2.1 shows a Venn diagram in the space of possible outcomes for these two events. The red region represent the probability of $X$ occurring ($\rho(x)$), while the green region represents the probability of $Y$ occurring ($\rho(y)$). The yellow region is where both occur ($\rho(x,y)$). From Figure 2.1, it is easy to see that

$$\rho(x|y) = \frac{\rho(x,y)}{\rho(y)}. \tag{2.7}$$

Indeed, $\rho(x,y)$ should be scaled with $\rho(y)$ since $Y$ has occurred. The same can be done for the reverse conditional distribution:

$$\rho(y|x) = \frac{\rho(x,y)}{\rho(x)}. \tag{2.8}$$

Combining Eqs.2.7 and 2.8 gives Bayes' rule:

$$\rho(x|y) = \frac{\rho(y|x)\,\rho(x)}{\rho(y)}. \tag{2.9}$$

Since the complete state is accompanied by the initial and boundary conditions, Eq.2.6 is slightly more involved. Consider the joint pdf:

$$\rho(\boldsymbol{\phi},\boldsymbol{\phi}_0,\boldsymbol{\phi}_b) = \rho(\boldsymbol{\phi}_0)\,\rho(\boldsymbol{\phi}_b|\boldsymbol{\phi}_0)\,\rho(\boldsymbol{\phi}|\boldsymbol{\phi}_0,\boldsymbol{\phi}_b). \tag{2.10}$$

Assuming the boundary conditions and initial conditions are independent:

$$\rho(\boldsymbol{\phi}_b|\boldsymbol{\phi}_o) = \rho(\boldsymbol{\phi}_b),$$

Eq.2.10 becomes:

$$\rho(\boldsymbol{\phi},\boldsymbol{\phi}_0,\boldsymbol{\phi}_b) = \rho(\boldsymbol{\phi}_0)\,\rho(\boldsymbol{\phi}_b)\,\rho(\boldsymbol{\phi}|\boldsymbol{\phi}_0,\boldsymbol{\phi}_b). \tag{2.11}$$

Using Bayes' rule, we get the following posterior distribution:

$$\rho\left(\boldsymbol{\phi},\boldsymbol{\phi}_0,\boldsymbol{\phi}_b|\mathbf{d}\right) \propto \rho\left(\boldsymbol{\phi}|\boldsymbol{\phi}_0,\boldsymbol{\phi}_b\right)\rho\left(\boldsymbol{\phi}_0\right)\rho\left(\boldsymbol{\phi}_b\right)\rho\left(\mathbf{d}|\boldsymbol{\phi}\right). \tag{2.12}$$

The term $\rho\left(\boldsymbol{\phi}|\boldsymbol{\phi}_0,\boldsymbol{\phi}_b\right)$ gives the prior density for the model solution *given* the initial and boundary conditions. Eq.2.12 is the general formulation to fuse the measurement model from Eq.2.4 with the prior physical model from Eqs.2.1-2.3. Though it looks simple, evaluating these expressions for general distributions and a nonlinear physical model is complicated and computationally expensive, especially for high dimensional problems. These are indeed of interest considering that we are investigating time varying three-dimensional flow fields. Therefore, a number of assumptions need to be made to allow for simplifications, consequently resulting in computationally tractable procedures. A key example is Gaussian processes, where we deal with multivariate normals and linear(ised) model operators. The next two sections describe a number of data assimilation methods that can result from such simplifications.

A vast number of data assimilation methods have been developed and there are various ways to classify them. For example, methods for which the underlying physical model is linear versus nonlinear. Another approach is to categorize into methods based on estimation theory, control theory, direct minimization, and stochastic approaches (Robinson and Lermusiaux, 2000). We do not prefer such a categorization since there is a strong overlap between these groups. Instead, in the present work we choose a classification that follows the type of volumetric velocity measurements available, namely single instantaneous measurements versus time-resolved measurements.

## 2.3. ASSIMILATING SPATIAL FIELDS

Removing the temporal dimension, the first obvious resulting simplification is that the state only depends on the spatial location: $\boldsymbol{\phi}(\mathbf{x})$. Therefore, there is no time-derivative and no initial condition. In the following, we assume Gaussian processes for the prior $\rho\left(\boldsymbol{\phi}\right)$ and a Gaussian measurement model for the likelihood $\rho\left(\mathbf{d}|\boldsymbol{\phi}\right)$. This assumption is not a major limitation and in fact will eventually lead to a powerful data fitting procedure known as *Gaussian process regression* (GPR) (Rasmussen, 2006). Additionally, we discretize the process at $n$ spatial locations.

### 2.3.1. THE PRIOR

Consider an unobservable Gaussian process with mean $\mu$ and covariance function $\psi$ (Rasmussen, 2006):

$$\Phi \sim \mathscr{GP}\left(\mu\left(\mathbf{x}\right),\psi\left(\mathbf{x},\mathbf{x}^i;\boldsymbol{\chi}\right)\right),$$

where $\mathbf{x} \in \mathbb{R}^3$ is the spatial coordinate and $\boldsymbol{\chi} \in \mathbb{R}^h$ are hyperparameters. Discretizing the process at $n$ spatial locations, gives a multivariate normal random-variable:

$$\begin{aligned}
&\boldsymbol{\Phi} \sim \mathscr{N}\left(\boldsymbol{\mu},P\right),\\
&\mu_i = \mu\left(\mathbf{x}^i\right),\\
&P^{ij} = \psi\left(\mathbf{x}^i,\mathbf{x}^j;\boldsymbol{\chi}\right), \quad i,j = 1,\dots,n.
\end{aligned} \tag{2.13}$$

Its realization is the state vector $\boldsymbol{\phi} \in \mathbb{R}^n$. We assume a stationary process, i.e. the correlation between two points $\mathbf{x}^i$ and $\mathbf{x}^j$ depends only on their separation, $\mathbf{x}^i - \mathbf{x}^j$, and not their absolute positions. The covariance function then simplifies to $\psi\left(r^{ij}\right)$, where $r^{ij}$ is the Euclidean distance between the points $\mathbf{x}^i$ and $\mathbf{x}^j$, defined as:

$$(r^{ij})^2 = \sum_{k=1}^{3} \left( \frac{x_k^i - x_k^j}{\gamma_k} \right)^2 , \tag{2.14}$$

where parameter $\gamma_k$ is the correlation length in the direction $k$.

### 2.3.2. THE MEASUREMENT MODEL

The measurement model is chosen as $\mathbf{d} = H\boldsymbol{\phi} + \boldsymbol{\epsilon}$, where $\mathbf{d} \in \mathbb{R}^m$ contains the $m$ observations, $H$ is the observation matrix and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, R)$ (assuming unbiased noise). Therefore, $\mathbf{D}|\boldsymbol{\phi} \sim \mathcal{N}\left(H\boldsymbol{\phi}, R\right)$. The matrix $R$ is known as the *observation error covariance matrix*, representing the measurement uncertainty. By using the matrix $H$, we assume the relation between the state and the observations is linear. This is acceptable since the measurements in the present work are velocity measurements, and the state is the velocity (or partial derivatives of the velocity potential). In addition, the PIV-filtering mentioned in Chapter 3.1 is a linear operation.

### 2.3.3. THE POSTERIOR

Due to exclusion of the time evolution model, the posterior distribution given by Eq.2.12 reduces to:

$$\rho\left(\boldsymbol{\phi}|\mathbf{d}\right) \propto \rho\left(\boldsymbol{\phi}\right) \rho\left(\mathbf{d}|\boldsymbol{\phi}\right). \tag{2.15}$$

Since both the prior and measurement model are assumed Gaussian and all operators are linear, the posterior will be Gaussian as well (Wikle and Berliner, 2007). The posterior mean and covariance are:

$$\mathbb{E}\left(\boldsymbol{\Phi}|\mathbf{d}\right) = \boldsymbol{\mu} + PH'\left(R + HPH'\right)^{-1}\left(\mathbf{d} - H\boldsymbol{\mu}\right), \tag{2.16}$$

$$\Sigma\left(\boldsymbol{\Phi}|\mathbf{d}\right) = \left(I - PH'\left(R + HPH'\right)^{-1} H\right) P. \tag{2.17}$$

The term $R + HPH'$ will be referred to as the *gain matrix* $A$, i.e.

$$A = R + HPH'. \tag{2.18}$$

### 2.3.4. VARIATIONS AND RELATED METHODS

Depending on the field of application, the described method goes by different names. In machine learning it is most well known as Gaussian process regression (Rasmussen, 2006), which we have called it here as well to emphasize our use of a Gaussian prior and measurement model. In geostatistics it is better known as *Kriging* (Krige, 1951; Matheron, 1971). Depending on the prior mean used, there are a number of variations. For example, *simple Kriging* takes a known prior mean, *ordinary Kriging* takes a prior mean that is constant but unknown and *universal Kriging* takes a linear function of covariates for the mean (Cressie, 2015). In the fields of meteorology and oceanography, the method is better known as *Optimal Interpolation* (Talagrand, 1997). We specifically mention the work by de Baar (2014), who focused on the application of Kriging to both experimental fluid dynamics and computational fluid dynamics and in addition investigated the use of *gradient-enhanced Kriging*.

It is interesting to mention radial basis function (RBF) interpolation, a widely used method for scattered data approximation (Wendland, 2005). There are some equivalences with GPR (Anjyo and Lewis, 2011; Fasshauer, 2011; Forrester et al, 2008). Rather than expressing covariances between points using a covariance function, RBF interpolation expresses the interpolant in terms of a sum of radial basis functions. Practically though, they lead to the same prior covariance matrix $P$. Also, with RBF interpolation the data are interpolated. With GPR, the sample points are treated as having measurement uncertainty, which is expressed through the observation error covariance matrix $R$. In essence then, RBF interpolation sets $R = 0$.

Finally, we present a more compact version of the posterior mean that will also be useful in the following chapters. First of all, we set the prior mean equal to zero. This is equivalent to simple Kriging. From Eqs.2.16 and 2.18 we can write:

$$\mathbf{c} = A^{-1}\mathbf{d}. \tag{2.19}$$

The vector **c** will also be referred to as the *weight vector*. Given a set of locations at which we want to evaluate the posterior, we can compactly write this as:

$$\boldsymbol{\phi} = C\mathbf{c}, \tag{2.20}$$

where

$$C^{ij} = \psi\left(\mathbf{x}^i, \mathbf{x}^j\right), \tag{2.21}$$

with $\mathbf{x}^i$ an evaluation point and $\mathbf{x}^j$ a measurement location.

### 2.3.5. 1D EXAMPLE OF GAUSSIAN PROCESS REGRESSION

We present an artificial one dimensional fitting problem. Consider the true function

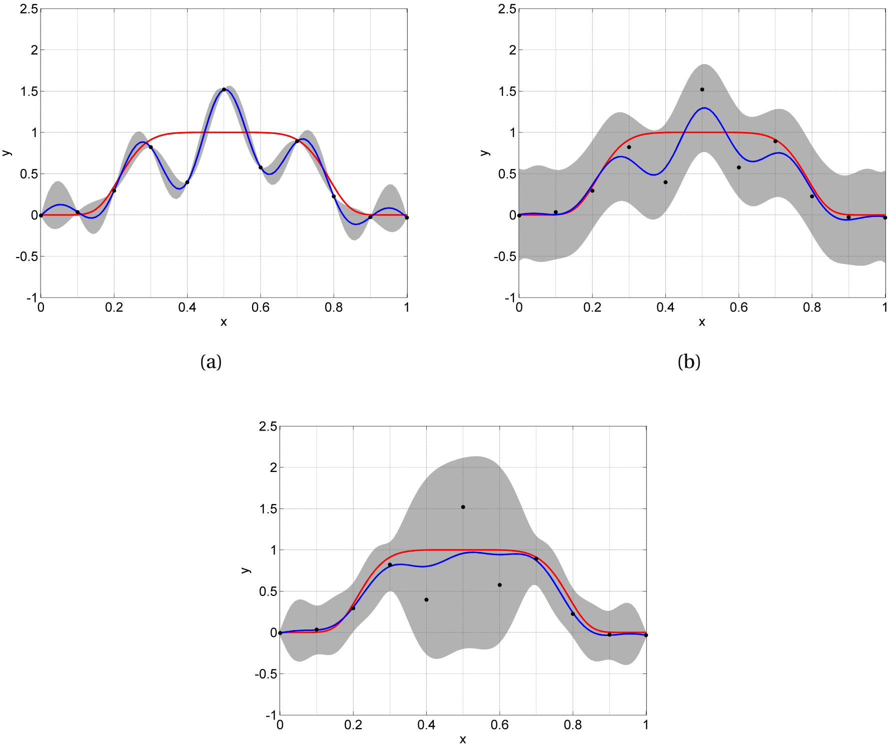
$$h(x) := \exp\left[-54\,(x-0.5)^6\right],$$

observed at eleven equally spaced measurement locations on the interval $[0,1]$ with added noise. The measurements do not all have the same noise level (a common situation in e.g. PIV). For simplicity, we set the measurement uncertainty $R$ to a diagonal matrix (i.e. uncorrelated noise), where the elements on the diagonal are represented by the vector

$$\begin{aligned}\mathbf{r} =& (10^{-4}, 10^{-2}, 10^{-2}, 10^{-2}, 0.25, 0.25, 0.25, \ldots\\ & 10^{-2}, 10^{-2}, 10^{-2}, 10^{-4}).\end{aligned}$$

Notice that the measurements at the boundaries of the interval are more accurate than measurements in the center. We use a Gaussian covariance function

$$\psi_G\left(\mathbf{x}^i, \mathbf{x}^j; \gamma\right) := \exp\left[-\left(\frac{x^i - x^j}{\gamma}\right)^2\right],$$



(a)

(b)

(c)

Figure 2.2: Reconstructions of the function $h(x)$ (red line) based on eleven measurements (black dots), assuming perfect observations (a), uniform measurement uncertainties (b) and spatially varying measurement uncertainties (c). A reconstruction is given by a mean field (blue lines) with confidence intervals of $\pm 3$ standard deviations (grey regions).

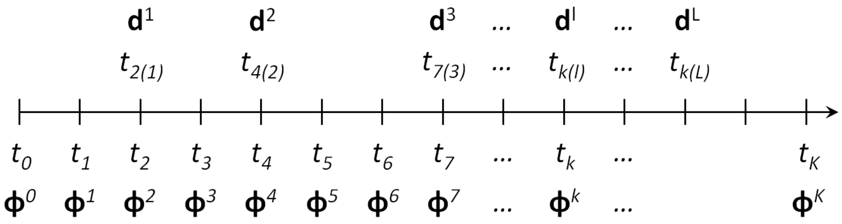


Figure 2.3: Discretization of the state $\boldsymbol{\phi}^k$ and the availability of the measurements $\mathbf{d}^l$ in time (adapted from Evensen (2009).

with $\gamma = 0.15$. Eqs.2.16 and 2.17 are then used to reconstruct the true function given the eleven measurements, and the (known) error covariance. Figure 2.2 shows the result. We have taken $n = 101$, uniformly distributed over the domain, so the function is evaluated at significantly more points than the measurement grid. $H$ is a sparse matrix with a few 1's picking the 11 measurement points from the 101 points. We have considered three cases in terms of our prior knowledge of the measurement uncertainty: (i) $R = 0$ (see Figure 2.2(a)), wrongfully assuming that the measurements are perfect. As a result, the reconstructed function interpolates the observed points. At these locations, the posterior uncertainty is zero. Notice that due to our over-confidence in the accuracy of the measurements, the true function is well outside the ±3 standard deviations region for a large part of the interval; (ii) (see Figure 2.2(b)), taking into account measurement uncertainty but wrongfully assuming that all measurements have equal uncertainty. The reconstructed function is more accurate than in the first case. Furthermore, the posterior uncertainty does not become zero at the measurement locations. Since we assume homogeneous measurement uncertainty and the observations are equally spaced, the confidence interval is very homogeneous throughout the interval; (iii) (see Figure 2.2(c)), taking into account the correct measurement uncertainty. This gives the most accurate reconstructed function of all three cases. Notice also that the process realizes the large uncertainty in the center of the interval and as a result, relies more on the prior. This is reflected by the large bounds of the confidence interval in this region.

## 2.4. ASSIMILATING SPATIO-TEMPORAL FIELDS

Since in practice the measurements will be available at a finite number of times and the model will also be integrated in time in a discrete way, we will represent the model state discretized in time. So, $\boldsymbol{\phi}(\mathbf{x}, t)$ is represented as $\boldsymbol{\phi}^k(\mathbf{x}) = \boldsymbol{\phi}(\mathbf{x}, t_k)$ with $k = 0, 1, ..., K$. The measurements $\mathbf{d} \in \mathbb{R}^m$ are available at times $t_{k(l)}$, where $l = 1, ..., L$, and $\mathbf{d}^l \in \mathbb{R}^{m_l}$. Figure 2.3 shows how the state and measurements are available in time. It is common to assume that the model is a first-order Markov process, meaning that the state at a new time $t_k$ only depends on the state at the previous time step $t_{k-1}$. We will use such a model in Chapter 5. Indeed, assuming such a model is not a strong assumption or simplification. Even if the assumption does not hold, meaning that the model errors are time correlated, the problem can be reformulated as a first-order Markov process again by augmenting the model errors to the model state vector (Evensen, 2003; Reichle et al, 2002). Due to the assumption of a first-order Markov process, the joint pdf given by Eq.2.11 can be expressed as follows:

$$\rho\left(\boldsymbol{\phi}^1,\ldots,\boldsymbol{\phi}^K,\boldsymbol{\phi}_0,\boldsymbol{\phi}_b\right)=\rho\left(\boldsymbol{\phi}_b\right)\rho\left(\boldsymbol{\phi}_0\right)\prod_{k=1}^{K}\rho\left(\boldsymbol{\phi}^k|\boldsymbol{\phi}^{k-1},\boldsymbol{\phi}_b\right). \tag{2.22}$$

If $\mathbf{d}^l$ only depends on $\boldsymbol{\phi}\left(t_{k(l)}\right)=\boldsymbol{\phi}^{k(l)}$ and the measurement errors are uncorrelated in time, the pdf for the measurement model can be expressed as:

$$\rho\left(\mathbf{d}|\boldsymbol{\phi}\right)=\prod_{l=1}^{L}\rho\left(\mathbf{d}^l|\boldsymbol{\phi}^{k(l)}\right). \tag{2.23}$$

From Bayes' rule the posterior becomes:

$$\rho\left(\boldsymbol{\phi}^1,\ldots,\boldsymbol{\phi}^K,\boldsymbol{\phi}_0,\boldsymbol{\phi}_b|\mathbf{d}\right)\propto\rho\left(\boldsymbol{\phi}_0\right)\rho\left(\boldsymbol{\phi}_b\right)\prod_{k=1}^{K}\rho\left(\boldsymbol{\phi}^k|\boldsymbol{\phi}^{k-1}\right)\prod_{l=1}^{L}\rho\left(\mathbf{d}^l|\boldsymbol{\phi}^{k(l)}\right). \tag{2.24}$$

As was done in Section 2.3, we will again assume Gaussian priors, making it possible to represent the pdf's using only the mean and covariance.

### 2.4.1. LINEAR DYNAMICS

In the case of linear dynamics, the Gaussian priors will remain Gaussian. Before measurements are included, the mean and covariance of the state are obtained from the model evolution. With the assumption of linear dynamics and due to the discretization, the nonlinear operator $\boldsymbol{\mathcal{M}}$ from Eq.2.1 becomes a matrix $M$. The model error is expressed as $\mathbf{q}\sim\mathcal{N}\left(0,C_{qq}\right)$. Therefore we can write:

$$\boldsymbol{\phi}^{k|k-1}=M_k\boldsymbol{\phi}^{k-1|k-1}, \tag{2.25}$$

$$P_{k|k-1}=M_kP_{k-1|k-1}M_k'+C_{qq,k}. \tag{2.26}$$

Eqs.2.25 and 2.26 are integrated forwards until a measurement becomes available. This is when the following updating equations are used:

$$\boldsymbol{\phi}^{k|k}=\boldsymbol{\phi}^{k|k-1}+P_{k|k-1}H_k'\left(R_k+H_kP_{k|k-1}H_k'\right)^{-1}\left(\mathbf{d}^l-H_k\boldsymbol{\phi}^{k|k-1}\right), \tag{2.27}$$

$$P_{k|k}=\left(I-P_{k|k-1}H_k'\left(R_k+H_kP_{k|k-1}H_k'\right)^{-1}H_k\right)P_{k|k-1}. \tag{2.28}$$

Compare Eqs.2.27-2.28 with Eqs.2.16-2.17; they are equivalent, since they follow from Bayes' rule, where all pdf's are Gaussian and all operators are linear. The procedure represented by Eqs.2.25-2.28 is known as the *Kalman filter* (Kalman, 1960). It is the *optimal* sequential data assimilation method for linear dynamics, given that the initial priors are Gaussian and unbiased.

### 2.4.2. NONLINEAR DYNAMICS

In the case of nonlinear dynamics, the initial Gaussian priors will not remain Gaussian, leading to the use of suboptimal methods. In this case, we will make a distinction between sequential methods that integrate the model step by step forwards in time, while assimilating observations as they become available, and variational methods that handle all observations at once.

### SEQUENTIAL METHODS

An alternative to the Kalman filter, known as the *Extended Kalman filter* (Anderson and Moore, 2012), works by approximating the nonlinear model operator by linearising it around the current state. This method is suboptimal and its usefulness depends on the strength of the nonlinearity of the model dynamics. The method can be improved by incorporating higher-order moments (Miller et al, 1994). However, this leads to larger storage and computational requirements.

Improved methods take an alternative approach, where the nonlinear dynamics is retained. Instead, the distribution of the state is approximated using a finite set of samples. The *Unscented Kalman Filter* (Julier and Uhlmann, 1997) chooses these samples deterministically and requires the number of sample points to be of the same order as the dimension of the state. The *Ensemble Kalman Filter* (Evensen, 1994) on the other hand chooses the samples randomly but requires a significantly smaller set of samples. According to Houtekamer and Mitchell (1998), an ensemble size of 100 members is often considered 'sufficient'.

### VARIATIONAL METHODS

Contrary to the sequential methods which update the state whenever measurements are available, for the variational methods the state depends on both future and past measurements. This is enabled by incorporating these measurements into a scalar cost function $J$ that subsequently needs to be minimised, therefore assimilating the observations at once. We explain how this cost function is obtained. Recall the components needed to calculate the posterior $\rho\left(\boldsymbol{\phi}^1,\ldots,\boldsymbol{\phi}^K,\boldsymbol{\phi}_0,\boldsymbol{\phi}_b|\mathbf{d}\right)$ (see Eq.2.24). For clarity, we also refer to the equations in Section 2.2 on which they are based:

(i) $\rho\left(\boldsymbol{\phi}_0\right)$, the pdf of the initial condition (originally from Eq.2.2);

(ii) $\rho\left(\boldsymbol{\phi}_b\right)$, the pdf of the boundary conditions (originally from Eq.2.3);

(iii) $\prod_{k=1}^{K}\rho\left(\boldsymbol{\phi}^k|\boldsymbol{\phi}^{k-1}\right)$, the pdf of the model evolution (originally from Eq.2.1);

(iv) $\prod_{l=1}^{L}\rho\left(\mathbf{d}^l|\boldsymbol{\phi}^{k(l)}\right)$, the pdf of the measurement model (originally from Eq.2.4).

We assume Gaussian priors for all components. This gives for the error terms:

(i) initial condition: $\mathbf{a}\sim\mathcal{N}\left(0,C_{aa}\right)$;

(ii) boundary condition at time $k$: $\mathbf{b}^k\sim\mathcal{N}\left(0,C_{bb,k}\right)$;

(iii) model evolution at time $k$: $\mathbf{q}^k\sim\mathcal{N}\left(0,C_{qq,k}\right)$;

(iv) observations at time $l$: $\boldsymbol{\epsilon}^l\sim\mathcal{N}\left(0,R_l\right)$.

We start with the initial condition:

$$\rho\left(\boldsymbol{\phi}_0\right)\propto\exp\left(-\frac{1}{2}\left(\boldsymbol{\phi}_0-\boldsymbol{\Phi}_0\right)'C_{aa}^{-1}\left(\boldsymbol{\phi}_0-\boldsymbol{\Phi}_0\right)\right), \tag{2.29}$$

where $\boldsymbol{\Phi}_0$ is the prior mean. Proceeding to the boundary condition:

$$\rho\left(\boldsymbol{\phi}_b\right)\propto\exp\left(-\frac{1}{2}\sum_{k=1}^{K}\left(\boldsymbol{\phi}_b^k-\boldsymbol{\Phi}_b^k\right)'C_{bb,k}^{-1}\left(\boldsymbol{\phi}_b^k-\boldsymbol{\Phi}_b^k\right)\right), \tag{2.30}$$

where $\boldsymbol{\Phi}_b^k$ is the prior mean of the boundary condition at time $t_k$. For the model evolution:

$$\prod_{k=1}^{K} \rho\left(\boldsymbol{\phi}^k | \boldsymbol{\phi}^{k-1}\right) \propto \exp\left(-\frac{1}{2}\sum_{k=1}^{K}\left(\frac{\boldsymbol{\phi}^k - \boldsymbol{\phi}^{k-1}}{\Delta t} - \boldsymbol{\mathcal{M}}\left(\boldsymbol{\phi}^k\right)\right)' C_{qq,k}^{-1}\left(\frac{\boldsymbol{\phi}^k - \boldsymbol{\phi}^{k-1}}{\Delta t} - \boldsymbol{\mathcal{M}}\left(\boldsymbol{\phi}^k\right)\right)\right), \tag{2.31}$$

Finally, the measurement model:

$$\prod_{l=1}^{L} \rho\left(\mathbf{d}^l | \boldsymbol{\phi}^{k(l)}\right) \propto \exp\left(-\frac{1}{2}\sum_{l=1}^{L}\left(\mathbf{d}^l - \boldsymbol{\mathcal{H}}^l\left(\boldsymbol{\phi}^{k(l)}\right)\right)' R_l^{-1}\left(\mathbf{d}^l - \boldsymbol{\mathcal{H}}_l\left(\boldsymbol{\phi}^{k(l)}\right)\right)\right), \tag{2.32}$$

Inserting Eqs.2.29-2.32 into Eq.2.24 gives:

$$\rho\left(\boldsymbol{\phi}^1, ..., \boldsymbol{\phi}^K, \boldsymbol{\phi}_0, \boldsymbol{\phi}_b | \mathbf{d}\right) \propto \exp\left(-\frac{1}{2} J\left(\boldsymbol{\phi}^1, ..., \boldsymbol{\phi}^K, \boldsymbol{\phi}_0, \boldsymbol{\phi}_b\right)\right), \tag{2.33}$$

where the cost function is:

$$\begin{aligned} J\left(\boldsymbol{\phi}^1, ..., \boldsymbol{\phi}^K, \boldsymbol{\phi}_0, \boldsymbol{\phi}_b\right) = {} & \left(\boldsymbol{\phi}_0 - \boldsymbol{\Phi}_0\right)' C_{aa}^{-1}\left(\boldsymbol{\phi}_0 - \boldsymbol{\Phi}_0\right) \\ & + \sum_{k=1}^{K}\left(\boldsymbol{\phi}_b^k - \boldsymbol{\Phi}_b^k\right)' C_{bb,k}^{-1}\left(\boldsymbol{\phi}_b^k - \boldsymbol{\Phi}_b^k\right) \\ & + \sum_{k=1}^{K}\left(\frac{\boldsymbol{\phi}^k - \boldsymbol{\phi}^{k-1}}{\Delta t} - \boldsymbol{\mathcal{M}}\left(\boldsymbol{\phi}^k\right)\right)' C_{qq,k}^{-1}\left(\frac{\boldsymbol{\phi}^k - \boldsymbol{\phi}^{k-1}}{\Delta t} - \boldsymbol{\mathcal{M}}\left(\boldsymbol{\phi}^k\right)\right) \\ & + \sum_{l=1}^{L}\left(\mathbf{d}^l - \boldsymbol{\mathcal{H}}_l\left(\boldsymbol{\phi}^{k(l)}\right)\right)' R_l^{-1}\left(\mathbf{d}^l - \boldsymbol{\mathcal{H}}_l\left(\boldsymbol{\phi}^{k(l)}\right)\right) \end{aligned} \tag{2.34}$$

Since we are using Gaussian priors, maximisation of the posterior is equivalent to minimising the cost function $J$. Therefore, the variational methods typically find the mode of the posterior. If the posterior is Gaussian, then the mode is equal to the mean. This happens in the case of linear dynamics, where the sequential and variational methods are equivalent.

The cost function $J$ has a global minimum, but it may not be unique in the case of nonlinear dynamics. It may also have a number of local minima. Therefore, there may be a risk that we do not converge to the global minimum. Taking a closer look at the cost function, we see that in the absence of measurements, the minimum is obtained at $J = 0$. In this case, the solution is equal to the prior model solution. On the other hand, in the presence of measurements, and more specifically *accurate* measurements, the posterior state will rely less on the simulation. This again emphasizes the inherently scientific approach of the Bayesian framework: if we do not have measurements, we completely rely on our prior physical knowledge. If we do have measurements, we will rely more on the measurements provided that they are accurate.

Comparing the variational with the sequential methods further, the first we note is that the variational method incorporates all time steps simultaneously. Even though we assumed the model errors to be uncorrelated in time and the measurement errors to be uncorrelated in time as well, the variational method does not require this. An additional difference between the two methods is that the variational method does not automatically return the uncertainty associated with the predicted state. Instead, as stated earlier, the solution we find is the mode of the posterior distribution. If gradients can be calculated efficiently using the adjoint approach, requiring one backward model integration, and assuming that the convergence is achieved quickly in the minimisation, variational methods can be more efficient than sequential methods.

# 3

# SOLENOIDAL FILTERING OF TOMOGRAPHIC PIV DATA

*This chapter investigates the use of Gaussian process regression to filter spurious velocity divergence from tomographic PIV data, returning analytically solenoidal velocity fields. We denote the filter Solenoidal Gaussian Process Regression (SGPR) and formulate it within the Bayesian framework to allow a natural inclusion of measurement uncertainty. In Section 3.2, we present a number of solenoidal filters proposed in the literature that have been applied to filter out spurious divergence from volumetric velocity measurements. In Section 3.3 we derive SGPR. The practical implementation of SGPR is subsequently discussed in Section 3.4. SGPR and two other recently proposed solenoidal filters are applied to synthetic test cases in Section 3.5 and experimental data in Section 3.6. For the synthetic test cases, we find that SGPR consistently returns more accurate velocity, vorticity and pressure fields. From the experimental test cases we draw two important conclusions. Firstly, it is found that including an accurate model for the local measurement uncertainty further improves the accuracy of the velocity field reconstructed with SGPR. Secondly, it is found that all solenoidal filters result in an improved reconstruction of the pressure field, as verified with microphone measurements. The results obtained with SGPR are insensitive to correlation length, demonstrating the robustness of the filter to its parameters. Conclusions are drawn in Section 3.7.*

## 3.1. INTRODUCTION

From the mass conservation equation for incompressible flows it follows that the velocity field is divergence-free: $\nabla \cdot \mathbf{u} = 0$. In the present chapter, we are concerned with PIV data, where the velocity field is obtained using the cross-correlation technique. A suitable model for the workings of the cross-correlation based PIV method is a weighted

---

Parts of this chapter have been published in Experiments in Fluids (Azijli and Dwight, 2015).

mean over the particles in the interrogation window:

$$\hat{\mathbf{u}} = \sum_{i=1}^{N} \rho_i \mathbf{u}(\mathbf{x}_i), \tag{3.1}$$

where $\mathbf{u}(\mathbf{x}_i)$ is the incompressible velocity field evaluated at $\mathbf{x}_i$, $\rho_i$ is the weighting at $\mathbf{x}_i$, $N$ is the number of particles in the window over which we sum and $\hat{\mathbf{u}}$ is the final velocity corresponding to the particular window. The question arises whether the obtained velocity $\hat{\mathbf{u}}$ is still divergence-free. To this end, we take the divergence of Eq.3.1:

$$\nabla \cdot \hat{\mathbf{u}} = \nabla \cdot \left( \sum_{i=1}^{N} \rho_i \mathbf{u}(\mathbf{x}_i) \right) = \sum_{i=1}^{N} \rho_i \nabla \cdot \mathbf{u}(\mathbf{x}_i) = \sum_{i=1}^{N} \rho_i \cdot 0 = 0. \tag{3.2}$$

If the divergence is evaluated analytically and no noise is present in the measurements, the low pass filtered divergence-free field is therefore still divergence-free. In practice though, the divergence is evaluated discretely and there is measurement noise. Therefore, the measured fields will have spurious non-zero divergence. It is possible to reduce this spurious divergence somewhat through ad hoc smoothing of the data, as shown by Zhang et al (1997). However, this results in a loss of resolution, without achieving a perfectly divergence-free velocity field. A more physically consistent approach is to enforce mass conservation in the filter itself.

It can be particularly advantageous to include this constraint when calculating physics-based derived quantities. For example, Sadati et al (2011) found a significant improvement in the calculation of stress and optical signal fields from velocity measurements of a viscoelastic flow when enforcing the divergence-free constraint. In this chapter, we demonstrate improvements in the calculation of pressure fields. Additionally, it is essential when the measurements are to be used as input for numerical simulations, e.g. for temporal flow reconstruction. In particular, in the field of PIV, Sciacchitano et al (2012a) proposed a method that uses the unsteady incompressible Navier-Stokes equations to fill spatial gaps. Similarly Schneiders et al (2014) proposed a vortex-in-cell method to fill temporal gaps. These methods currently build a solenoidal velocity field from measurement data in an ad hoc way, without accounting for measurement noise. This introduces an error in their input which pollutes the temporal reconstruction. Both methods could benefit from a preprocessing step that handles the noisy data and imposes mass conservation consistently.

## 3.2. SOLENOIDAL FILTERS

We have identified three general approaches to filter spurious divergence in the literature, based on three different principles.

### 3.2.1. HELMHOLTZ REPRESENTATION THEOREM

The first approach is based on the Helmholtz Representation Theorem. This states that any finite, twice differentiable vector field $\mathbf{u}$ can be decomposed into a solenoidal vector field $\mathbf{u}_{\text{sol}}$ plus an irrotational vector field $\mathbf{u}_{\text{irrot}}$ (Segel, 2007):

$$\mathbf{u} = \mathbf{u}_{\text{sol}} + \mathbf{u}_{\text{irrot}} = \nabla \times \mathbf{a} + \nabla \psi, \tag{3.3}$$

where **a** is a vector potential and $\psi$ is a scalar potential. Taking the divergence on both sides of Eq.3.3 and applying $\nabla \cdot \mathbf{u}_{\text{sol}} = 0$ gives a Poisson equation:

$$\nabla^2 \psi = \nabla \cdot \mathbf{u}. \tag{3.4}$$

Solving Eq.3.4 gives $\psi$, from which the solenoidal velocity field can be obtained using Eq.3.3. Difficulties arise in the specification of the boundary conditions, as in the absence of walls the boundary conditions are unknown. One extreme is to specify $\nabla\psi \cdot \mathbf{n} = \mathbf{u} \cdot \mathbf{n}$, i.e. the irrotational velocity normal to the boundary is equal to the measured velocity. As a consequence, the solenoidal velocity field normal to the boundary will be zero. The other extreme is to specify $\nabla\psi \cdot \mathbf{n} = 0$; the irrotational velocity normal to the boundary is zero, meaning that the solenoidal velocity normal to the boundary is equal to the measured velocity. This will result in an ill-posed PDE if the measured velocity field is not globally mass conserving. An additional complication arises when the vorticity field is to be calculated, because the vorticity of the solenoidal velocity is equal to that of the measured velocity (an irrotational velocity field has vanishing curl). Since the measured velocity field is contaminated by noise, the vorticity field of the solenoidal field will not be improved. A number of authors have followed the approach of the Helmholtz Representation Theorem in filtering velocity measurements Song et al (1993); Suzuki et al (2009); Worth (2012); Yang et al (1993).

### 3.2.2. LEAST-SQUARES VARIATIONAL FILTERS

The second approach is to define a minimisation problem. Liburdy and Young (1992) and Sadati et al (2011) defined an unconstrained minimisation problem, where the objective function consists of three terms:

(i) the discrepancy between the measured and filtered velocity field, expressed as a sum of squared differences;

(ii) the smoothness of the velocity gradients;

(iii) the divergence of the velocity field.

The latter two terms are multiplied with weighting parameters that must be supplied by the user. Sadati et al (2011) proposed using generalized cross validation to find optimum weights.

de Silva et al (2013) defined a constrained minimisation problem, thereby avoiding the use of weights in the objective function. They called their method the *divergence correction scheme* (DCS). The objective function is only the discrepancy between the measured and unknown filtered velocity fields, expressed as a sum of squared differences. The divergence-free condition is introduced as a linear equality constraint. The method was applied to a tomographic PIV experiment and an improvement in the prediction of turbulence statistics was shown. Formally, it defines the following optimisation problem:

$$\min_{\mathbf{u}_{\text{sol}}} \frac{1}{m} \sum_{i=1}^{m} \left[ \sum_{j=1}^{3} \left( u^i_{\text{sol},j} - u^i_{\text{PIV},j} \right)^2 \right] \quad \text{subject to} \tag{3.5}$$

$$\nabla \cdot \mathbf{u}_{\text{sol}} = 0. \tag{3.6}$$

The divergence-free condition in Eq.3.6 is approximated using a second-order central difference scheme on the regular measurement grid. To solve the optimisation problem, de Silva et al (2013) use MATLAB's solver *fmincon*. It possesses the two most powerful algorithms for solving general constrained nonlinear multivariable minimisation problems, namely sequential quadratic programming (SQP) and the interior-point method (Wright and Nocedal, 1999). However, a closer look at the defined optimisation problem reveals that it can be restated as a quadratic programming problem with linear equality constraints (Wright and Nocedal, 1999), therefore avoiding the need for either SQP or the interior-point method. Instead, it will results in one linear system of equations that needs to be solved. The quadratic programming problem can be stated as follows:

$$\min_{\mathbf{u}_{\mathrm{sol}}} \frac{1}{2}\mathbf{u}'_{\mathrm{sol}} Q \mathbf{u}_{\mathrm{sol}} + \mathbf{d}'\mathbf{u}_{\mathrm{sol}} \quad \text{subject to} \tag{3.7}$$

$$D\mathbf{u}_{\mathrm{sol}} = 0, \tag{3.8}$$

where $\mathbf{u}_{\mathrm{sol}} = \left(\mathbf{u}_{\mathrm{sol},1}\ \ \mathbf{u}_{\mathrm{sol},2}\ \ \mathbf{u}_{\mathrm{sol},3}\right)'$, $Q = (2/m)\,I$ ($I$ is the identity matrix),

$$\mathbf{d} = -\,(2/m)\left(\mathbf{u}_{\mathrm{PIV},1}\ \ \mathbf{u}_{\mathrm{PIV},2}\ \ \mathbf{u}_{\mathrm{PIV},3}\right)',$$

and $D$ is the discrete divergence operator. The solution to the optimisation problem can be obtained by solving the following linear system of equations:

$$\begin{bmatrix} Q & D' \\ D & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u}_{\mathrm{sol}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} -\mathbf{d} \\ 0 \end{bmatrix}, \tag{3.9}$$

where $\boldsymbol{\lambda}$ are Lagrange multipliers. The system matrix is sparse indefinite and can be solved using SYMMLQ or MINRES (Paige and Saunders, 1975).

Our reformulation of DCS as a quadratic programming problem with linear equality constraints reveals that it is equivalent to the Helmholtz Representation Theorem. To show this, we extract the first row from Eq.3.9:

$$Q\mathbf{u}_{\mathrm{sol}} + D'\boldsymbol{\lambda} = -\mathbf{d}, \tag{3.10}$$

which can be rewritten as

$$\mathbf{u}_{\mathrm{sol}} = \mathbf{u}_{\mathrm{PIV}} - D'\boldsymbol{\lambda}. \tag{3.11}$$

The transpose of the divergence operator ($D'$) is the nabla operator or gradient $G$ (Ma et al, 2013):

$$\mathbf{u}_{\mathrm{sol}} = \mathbf{u}_{\mathrm{PIV}} - G\boldsymbol{\lambda}. \tag{3.12}$$

Taking the curl on both sides of Eq.3.12 and recalling the vector identity that the curl of the gradient operator is zero, we find that the vorticity of the solenoidal velocity field is equal to the vorticity of the measured field. As stated in Section 3.2.1, this is equivalent to the result obtained through The Helmholtz Representation Theorem. Despite this, the advantage of DCS is that it does not require the user to specify boundary conditions. DCS is one of the solenoidal filters that will be assessed in the present chapter. For it to work optimally, one could apply smoothing to the noisy velocity field first before introducing the solenoidal filter itself. However, we have decided to use it as it was defined by de Silva et al (2013).

### 3.2.3. RECONSTRUCTION WITH A SOLENOIDAL BASIS

A third approach is to use a solenoidal basis, which by construction will return solenoidal velocity fields. There are several possible choices of basis. Solenoidal wavelets were proposed by Battle and Federbush (1993). Narcowich and Ward (1994) devised solenoidal radial basis functions (RBF). Lowitzsch (2005) subsequently derived a density theorem stating that any sufficiently smooth solenoidal function can be approximated arbitrarily closely by a linear combination of solenoidal RBFs. Therefore any incompressible flow may be represented by this construction. Solenoidal wavelets and RBFs have been applied in computational fluid dynamics to solve incompressible flows (Deriaz and Perrier, 2006; Schräder and Wendland, 2011; Urban, 1996; Wendland, 2009). Applications of these methods to reconstruct solenoidal velocity fields from velocity measurements have been reported as well (Busch et al, 2013; Ko et al, 2000; Vennell and Beatson, 2009).

Another choice of basis is described in the work by Schiavazzi et al (2014), who introduced a method, referred to in the following as *solenoidal waveform reconstruction* (SWR). It works by first defining a Voronoi tessellation of the measurement grid. Then, the observed velocities $\mathbf{u}_{\mathrm{PIV}}$ are converted to face fluxes $\mathbf{g}$. Vortices with unknown strengths $\boldsymbol{\alpha}$ are defined around all edges of the grid (in 2D, vortices are defined around nodes). This ensures that the reconstructed velocity field, which is a sum of these vortices, will be solenoidal. The vortex strengths $\boldsymbol{\alpha}$ are found by solving the system $W\boldsymbol{\alpha} = \mathbf{g}$ in the least-squares sense, where the sparse matrix $W$ contains the influence of the solenoidal waveforms in terms of the flux they generate. Schiavazzi et al (2014) propose a sequential matching pursuit algorithm to solve the underdetermined system. Here, we use LSQR (Paige and Saunders, 1982), a conjugate-gradient type method for sparse least-squares systems. Once the vortex strengths have been found, the fluxes $\mathbf{g}^*$ are used to reconstruct the divergence-free velocities $\mathbf{u}_{\mathrm{sol}}$. The basis functions used in this method are the smallest possible solenoidal waveforms compatible with the measurement grid. The advantage of this is that large local measurement errors only disturb reconstructed velocities at neighboring grid points. The disadvantage of this minimal support is that it becomes difficult to reconstruct gappy fields, i.e. regions with missing data.

## 3.3. SOLENOIDAL GAUSSIAN PROCESS REGRESSION

This section describes a modification to Gaussian process regression, described in Chapter 2.3, that enforces the divergence-free constraint *analytically*. In the context of reconstructing measured velocity fields using Gaussian process regression, previous works took the route of reconstructing the velocity components independently from one another, precluding enforcement of mass conservation or other physical constraints (de Baar et al, 2014; Gunes et al, 2006; Inggs and Lord, 1996; Lee et al, 2008). In these works the state is chosen as

$$\boldsymbol{\phi} = \begin{pmatrix}\mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3\end{pmatrix}' \in \mathbb{R}^{3n},$$

where $\mathbf{u}_k$ are the Cartesian velocity components. The $m$ observations are defined by $\mathbf{y} = \begin{pmatrix}\mathbf{u}_{\mathrm{PIV},1} & \mathbf{u}_{\mathrm{PIV},2} & \mathbf{u}_{\mathrm{PIV},3}\end{pmatrix}' \in \mathbb{R}^{3m}$. The covariance matrix $P \in \mathbb{R}^{3n\times 3n}$ is a $3\times 3$ block diagonal matrix, with elements $P_k$, $k = 1,2,3$ on the diagonal (see the last expression of Eq.2.13). The off-diagonal blocks are zero by assumption of uncorrelated velocity components.

To include the divergence-free constraint, we use the vector calculus identity which states that the curl of a vector potential $\mathbf{a}$ is divergence-free. We therefore choose the state vector

$$\boldsymbol{\phi} = \begin{pmatrix}\mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3\end{pmatrix}' \in \mathbb{R}^{3n},$$

where $\mathbf{a}_k$ are the Cartesian vector potential components. The observation matrix $H \in \mathbb{R}^{3m\times 3n}$, when multiplied with the vector potential, should return its curl, therefore ensuring that the resulting vector will be divergence-free:

$$H = \begin{bmatrix} 0 & -\partial_3\cdot & \partial_2\cdot \\ \partial_3\cdot & 0 & -\partial_1\cdot \\ -\partial_2\cdot & \partial_1\cdot & 0 \end{bmatrix}, \tag{3.13}$$

where $\partial_k$ is the partial derivative with respect to $x_k$. The covariance matrix $P \in \mathbb{R}^{3n\times 3n}$ is a $3\times 3$ block diagonal matrix, with elements $P_k$, $k = 1,2,3$ on the diagonal, representing the covariance between the components of the vector potential. The off-diagonals of the covariance matrix are zero since the different components of the vector potential are assumed to be uncorrelated. In the most general case there is no *a priori* physical knowledge to assume that there *is* a relation between them. The component $HPH'$ of the gain matrix $A$ (see Eq.2.18) is equal to

$$\begin{bmatrix} \partial_{3,3}P_2 + \partial_{2,2}P_3 & -\partial_{1,2}P_3 & -\partial_{1,3}P_2 \\ -\partial_{1,2}P_3 & \partial_{3,3}P_1 + \partial_{1,1}P_3 & -\partial_{2,3}P_1 \\ -\partial_{1,3}P_2 & -\partial_{2,3}P_1 & \partial_{2,2}P_1 + \partial_{1,1}P_2 \end{bmatrix}. \tag{3.14}$$

It can easily be verified that the columns and rows of Eq.3.14 are divergence-free. By taking the same covariance function for all directions of the vector potential ($P_1 = P_2 = P_3$) and assuming perfect measurements ($R = 0$), the gain matrix reduces to:

$$\left(\Delta I - \nabla\nabla'\right) P_1. \tag{3.15}$$

This turns out to be proportional to the operator constructed by Narcowich and Ward (1994) in the context of RBF interpolation. Our method can therefore be seen as a generalization of their method, that allows different covariance functions to be used for the different vector potential components, and that includes measurement uncertainty in the reconstruction via $R$. This latter property is a natural consequence of the Bayesian approach we followed.

## 3.4. EFFICIENT IMPLEMENTATION

The practical implementation of SGPR is discussed in this section, in which the primary concern is the storage, conditioning and inversion of the (in general) dense gain matrix $A$. This is critical given that tomographic data sets containing $10^5 - 10^6$ vectors are typical (Scarano, 2013). By exploiting the Toeplitz structure of the gain matrix we make the method highly efficient.

### 3.4.1. COMPUTATIONAL COST

To evaluate the divergence-free velocity field at any point one first has to solve the linear system $A\mathbf{c} = \mathbf{y}$, where $A \in \mathbb{R}^{3m\times 3m}$ (see also Eq.2.19). This is the most expensive component of SGPR. The gain matrix is per definition symmetric. For scattered observations, this requires storing $3m(3m+1)/2$ entries. If the covariance function used is positive definite, the matrix will also be positive definite (Narcowich and Ward, 1994). The Cholesky factorization, with a computational complexity of $O(27m^3)$, can therefore be used as a direct solver. Alternatively, the conjugate gradient (CG) method, with a computational complexity of $O(9k \cdot m^2)$ ($k$ is the number of iterations required to reach a given tolerance), can be used as an iterative solver. The dominant operation per iteration is the matrix-vector multiplication $A\mathbf{c}$. For large data sets, the cost of SGPR therefore becomes prohibitive. One approach to deal with this is to use a covariance function with local support: $\phi(r \geq r_d) = 0$, where $r_d$ is the support radius. If the support radius is much smaller than the size of the measurement domain, the gain matrix will be sparse, resulting in storage and computational savings. However, the optimum support radius may be large for a given flow field (Schaback, 1995), and we do not wish to be limited by computational considerations.

If the measurements are (i) available on a regular grid, as is the case in scanning PIV and tomographic PIV; and (ii) all observations have equal measurement uncertainty, then the gain matrix will have a small displacement rank (Kailath et al, 1979), allowing memory and time savings. To be more specific, the gain matrix will have a Toeplitz structure. For the present problem, it is a $3 \times 3$ block matrix where each block is a 3-level symmetric Toeplitz matrix. It is a 3-level Toeplitz matrix as we are dealing with a three-dimensional space, and the $3 \times 3$ block structure is due to the structure of Eq.3.14. A 3-level symmetric Toeplitz matrix $T$ is defined as follows:

$$T = \begin{bmatrix} T_1 & T_2 & \cdots & \cdots & T_{m_3} \\ T_2 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & T_2 \\ T_{m_3} & \cdots & \cdots & T_2 & T_1 \end{bmatrix}, \quad T_i = \begin{bmatrix} t_1 & t_2 & \cdots & \cdots & t_{m_2} \\ t_2 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & t_2 \\ t_{m_2} & \cdots & \cdots & t_2 & t_1 \end{bmatrix}, \quad t_i = \begin{bmatrix} \tau_1 & \tau_2 & \cdots & \cdots & \tau_{m_1} \\ \tau_2 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \tau_2 \\ \tau_{m_1} & \cdots & \cdots & \tau_2 & \tau_1 \end{bmatrix}, \tag{3.16}$$

where $m = m_1 m_2 m_3$ and $m_k$, $k = 1,2,3$, is the number of observations in the $k$-direction. The gain matrix can now be stored using $3m$ entries. Fast and superfast direct solvers of complexity $O(9m^2)$ and $O(3m\log^2(3m))$, respectively, are available (Ammar and Gragg, 1988).

If the measurement uncertainty is not the same at all points, the gain matrix will have arbitrary displacement rank and no fast direct solvers are available for such a system (Ng and Pan, 2010). However, when the CG method is used, accelerations can be achieved. First, the gain matrix $A$ is decomposed into a diagonal matrix $D$, containing the unequal measurement uncertainties, and a $3 \times 3$ block matrix $G$, where each block $G_i$ is a 3-level symmetric Toeplitz matrix:

$$A = D + \begin{bmatrix} G_1 & G_4 & G_6 \\ G_4 & G_2 & G_5 \\ G_6 & G_5 & G_3 \end{bmatrix}. \tag{3.17}$$

The multiplication $D\mathbf{c}$ can be carried out in $O(3m)$ operations. The matrix-vector multiplication $G\mathbf{c}$ can be accelerated by embedding each 3-level Toeplitz matrix $G_i \in \mathbb{R}^{m \times m}$

Table 3.1: Relative correlation length $\gamma/L$ at which $\kappa(A) = 10^{15}$. The spatial dimension is 3 and the data points are defined on a regular $15\times15\times15$ grid with size $L\times L\times L$.

| | $\phi_{3,1}$ | $\phi_{3,2}$ | $\phi_{3,3}$ | $\phi_{3,4}$ | $\phi_G$ |
|---|---|---|---|---|---|
| GPR | 508.35 | 21.617 | 6.1209 | 3.2370 | 0.5175 |
| SGPR | $4.4558\times10^9$ | 408.63 | 18.503 | 5.4604 | 0.4884 |

into a 3-level circulant matrix $C_i \in \mathbb{R}^{8m\times8m}$ and then using 3D FFTs to carry out matrix-vector multiplications at a cost of $O\big(8m\log(8m)\big)$ (Chan and Jin, 2007). The computational complexity of the CG algorithm now becomes $O\big(72k\cdot m\log(8m)\big)$. Normally, the CG method is used in combination with a preconditioner to speed up convergence. Unfortunately, Capizzano (2002) has shown that preconditioners for multilevel Toeplitz matrices are not super linear. So in the present work we have implemented SGPR without a preconditioner.

Note that the assumption of observations on a regular grid should not be seen as a significant limitation. Consider two common situations where we have (i) a regular measurement grid with missing data (e.g. from removed outliers); (ii) an irregularly scattered data set that can be approximated as a subset of a regular grid. In these cases, we can set the values of the unobserved points equal to the mean of the observations (either global or local) and specify a large measurement uncertainty at these points. The size of the gain matrix increases, but this may be out weighted by the benefits of having a Toeplitz structure.

### 3.4.2. CONDITIONING

The most important factors that influence the condition number of the gain matrix, $\kappa(A)$, are the covariance function, the separation distance of the data, the correlation length and the observation error (Davis and Morris, 1997; Narcowich and Ward, 1994). Decreasing the separation distance, increasing the correlation length and decreasing the measurement uncertainty all increase the condition number. An ill-conditioned gain matrix causes inaccurate results (in the form of numerical noise in the reconstruction), and reduces the convergence speed of the CG method. In fact, the gain matrix can become so ill-conditioned that it stops being *numerically* positive definite (i.e. in finite-precision arithmetic), even though it will always be analytically positive given a positive definite covariance function (de Baar, 2014). Since our solution methods, Cholesky and CG, both rely on positive-definiteness, they cannot be applied. We therefore aim to avoid ill-conditioning by a suitable choice of the covariance function $\phi$. Table 3.1 summarizes at which correlation length $\kappa(A)$ becomes $10^{15}$. The functions $\phi_{d,k}$, $k = 1,2,3,4$ are the Wendland functions with smoothness $C^{2k}$ and spatial dimension $d$ (Wendland, 2005). The Gaussian ($\phi_G$), an infinitely smooth function, is defined as $\phi_G(r) = \exp\left(-\alpha^2 r^2\right)$. The constant $\alpha$ was set to 3.3 to make it resemble $\phi_{d,3}$ as closely as possible. Unlike the Gaussian, the Wendland functions have compact support: $\phi_{d,k} = 0$ for $r \geq 1$. The Gaussian covariance function appears to be the most ill-conditioned, a property attributed to the fact that it is infinitely differentiable (Davis and Morris, 1997). The smoother the Wendland function, the more it approaches the Gaussian (Chernih et al, 2014). The idea that the level of differentiability is related to the conditioning of the gain matrix (Chernih et al, 2014), is supported by the observation that for the Wendland functions, the gain matrix for SGPR becomes numerically not-positive definite at a larger $\gamma$. The reverse happens for the Gaussian. For SGPR, the covariance function is differentiated twice (see Eq.3.14), reducing the level of differentiability of the resulting function for the Wendland functions, explaining the better conditioning of the matrix. Clearly, $\phi_G$ limits our choice of correlation length, whereas Wendland functions are always suitable. Appendix B gives the analytical expressions of the Wendland functions ($k = 1, 2, 3, 4$), together with their second derivatives.

## 3.5. APPLICATION TO SYNTHETIC TEST CASES

The performance of SGPR is assessed using the two synthetic test cases of incompressible flow described in Appendix A. It is compared with three other filters. The first does not enforce the divergence-free constraint, and works through a convolution of the data using a $3 \times 3$ smoothing kernel with equal weights ($\text{BOX}_{3\times3}$), a simple yet widely used approach to dealing with noisy data (Raffel et al, 1998). The two solenoidal filters we compare with are DCS and SWR. They were described in Section 3.2.2 and 3.2.3, respectively. The covariance function we use when applying SGPR is the Wendland function with order $k = 2$ and smoothness $C^4$: $\phi_{3,2}(r)$ (see Eq.B.5). The choice for using this function is computational efficiency; it is the lowest-order Wendland function with continuous second order velocity components, which are needed for calculating the pressure field, and as was concluded in Section 3.4.2, using lower order covariance functions improves the conditioning of the gain matrix and as a result improves the convergence speed of the conjugate gradient method. Recall from Eq.2.14 that the radial distance $r$ is scaled with a correlation length $\gamma$. The user can either specify its value *a priori* or its optimum value can be pursued by using an optimisation algorithm. Choosing between the manual and automatic approach is a trade-off between computational cost and accuracy. In this chapter, we suggest to set this parameter *a priori*. The reasoning is that in practice, for a tomographic PIV data set, an experimenter will already know what the flow looks like before applying SGPR. So the parameter could be set equal to a representative flow feature, like the size of a vortex. In Chapters 4 and 5, where the experimental data such as from 3D-PTV is sparse, we will use an optimisation algorithm to determine the optimum parameter. In this section and the next, we will investigate how the reconstructed fields are influenced by the choice of the parameter and then determine how the *a priori* choice compares with the actual optimum value.

We set $R = \lambda I$, where $\lambda$ simply acts as a regularization parameter. Many different methods have been proposed in the literature to choose this parameter, however there is no generally accepted approach. For an extensive review of available approaches, the reader is referred to Bauer and Lukas (2011). We chose $\lambda = 10^{-2}$ for reasons of computational efficiency; satisfactory convergence speeds were then obtained for all test cases when using the CG method.

To simulate PIV noise, we add *spatially correlated* Gaussian noise to the true velocity field. However, the error of one velocity component is assumed to be uncorrelated from the error of the other velocity components. This is typically the case in PIV. Therefore, noise will be added to each velocity component independently. For each velocity com-

Table 3.2: Taylor-vortex: noise reduction $Q$ of velocity magnitude, vorticity and pressure.

| | $\text{BOX}_{3\times3}$ | DCS | SWR | SGPR |
|---|---|---|---|---|
| velocity | 14.9% | 29.6% | 35.8% | 59.2% |
| vorticity | 26.3% | 0% | 15.8% | 70.4% |
| pressure | 42.7% | 83.0% | 88.1% | 92.3% |

ponent, we express the covariance of the measurement uncertainty between point $i$ and $j$ as:

$$S^{ij} = \text{Cov}\left[u^i_{\text{PIV}}, u^j_{\text{PIV}}\right] = \sigma^i \zeta^{ij} \sigma^j,$$

where $\sigma^i$ and $\sigma^j$ are the standard deviations of the measurement noise at points $i$ and $j$, respectively. We take this standard deviation equal to a fraction of the local velocity magnitude. For point $i$:

$$\sigma^i = \alpha ||\mathbf{u}^i_{\text{PIV}}||_2,$$

where we have taken $\alpha = 0.1$, i.e. the standard deviation is equal to 10% of the local velocity magnitude. Therefore, the measurement uncertainty is spatially varying. This is representative of the situation in a real PIV experiment, see e.g. Figure 3.5. The correlation $\zeta^{ij}$ we use is based on results obtained by Wieneke and Sciacchitano (2015), who investigated the spatial correlation of measurement noise between PIV velocity vectors as a function of overlap and interrogation window size. We used the results for an overlap of 75% and an interrogation window size of 32 voxels in each direction, which are the most frequently used settings for tomographic PIV (Scarano, 2013). To sample noise with the desired spatial covariance structure, we compute the Cholesky decomposition of $S = LL'$, and multiply $L$ with a vector containing *uncorrelated* noise from a standard Gaussian distribution. This noise is added to the analytical velocity fields.

The filters we will use work on the velocity field. To get the vorticity field, we take the curl. With SGPR we can calculate this field analytically due to the use of the covariance function. For the other filters we use finite differencing, as velocities are only available on grid nodes. To allow for a fair comparison, we also use finite differencing to compute curl when using SGPR.

For the first test case we will also calculate the pressure field. The pressure field is obtained from the velocity field by using the Navier-Stokes equations. Chapter 6 describes in detail how this can be achieved. Neumann boundary conditions are imposed, and the reference pressure in the center is set to act as a Dirichlet boundary condition. All the derivatives are calculated using second-order accurate finite difference schemes.

The metric we use to compare the filters is a noise reduction percentage ($Q$), defined as:

$$Q = \left(\frac{\varepsilon^{\text{o}} - \varepsilon^{\text{f}}}{\varepsilon^{\text{o}}}\right) \times 100\%, \tag{3.18}$$

where $\varepsilon^{\text{o}}$ is the root-mean-square error of the measured (observed) field and $\varepsilon^{\text{f}}$ is the root-mean-square-error of the filtered field.

### 3.5.1. TAYLOR-VORTEX

We use a regular grid (tomographic PIV case) with 101 points in both directions, totaling 10,201 observation points. For simplicity, we use the same correlation length in both directions. Since the whole domain contains one vortex, we set its value equal to the size of the domain: $\gamma = \gamma_1 = \gamma_2 = 2L_0$. Table 3.2 shows the noise reduction $Q$ for the velocity magnitude, vorticity and pressure fields, averaged over the 26 time frames. Comparing SGPR with the other filters, we see that it provides the highest noise reduction for all fields. Comparing BOX$_{3\times3}$ with DCS and SWR, we see that BOX$_{3\times3}$ returns a more accurate vorticity field, but less accurate velocity and pressure fields. Notice that for DCS the noise in the vorticity field is not reduced, so the vorticity of the solenoidal field is equal to that of the observed field. This confirms that it is equivalent to the Helmholtz Representation Theorem. Comparing the three solenoidal filters, SGPR has the highest noise reduction, followed by SWR and then DCS.

To get a qualitative impression of the reconstructed fields, we refer to Figure 3.1. First of all, notice that the noise we have added to represent the PIV noise is indeed spatially correlated. The fields returned by SGPR do indeed look better than the ones obtained with the other filters, especially when looking at the velocity and vorticity fields. The improvement in the pressure field is less noticeable, as verified from Table 3.2.

The correlation length we used was determined based on the fact that the field contains one vortex that encompasses the entire domain, so we set $\gamma/L_0 = 2$. Figure 3.2 shows how the noise reduction for the velocity, vorticity and pressure fields is influenced by the correlation length. The three graphs share the following characteristics:

(i) there is an optimum noise reduction inside the interval;

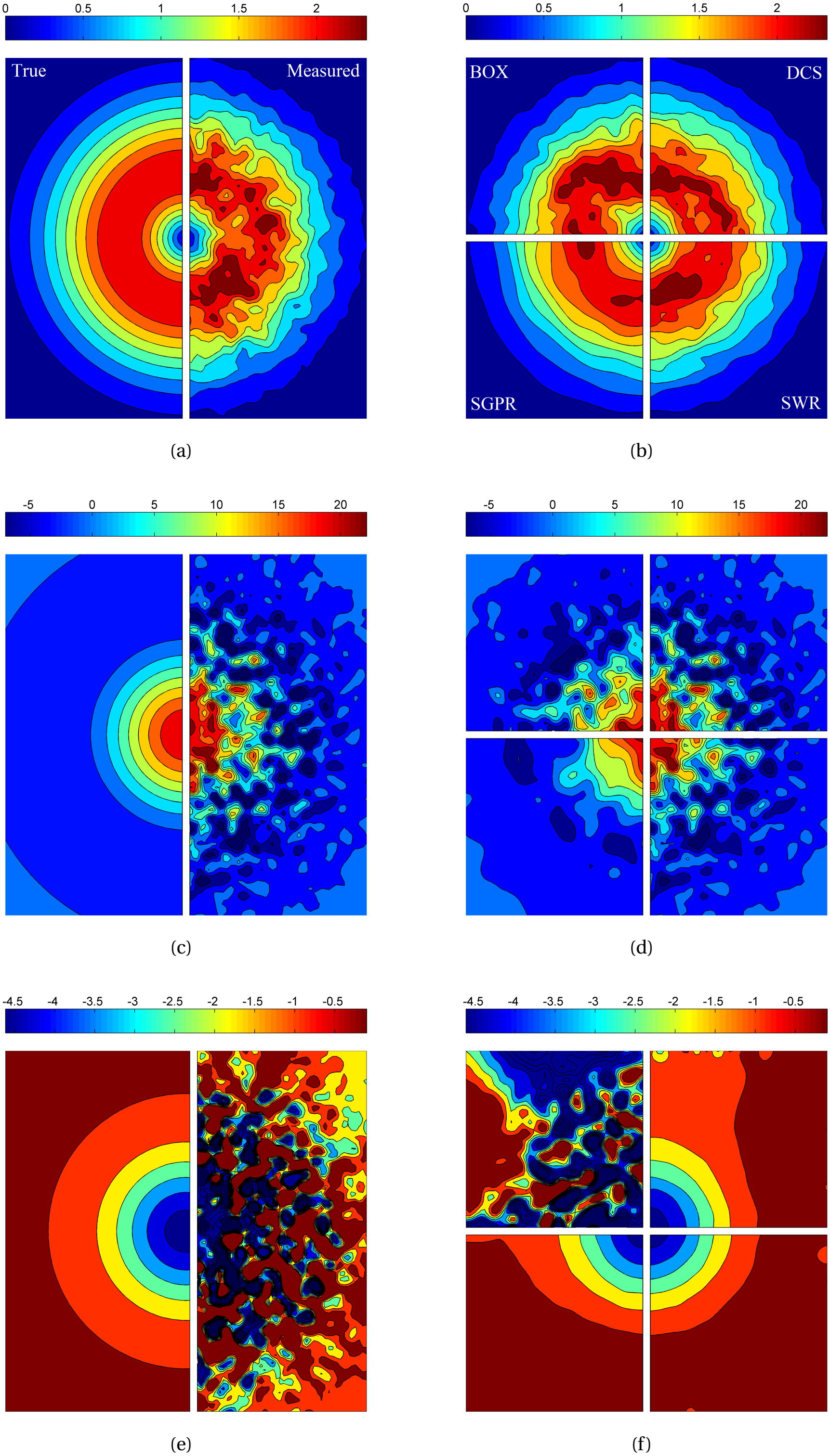
(ii) small values of the correlation length lead to an abrupt decrease in noise reduction;

(iii) around the optimum, the graph is relatively flat.

The optimum correlation length is around $\gamma/L_0 = 9$, resulting in noise reductions of 78%, 93% and 93% for velocity, vorticity and pressure, respectively. Though our choice of correlation length was therefore suboptimal, the *a priori* physical knowledge we used has still given us better noise reduction than the other filters. It is only when we would have chosen a length smaller than $\gamma/L_0 = 0.45$ that SWR would have been better. The fact that the graphs are relatively flat around the optimum is a desirable property as it implies robustness of SGPR. As long as the user makes an educated guess for the correlation length based on *a priori* physical knowledge, SGPR is relatively insensitive to the exact value chosen.

### 3.5.2. VORTEX RING

We take a regular grid with 21 data points in the 1-direction and 41 data points in the 2- and 3-directions. So there are 35,301 data points in total. We use a correlation length of $10R_0$ in each direction. So the support radius spans the complete vortex ring. Table 3.3 shows the noise reduction for the velocity and vorticity magnitude. Again, BOX$_{3\times3}$ results in the lowest noise reduction for the velocity field. However, as for the previous test case, it returns more accurate vorticity fields than DCS and SWR, where DCS again leaves the



(a) (b)

(c) (d)

(e) (f)

Figure 3.1: Taylor-vortex: true (left side of (a), (c), (e)), measured (right side of (a), (c), (e)) and filtered ((b), (d), (f)) velocity magnitude (top), vorticity (center) and pressure (bottom) fields. Filters (clock-wise, starting at top left): $\mathrm{BOX}_{3\times3}$, DCS, SWR, SGPR. The second frame is shown.

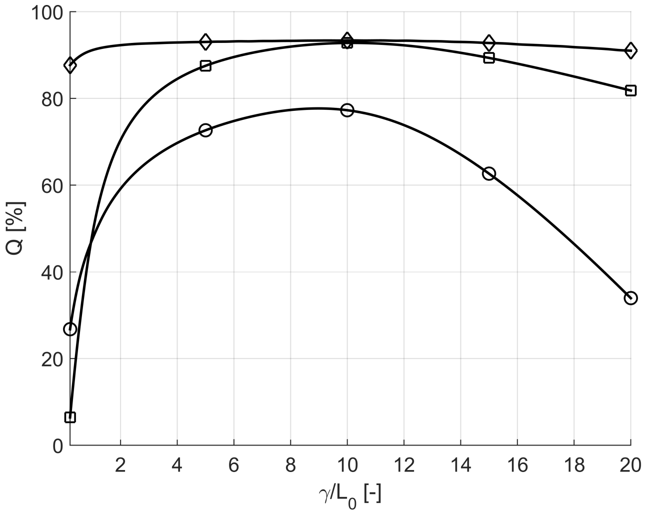


Figure 3.2: Taylor-vortex: noise reduction of the velocity (*circle*), vorticity (*square*) and pressure (*diamond*) fields as a function of correlation length using SGPR.

Table 3.3: Noise reduction $Q$ of velocity magnitude and vorticity for the vortex ring.

| | $BOX_{3\times3}$ | DCS | SWR | SGPR |
|---|---|---|---|---|
| velocity | 16.6% | 17.6% | 27.5% | 27.4% |
| vorticity | 18.5% | 0% | 15.1% | 20.7% |

measured vorticity field unchanged. SGPR results in the highest noise reduction for the vorticity field, but for the velocity field SWR is marginally better.

To get a qualitative impression of the methods, Figure 3.3 compares the true vortex ring with the filtered ones. Figure 3.4 shows how the noise reduction is influenced by the choice of correlation length. The value we chose ($\gamma/R_0 = 10$) is close to the optimum at $\gamma/R_0 = 14$. At this value, the noise reduction of velocity and vorticity is 30.4% and 23.2%, respectively. Similar to the Taylor-vortex, the current graph has three characteristics, namely an optimum inside the interval, a large decrease in noise reduction for small correlation lengths and a relatively flat curve around the optimum.

## 3.6. APPLICATION TO EXPERIMENTAL TEST CASES

We will not consider the non-solenoidal $BOX_{3\times3}$ filter, since such filtering steps have already been applied to obtain the velocity fields from the particle images, especially the effect of the PIV interrogation windows. To assess the reconstruction accuracy of the solenoidal filters, we use different approaches for the two test cases: for the circular jet in water, we use *a posteriori* estimates of random uncertainty for the measurements in terms of the velocity field and investigate to what extent the filtered velocities are in-

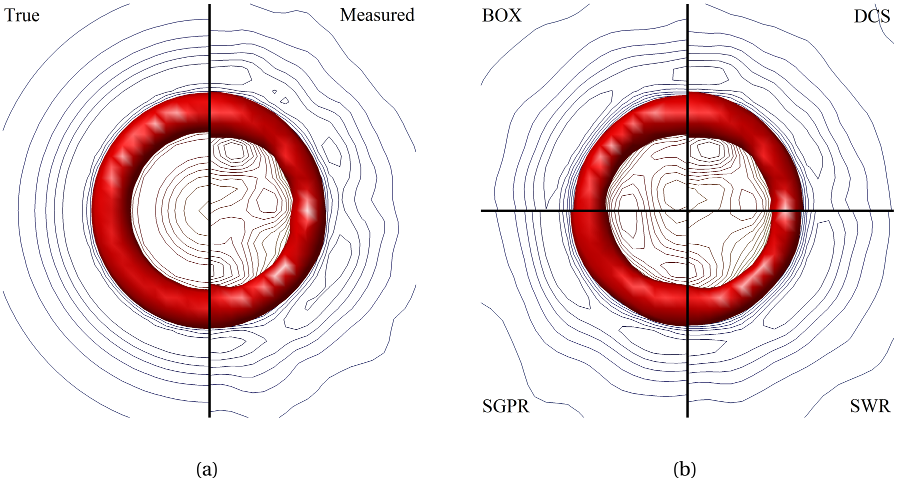


Figure 3.3: Vortex ring: top view and contours of the out-of-plane velocity. True (left side of (a)), observed (right side of (a)) and filtered (b) fields.

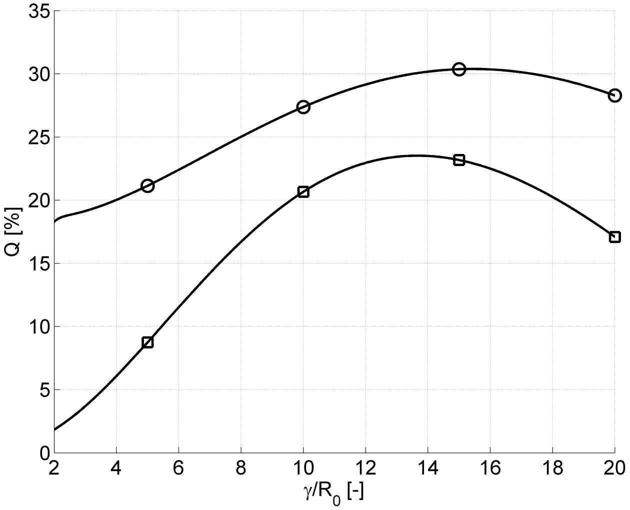


Figure 3.4: Vortex ring: noise reduction of the velocity (*circle*) and vorticity (*square*) fields as a function of correlation length using SGPR.

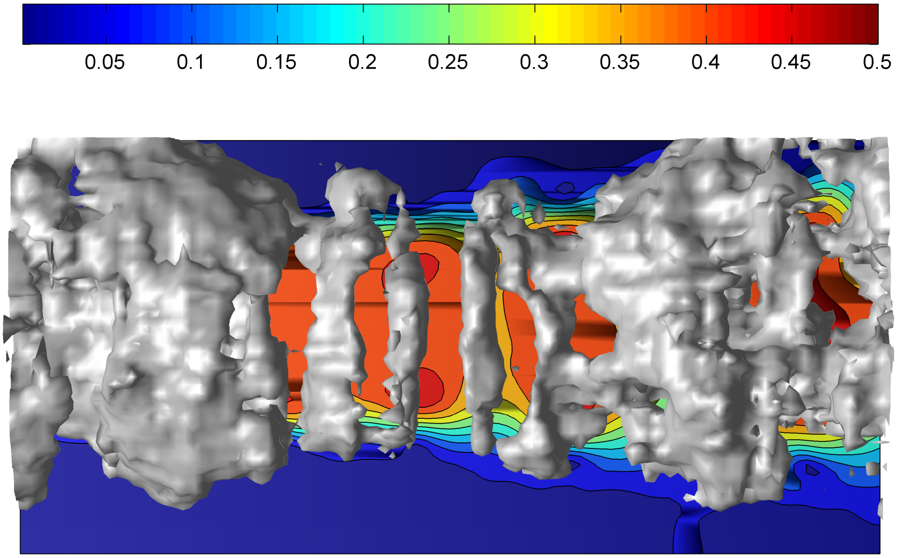


Figure 3.5: Circular jet in water: velocity magnitude slice in the axial plane. Isosurfaces of measurement uncertainty in terms of velocity magnitude standard deviation at 0.035 m/s.

side the uncertainty bounds; for the turbulent boundary layer in air, we reconstruct the pressure field from the velocity data and compare with microphone measurements.

### 3.6.1. CIRCULAR JET IN WATER

Since we are considering experimental data, there is no perfect reference solution. To make a quantitative performance assessment of the solenoidal filters, we use a recent method that estimates *a posteriori* random uncertainty for measurements locally (Sciacchitano, 2015). It is also applicable to tomographic PIV. It expresses the distribution of error for each vector component in terms of standard deviation ($1\ \sigma$). Figure 3.5 shows isosurfaces of the standard deviation at a velocity magnitude of 0.035 m/s. Clearly, the measurement uncertainty varies in space. Also notice how the uncertainty is localized inside the jet, as opposed to outside where the flow is practically at rest. In particular, we can identify ring shapes (compare with the vortex rings in Figure A.3(a)). These are regions with strong velocity gradients so it is reasonable to expect larger uncertainties there (Sciacchitano et al, 2013).

When using SGPR, we consider two scenarios. In the first, we do not include the *a posteriori* measurement uncertainty in the reconstruction. In the second scenario we take it into account in the $R$ matrix. The covariance function we use is the Wendland function with order 2 (see Eq.B.5). The correlation length spans 15 vectors in each direction, which is approximately the diameter of a vortex ring, i.e. $\mathrm{d}_{\mathrm{vortex}} = 15h$. To investigate the influence of the correlation length on the reconstructed velocity field, we under sample the data set by halving the measurement resolution in each direction. The data points in between are taken as validation, which we then use to calculate the root-mean-

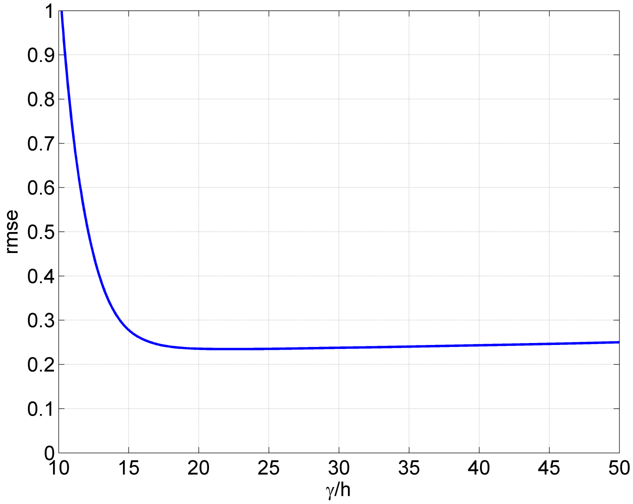
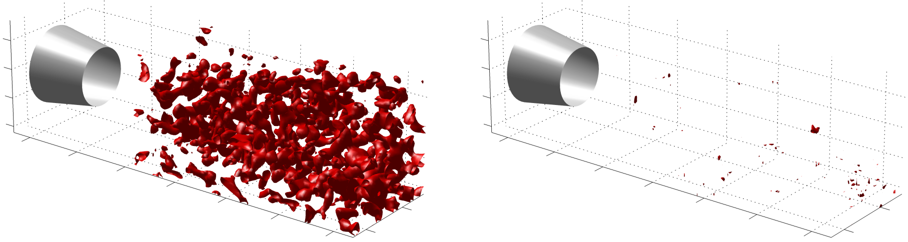


Figure 3.6: Circular jet in water: root-mean-square error as a function of correlation length.

square error. Figure 3.6 shows the result. The behaviour is very similar to what we found for the synthetic data sets in the previous section: (i) there is an optimum correlation length inside the interval; (ii) small values of the correlation length lead to an abrupt increase in root-mean-square error; (iii) around the optimum, the graph is relatively flat. Again, we emphasize that the latter characteristic is a desirable one from a robustness point of view. It turns out that our choice of correlation length $\gamma/\mathrm{d}_{\mathrm{vortex}} = 1$ was a very good one. Not only are we in the flat region, we are at the beginning of it. This is advantageous since a smaller correlation length leads to a better conditioned gain matrix, resulting in faster convergence. Note that we have only under sampled to understand the relation between reconstruction accuracy and correlation length. In the following, we use the full data set. Figure 3.7(a) shows isosurfaces of the velocity divergence at 15 1/s. Before removing the spurious divergence, we first quantify how far from divergence-free the data set is. The following metric is due to Zhang et al (1997):

$$\delta = \frac{(\partial u_1/\partial x_1 + \partial u_2/\partial x_2 + \partial u_3/\partial x_3)^2}{(\partial u_1/\partial x_1)^2 + (\partial u_2/\partial x_2)^2 + (\partial u_3/\partial x_3)^2}. \tag{3.19}$$

The mean of $\delta$ will lie between 0 and 1. When $\bar{\delta} = 0$, the velocity field is exactly divergence-free. When $\bar{\delta} = 1$, the velocity components are independent, random variables. Zhang et al (1997) showed that smoothing of the data can significantly reduce this value. For a turbulent flow measurement in a square duct using hybrid holographic PIV, they showed how using a Gaussian filter decreased $\bar{\delta}$ from 0.74 to 0.50. For the dataset used in the present paper, we found that $\bar{\delta} = 0.66$. Figure 3.7(b) shows the isosurfaces of velocity divergence after applying SGPR. The mean of the normalized divergence is $\bar{\delta} = 0.09$. Note that even though SGPR defines an analytically divergence-free field, we see some spurious divergence. This is because divergence was calculated using finite differencing to



(a) (b)

Figure 3.7: Circular jet in water: isosurfaces of velocity divergence at 15 1/s of the original data set (a) and after applying SGPR (b).

compare with the original data set. Therefore, the spurious divergence we observe for SGPR is completely due to truncation error. We can conclude that the spurious divergence observed in the data set is for a large part not due to truncation error, but other sources that arise from measurements, like random measurement errors. The reason that we do not show the velocity divergence after applying DCS or SWR is because they are defined to return divergence-free fields on a finite-difference and finite-volume level, respectively. So, the divergence calculated with these methods at the data points is zero.

We quantify the performance of the filters by using the following metric:

$$\xi_i(\mathbf{x}) = \frac{\left|u_{\mathrm{sol},i}(\mathbf{x}) - u_{\mathrm{exp},i}(\mathbf{x})\right|}{\sigma_i(\mathbf{x})}, \quad i = 1,2,3 \tag{3.20}$$

where $u_{\mathrm{sol},i}(\mathbf{x})$ is the solenoidal velocity field in the $i$-direction, obtained with DCS, SWR or SGPR and $u_{\mathrm{exp},i}(\mathbf{x})$ is the experimental velocity in the $i$-direction. The *a posteriori* standard deviation of the random uncertainty in the $i$-direction is given by $\sigma_i(\mathbf{x})$. Ideally, all reconstructed data points should have $\xi_i \leq 3$, i.e. the error should be within 3 standard deviations of the measurement uncertainty (99.7% certainty). Figure 3.8 shows the cumulative distribution of $\xi_1$. The plots for the other two directions look very similar. The results for DCS and SWR are practically identical. When the *a posteriori* measurement uncertainty information is not included (i), SGPR is only slightly better than the other filters. However, when this information is included in the reconstruction (ii), the method is clearly better in reconstructing a solenoidal velocity field within the measurement uncertainty. At $\xi_i = 3$, $F(\xi_i)$ is 1, whereas for DCS and SWR the value is around 0.9, meaning that 10% of the data points have a deviation from the measurements greater than 3 standard deviations. With regards to the average $\xi_1$, DCS and SWR are equal to 2.0. For SGPR, it is 1.8 when no measurement uncertainty is included, but it reduces to 0.3 when this information is incorporated. In conclusion, when including local measurement uncertainty, SGPR is able to return a divergence-free velocity field that follows the measured velocity field much more faithfully than any other known technique.

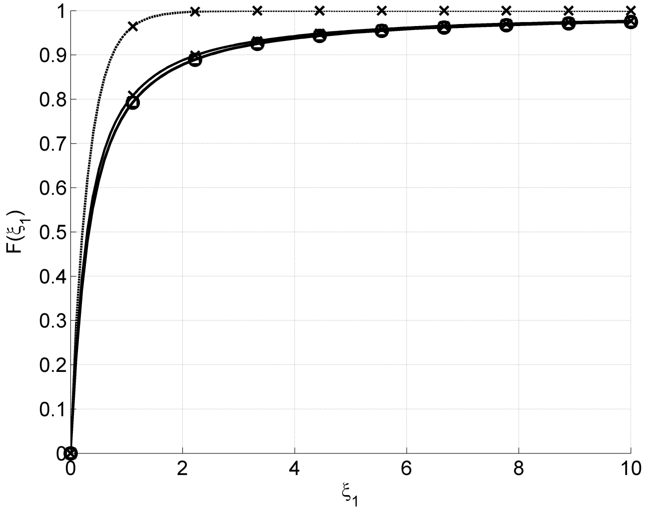


Figure 3.8: Circular jet in water: cumulative distribution of $\xi_1$ using DCS (*circle*), SWR (*square*) and SGPR (i) (*times, solid*) and SGPR (ii) (*times, dashed*).

### 3.6.2. TURBULENT FLAT PLATE BOUNDARY LAYER IN AIR

Since the particle images were not available to us at the moment of the calculations, we were unable to use the approach by Sciacchitano (2015) to estimate the measurement uncertainty, as was done in the first experimental test case. Therefore, for SGPR the observation error covariance matrix $R$ is set to the identity matrix multiplied with $\lambda = 10^{-2}$. We again use the Wendland function with order 2 as the covariance function. Recall that for the circular jet in water, we chose the correlation length equal to the diameter of the vortex ring. The present turbulent boundary layer does not have such a distinguishable feature. One can suggest using the boundary layer thickness $\delta$ and set the correlation length equal to it. However, the boundary layer thickness is approximately 9.4 mm and the height of the measurement volume is 4.2 mm, i.e. the measurement volume is completely immersed in the boundary layer. For this reason, we simply chose an *a priori* correlation length equal to the height of the measurement volume $H$. Indeed, as was explained before, larger correlation length results in a more ill-conditioned system matrix, which increases the computational time.

To investigate the influence of the correlation length on the reconstructed velocity field, we again under sample the data set by halving the measurement resolution in each direction. The data points in between are taken as validation, which we then use to calculate the root-mean-square error. Figure 3.9 shows the result. We observe that our *a priori* correlation length of $\gamma/H = 1$ is close to the optimum. Before applying SGPR, $\bar{\delta} = 0.74$. After application of this filter, it reduces to $\bar{\delta} = 0.03$.

Contrary to the previous experimental test case, no *a posteriori* estimates of random uncertainty for the velocity measurements are available. Instead, to quantify the reconstruction accuracy of the solenoidal filters, we compute pressure fields from the

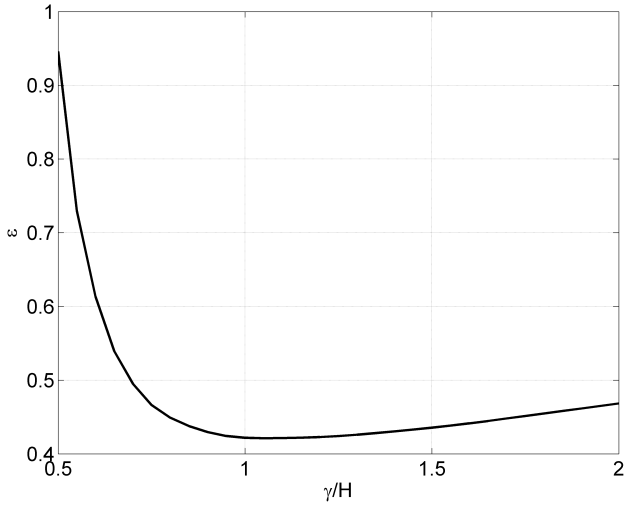


Figure 3.9: Turbulent boundary layer in air: root-mean-square error as a function of correlation length.

time-resolved velocity field measurements and compare the results with a pinhole microphone located on the wall. The boundary conditions imposed are the same as discussed in Pröbsting et al (2013), so Neumann boundary conditions at all sides except the top side closest to the free stream, where the pressure is prescribed (Dirichlet boundary condition).

Three approaches are followed to evaluate the acceleration, required for the pressure calculation:

(i) the Eulerian scheme (eul), where the material acceleration is evaluated with respect to a stationary reference frame;

(ii) the 'standard' Lagrangian scheme (lag), where the material acceleration is calculated by following a fluid particle. The specific method used is that mentioned by van Oudheusden (2013), i.e. a first-order reconstruction of the particle track with a central difference scheme for the material acceleration at the next and previous time instances;

(iii) a least-squares regression of the fluid parcel's velocity using a first-order polynomial basis ($\text{lsr}_N$), as proposed by Pröbsting et al (2013). Here, $N$ is the number of velocity fields over which the particle trajectory is followed.

The two latter approaches are both Lagrangian methods. According to Violato et al (2011), Lagrangian methods are more accurate for convection dominated flows, like boundary layer flows. For the synthetic test case of the (non-convecting) Taylor-vortex (Section 3.5.1), we only used the Eulerian scheme.

To determine the microphone location on the wall (the particle images were not available), we cross-correlated the time series of the microphone measurement with the

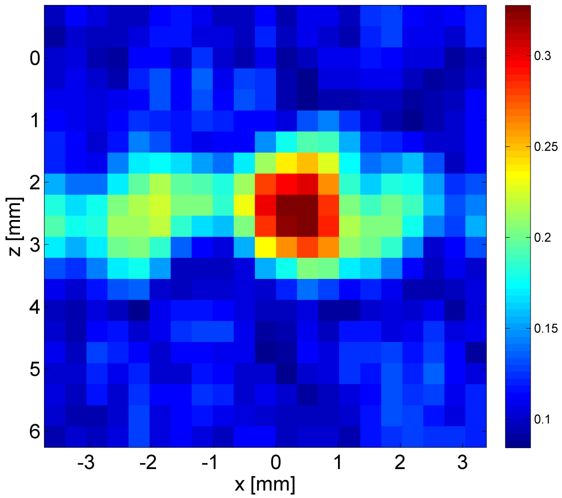
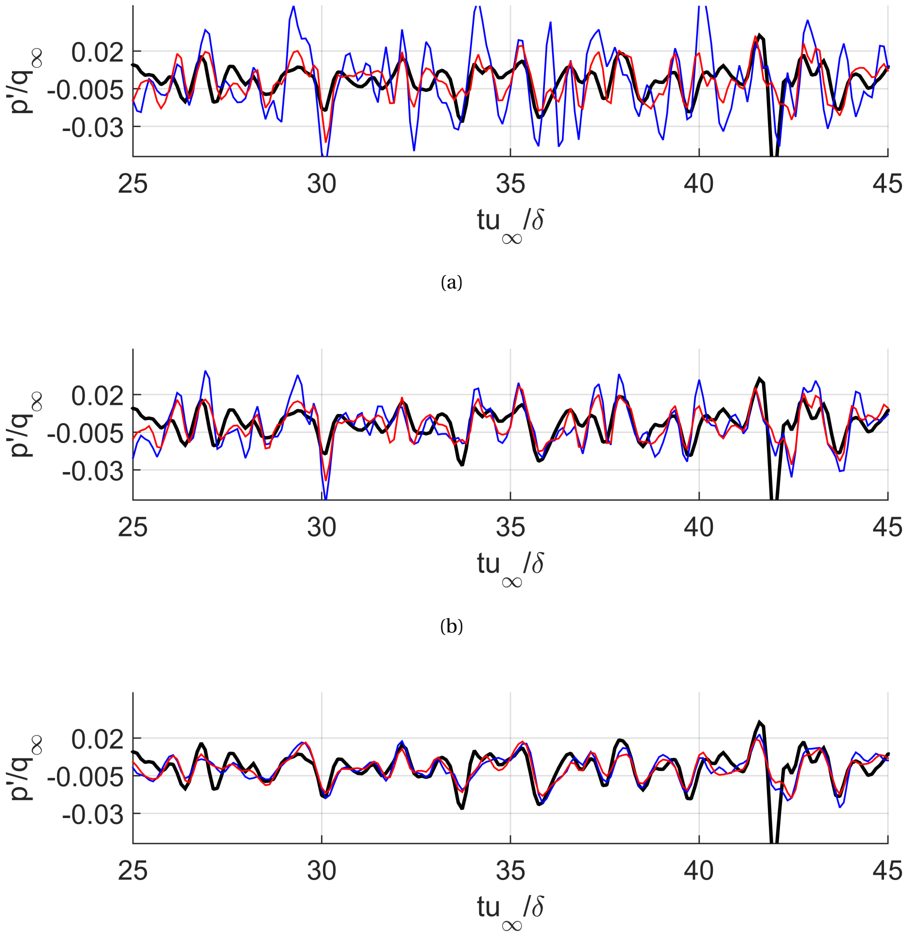


Figure 3.10: Turbulent boundary layer in air: cross-correlation between microphone pressure time series and PIV-reconstructed pressure time series, as a function of location on the wall.

PIV-based pressure fields. Figure 3.10 shows the cross-correlation coefficient taken at a region where we believed the microphone was located. We clearly see a peak and this is where the microphone is most likely located.

Following Pröbsting et al (2013), we show the pressure time series for a subset of the data in Figure 3.11. Both the microphone and PIV-based pressure signals were band pass-filtered over the range 300 Hz to 3 kHz. We have not shown the signals obtained with the other solenoidal filters to avoid obscuring the plots, however they are very similar to SGPR. As expected, the worst reconstruction is obtained with the Eulerian scheme. Improvements are obtained by switching to the Lagrangian scheme. The best results are obtained when using the least squares regressor, where 9 velocity fields were used for the particle trajectory. Comparing the results before and after applying SGPR, we clearly see that SGPR improves the reconstruction. The improvement is most noticeable for the Eulerian scheme and the least for the least squares regressor with 9 velocity fields, in which case the random errors seem to be sufficiently suppressed, leaving out mainly bias errors. However, we emphasize that using many velocity fields to reconstruct a particle trajectory is in principle undesirable since this results in a larger number of particles to fall outside the measurement domain, potentially cropping a large part of the reconstructed pressure field. Therefore, it is indeed an important advantage that the solenoidal filters result in a significant improvement when using the Eulerian and standard Lagrangian schemes.

For completeness, Table 3.4 shows the cross-correlation coefficients achieved using the various solenoidal filters and the various acceleration calculation schemes. As stated previously, the solenoidal filters return similar results and they always result in an improved pressure signal. This is independent of which scheme is used to calculate the



(a)

(b)

(c)

Figure 3.11: Turbulent boundary layer in air: pressure time series obtained with the microphone (black), tomo PIV with the original data (blue) and SGPR (red) using the Eulerian (a), standard Lagrangian (b) and least-squares regression ($\mathrm{lsr}_9$) (c) approaches. $\delta$ is the boundary layer thickness, $u_\infty$ is the free stream velocity, $q_\infty$ is the free stream dynamic pressure and $p'$ the pressure fluctuation.

Table 3.4: Turbulent boundary layer in air: cross-correlation coefficient achieved with the original data (org) and the solenoidal filters.

| | eul | lag | $\text{lsr}_3$ | $\text{lsr}_5$ | $\text{lsr}_7$ | $\text{lsr}_9$ |
|---|---|---|---|---|---|---|
| org | 0.38 | 0.46 | 0.46 | 0.55 | 0.60 | 0.62 |
| DCS | 0.51 | 0.55 | 0.56 | 0.62 | 0.63 | 0.65 |
| SWR | 0.51 | 0.54 | 0.55 | 0.61 | 0.62 | 0.63 |
| SGPR | 0.51 | 0.55 | 0.56 | 0.62 | 0.63 | 0.65 |

acceleration, though the gain becomes less noticeable as more velocity fields are used to reconstruct a particle trajectory using lsr.

## 3.7. CONCLUSION

In this chapter, we investigated the use of an analytically solenoidal filter based on Gaussian process regression (SGPR) to remove spurious divergence from tomographic PIV measurements. We formulated the filter from a Bayesian perspective. Measurement uncertainty can therefore be included naturally. To allow filtering large data sets, we have exploited the Toeplitz structure of the resulting gain matrix to speed up matrix-vector multiplications in the CG method. This can be done when the measurements are on a (near) regular grid. We used two synthetic test cases to assess the accuracy of the filtered velocity and derived vorticity and pressure fields and compared with conventional (non-physics-based) filtering and two other recently proposed solenoidal filters that have been applied to volumetric velocity measurements as well. We have used a realistic model of PIV noise with *correlated* random errors. We found that SGPR has better noise reduction properties. The advantage of the other solenoidal filters is that they do not require a parameter. SGPR on the other hand does, but we found that as long as the user makes use of his *a priori* physical knowledge of the field of interest, the results are quite insensitive to the exact value chosen. This demonstrates that the method is robust to its parameters. The first experimental data set showed that by including the measurement uncertainty, SGPR is able to reconstruct a solenoidal velocity field that more faithfully follows the measurements. The second experimental data set showed that all three solenoidal filters improve the reconstructed pressure signal, independent of which scheme is used to evaluate flow acceleration.

# 4

# SOLENOIDAL INTERPOLATION OF SPARSE 3D-PTV DATA

*This chapter investigates whether enforcing the solenoidal (divergence-free) constraint can improve the interpolation of velocity (and derived) fields obtained from sparse three-dimensional particle tracking velocimetry (3D-PTV) of incompressible flows. We investigate two approaches to enforce a solenoidal velocity field: (i) a two-step approach, where initially, the velocity components are interpolated independently from one another onto a (fine) regular grid, followed by the application of a solenoidal filter, enforcing a divergence-free velocity field on a discrete level; (ii) a one-step approach, where solenoidal Gaussian process regression (SGPR) is used, returning analytically solenoidal velocity fields (see Chapter 3). In the first step of the two-step approach, we consider both linear interpolation and Gaussian process regression (GPR). Since GPR and SGPR have parameters that need to be specified, we propose an efficient methodology in Section 4.2 that automatically finds the values for optimum interpolation based on the information provided by the measurement itself. We apply the interpolation methods to synthetic data in Section 4.3 and experimental data in Section 4.4, and show that enforcing the divergence-free constraint indeed results in consistently more accurate velocity and derived fields. Specifically, we find that the one-step approach of SGPR is superior to the two-step approach. Conclusions are drawn in Section 4.5.*

## 4.1. INTRODUCTION

3D particle tracking velocimetry (3D-PTV) returns velocity vectors non-uniformly *scattered* over the domain of interest, corresponding to the location of individual tracer particles at a single time-instant (Maas et al, 1993). To carry out flow field visualization or obtain derived quantities such as vorticity, vortex identification metrics or pressure, the velocity measurements need to be converted to a regular grid, for which a variety of data interpolation methods can be used. The simplest approach is to construct a triangulation (for example, a Delaunay triangulation) based on the measurement points, followed by piecewise (nearest-neighbor, natural-neighbor, linear) interpolation. Other methods popular in the field of 3D-PTV are the adaptive Gaussian window (Agüí and Jimenez, 1987) and local trivariate polynomial interpolation (Malik and Dracos, 1995). The use of radial basis function (RBF) interpolation (Casa and Krueger, 2013) as well as Gaussian process regression (GPR) or Kriging (de Baar et al, 2014; Gunes et al, 2006; Venturi and Karniadakis, 2004) has also been explored. These two methods do not require a triangulation or grid to be defined.

None of the above methods enforce the velocity field to be divergence-free. Including this constraint can potentially improve the accuracy of the interpolated velocity and derived fields, especially if the number of measurement points is low, i.e. *sparse* data sets. This has become particularly relevant with the recent trend to move tomographic PIV to large scale applications, enabled by the use of helium-filled soap bubbles (HFSB) as seeding material (Scarano et al, 2015). The major problem with this type of tracers is obtaining sufficient seeding concentration. The low seeding density makes the use of particle tracking a preferred option and results in sparse non-uniform data sets. A variant of the adaptive Gaussian window that incorporates the divergence-free constraint was proposed by Zhong et al (1992). However, their method only imposes this constraint at pre-selected points. It was Narcowich and Ward (1994) who introduced a method that returns an *analytically* divergence-free vector field throughout the domain. In fact, as stated by the authors, it is impossible to produce such a vector field with scalar basis functions. Chapter 3 proposed solenoidal Gaussian process regression (SGPR) to filter spurious divergence obtained from tomographic PIV data, where the main concern was not the spatial resolution of the measurements but rather the measurement noise. This method returns analytically divergence-free velocity fields.

The focus of the present chapter is to investigate whether the interpolation accuracy can be improved by enforcing the divergence-free constraint . Especially as the spatial resolution of the measurements decreases, incorporating this constraint may result in an increased benefit. We investigate two approaches to enforce a solenoidal velocity field:

(i) a two-step approach: the velocity components are interpolated independently from one another onto a (fine) regular grid, followed by the application of a solenoidal filter, enforcing a divergence-free velocity field on a discrete level;

(ii) a one-step approach: SGPR.

The second approach is preferred preferred since it works in one step and returns analytically divergence-free velocity fields.

Though the formulation of SGPR as presented in Chapter 3 does not change, the two important differences when applying it to 3D-PTV data instead of tomo-PIV data are:

(i) the choice of the correlation length or width of the covariance function used;

(ii) the procedure for solving the system matrix.

With regards to the first difference: for a well-resolved flow field, this length can be chosen *a priori* based on the size of relevant flow features, like vortices. In Chapter 3, it was found for a number of test cases that the accuracy is insensitive to the exact choice, indicating robustness of SGPR. However, if the data set is sparse, choosing a correlation length based on *a priori* knowledge is not straightforward. Increasing the width usually improves the accuracy significantly (Fornberg and Wright, 2004). However, this results in dense system matrices and subsequently larger computational time and memory required. Also, it is by no means a general fact that larger widths always result in a more accurate interpolation. For these reasons, this chapter proposes an automatic method for finding the optimum correlation length. With regards to the second difference: when applying SGPR to 3D-PTV we cannot exploit a Toeplitz structure of the system matrix, since the measurements are not available on a regular grid. With measurements distributed randomly, we are forced to store the full matrix. However, memory and computational time savings can be achieved by using compact covariance functions instead of covariance functions with global support. Following Chapter 3, this chapter will also use Wendland functions. These two differences seem to indicate that SGPR is less suited for 3D-PTV than it was for tomo-PIV. However, that is not necessarily the case. Indeed, the number of measurements from 3D-PTV is significantly smaller than tomo-PIV data, especially when using HFSBs, a primary motivation for the present chapter.

## 4.2. PARAMETER DETERMINATION

To find the optimum parameters (correlation length $\boldsymbol{\gamma}$ from Eq.2.14 and the regularization parameter $\lambda$ discussed in Section 3.5), we follow an approach popular in machine learning (Suthaharan, 2015). With $N_p$ measurement points coming from the tracked particles:

$$\mathbf{x}_{p_j} \in \mathbb{R}^3, \;\; j \in P = \{j \in \mathbb{N} | 1 \le j \le N_p\},$$

the set $P$ is split into two disjoint sets $T$ and $V$, representing the training set and validation set, respectively. The $N_{\text{train}}$ points belonging to the training set are denoted $\hat{\mathbf{x}}_p$ and the $N_{\text{val}}$ points belonging to the validation set are denoted $\check{\mathbf{x}}_p$. The training set is used to fit the interpolation method to the velocity measurements in the training set, $\hat{\mathbf{u}}_p$. The resulting interpolation is then evaluated at the locations of the validation set, giving the predicted velocity $\check{\mathbf{u}}_p^*$. We define the following objective function:

$$J(\boldsymbol{\gamma}, \lambda) = \sqrt{\frac{1}{N_{\text{val}}} \sum_{k=1}^{N_{\text{val}}} \left(\check{\mathbf{u}}_{p,k}^* - \check{\mathbf{u}}_{p,k}\right)' \left(\check{\mathbf{u}}_{p,k}^* - \check{\mathbf{u}}_{p,k}\right)},$$

which simply represents the root-mean-square error between the predicted velocity $\check{\mathbf{u}}_p^*$ and measured velocity $\check{\mathbf{u}}_p$ of the validation set. The arguments of the objective function are the parameters we are looking for. The *optimum* parameters are found by minimising $J$. The reason for using a validation set, separate from the training set, is to avoid overfitting. Figure 4.1 illustrates how the objective function is calculated.

With regards to choosing the relative size of the training and validation set, there is no universal rule. A common approach is 80% for the training set and 20% for the validation set (Suthaharan, 2015). This is known as the *Pareto principle* and is the rule we have followed for the synthetic cases in Section 4.3.

For the experimental case in Section 4.4 we have defined a training set in addition to the fitting and validation sets. After finding the optimum parameters, the velocity field is evaluated at the locations of the test set and compared to the test set measurements,

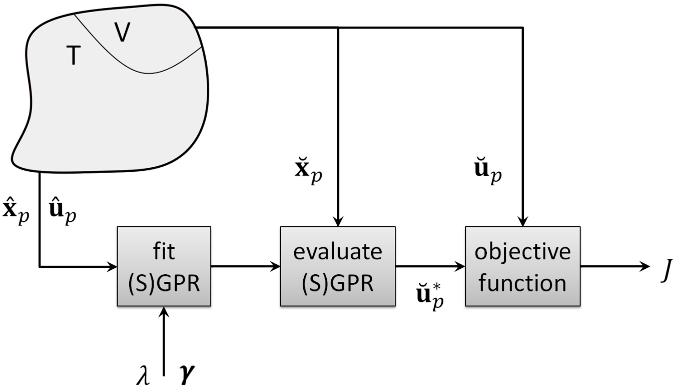


Figure 4.1: Calculation of the objective function $J$ to be minimised. The measurement set is randomly split into the disjoint sets $T$ and $V$, containing the training and validation points, respectively.

from which we can calculate the root-mean-square error. This gives a final metric for the quality of the interpolation and allows us to compare with other interpolation methods. We have chosen the following subdivision for the experimental case: 70% for the training set, 15% for the validation set and 15% for the test set. We therefore stay relatively close to the Pareto principle. Of course, for the synthetic cases there is also test set, but since we can evaluate the true fields anywhere in the domain, we can choose a large number of points for this, without taking them from the original 'measurement set'.

As stated earlier, large values of the correlation length result in dense system matrices and subsequently larger computational time and memory required. For that reason, we suggest not to use a gradient-free optimisation algorithm that requires bracketing the interval, since we would like to avoid evaluating at large correlation lengths. Using a gradient-based method is preferred since it requires an initial *point*. Obviously, choosing the initial point at the lower bound of the interval is ideal. For the correlation length, this would be the mean distance between the locations in the training set. For the regularization parameter, one could choose a number in the order of machine-precision, resulting in interpolation. With regards to the type of optimisation algorithm, Newton's method is better than the steepest descent method. Not only does it show quadratic convergence (Wright and Nocedal, 1999), it does not carry out a one-dimensional line search, which again requires bracketing the interval.

This approach of splitting the data for parameter determination is not necessarily better than maximum likelihood estimation (Cressie, 2015). The main reason for choosing it is computational efficiency. Indeed, by splitting the measurement set into a training and validation set, we decrease the size of the training set and subsequently computational time. Maximum likelihood on the other hand uses all data points.

## 4.3. APPLICATION TO SYNTHETIC TEST CASES

The synthetic test cases are the same as described in Appendix A and also used in Chapter 3. The following interpolation methods are considered:

(i) linear interpolation (LIN), where the velocity components are interpolated independently from one another;

(ii) linear interpolation followed by application of a solenoidal filter (LIN + DCS). First, the measurement points are interpolated onto a fine regular grid using linear interpolation. Then, the solenoidal filter proposed by de Silva et al (2013), known as the *divergence correction scheme*, is applied creating a solenoidal velocity field on a finite-difference level at the resolution of the regular grid;

(iii) Gaussian process regression (GPR), where the velocity components are interpolated independently from one another;

(iv) Gaussian process regression followed by application of the divergence correction scheme (GPR + DCS);

(v) solenoidal Gaussian process regression (SGPR).

The results will be presented as an interpolation improvement with respect to linear interpolation in percentages, denoted $Q$. The metric is defined as follows:

$$Q = \left( \frac{\epsilon_{\text{lin}} - \epsilon_{\text{interp}}}{\epsilon_{\text{lin}}} \right) \times 100\%, \tag{4.1}$$

where $\epsilon_{\text{lin}}$ is the root-mean-square error of the field obtained with linear interpolation and $\epsilon_{\text{interp}}$ is the root-mean-square error of the field obtained with the other interpolation methods.

### 4.3.1. TAYLOR-VORTEX

The first synthetic test case is the two-dimensional Taylor-vortex (see Appendix A.1.1), for which analytical expressions of the velocity, vorticity and pressure fields are available. Figures 4.2-4.4 show how the improvement with respect to linear interpolation changes with the number of measurement points $N_p$ for the velocity, vorticity and pressure fields, respectively. The number of measurement points we consider is between 10 and 200. Starting with LIN + DCS, i.e. application of the discrete solenoidal filter after linear interpolation, we see a consistent improvement in the velocity field (see Figure 4.2) of around 10%. The vorticity field however (see Figure 4.3) is unaltered. This was already concluded in Chapter 3, where it was found that DCS leaves the vorticity field unaltered, equivalent with the Helmholtz Representation Theorem. The improvement in the pressure field (see Figure 4.4) starts out at approximately 10% for the lowest number of measurements and gradually increases to around 70%.

Proceeding to GPR, we observe that it actually produces less accurate fields than linear interpolation at the sparsest measurements. This does not completely come as a surprise considering that we split the measurement points into a training and validation set for GPR to find the optimum parameters. Therefore, fewer measurement points are

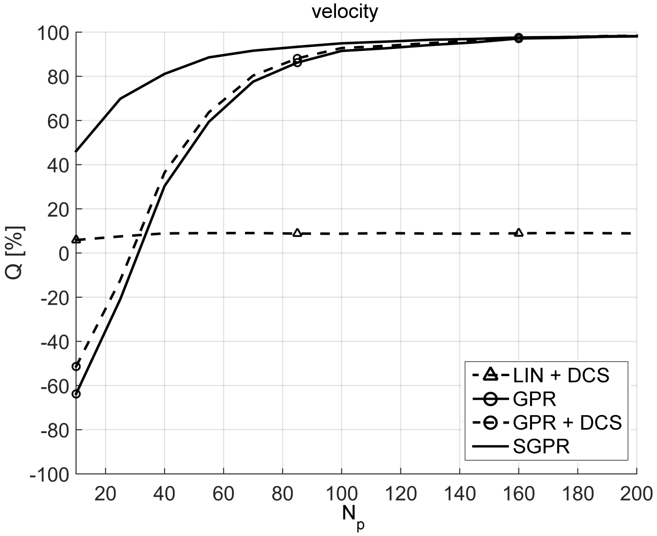


Figure 4.2: The Taylor-vortex: improvement with respect to linear interpolation (Q) as a function of number of measurement points ($N_p$) for the **velocity field**.

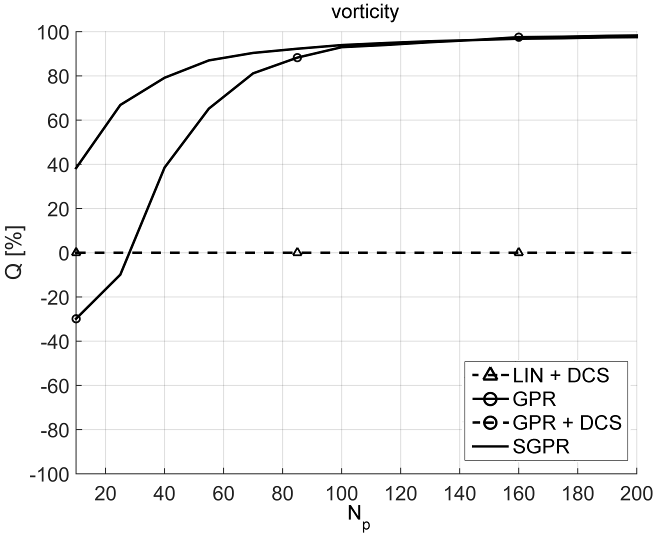


Figure 4.3: The Taylor-vortex: improvement with respect to linear interpolation (Q) as a function of number of measurement points ($N_p$) for the **vorticity field**.

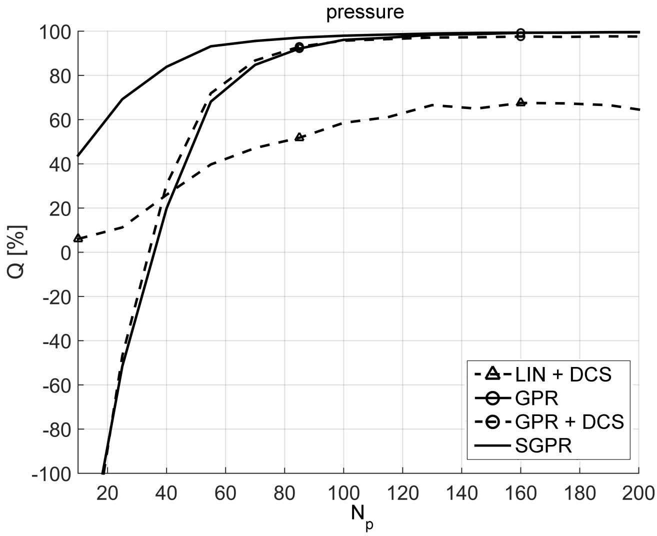
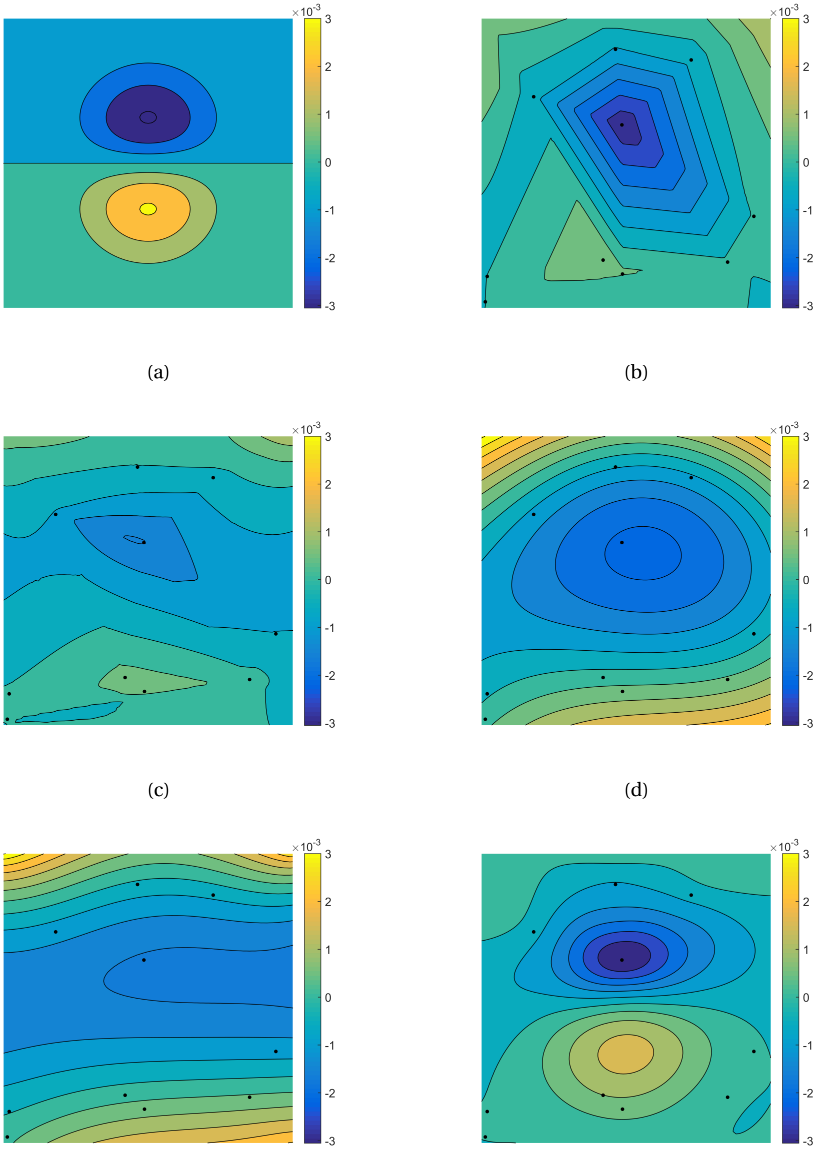


Figure 4.4: The Taylor-vortex: improvement with respect to linear interpolation (Q) as a function of number of measurement points ($N_p$) for the **pressure field**.

actually used for the interpolation. Linear interpolation on the other hand does not require parameters, so we use both the training and validation as one set for interpolation. However, when we use approximately 30 measurement points and more, GPR becomes more accurate. As we further increase the number of measurement points, the improvement of GPR becomes higher, until it almost reaches 100%. Again, this behaviour is expected since GPR exhibits spectral convergence. Application of the solenoidal filter after GPR (GPR + DCS) consistently results in more accurate velocity and pressure fields. The vorticity field however is not affected, as explained earlier.

Finally, we investigate SGPR. Contrary to GPR, which results in a poorer field than linear interpolation for the sparsest measurements, SGPR always produces more accurate fields. At the lowest number of measurements, the velocity, vorticity and pressure fields are approximately 40% more accurate. Additionally, it is always better than GPR or GPR + DCS. However, as more measurements are added, the improvement of SGPR with respect to GPR and GPR + DCS decreases. These observations indeed confirm that it becomes more beneficial to include the divergence-free constraint for very sparse measurements.

Figure 4.5 shows the velocity component in the horizontal direction for the true velocity field (see Figure 4.5(a)) and resulting from the interpolation methods (Figures 4.5(b)-4.5(f)) for the lowest number of measurement points considered, namely $N_p = 10$. The black dots are the measurement points. SGPR is the only method that captures the vortex.

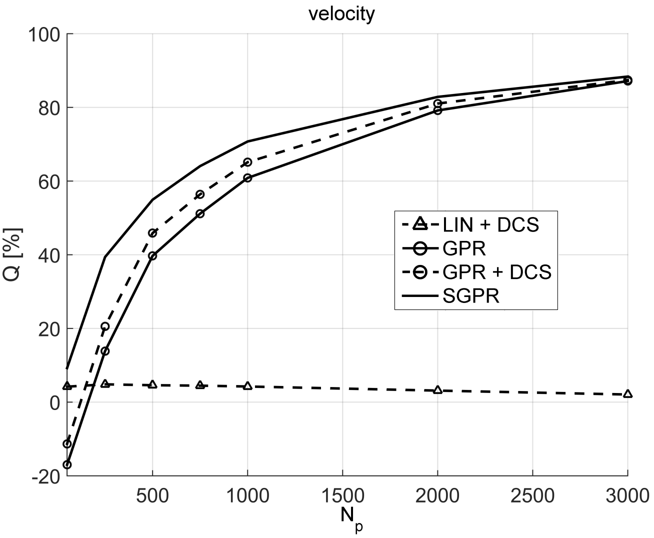


(a) (b)

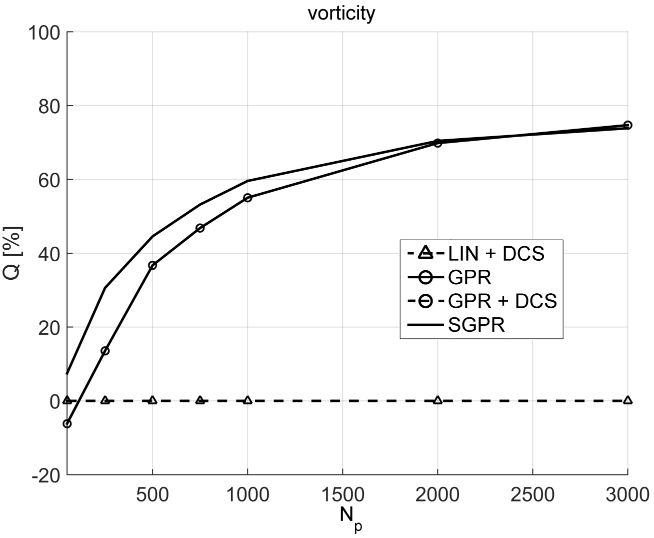
(c) (d)

(e) (f)

Figure 4.5: The Taylor-vortex: true velocity in the horizontal direction (a); velocity in the horizontal direction with 10 measurement points using (b) LIN, (c) LIN + DCS, (d) GPR, (e) GPR + DCS and (f) SGPR.



(a)



(b)

Figure 4.6: The vortex ring: improvement with respect to linear interpolation (Q) as a function of number of measurement points ($N_p$) for the velocity (a) and vorticity (b) fields.

### 4.3.2. VORTEX RING

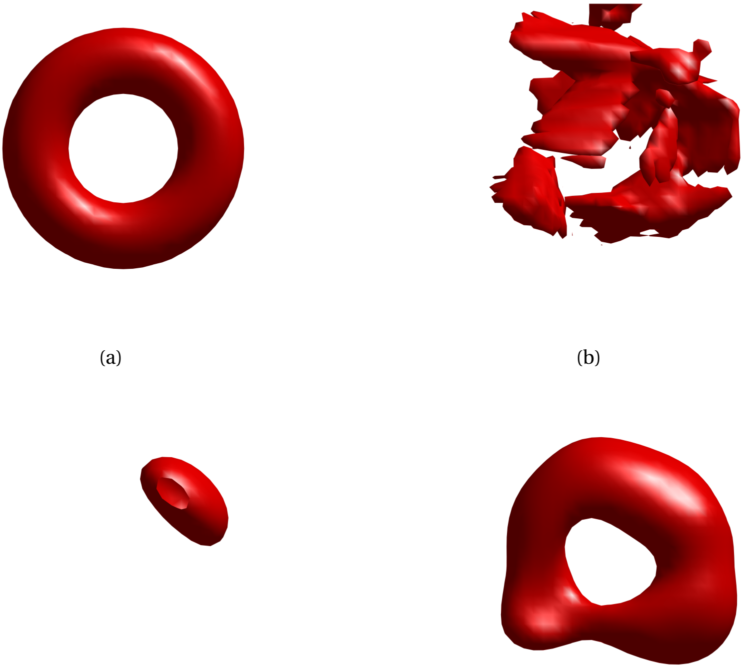
The second synthetic test case is the three-dimensional vortex ring, for which analytical expressions of the velocity and vorticity fields are available (see Appendix A.1.2). Figure 4.6 shows how the improvement with respect to linear interpolation changes with the number of measurement points. The number of measurement points we consider is between 50 and 3000. Starting with LIN + DCS, we see a consistent improvement in the velocity field around 5%. Again, no improvement is obtained in the vorticity field. Proceeding to GPR, we again observe for the sparsest data set that it is worse than linear interpolation. However, around 100 measurement points, the two methods give the same results. With increasing number of sample points, the improvement obtained with GPR increases monotonically. Looking at the velocity field, applying the DCS filter (GPR + DCS) consistently results in an improvement, although the improvement decreases with increasing number of sample points. Finally, SGPR consistently returns the most accurate velocity and vorticity, although the gain with respect to GPR or GPR + DCS decreases as the data set becomes larger.

Figure 4.7 shows isosurfaces of the vorticity magnitude, viewing the vortex ring from the top. We show the results for 50 measurement points, i.e. the lowest number of measurement points considered. We do not show the result when DCS is included, since it leaves the vorticity field unchanged. Notice that SGPR is the only method that returns the vortex ring.

## 4.4. APPLICATION TO EXPERIMENTAL DATA

We use the experiment of the circular jet in water, described in Appendix A.2.1. The data set used in Chapter 3 was obtained using a cross-correlation based technique, resulting in measurement points on a regular grid. In this chapter, we use a particle tracking algorithm instead (Schneiders et al, 2015a). The data set contains 5,678 velocity vectors scattered over the measurement domain. This translates to a seeding concentration of $9.5 \times 10^{-5}$ particles per voxel (ppv). Neutrally buoyant polyamide particles were used as tracers for the experiment, resulting in relatively high seeding concentration as compared to an experiment where helium-filled soap bubbles are used. The advantage of using this experiment though is that we can investigate the influence of the seeding concentration on the interpolation accuracy by deliberately reducing the size of the measurement set. The data is randomly divided into a training (70%), validation (15%) and test set (15%). To investigate the influence of the seeding concentration on the interpolation accuracy, we have also reconstructed velocity fields by randomly leaving out fractions of the training and validation data. We went as low as effectively reducing the ppv with an order of magnitude. Figure 4.8 shows the improvement with respect to linear interpolation, evaluated on the test test. Application of the solenoidal filter after linear interpolation results in an improvement around 10%. Using GPR results in an improvement of 20% for the lowest seeding concentration to around 45% for the highest concentration considered. Application of the solenoidal filter to GPR results in a further improvement of around 5%. Finally, SGPR consistently returns the most accurate velocity fields, with improvements of around 35% for the lowest concentration to approximately 55% for the highest concentration.

To get a qualitative understanding of the interpolation accuracies, Figure 4.9 shows



(a) (b) (c) (d)

Figure 4.7: The vortex ring: isosurfaces of vorticity magnitude, viewing the vortex ring from the top. The true field (a), results obtained with LIN (b), GPR (c) and SGPR (d). 50 measurement points are used.

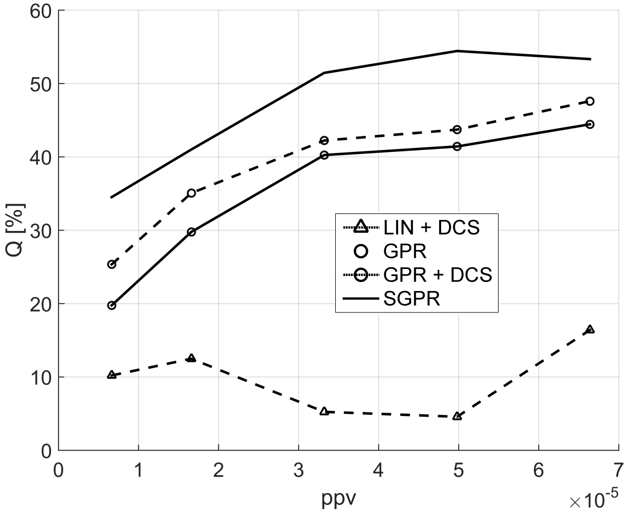
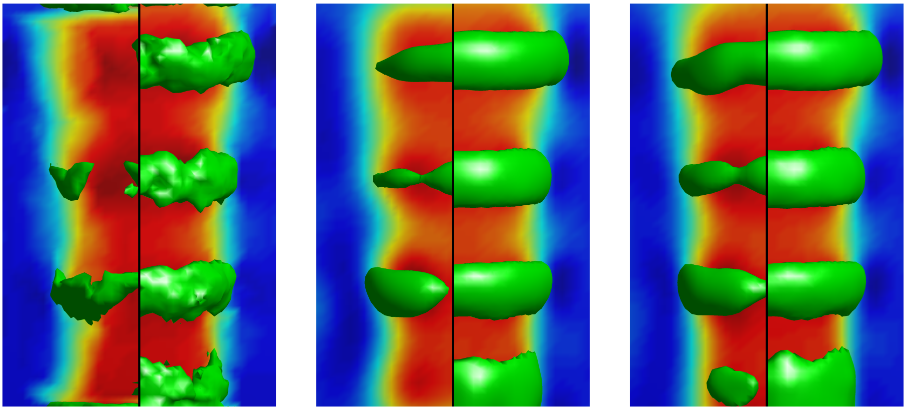


Figure 4.8: Circular jet in water: improvement evaluated on the test set with respect to linear interpolation, as a function of the seeding concentration (ppv).

the resulting fields using the interpolation methods at the lowest (left-hand side) and highest (right-hand side) particle concentrations. From Appendix A.2.1 (Figure A.3) we see that vortex rings are shed from the nozzle. For the lowest concentration, linear interpolation does not reproduce all vortex rings. GPR returns the top three rings, but not the bottom ring. SGPR on the other hand returns all rings. For the highest seeding concentration, all methods show four vortex rings. GPR and SGPR return smoother fields, as expected, but there is little difference between them.

## 4.5. CONCLUSION

We investigated whether enforcing the divergence-free constraint can improve the interpolation of velocity (and derived) fields obtained from sparse 3D-PTV data of incompressible flows. Since GPR and SGPR have parameters that need to be specified, we proposed an efficient methodology that automatically finds the values for optimum interpolation based on the information provided by the measurement itself. We found that incorporating the constraint analytically using SGPR results in more accurate interpolations than applying a discrete solenoidal filter to a linearly interpolated field or a field obtained using GPR. In addition, we found that SGPR consistently returns the most accurate velocity, vorticity and pressure fields. The improvement is most pronounced for the sparsest data sets. These observations were confirmed using both synthetic test cases and an experimental data set.



(a) (b) (c)

Figure 4.9: Circular jet in water: interpolated axial velocity (contour-plots) and vorticity magnitude (isosurfaces) fields for the lowest (left-hand plots) and highest (right-hand plots) seeding concentrations using LIN (a), GPR (b) and SGPR (c).

# 5

# SPATIO-TEMPORAL INTERPOLATION OF SPARSE 3D-PTV DATA

*This chapter investigates how incorporating the temporal information of particle tracks in 3D-PTV can improve the reconstructed velocity and vorticity field. To incorporate temporal information, we use the momentum conservation equation formulated in the vorticity-velocity form via the vorticity transport equation, in addition to mass conservation. We model these equations using the vortex-in-cell (VIC) solver. Section 5.2 presents a mathematical formulation of the data-assimilation problem, using the results obtained in Chapter 2 as the starting point. As was explained in that chapter, inevitably some assumptions and simplifications need to be made to allow for a practical implementation. The resulting algorithm is formulated as an optimisation problem, i.e. the solution to the data-assimilation problem lies in finding the variables that minimize a cost function. Section 5.3 describes the optimisation algorithm we use and Section 5.4 shows how the cost function is calculated. The optimisation algorithm requires the gradient of the cost function. Section 5.5 shows how it can efficiently and exactly be obtained. Section 5.6 applies the algorithm to an experimental data set and compares it with solenoidal interpolation as explained in Chapter 4, which interpolates the snapshots independently from one another, therefore not taking the temporal information into account. We find clear improvements in the reconstructed field when incorporating temporal information. Conclusions are drawn in Section 5.7.*

## 5.1. INTRODUCTION

Recall how we classified data-assimilation methods in Chapter 2: in Section 2.3, we described methods that assimilate spatial fields; in Section 2.4, we described methods that assimilate spatio-temporal fields. In the previous two chapters, we considered the former methods: in Chapter 3, we reduced noise in tomographic PIV data and in Chapter

4 we interpolated sparse 3D-PTV data. In both cases, we did not require time-resolved measurements and only included the divergence-free constraint as the physical knowledge.

Naturally, one could ask whether spatio-temporal data-assimilation methods can provide a further improvement in noise reduction and spatial resolution of volumetric velocity measurements.

The topic of noise reduction has recently been investigated by Schneiders et al (2015b). Their goal was to reduce the random error of instantaneous velocity fields. The method they used was to combine a short-sequence of time-resolved tomographic PIV measurements with multiple time-marching simulations, in which these simulations were started at the various snapshots of the measurements and integrated (forwards and backwards) towards the instantaneous time of interest. The simulations, or physical knowledge, was based on the momentum conservation equation formulated by the vorticity transport equation (see Eq.1.3) and solved using the vortex-in-cell method. With regards to improving spatial resolution, this has recently been investigated by Schneiders et al (2015a), where again the vortex-in-cell method was used as the flow solver. The present chapter is a proof-of-concept study that focuses on the topic of increasing spatial resolution in sparse volumetric velocity measurements. The method used follows from the mathematical formulation of the data-assimilation problem outlined in Chapter 2, where a number of assumptions and simplifications have been made to allow for a practical implementation. By making additional simplifications, the method presented by Schneiders et al (2015a) follows.

Before proceeding, it is worth mentioning other works in the field of experiments in fluids that have explicitly exploited the temporal information from time-resolved velocity measurements. For correlation-based techniques, a number of methods have been proposed to reduce random and bias errors and enhance measurement robustness (Jeon et al, 2014; Lynch and Scarano, 2013; Sciacchitano et al, 2012b). These methods are not limited to volumetric measurements and can be applied to 2D PIV as well. Other methods have explicitly been developed for volumetric velocity measurements (Lynch and Scarano, 2015; Novara et al, 2010; Schanz et al, 2013). These methods do not take into account the physical conservation laws. In the field of optical flow, a number of authors have investigated the use of physics-based constraints to reconstruct velocity fields, directly from particle images. We refer to the review paper by Heitz et al (2010) and the references therein. At the end, one of the perspectives they provide is the following:

> *This move to three dimensions plus eventually time will likely enable physics-based models and methods to provide accurate inspection tools for experimental fluid mechanics.*

The present chapter is aimed as a contribution towards that goal.

## **5.2.** MATHEMATICAL FORMULATION

A particle tracking algorithm returns the location of a particle $p_j$ at snapshot-time $t_l$:

$$\mathbf{x}^l_{p_j} \in \mathbb{R}^3,$$

where $j = 1, ..., N_p$ and $l = 1, ..., N_t$, i.e. there are $N_p$ particles and $N_t$ snapshots. Polynomials can be fitted through each particle track, resulting in a reduction of the random measurement error (Schanz et al, 2013). Taking the time derivative of a particle track gives the particle velocity, which is considered the measured velocity

$$\mathbf{U}_{p_j}^{l} \in \mathbb{R}^3.$$

The state we are interested in is the velocity field as a function of space and time: $\mathbf{u}(\mathbf{x}, t)$. The time evolution of the state follows from the Navier-Stokes equations, as expressed by Eq.1.2. In the present chapter however, we use the vorticity-velocity formulation of the Navier-Stokes equations, since it does not contain the pressure. Therefore, it is more suitable in the present work where the measurements are of velocity. The vorticity-transport equation was given by Eq.1.3. For completeness, we state all the relevant equations below:

$$\frac{\partial \boldsymbol{\omega}}{\partial t} = -(\mathbf{u} \cdot \nabla)\boldsymbol{\omega} + (\boldsymbol{\omega} \cdot \nabla)\mathbf{u} + \nu \Delta \boldsymbol{\omega}, \tag{5.1}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{5.2}$$

$$\nabla \times \mathbf{u} = \boldsymbol{\omega}. \tag{5.3}$$

Eq.5.3 simply states that the vorticity $\boldsymbol{\omega}$ is the curl of the velocity field. The above system is completed with initial and boundary conditions:

$$\mathbf{u}(\mathbf{x}, t_0) = \mathbf{u}_0(\mathbf{x}), \tag{5.4}$$

$$\mathbf{u}(\mathbf{x}, t)\,|_{\Gamma} = \mathbf{u}_b\left(\boldsymbol{\xi}, t\right), \tag{5.5}$$

$$\boldsymbol{\omega}(\mathbf{x}, t)\,|_{\Gamma} = \boldsymbol{\omega}_b\left(\boldsymbol{\xi}, t\right). \tag{5.6}$$

Note that in addition to time-varying boundary conditions for velocity, the vorticity-velocity formulation also requires time-varying boundary conditions for the vorticity. Section 5.4.3 shows what happens if we omit the time-varying vorticity boundary conditions. We proceed by discretising the field in space and time, where we define $n$ spatial locations and $K$ time intervals. The discretisation is carried out on a regular grid for the vortex-in-cell method, where the volume of each cell is

$$V = \prod_{l=1}^{3} \Delta x_l. \tag{5.7}$$

The subscripts $l = 1, 2, 3$ represent the three Cartesian coordinate directions. To distinguish the continuous variables from the discretized variables, we use the symbol $\hat{}$ for the latter. So we have

$$\hat{\mathbf{u}}^k \in \mathbb{R}^{3n},$$

where $k = 0, 1, ..., K$. In light of the velocity and vorticity boundary conditions from Eqs.5.5 and 5.6, respectively, we augment the complete state with $\hat{\mathbf{u}}_b^k$ and $\hat{\boldsymbol{\omega}}_b^k$, with $k = 1, ..., K$.

Considering the nonlinearity of the vorticity transport equation, we can either use a sequential or variational method as the data-assimilation algorithm (see Section 2.4.2). We choose the latter because of two reasons. Firstly, we can process all measurements simultaneously since the particle tracks will all be available from the particle tracking algorithm: no on-line assimilation is required. Secondly, in the present chapter we are not interested in quantifying or propagating the uncertainty of the state. Rather, our goal is to increase the spatial resolution of the measurements. Adding to this the fact that in Section 5.5 we will describe an efficient method to calculate gradients of the cost function with respect to the state, the variational method is preferred.

We proceed by stating some simplifications. First of all, we reduce the state to the initial velocity field and the time-dependent velocity and vorticity boundary conditions:

$$\boldsymbol{\phi} = \left(\hat{\mathbf{u}}^0, \hat{\mathbf{u}}_b^k, \hat{\boldsymbol{\omega}}_b^k\right),$$

where $k = 1,...,K$. This means that we assume the underlying model is perfect. In the field of weather forecasting, this results in the well-known variational algorithm known as *4DVAR* (4-dimensional variational method) (Talagrand and Courtier, 1987). The disadvantage of this simplification is that it limits the time interval over which the model can be integrated, i.e. the number of snapshots that can be assimilated. This is also a well-known limitation of 4DVAR. The second simplification is removing the terms containing the discrepancy with the priors of the state. Practically, these terms tend to avoid that the solution will deviate too much from the prior or initial guess. From an optimisation point of view, they aim to replace an originally constrained problem by an unconstrained one by acting as penalty terms (Wright and Nocedal, 1999). The advantage is that unconstrained optimisation algorithms can be used to solve the problem. The disadvantage is that one needs to specify how to weigh these terms with the other terms in the cost function. From a Bayesian point of view, these weights are expressed in terms of the inverse of the covariance matrices (see Eq.2.34). Taking the mentioned simplifications into account, the cost function from Eq.2.34 that needs to be minimised becomes:

$$J\left(\hat{\mathbf{u}}^0, \hat{\mathbf{u}}_b^{\{1:K\}}, \hat{\boldsymbol{\omega}}_b^{\{1:K\}}\right) = \sum_{l=1}^{N_t} \sum_{j=1}^{N_p} \mathbf{z}_j^{l\,\prime} \frac{1}{{\sigma_j^l}^2} \mathbf{z}_j^l, \tag{5.8}$$

where the vector $\mathbf{z}_j^l$ is defined as:

$$\mathbf{z}_j^l = \mathbf{U}_{p_j}^l - \boldsymbol{\mathcal{H}}\left(\hat{\mathbf{u}}^l; \mathbf{x}_{p_j}^l\right). \tag{5.9}$$

The only term in the cost function that has remained is the term representing the discrepancy of the model with the measurements. The standard deviation of the measurement uncertainty at location $j$ and time step $t_l$ is $\sigma_j^l$, and we have assumed that the measurement uncertainty of the different particles is uncorrelated with each other. Recall that for PIV data, where overlapping interrogation windows are used, it naturally follows that the measurement uncertainty of neighboring points will be correlated. With particle tracking on the other hand, the particles are tracked individually. We therefore assume the measurements to be uncorrelated. From this it follows that the measurement uncertainty matrix is diagonal, resulting in the scalar term in Eq.5.8. The operator $\boldsymbol{\mathcal{H}}$ in Eq.5.9 uses the velocity field on the regular grid $\hat{\mathbf{u}}^l$ at time $t_l$ and carries out an interpolation to obtain the velocity at the measurement position $\mathbf{x}_{p_j}^l$. We will also denote

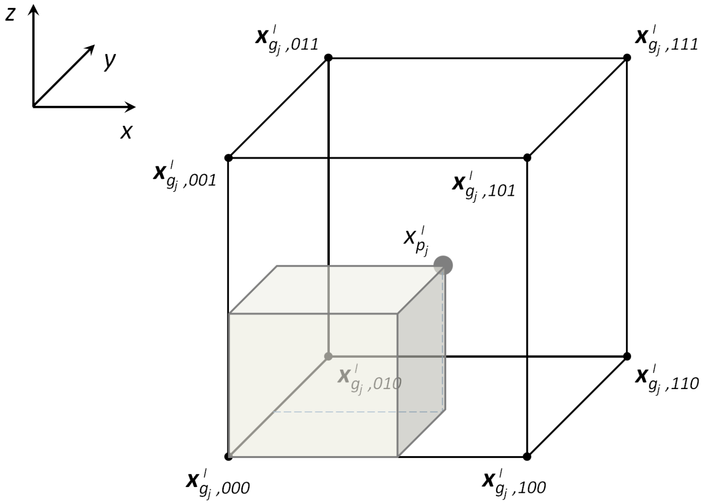


Figure 5.1: The interpolated value at $\mathbf{x}^l_{p_j}$ is obtained from the vertices of the cell containing $\mathbf{x}_{p_j}$ using trilinear interpolation.

it as:

$$\hat{\mathbf{U}}^l_{p_j} = \mathcal{H}\left(\hat{\mathbf{u}}^l;\mathbf{x}^l_{p_j}\right). \tag{5.10}$$

Specifically, since the interpolation is carried out from a relatively fine regular grid onto a sparse set of randomly distributed points, we use trilinear interpolation:

$$\begin{aligned}\hat{\mathbf{U}}^l_{p_j} =& \frac{V^l_{111}}{V}\hat{\mathbf{u}}^l_{pj,000} + \frac{V^l_{011}}{V}\hat{\mathbf{u}}^l_{pj,100} + \frac{V^l_{001}}{V}\hat{\mathbf{u}}^l_{pj,110} + \frac{V^l_{101}}{V}\hat{\mathbf{u}}^l_{pj,010} + \dots \\ & \frac{V^l_{110}}{V}\hat{\mathbf{u}}^l_{pj,001} + \frac{V^l_{010}}{V}\hat{\mathbf{u}}^l_{pj,101} + \frac{V^l_{000}}{V}\hat{\mathbf{u}}^l_{pj,111} + \frac{V^l_{100}}{V}\hat{\mathbf{u}}^l_{pj,011}\end{aligned} \tag{5.11}$$

We use Figure 5.1 to explain the components. Given a particle location $\mathbf{x}^l_{p_j}$ we obtain its velocity from the grid nodes of the cell in which it resides. $\hat{\mathbf{u}}^l_{pj,abc}$ is the velocity at the point $\mathbf{x}^l_{gj,abc}$ of the regular grid. $V^l_{def}$ is the volume of the rectangle opposite $\mathbf{x}^l_{gj,abc}$, across the diagonal of the cell. As an example we have plotted the transparent rectangle in Figure 5.1, which has a volume $V^l_{000}$. It lies opposite $\mathbf{x}^l_{gj,111}$ across the diagonal of the cell.

Before proceeding, it is worth mentioning the cost function used by Schneiders et al (2015a).

$$J = J_u + J_{Du} + J_r. \tag{5.12}$$

The first component $J_u$ is similar to Eq.5.8, with two differences: in the present chapter multiple time frames can be assimilated. In other words, $N_t = 1$ in Schneiders et al (2015a). The second difference is that in the present chapter, the velocity measurement uncertainty has explicitly been taken into account. The second component $J_{Du}$ represents the discrepancy of the material acceleration between the model and the measurements. The third and final component contains the sum of squares of the velocity divergence and the vorticity divergence. Due to the parameterization discussed in Section 5.4.2, the method used in the present chapter does not require explicitly defining the third component.

## 5.3. OPTIMISATION ALGORITHM

To find the minimum of the cost function $J$, we use a quasi-Newton method, known as limited-memory BFGS (L-BFGS) (Liu and Nocedal, 1989). Quasi-Newton methods can be seen as providing a good compromise between steepest descent methods and Newton's method. They only require the gradient of the cost function $J$ like the steepest descent methods. Therefore they are computationally cheaper than Newton's method, which requires second derivatives but shows quadratic convergence behaviour near the minimum. However, since quasi-Newton methods attempt to approximate the second derivatives of the objective function they show better convergence properties (superlinear) than the steepest descent method, which shows linear convergence behaviour. The specific method we use to approximate the second derivatives with the gradient information alone is the BFGS method, which is the most popular method (Wright and Nocedal, 1999). The limited-memory variant is useful for the present problem, since the initial condition and time-dependent boundary conditions together result in a large state. Like all local optimisation algorithms, L-BFGS requires an initial point to be supplied. Due to the nonlinearity of the cost function, there may be multiple local minima. Choosing a suitable initial condition is therefore crucial. To this end, we interpolate each frame of the sparse measurement independently from one another, as described in Chapter 4.

## 5.4. COST FUNCTION CALCULATION

Figure 5.2 illustrates how the cost function is calculated. The boxes called 'box' apply the constraints, as discussed in Section 5.4.1. The boxes with '$\mathrm{C_0}$' and '$\mathrm{C_k|G_k}$' represent an (S)GPR transformation, where the inputs are weightings (akin to $\mathbf{c}$ in Eq.2.19) and the outputs are the interpolated fields (akin to $\boldsymbol{\phi}$ in Eq.2.20). They are explained in more detail in Section 5.4.2, together with the inputs in red. The box 'VIC' is the vortex-in-cell method we use to evolve the velocity field forward in time and is described in Section 5.4.3. The box '$\boldsymbol{\mathcal{H}}$' represents the trilinear interpolation from Eq.5.11 and evaluates the simulated velocity field from the regular grid at the particle locations. Finally, 'sse' evaluates the sum of squared errors or discrepancy between the simulated and measured velocities, as expressed by Eq.5.8.

### 5.4.1. CONSTRAINT HANDLING

For robustness and computational efficiency, it is desired to constrain the optimisation algorithm. With regards to robustness, we need to avoid that it converges to an unphysical field. With regards to computational efficiency, we need to avoid that it wastes time on evaluations in unphysical domains. In 4DVAR this is typically achieved without ex-

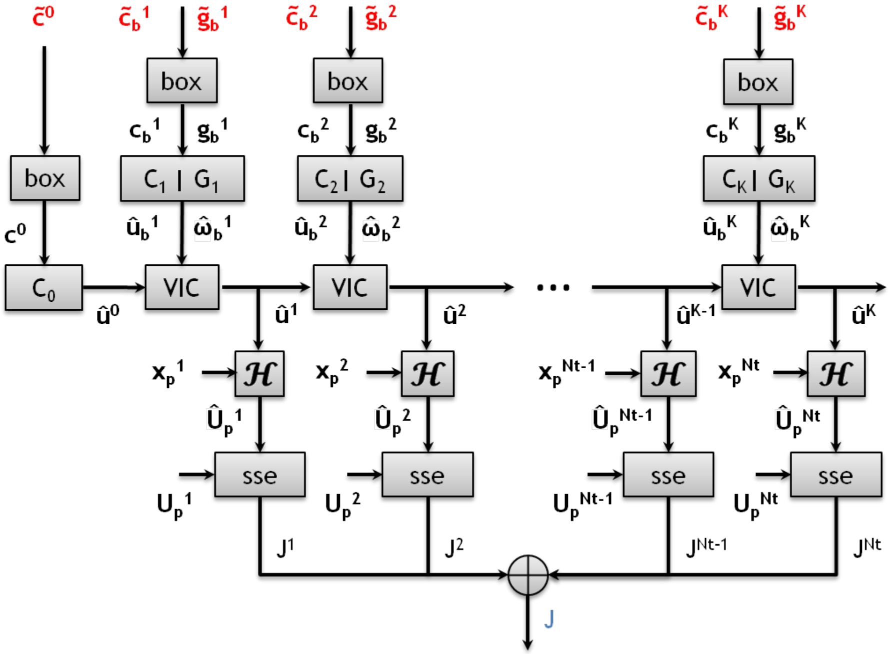


Figure 5.2: Diagram to calculate the cost function $J$ from Eq.5.8.

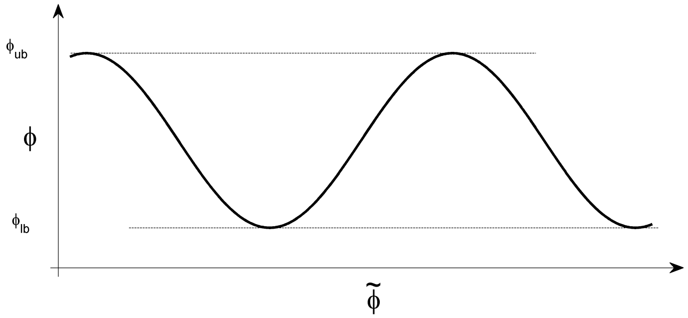


Figure 5.3: Transformation to ensure the unconstrained problem with $\tilde{\phi}$ results in $\phi_{lb} \le \phi \le \phi_{ub}$.

plicitly using a constrained optimisation algorithm. Instead, the first two terms in the cost function from Eq.2.34, representing the prior initial condition and prior boundary conditions, are used for this. From an optimisation point of view they act as penalisation terms, therefore hindering the optimisation algorithm to converge to solutions 'too far' from the prior. The disadvantage is the need to specify how to weight the different terms in Eq.2.34. We constrain the optimisation by applying a transformation of the variables. The domain in which the optimisation algorithm works is unbounded but the transformation ensures that the result will be within the specified bounds. The advantages are that we still can use the unconstrained optimisation algorithm L-BFGS and we can remove the penalization terms in the original cost function in the form of the priors (see Eq.2.34), therefore avoiding the need to specify weightings. Indeed, the cost function from Eq.5.8 that we minimise does not contain the prior terms.

The transformation works as follows. We require a variable $\phi$ to remain within a predefined interval:

$$\phi_{lb} \le \phi \le \phi_{ub},$$

where $\phi_{lb}$ and $\phi_{ub}$ are the lower and upper bounds, respectively. We replace this constrained variable by an unconstrained variable $\tilde{\phi}$ and introduce the following transformation:

$$\phi = \phi_{lb} + \frac{1}{2}\left(\phi_{ub} - \phi_{lb}\right)\left(\sin\tilde{\phi} + 1\right). \tag{5.13}$$

We can now use an unconstrained optimisation algorithm with variable $\tilde{\phi}$ with the guarantee that the optimum will be inside the defined interval (see also Figure 5.3).

### **5.4.2.** PARAMETERIZATION

The reduced state, containing the initial velocity field $\hat{\mathbf{u}}^0$, the time varying velocity boundary condition $\hat{\mathbf{u}}_b^{\{1:K\}}$ and the time varying vorticity boundary condition $\hat{\boldsymbol{\omega}}_b^{\{1:K\}}$, is defined on a regular grid as required by the VIC solver. Initially, we had parameterized the state by directly defining the (velocity and vorticity) components on the nodes of the regular grid. This resulted in the optimisation routine returning unphysical initial and boundary conditions, containing peaks. This approach was also followed by Schneiders et al

(2015a). To avoid unphysical results, we therefore define the state through a parameterization using SGPR, ensuring a smooth solenoidal field. SGPR therefore acts as a regularization term. In this section, we both explain the parameters we defined and how we find a first-guess for the optimisation algorithm.

We give a detailed explanation for the initial velocity field. The procedure for the velocity and vorticity boundary conditions is the same. The initial velocity field is defined as:

$$\hat{\mathbf{u}}^0 = C_0\mathbf{c}^0. \tag{5.14}$$

Covariance functions are placed at each node on the regular grid. Therefore, $C_0$ is the 3-level symmetric Toeplitz matrix from Chapter 3 and defined by Eq.3.16. The parameter for the initial velocity field is $\mathbf{c}^0$, which we denote a *weight* vector. To find a first-guess for it, we interpolate the velocity measurements at the initial snapshot $\mathbf{U}_p^0$ onto the regular grid. Following Eq.2.19, which showed how a weight vector can be obtained from measurements, we write:

$$\bar{\mathbf{c}}^0 = A_0^{-1}\mathbf{U}_p^0, \tag{5.15}$$

where $A_0$ is the solenoidal gain matrix from Eq.3.14 corresponding to the measurement locations $\mathbf{x}_p^0$. Once we have found $\bar{\mathbf{c}}^0$ from Eq.5.15, we interpolate onto the fine grid. Following Eq.2.20, we get:

$$\hat{\mathbf{u}}^0 = \bar{C}_0\bar{\mathbf{c}}^0, \tag{5.16}$$

where $\bar{C}_0$ is simply the solenoidal covariance matrix representing the covariance between the points on the regular grid and the measurement locations $\mathbf{x}_p^0$. Finally, we obtain a first-guess for $\mathbf{c}^0$ by solving the linear system:

$$\mathbf{c}^0 = C_0^{-1}\hat{\mathbf{u}}^0, \tag{5.17}$$

As stated before, the procedure for the velocity and vorticity boundary conditions is the same. Starting with the velocity boundary condition at time $t_k$, it is defined as:

$$\hat{\mathbf{u}}_b^k = C_k\mathbf{c}_b^k. \tag{5.18}$$

The parameter for the velocity boundary condition is the weight vector $\mathbf{c}_b^k$. The covariance functions are placed on the nodes of the regular grid corresponding to the boundary. Finding a first guess for $\mathbf{c}_b^k$ can be achieved in the same way as was done for the initial velocity field. Finally, the vorticity boundary condition:

$$\hat{\boldsymbol{\omega}}_b^k = D_k\mathbf{g}_b^k. \tag{5.19}$$

The parameter for the vorticity boundary condition is the weight vector $\mathbf{g}_b^k$. The covariance functions are placed on the nodes of the regular grid corresponding to the boundary. The same procedure as followed for the velocity boundary condition is used to find a first guess for $\mathbf{g}_b^k$, except that we now take the curl of the interpolated velocity fields. Summarizing, the optimisation variables are $\mathbf{c}^0$, $\mathbf{c}_b^{\{1:K\}}$ and $\mathbf{g}_b^{\{1:K\}}$. With the constraint handling described in Section 5.4.1, the optimisation problem becomes:

$$\min_{\tilde{\mathbf{c}}^0,\tilde{\mathbf{c}}_b^{\{1:K\}},\tilde{\mathbf{g}}_b^{\{1:K\}}} J\left(\tilde{\mathbf{c}}^0,\tilde{\mathbf{c}}_b^{\{1:K\}},\tilde{\mathbf{g}}_b^{\{1:K\}}\right). \tag{5.20}$$

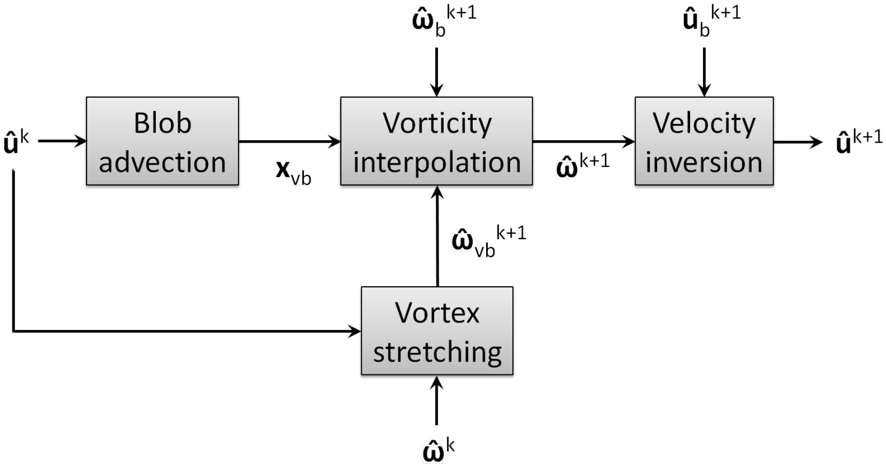


Figure 5.4: One time integration step of the VIC solver.

### 5.4.3. MODEL EVOLUTION

To calculate the time evolution of the velocity field, we use the vortex-in-cell (VIC) solver (Cottet and Koumoutsakos, 2000). We neglect viscous diffusion from the vorticity transport equation by removing the term $\nu\Delta\boldsymbol{\omega}$ in Eq.5.1, making the equation time-reversible. This will later allow efficient calculation of the cost function gradient by the adjoint method (see Section 5.5.1). Neglecting the viscous term is not a major limitation in case the time intervals we consider are small, no wall is present in the domain and the flow is convection dominated. In the context of volumetric velocity measurements, the VIC solver has been used before by Schneiders et al (2014) to *increase* the temporal resolution of tomographic PIV. For two experimental test cases namely a *turbulent* wake at a free stream velocity of 14 m/s and the circular jet from Appendix A.2.1 they show that the method is able to reconstruct detailed temporal dynamics using data subsampled at a rate far below the Nyquist frequency.

Figure 5.4 shows one time integration step of the VIC solver. We start out with a velocity field $\hat{\mathbf{u}}^k$ and its corresponding vorticity field $\hat{\boldsymbol{\omega}}^k$. As stated earlier, velocity and vorticity boundary conditions, $\hat{\mathbf{u}}_b^{k+1}$ and $\hat{\boldsymbol{\omega}}_b^{k+1}$ respectively, are also needed to get the velocity $\hat{\mathbf{u}}^{k+1}$ and vorticity $\hat{\boldsymbol{\omega}}^{k+1}$ fields at the new time step. The VIC solver works on a regular grid and defines vortex blobs at each node with a vorticity that corresponds to that node.

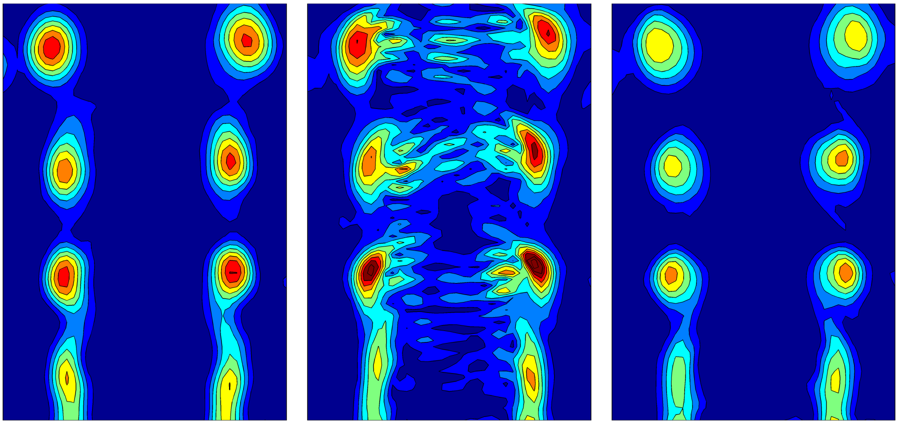
#### VORTEX BLOB ADVECTION

The first step illustrated in Figure 5.4 is vortex blob advection[1]. Each node or vortex blob on the regular grid is advected according to the current underlying velocity field:

$$\frac{d\mathbf{x}_{vb}}{dt} = \mathbf{u}(\mathbf{x}, t)\,.$$

---

[1] Normally, the term *particle* is used instead of *blob*. We use the term blob to avoid confusion with the tracer particles used in PTV.



(a) (b) (c)

Figure 5.5: Vorticity magnitude of the circular jet in water (a), simulated with a finite difference discretization of the vorticity transport equation (b) and with the VIC solver (c). Notice the growth of numerical instabilities in (b) due to the CFL-condition violation.

We use a first-order Euler scheme to approximate the above expression:

$$\mathbf{x}_{vb} = \mathbf{x}_g + \Delta t \cdot \hat{\mathbf{u}}^k, \tag{5.21}$$

where $\mathbf{x}_g$ are the coordinates of the nodes on the regular grid and $\mathbf{x}_{vb}$ are their locations after they have been advected along the velocity field $\hat{\mathbf{u}}^k$. $\Delta t$ is the time integration step. Treating the advection part of the vorticity transport equation by advecting the particles gives the VIC solver an important advantage in terms of stability. It is linearly unconditionally stable, while for nonlinear stability the requirement is that the particles do not collide (Cottet and Poncet, 2004). Therefore, traditional CFL-type stability conditions do not impose a limitation for the VIC solver, allowing larger time steps for the time integration. Figure 5.5 compares a time-integration of the VIC solver with a finite-difference discretization of the vorticity transport equation with the same time step. Figure 5.5(a) shows the final vorticity magnitude at the symmetry plane from the reference solution (see Appendix A.2.1). Both the finite-difference and VIC approaches have been integrated for 28 time steps. The CFL number reaches a value around 1.2 in the high velocity region. The finite-difference solver, given by Figure 5.5(b) has formed unphysical oscillations, while the VIC solver, given by Figure 5.5(c) is able to accurately reproduce the reference solution.

### VORTEX STRETCHING

In a two dimensional inviscid flow,

$$\frac{D\boldsymbol{\omega}}{Dt} = 0,$$

where $D\cdot/Dt$ is the material derivative. The initial vorticity field is simply advected with the underlying velocity field, meaning that the vorticity associated with each advected vortex blob stays constant. For a three dimensional flow however, this is not the case due to the term $(\boldsymbol{\omega}\cdot\nabla)\,\mathbf{u}$ in the vorticity transport equation, which represents vortex stretching. The new vorticity field is calculated as follows:

$$\frac{d\boldsymbol{\omega}}{dt} = \boldsymbol{\omega}\cdot\nabla\mathbf{u}.$$

Again we use a first-order Euler scheme for the above expression:

$$\hat{\boldsymbol{\omega}}_{vb}^{k+1} = \hat{\boldsymbol{\omega}}^{k} + \Delta t\cdot\hat{\boldsymbol{\omega}}^{k}\cdot\nabla\hat{\mathbf{u}}^{k}. \tag{5.22}$$

The new vorticity values correspond to the vortex blob locations at their new position.

### VORTICITY INTERPOLATION

The VIC solver requires that the new vorticity field is evaluated at the grid nodes of the regular grid to enable calculation of the new velocity field in the velocity inversion step. To this end, the vorticity field should be interpolated from the new vortex blob location $\hat{\boldsymbol{\omega}}_{vb}^{k+1}$ to the regular grid $\hat{\boldsymbol{\omega}}^{k+1}$:

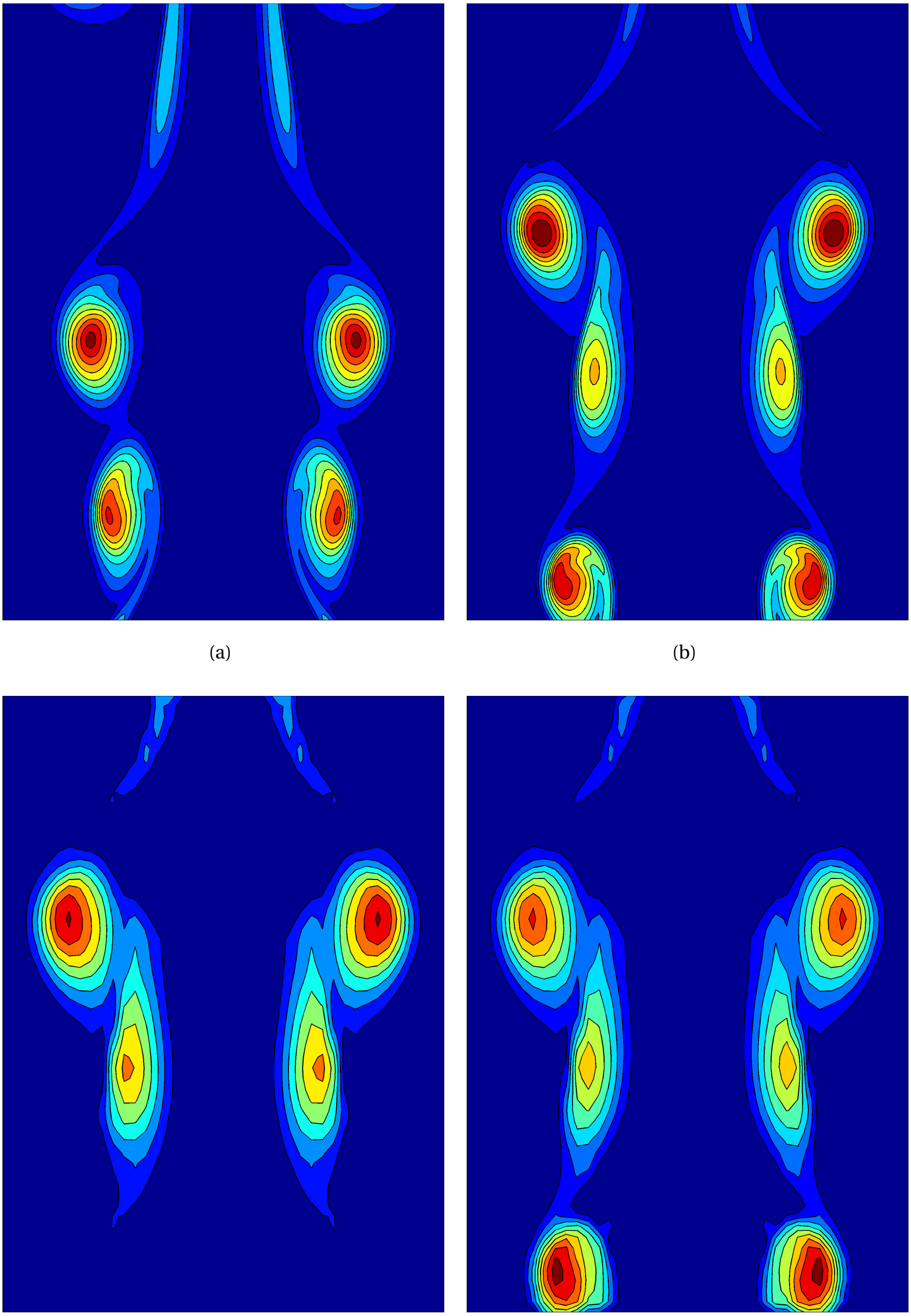
$$\hat{\boldsymbol{\omega}}_{j}^{k+1} = \frac{1}{V}\sum_{i=1}^{N_p}\hat{\boldsymbol{\omega}}_{vb,i}^{k+1}\prod_{l=1}^{3}\zeta_l\left(\frac{x_{l,g,j}-x_{l,vb,i}}{\Delta x_l}\right), \tag{5.23}$$

where $V$ is the volume of a cell in the regular grid (see Eq.5.7). The function $\zeta_l$ we use is known as the $M_4'$ kernel (Monaghan, 1985):

$$\zeta_l(\alpha) = \left\{\begin{array}{cc} 0 & |\alpha|\geq 2 \\ \frac{1}{2}(2-|\alpha|)^2(1-|\alpha|) & 1<|\alpha|<2 \\ 1-\frac{5}{2}\alpha^2+\frac{3}{2}|\alpha|^3 & |\alpha|\leq 1 \end{array}\right\}. \tag{5.24}$$

The advantage of this function is that it conserves total circulation and linear and angular impulse. Note from Figure 5.4 that next to the new vortex blob locations and their vorticity, the *Vorticity interpolation* block also takes vorticity boundary conditions as input. It is overlaid on the boundary after the interpolation.

Figure 5.6 aims to illustrate the importance of also imposing the vorticity boundary condition. Using the same geometric and flow parameters as the circular jet in water experiment discussed in Appendix A.2.1, we simulated the flow using the commercial package *Comsol*. It uses the Finite Element Method, with the Navier-Stokes equations in their standard form (see Eq.1.2), i.e. velocity and pressure. The results obtained are used as the reference solution. We then used the VIC solver and imposed the initial condition seen in Figure 5.6(a), which is a subdomain of what we simulated in Comsol. The final condition from Comsol can be seen in Figure 5.6(b). Figure 5.6(c) shows the final condition from the VIC solver if we only impose velocity boundary conditions. We clearly see that the new vortex ring from the bottom does not convect into the domain. Figure 5.6(d) shows the results if we impose both velocity and vorticity boundary conditions. Specifying the vorticity boundary condition is therefore important, especially if we integrate forwards over multiple time steps. Though this is in many ways not surprising, it is important to note considering that the velocity-pressure formulation is the most commonly used approach in computational fluid dynamics.



(a) (b)

(c) (d)

Figure 5.6: Simulated circular jet in water: True initial (a) and final (b) vorticity fields; final vorticity fields from imposing velocity boundary condition alone (c) and both velocity and vorticity boundary conditions (d). Flow direction: bottom to top.

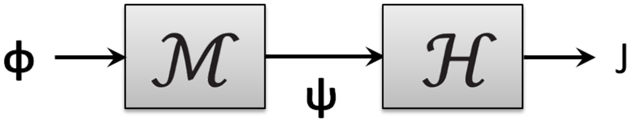


Figure 5.7: The original (nonlinear) forward code to obtain the cost function $J$ from the input state $\boldsymbol{\phi}$, via the intermediate $\boldsymbol{\psi}$.

### VELOCITY INVERSION

The final step is to obtain the new velocity field on the regular grid. Taking the curl on both sides of Eq.5.3, which simply states that the vorticity is the curl of the velocity, gives the following:

$$\nabla \times (\nabla \times \mathbf{u}) = \nabla \times \boldsymbol{\omega}. \tag{5.25}$$

The curl of the curl is:

$$\nabla \times (\nabla \times \mathbf{u}) = \nabla (\nabla \cdot \mathbf{u}) - \nabla^2 \mathbf{u},$$

and since the velocity field is divergence free, it reduces to:

$$\nabla \times (\nabla \times \mathbf{u}) = -\nabla^2 \mathbf{u}. \tag{5.26}$$

Combining Eqs.5.25 and 5.26 gives a Poisson equation which allows to calculate the new velocity field from the new vorticity field:

$$\nabla^2 \mathbf{u} = -\nabla \times \boldsymbol{\omega}. \tag{5.27}$$

Of course, velocity boundary conditions need to be supplied to solve the Poisson equation, as illustrated in Figure 5.4.

## 5.5. COST FUNCTION GRADIENT CALCULATION

To find the minimum of the cost function $J$, the optimisation algorithm L-BFGS requires the gradient $\nabla J$. The easiest method to calculate the gradient is finite differencing, since it treats the evaluation of the cost function as a black box and therefore does not require any specific further code development from the user. Figure 5.7 shows a simplified flow diagram to obtain the cost function $J \in \mathbb{R}$ from the input state $\boldsymbol{\phi} \in \mathbb{R}^n$. The intermediate state $\boldsymbol{\psi} \in \mathbb{R}^m$ is related to the input state through the nonlinear model operator $\mathscr{M}$:

$$\boldsymbol{\psi} = \mathscr{M}(\boldsymbol{\phi}). \tag{5.28}$$

The cost function $J$ is calculated from the intermediate state using the nonlinear cost function operator $\mathscr{H}$:

$$J = \mathscr{H}(\boldsymbol{\psi}). \tag{5.29}$$

With respect to a reference state $\boldsymbol{\phi}_0$, calculating the $i$-th component of the gradient with first-order finite differencing simply requires the evaluation:

$$\left.\frac{\partial J}{\partial \phi_i}\right|_{\boldsymbol{\phi}_0} \approx \frac{J\left(\mathcal{H}\left(\mathcal{M}\left(\boldsymbol{\phi}_0 + h\mathbf{e}_i\right)\right)\right) - J\left(\mathcal{H}\left(\mathcal{M}\left(\boldsymbol{\phi}_0\right)\right)\right)}{h}, \tag{5.30}$$

where $h$ is the user-specified step size and $\mathbf{e}_i$ is the standard unit basis vector with 1 on the $i$-th entry and 0 everywhere else. The full gradient is calculated by evaluating Eq.5.30 $n$ times. So, the code from Figure 5.7 needs to be evaluated a total of $n+1$ times. Despite its simplicity, finite differencing has two drawbacks:

(i) the calculated gradient is an *approximation*, dependent on the chosen step size $h$;

(ii) the cost of the gradient calculation scales with the size of the state, making it prohibitively *expensive* for our purposes.

. The first drawback can be negated by using an exact method. With this we do not mean symbolic differentiation of the code shown in Figure 5.7, which would lead to inefficient code and difficulties in converting the computer program into a single expression. Instead, we mean *automatic differentiation* (AD). It is based on two important concepts:

5

(i) every computer program, irrespective of its complexity, can be reduced to a sequence of elementary arithmetic operations and functions;

(ii) the chain rule.

So, for the flow diagram in Figure 5.7, we can write the $i$-th component of the gradient as:

$$\left.\frac{\partial J}{\partial \phi_i}\right|_{\boldsymbol{\phi}_0} = \sum_{j=1}^{m} \left.\frac{\partial J}{\partial \psi_j}\right|_{\boldsymbol{\phi}_0} \left.\frac{\partial \psi_j}{\partial \phi_i}\right|_{\boldsymbol{\phi}_0}.$$

Practically, there are two approaches to obtain the gradient using automatic differentiation: the forward mode and the reverse mode. Both approaches require access to and modification of the original code. The resulting codes are better known as the tangent-linear and adjoint code for the forward and reverse modes, respectively. There are algorithms available that can automatically generate these codes from the original code[2], though they will not necessarily generate the most efficient code. In the present work however, we have generated these codes manually for optimal efficiency.

### 5.5.1. THE TANGENT-LINEAR CODE (FORWARD MODE)

Figure 5.8 shows the tangent-linear code of the original nonlinear code from Figure 5.7. The linearisation is with respect to the reference state $\boldsymbol{\phi}_0$. The figure shows how a disturbance in the input state, $\delta\boldsymbol{\phi}$, propagates into a disturbance in the cost function $\delta J$. Mathematically,

$$\delta\boldsymbol{\psi} = M\delta\boldsymbol{\phi}, \tag{5.31}$$

---

[2] `http://www.autodiff.org/?module=Tools`

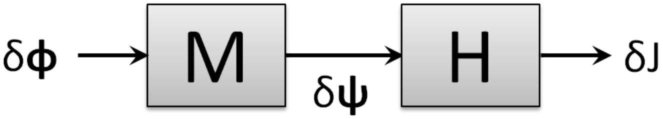


Figure 5.8: The tangent-linear code of the original nonlinear code from Figure 5.7.

where $M \in \mathbb{R}^{m\times n}$ is the matrix of the linearized model operator $\mathscr{M}$. Its row $j$ and column $i$ are defined as:

$$M_{ji} = \left.\frac{\partial \psi_j}{\partial \phi_i}\right|_{\boldsymbol{\phi}_0}.$$

The second block represents the operation:

$$\delta J = H\delta\boldsymbol{\psi}. \tag{5.32}$$

$H \in \mathbb{R}^{1\times m}$ is the linearisation of the operator $\mathscr{H}$. Its element $j$ is defined as:

$$H_j = \left.\frac{\partial J}{\partial \psi_j}\right|_{\boldsymbol{\phi}_0}.$$

The full propagation from the input disturbance to the cost function disturbance can be written as follows:

$$\delta J = \sum_{i=1}^{n}\sum_{j=1}^{m} \left.\frac{\partial J}{\partial \psi_j}\right|_{\boldsymbol{\phi}_0} \left.\frac{\partial \psi_j}{\partial \phi_i}\right|_{\boldsymbol{\phi}_0} \delta\phi_i. \tag{5.33}$$

If we specify the input disturbance $\mathbf{e}_1$, the tangent-linear code returns:

$$\delta J = \sum_{j=1}^{m} \left.\frac{\partial J}{\partial \psi_j}\right|_{\boldsymbol{\phi}_0} \left.\frac{\partial \psi_j}{\partial \phi_1}\right|_{\boldsymbol{\phi}_0}.$$

This is the first element of the cost function gradient $\nabla J$. So, to obtain all elements of $\nabla J$, we need to go through the tangent-linear code $n$ times, supplying the inputs $\mathbf{e}_i$, $i = 1,\dots,n$. Therefore, the only advantage of this approach compared to finite differencing is that it is exact. The reverse mode, explained in Section 5.5.2 is more efficient. Before proceeding though, we explain how to setup the tangent-linear code for the present cost function given by Eq.5.8, since we will need the tangent-linear code to generate the adjoint code.

To explain the tangent-linear code of the VIC solver, we refer to Figure 5.4, where each variable $(\cdot)$ should be replaced by its disturbed version $(\delta\cdot)$.

### VORTEX BLOB ADVECTION

Recall that we used a first-order Euler scheme for vortex blob advection (see Eq.5.21). The tangent-linear version is simple, since the original code is already linear:

$$\delta\mathbf{x}_{vb} = \Delta t \cdot \delta\hat{\mathbf{u}}^k, \tag{5.34}$$

### VORTEX STRETCHING

The tangent-linear version of the vortex stretching block (see Eq.5.22) is more involved due to the nonlinear term $\hat{\boldsymbol{\omega}}^k \cdot \nabla \hat{\mathbf{u}}^k$:

$$\delta \hat{\boldsymbol{\omega}}_{vb}^{k+1} = \delta \hat{\boldsymbol{\omega}}^k + \Delta t \cdot \hat{\boldsymbol{\omega}}^k \cdot \nabla \delta \hat{\mathbf{u}}^k + \Delta t \cdot \delta \hat{\boldsymbol{\omega}}^k \cdot \nabla \hat{\mathbf{u}}^k. \tag{5.35}$$

Note that contrary to the tangent-linear of the vortex blob advection block, which did not require the current state, the vortex stretching does require it. Therefore, the original nonlinear code needs to be run next to the tangent-linear code to supply the intermediate states.

### VORTICITY INTERPOLATION

The vorticity interpolation block is also nonlinear (see Eq.5.23) due to multiplication of the vorticity with the $M_4'$ kernel, which contains the advected vortex blob position. The tangent-linear version is:

$$\delta \hat{\boldsymbol{\omega}}_j^{k+1} = \frac{1}{V} \sum_{i=1}^{N_p} \left( \delta \hat{\boldsymbol{\omega}}_{vb,i}^{k+1} \prod_{l=1}^{3} \zeta_l \left( \frac{x_{l,g,j} - x_{l,vb,i}}{\Delta x_l} \right) + \hat{\boldsymbol{\omega}}_{vb,i}^{k+1} Z_{ji} \right), \tag{5.36}$$

where:

$$Z_{ji} = \delta\zeta_{1,ji}\zeta_{2,ji}\zeta_{3,ji} + \zeta_{1,ji}\delta\zeta_{2,ji}\zeta_{3,ji} + \zeta_{1,ji}\zeta_{2,ji}\delta\zeta_{3,ji}. \tag{5.37}$$

The element $\zeta_{l,ji}$ is a shorthand for

$$\zeta_{l,ji} = \zeta_l \left( \alpha_{l,ji} \right) = \zeta_l \left( \frac{x_{l,g,j} - x_{l,vb,i}}{\Delta x_l} \right).$$

With regards to the perturbation $\delta\zeta_{l,ji}$, we have:

$$\delta\zeta_{l,ji} = \frac{\partial \zeta_l}{\partial \alpha} \frac{\partial \alpha}{\partial x_{l,vb,i}} \delta x_{l,vb,i}, \tag{5.38}$$

where

$$\frac{\partial \zeta_l}{\partial \alpha} = \left\{ \begin{array}{cc} 0 & |\alpha| \geq 2 \\ -\operatorname{sgn}\alpha\,(2-|\alpha|)\left(1-|\alpha|+\frac{1}{2}(2-|\alpha|)^2\right) & 1 < |\alpha| < 2 \\ -5\alpha + \operatorname{sgn}\alpha\,\frac{9}{2}|\alpha|^2 & |\alpha| \leq 1 \end{array} \right\}, \tag{5.39}$$

and

$$\frac{\partial \alpha}{\partial x_{l,vb,i}} = -\frac{1}{\Delta x_l}.$$

### VELOCITY INVERSION

Finally, the tangent-linear block of velocity inversion is again simple, since the Laplace and curl operators (see Eq.5.27) are linear. Therefore,

$$\nabla^2 \delta \mathbf{u} = -\nabla \times \delta \boldsymbol{\omega}. \tag{5.40}$$

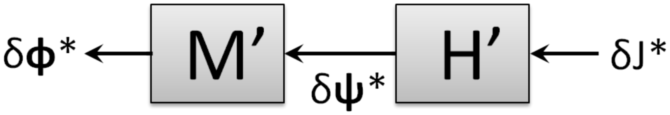


Figure 5.9: The adjoint code of the tangent-linear code from Figure 5.8.

### 5.5.2. THE ADJOINT CODE (REVERSE MODE)

As the name suggests, in the reverse mode of differentiation, we traverse the code from output to input (see Figure 5.9). The corresponding *adjoint* code can be derived from the tangent-linear code by converting the matrices by their transpose or adjoint. Note that all variables have been replaced by their adjoint counterpart by denoting them with (*). The operation from Eq.5.32 now becomes:

$$\delta\boldsymbol{\psi}^* = H'\delta J^*, \tag{5.41}$$

and the operation from Eq.5.31 becomes:

$$\delta\boldsymbol{\phi}^* = M'\delta\boldsymbol{\psi}^*, \tag{5.42}$$

So combining Eqs.5.41 and 5.42 we get:

$$\delta\boldsymbol{\phi}^* = M'H'\delta J^*. \tag{5.43}$$

Note that Eq.5.33 can also be written as:

$$\delta J = \nabla J'\delta\boldsymbol{\phi} = HM\delta\boldsymbol{\phi}. \tag{5.44}$$

Therefore, the gradient we are interested in is simply:

$$\nabla J = M'H'. \tag{5.45}$$

Comparing Eqs.5.43 and 5.45, we clearly see that if we set $\delta J^* = 1$ and go through the adjoint code once, we get the full gradient of the cost function. So, comparing the forward and reverse modes of differentiation we see that even though they both give the exact gradient, the reverse mode can achieve this with one model evaluation rather than $n$. For that reason we use the reverse mode in the present work.

It is important to note that the adjoint code comes with a disadvantage that the complete nonlinear trajectory needs to be stored, since we traverse the tangent-linear code from output to input. So, the benefit of reduced computational time of the adjoint method comes at the price of increased storage requirements. A number of methods have been proposed to deal with this, the most prominent being checkpointing (Griewank and Walther, 2000). The idea is simple and illustrated by Figure 5.10. Instead of storing the complete nonlinear trajectory from $t_0$ to $t_N$, we only store the states

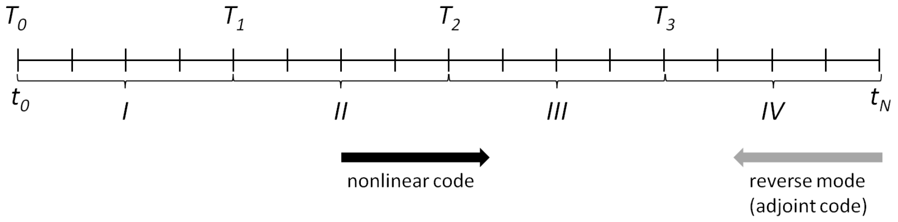


Figure 5.10: Checkpointing.

at $T_i$, $i = 0, ..., 3$. To start the adjoint code, we integrate the nonlinear code forward from $T_3$ and save all the intermediate states. Then we integrate the adjoint code backward to $T_3$ and afterwards delete all the states in interval $IV$. We then integrate the nonlinear code forward from $T_2$ and store all the states in the interval $III$, integrate the adjoint backwards from $T_3$ to $T_2$ and subsequently delete the states in $III$. This procedure is continued until we reach $t_0$, where we eventually get the full gradient. For the present problem though, where we have neglected viscosity, the vorticity transport equation is time-reversible. Therefore, we do not need to store the full nonlinear trajectory. Rather, we start with the final state and integrate it backwards in time by changing the sign of the velocity. In this way, the nonlinear trajectory is available for the adjoint code at each time instant.

Creating the adjoint code is simple but tedious. As explained earlier, one simply needs to transpose the matrices from the tangent-linear code (Giering and Kaminski, 1998). However, explicitly creating a matrix and then taking the transpose will lead to inefficient code, even though it is the simplest way to create the adjoint code. In addition, we have not explicitly created matrices for the operations in the tangent-linear code. Appendix C explains in detail how we created the adjoint code, in which we have avoided this simple approach.

## 5.6. APPLICATION TO EXPERIMENTAL DATA

In this section we apply the spatio-temporal interpolation method to the circular jet in water from Appendix A.2.1. We compare the method with SGPR, which does not include the temporal dimension and only enforces mass conservation. Figure 5.11(a) shows 4,868 particle tracks from 15 consecutive snapshots. The same particle tracking algorithm as applied in Chapter 3 was used (Schneiders et al, 2015a). The tracks have been colour-coded with velocity magnitude and one can clearly identify the jet issuing from the middle of the domain and the vortex rings forming at the shear layer. Figure 5.11(b) shows the *sparse* data set we consider, which only contains 487 particle tracks from 15 consecutive snapshots. SGPR interpolates the velocity field from each snapshot. The resulting fields are taken as the initial guess by the spatio-temporal method, as required by the optimisation algorithm. As explained earlier, the elements of the state vector are the velocity field at the first snapshot, and the velocity and vorticity boundary conditions at the subsequent snapshots. Figures 5.12(a) and 5.12(d) compare the first and final snapshots, respectively in terms of the axial velocity and iso-surfaces of the vorticity magni-

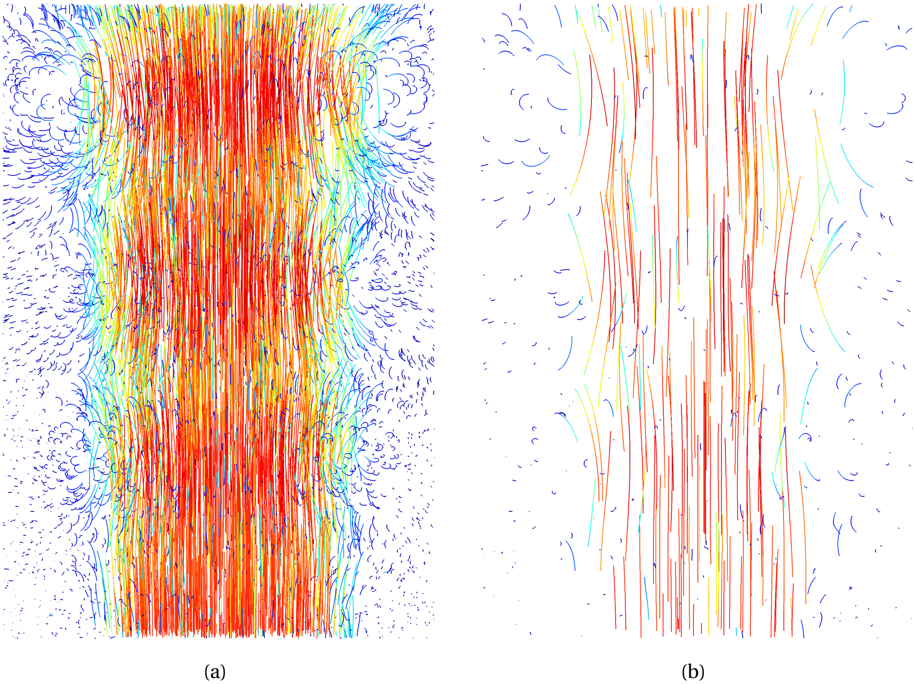
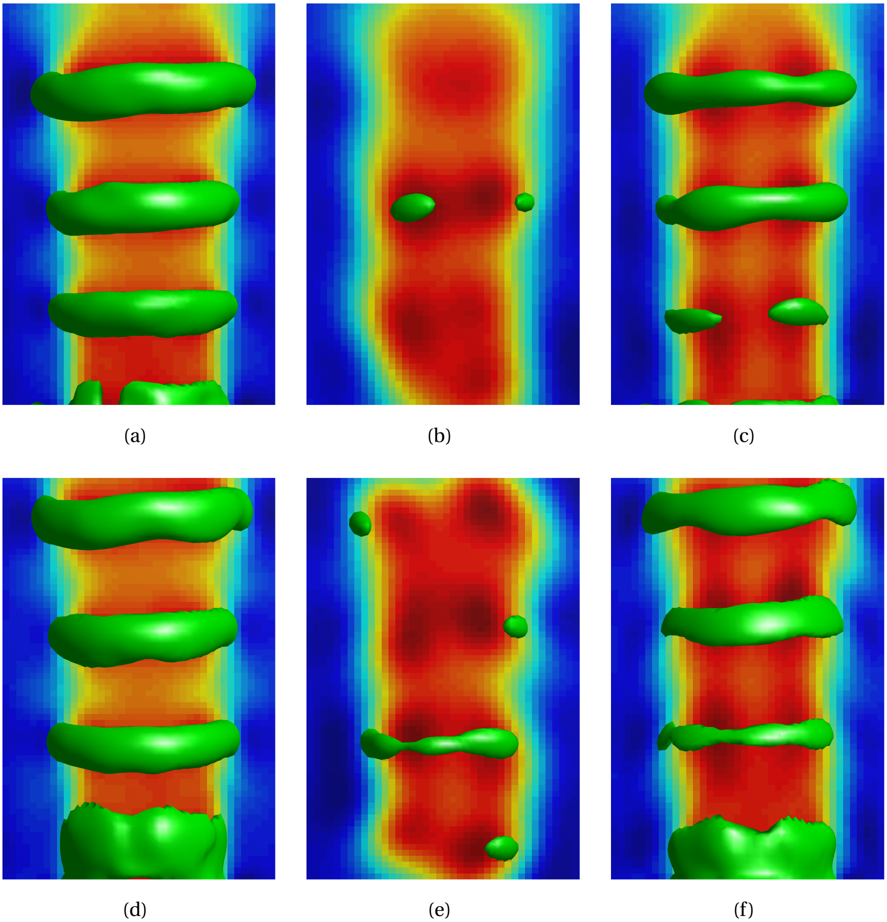


Figure 5.11: Circular jet in water: 4,868 (a) and 487 (b) particle tracks from 15 consecutive snapshots. Colour-coded with velocity magnitude.

tude, reconstructed with SGPR using all 4,868 particle tracks. Figures 5.12(b) and 5.12(e) show the same results, but with the reduced data set of 487 particle tracks. Finally, Figures 5.12(c) and 5.12(f) show the results achieved with the spatio-temporal method. The improvement achieved with the spatio-temporal method is evident. For completeness, Figure 5.13 shows a time history of the vorticity magnitude, evaluated at a point in the shear layer. The spatio-temporal method is indeed closer to the reference solution. In addition, notice the significant improvement in the temporal coherence. Of course, this is to be expected since the time history is obtained by integrating the VIC solver from the guessed initial field.

## 5.7. CONCLUSION

We investigated how incorporating the temporal information of particle tracks in 3D-PTV can improve the reconstructed velocity and vorticity field. To incorporate this information, we have used the momentum conservation equation formulated in the vorticity-velocity form via the vorticity transport equation, in addition to mass conservation. We modeled this using the vortex-in-cell (VIC) solver. To combine the physical knowledge from the model and the measured tracks we used a variational method. Starting from the Bayesian framework, we formulated all the assumptions that eventually lead to the algorithm used in the present chapter. The result was an optimisation problem, which we solved using a quasi-Newton method. To calculate gradients efficiently, we exploited the reverse mode of differentiation, which we hand coded. The assimilation algorithm was applied to an experimental test case of a circular jet in water. We compared it with solenoidal interpolation, which interpolates the snapshots independent from one another, therefore not taking the temporal information into account. We found clear improvements in the reconstructed field when incorporating temporal information.



(a) (b) (c)

(d) (e) (f)

Figure 5.12: Circular jet in water: first (top) and final (bottom) fields obtained with the full data set (a,d), the sparse data set and interpolated with SGPR (b,e) and with the spatio-temporal method (c,f).

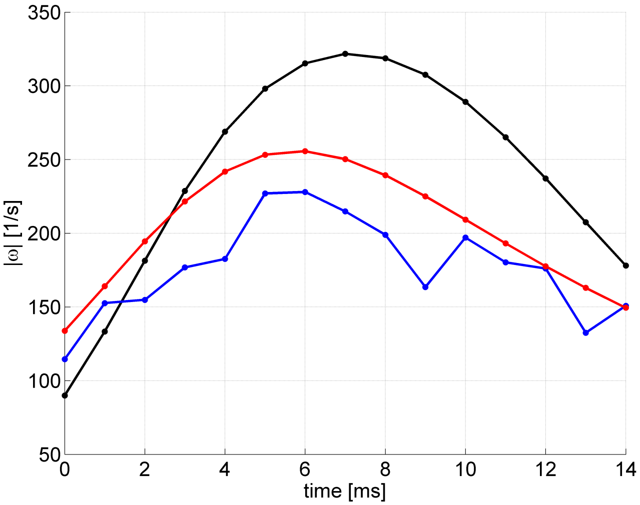


Figure 5.13: Circular jet in water: time history of the vorticity magnitude at a point in the shear layer from the reference solution using the full data set (black), SGPR (blue) and the spatio-temporal method (red).

# 6

# *A posteriori* UNCERTAINTY QUANTIFICATION OF PIV-BASED PRESSURE DATA

*This chapter proposes a methodology for a posteriori quantification of the uncertainty of pressure data retrieved from PIV. It relies upon the Bayesian framework introduced in Chapter 2. Section 6.2 introduces the proposed methodology. Section 6.3 presents a numerical assessment with an analytical test case, where the proposed method is compared with Monte-Carlo simulations and linear uncertainty propagation. We find excellent agreement with Monte-Carlo simulations, while linear uncertainty propagation underestimates the uncertainty in the pressure by up to* 30%. *In Section 6.4, the method is applied to an experimental test case of a turbulent boundary layer in air, obtained using time-resolved tomographic PIV. The pressure reconstructed from the tomographic PIV data is compared to a microphone measurement conducted simultaneously at the wall to determine the true error of the former. The comparison between true error and estimated uncertainty demonstrates the accuracy of the uncertainty estimates on the pressure. In addition, enforcing the divergence-free constraint is found to result in a significantly more accurate reconstructed pressure field. The estimated uncertainty confirms this result. Conclusions are drawn in Section 6.5.*

## 6.1. INTRODUCTION

PIV is nowadays recognized as a reliable tool for the determination of the instantaneous static pressure in two or three dimensional flow fields (van Oudheusden, 2013). The main advantage of the PIV-based pressure reconstruction with respect to conventional pressure probes (e.g. pressure transducers or microphones) is the instantaneous 2D or 3D pressure *field* determination, without the need of an expensive design and manufacturing of models where densely distributed sensors are installed (Ragni et al, 2011).

---

Parts of this chapter have been published in Experiments in Fluids (Azijli et al, 2016).

PIV-based pressure measurements are made possible for applications where probes cannot be installed, such as micro aerial vehicles. Additionally, they convey a quantitative visualization of the flow structures responsible for the aerodynamic forces acting on the model. Industrial tests typically require pressure information, for the evaluation of forces. Pressure fluctuations are important when dealing with fatigue studies and aeroacoustic source measurements (Haigermoser, 2009; Pröbsting et al, 2013).

The computation of the pressure field relies upon PIV measurement of the velocity and acceleration and is conventionally performed by spatial integration of the pressure gradient or by solution of the Poisson equation, provided appropriate boundary conditions. The formulation of the problem in terms of the Poisson equation is often employed because it avoids accumulation of errors in the reconstructed pressure (van Oudheusden, 2013). Applying the divergence operator to the momentum equation and assuming incompressible flow, the following Poisson equation for pressure follows:

$$\nabla^2 p = -\rho \nabla \cdot (\mathbf{u} \cdot \nabla)\, \mathbf{u}. \tag{6.1}$$

To solve Eq.6.1, appropriate boundary conditions on the pressure (Dirichlet) or its spatial gradient (Neumann) must be specified. It is important to remark that, although no temporal derivative appears in Eq.3.4, its boundary conditions are time dependent and require the computation of the flow acceleration at the boundary of the computational domain. In practice, the discretization of Eq.3.4 results in an algebraic system of equations:

$$A\mathbf{p} = \mathbf{f}, \tag{6.2}$$

which is then solved. $A$ is a discrete representation of the Laplace operator, obtained for example by finite differences and **f** will be referred to as the *source term*.

When solving Eq.6.2 to determine the static pressure field, several factors contribute to the overall uncertainty. According to van Oudheusden (2013), the most relevant for the accuracy of the reconstructed pressure field are:

- the accuracy of the underlying velocity data;
- the spatial and temporal resolutions, which affect the accuracy of velocity derivatives;
- the approach used to evaluate the velocity material acceleration (Eulerian or Lagrangian approach);
- the boundary conditions;
- the pressure-gradient integration procedure;
- for PIV-based pressure reconstruction of planar PIV data, application of a 2D model to a 3D flow.

As a result, the uncertainty of the static pressure is a complex function of the parameters above. Quantifying the uncertainty of the pressure measurement is of primary importance for the determination of a confidence interval where the true pressure value lies.

Furthermore, it constitutes the basis for estimating the uncertainty of the aerodynamic forces (e.g. lift and drag) acting on a model.

Several works have focused on evaluating the contribution of the above mentioned factors to the error of the reconstructed pressure. For example van Oudheusden (2013) derived an analytical expression of the error of the material acceleration for both Eulerian and Lagrangian approaches. In both approaches, the laser pulse-separation $\Delta t$ has opposite effects on truncation and random error. Increasing $\Delta t$ increases truncation error, but decreases the relative random error. Furthermore, the error of the material acceleration is typically smaller when it is calculated with the Lagrangian approach. Violato et al (2011) introduced a criterion for the selection of the minimum and maximum $\Delta t$ that accounts for the maximum allowed error on the material acceleration and the effect of the out-of-plane displacement of the particles. The authors found that the Lagrangian approach features a random error about 1.5 times smaller than that estimated for the Eulerian approach. The difference between the two approaches is less pronounced when looking at the resulting pressure field, due to the error suppression during the pressure gradient integration. Several schemes for the integration of the pressure gradient have been proposed, including the space-marching integration (Baur and Köngeter, 1999), the omni-directional integration (Liu and Katz, 2006) and the Poisson equation for pressure (Gurka et al, 1999). In de Kat and van Oudheusden (2010) a comparative assessment of pressure integration methods is conducted. Although the upshot was that the integration method has minor impact, and only the space-marching approach is clearly inferior, their conclusions were based on statistical analysis and not on the instantaneous pressure uncertainty. In fact, no quantitative information about the uncertainty of computed pressure fields was provided in these early studies. Charonko et al (2010) reported that the error of the velocity field has a major influence on the accuracy of the reconstructed pressure. In their work, the pressure result became 'unusable' as soon as very small levels of random error were introduced. Extensive data post-processing was required to reduce the velocity error and make accurate pressure reconstructions feasible. However, Murai et al (2007) reported that a fairly simple smoothing of the velocity data is sufficient for accurate pressure results. Charonko et al (2010) also investigated the effect of off-axis measurement and out-of-plane velocity. They found that schemes performing well in 2D conditions continued to do so for misalignments up to 30 degrees.

In turbulent flows, PIV-based pressure reconstruction presents two major challenges: the small magnitude of the pressure fluctuations and the large dynamic range. Ghaemi et al (2012) conducted time-resolved PIV (TR-PIV) measurements to investigate the pressure fluctuations in a turbulent boundary layer at $Re_\theta = 2400$. The agreement between the PIV-based pressure and that measured by a surface microphone was evaluated via the cross-correlation coefficient, which was equal to 0.6. The probability density functions of the two signals agreed up to 3 kHz for tomographic PIV data and up to 1 kHz for planar data. Similarly Pröbsting et al (2013) carried out the PIV-based pressure reconstruction in a turbulent boundary layer at $Re_\theta = 730$. The data were validated by comparison with a surface microphone and with DNS data. Also in this case, the PIV-based pressure signal showed fair agreement with the microphone signal, but the cross-correlation coefficient between the two did not exceed 0.6. Despite the relatively high correlation coefficient achieved, those studies opened several doubts on the accuracy of the PIV-based pressure reconstruction. Acoustic emissions of the flow over a rectangular cavity were investigated by Koschatzky et al (2011). The fluctuating pressure fields computed from TR-PIV were used as input for the prediction of the sound emission. The authors found that the hydrodynamic pressure computed from the PIV data had a frequency spectrum comparable to that of the direct pressure measurement at the cavity walls. The main frequency tone and the first harmonic were correctly captured, although the amplitude was significantly underestimated. At locations where the pressure fluctuations were weak, the PIV measurement uncertainty strongly affected the accuracy of the reconstructed pressure.

From the discussion above, it emerges that PIV-based pressure is often highly inaccurate due to the many error sources that contribute to the total error. As a result, quantifying the uncertainty of the reconstructed pressure is of primary importance to the future use of these methods. Furthermore, *a posteriori* uncertainty quantification could be used to inform the choice of experimental and postprocessing parameters, along with the pressure reconstruction approach, to maximize the precision of the pressure field. The present chapter discusses the mathematical framework for the uncertainty propagation from the velocity field to the pressure field. It uses the Bayesian framework, explained in Chapter 2. The measurement uncertainty of the velocity field from PIV can therefore be naturally combined with prior knowledge of the velocity field (e.g. divergence-free) (Wikle and Berliner, 2007). A number of *a posteriori* approaches have recently been proposed to quantify the PIV measurement uncertainty. We refer to Sciacchitano et al (2015) for a comparison of these methods. The present framework can also propagate the uncertainty of the velocity spatial- and temporal-derivatives, when information on the latter is available. The uncertainty due to the application of a 2D model to a 3D flow – i.e. pressure from planar PIV – is not addressed here.

## 6.2. METHODOLOGY

The method to propagate the uncertainty from the velocity field to the pressure field is split into three successive steps and explained in their respective sections:

- characterize the uncertainty in the velocity field $\mathbf{u}$ (Section 6.2.1);
- propagate the uncertainty from the velocity field to the source term $\mathbf{f}$ (Section 6.2.2);
- propagate the uncertainty from the source term to the pressure field $\mathbf{p}$ (Section 6.2.3).

Section 6.2.4 illustrates the complete methodology for a simple one dimensional example. The uncertainty of a variable will be expressed through a probability density function (pdf) $\rho\left(\cdot\right)$.

### 6.2.1. THE VELOCITY FIELD UNCERTAINTY

To characterize the uncertainty in the velocity field we follow the approach described in Chapter 2, where we define a statistical model for the prior $\rho_0\left(\mathbf{u}\right)$, the PIV measurement $\rho\left(\mathbf{u}_{\mathrm{PIV}}|\mathbf{u}\right)$ and obtain the posterior $\rho\left(\mathbf{u}|\mathbf{u}_{\mathrm{PIV}}\right)$ using Bayes' rule.

### THE PRIOR ON THE VELOCITY FIELD

We discretize the velocity field at $n$ spatial locations of interest $\mathbf{x} \in \mathbb{R}^{d \cdot n}$. This gives a vector $\mathbf{u} \in \mathbb{R}^{d \cdot n}$, describing the $d$ dimensional velocity components at these locations. Due to the assumption of Gaussian processes and the discretization, we have the following multivariate normal distribution:

$$\mathbf{u} \sim \mathcal{N}\left(\boldsymbol{\mu}_0, P\right),$$

with $\boldsymbol{\mu}_0$ and $P$ the prior mean and prior covariance matrix, respectively. We start the discussion with the prior mean $\boldsymbol{\mu}_0 \in \mathbb{R}^{d \cdot n}$. For simplicity, we assume each velocity component to be a constant field. This is not a limiting simplification since the complexity of the field that can be reconstructed is contained in the prior covariance used. The prior covariance matrix is defined as:

$$P^{ij} = \tau^2 \phi\left(r^{ij}\right), \tag{6.3}$$

where $r^{ij}$ represents the (scaled) distance between $\mathbf{x}^i$ and $\mathbf{x}^j$:

$$\left(r^{ij}\right)^2 = \sum_{k=1}^{d} \left(\frac{x_k^i - x_k^j}{\theta_k}\right)^2, \tag{6.4}$$

where $\theta_k$ is the correlation length in the direction $k$. $\phi\left(r^{ij}\right)$ is a covariance function and represents the correlation between $\mathbf{x}^i$ and $\mathbf{x}^j$. It is a function of their separation distance due to the assumption of a *stationary* process. We use the Wendland function of order 2 (see Eq.B.5).

Ultimately, one can either setup a prior covariance matrix for each velocity component independently (de Baar et al, 2014) or a single covariance matrix that constrains the velocity components according to mass conservation. The latter approach was introduced in Chapter 3. Whichever method is used, two parameters have resulted that need to be determined, namely the prior variance $\tau^2$ from Eq.6.3 and the correlation length $\theta_k$ from Eq.6.4. They can either be specified *a priori* or through a maximum likelihood optimization. In Chapter 3, we followed the *a priori* approach. This was possible because the PIV data had sufficient spatial resolution, which is the same situation in the present chapter.

### THE PIV MEASUREMENT UNCERTAINTY MODEL

We define the vector $\mathbf{u}_{\text{PIV}} \in \mathbb{R}^{d \cdot m}$, since there are $m$ measurements. When defining the prior, we could have either taken $n$ equal to or larger than $m$. If equal to $m$, we will obtain the pressure field on the same mesh as the PIV field. If larger than $m$, we attempt to obtain the pressure field on a finer mesh. The measurement uncertainty is also assumed to come from a Gaussian distribution, where the mean represents the bias error and the standard deviation represents the random error (Sciacchitano et al, 2015). Once quantified, the bias error can be subtracted from the measurements. Therefore, the measurements

$$\mathbf{u}_{\text{PIV}} = H\mathbf{u} + \boldsymbol{\epsilon} \in \mathbb{R}^{d \cdot m}$$

can be represented as a multivariate Gaussian distribution, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, R)$ is the measurement error of the velocity field, assumed to be Gaussian noise with zero mean and covariance matrix $R$. The matrix $H$ maps the velocities from the $n$ locations of interest to the $m$ measurement locations. In the most general case, $H$ would represent an arbitrary interpolation matrix (representing for example polynomial or radial basis function interpolation). In the present work, the measurement locations are taken to be a subset of the locations of interest, therefore $H$ is a Boolean matrix. In particular, if $m = n$ then $H$ is the identity matrix, giving

$$\mathbf{u}_{\mathrm{PIV}} = \mathbf{u} + \boldsymbol{\epsilon}.$$

This situation where the pressure field is evaluated at the same locations where the velocity measurements are available is the most common (van Oudheusden, 2013).

### THE POSTERIOR ON THE VELOCITY FIELD

Having defined the prior and carried out the measurements, they are combined to obtain the posterior distribution. The prior and likelihood are both Gaussian distributed, and since all operators defined are linear, the posterior is Gaussian distributed as well. Its mean and covariance were given in Chapter 2, but we repeat them for clarity:

$$\mathbb{E}\left(\mathbf{u}|\mathbf{u}_{\mathrm{PIV}}\right) = \boldsymbol{\mu} + PH^T\left(R + HPH^T\right)^{-1}\left(\mathbf{u}_{\mathrm{PIV}} - H\boldsymbol{\mu}\right), \tag{6.5}$$

$$\Sigma\left(\mathbf{u}|\mathbf{u}_{\mathrm{PIV}}\right) = P - PH^T\left(R + HPH^T\right)^{-1}HP, \tag{6.6}$$

respectively. In the present chapter, we will also consider the situation where we construct the pressure field by completely relying on the PIV data, i.e. no prior knowledge is taken into account and the velocity is only assumed to be known at the locations coming from the measurements. Mathematically, this can be expressed by taking $P = \tau^2 I$, where $\tau = \infty$, and $H = I$. Eq.6.5 becomes

$$\mathbb{E}\left(\mathbf{u}|\mathbf{u}_{\mathrm{PIV}}\right) = \boldsymbol{\mu} + \tau^2 I\left(R + \tau^2 I\right)^{-1}\left(\mathbf{u}_{\mathrm{PIV}} - \boldsymbol{\mu}\right). \tag{6.7}$$

The term $\tau^2 I\left(R + \tau^2 I\right)^{-1}$ becomes $I$ for infinitely large prior variance, so Eq.6.7 reduces to

$$\mathbb{E}\left(\mathbf{u}|\mathbf{u}_{\mathrm{PIV}}\right) = \mathbf{u}_{\mathrm{PIV}}. \tag{6.8}$$

The posterior covariance from Eq.6.6 becomes

$$\Sigma\left(\mathbf{u}|\mathbf{u}_{\mathrm{PIV}}\right) = \tau^2 I - \tau^2 I\left(R + \tau^2 I\right)^{-1}\tau^2 I. \tag{6.9}$$

With the prior variance approaching infinity, Eq.6.9 can be rewritten to reveal

$$\Sigma\left(\mathbf{u}|\mathbf{u}_{\mathrm{PIV}}\right) = R. \tag{6.10}$$

In words, Eqs.6.8 and 6.10 state that the posterior mean and covariance are equal to the measured velocity field and measurement uncertainty, respectively. This makes sense intuitively, but it is important that it also follows naturally from the framework.

### 6.2.2. PROPAGATING THROUGH THE SOURCE TERM

The posterior distribution of the velocity (see Eqs.6.5 and 6.6) needs to be propagated through $\mathbf{f}$ to obtain the mean $\mathbb{E}(\mathbf{f})$ and covariance $\Sigma(\mathbf{f})$ of the source term. However, this is not straightforward since $\mathbf{f}$, derived from the Navier-Stokes equations, has a nonlinear dependence in the velocity. We consider two options to obtain the mean and covariance of the source term.

The first is through carrying out Monte-Carlo simulations (Metropolis and Ulam, 1949), a popular method for uncertainty propagation. Briefly, random realizations of the posterior distribution $(\mathbf{u}|\mathbf{u}_{\mathrm{PIV}})$ are generated. In practice, a random realization with the required covariance structure specified by $\Sigma(\mathbf{u}|\mathbf{u}_{\mathrm{PIV}})$ can be obtained by computing the Cholesky decomposition

$$\Sigma(\mathbf{u}|\mathbf{u}_{\mathrm{PIV}}) = LL^T,$$

followed by multiplying $L$ with a vector containing uncorrelated random realizations from a standard Gaussian distribution (Gibbs, 2011). Subsequently, for each realization the source term $\mathbf{f}$ is then evaluated. From this one can construct $\mathbb{E}(\mathbf{f})$ and $\Sigma(\mathbf{f})$.

The second method enables exact uncertainty propagation as opposed to the Monte-Carlo method. A closer look at the Navier-Stokes equations reveals that the nonlinear term contains a product of two variables. As a result, it is possible to calculate the expected value and covariance exactly. For generality, we define four random variables $a$, $b$, $c$ and $d$. For the expected value, one can use the definition of covariance:

$$\mathbb{E}(ab) = \mathbb{E}(a)\mathbb{E}(b) + \sigma(a,b), \tag{6.11}$$

where $\sigma(a,b)$ represents the covariance between $a$ and $b$. For the covariance between two products of random variables $ab$ and $cd$, Bohrnstedt and Goldberger (1969) derived the following expression:

$$\begin{aligned}\sigma(ab,cd) =& \mathbb{E}(a)\mathbb{E}(c)\sigma(b,d) + \ldots\\ & \mathbb{E}(a)\mathbb{E}(d)\sigma(b,c) + \ldots\\ & \mathbb{E}(b)\mathbb{E}(c)\sigma(a,d) + \ldots\\ & \mathbb{E}(b)\mathbb{E}(d)\sigma(a,c) + \ldots\\ & \sigma(a,c)\sigma(b,d) + \sigma(a,d)\sigma(b,c).\end{aligned} \tag{6.12}$$

Eq.6.11 is valid no matter what the distributions of $a$ and $b$ are, since it is simply a rewrite of the definition of covariance. Eq.6.12 on the other hand is valid if the distributions are Gaussian. Indeed, this is what we have done in the present work. The source term however is not Gaussian distributed anymore.

It is important to state that the exact method is only possible if the material acceleration is evaluated with respect to a stationary reference frame, i.e. the Eulerian approach. If the Lagrangian approach is used, products arise with more than two random variables. Brown and Alexander (1991) derived general expressions for the covariance of a product of $n$ random variables with a product of $m$ random variables. However, these are not usable unless the terms containing products with three terms and higher are neglected, resulting in an approximation. The Monte-Carlo method can straightforwardly be applied no matter which approach is used.

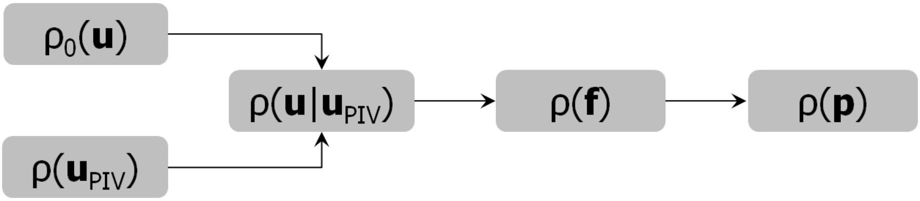


Figure 6.1: Flow diagram of the *a posteriori* uncertainty propagation method, giving the probability distributions function $\rho(\cdot)$ (pdf) of the different components.

### 6.2.3. THE PRESSURE FIELD UNCERTAINTY

The discretization error arising from solving the *discrete* system in Eq.6.2 contributes to the overall uncertainty in the calculated pressure. This error can be reduced by interpolating the measurements onto a finer numerical mesh, which is possible with GPR, and then solving the (larger) resulting system. However, there is a moment the interpolation error starts to dominate therefore limiting the total error reduction. Fortunately, with GPR the interpolation is accompanied by confidence intervals through the posterior covariance matrix $\Sigma(\mathbf{u}|\mathbf{u}_{\mathrm{PIV}})$. The interpolation error is therefore included in the pressure field uncertainty, provided that the system is solved on a sufficiently fine mesh.

The mean and covariance for the pressure follow rather straightforwardly from the identities of the expected value and covariance since the Laplacian operator given by the left-hand side of Eq.6.2, is a *linear* operator:

$$\mathbb{E}\left(\mathbf{p}\right) = A^{-1}\mathbb{E}\left(\mathbf{f}\right), \tag{6.13}$$

$$\Sigma\left(\mathbf{p}\right) = A^{-1}\Sigma\left(\mathbf{f}\right)A^{-T}. \tag{6.14}$$

Figure 6.1 summarizes the steps involved to propagate the uncertainty from the velocity to the pressure field.

### 6.2.4. A ONE DIMENSIONAL EXAMPLE

We use a simple one dimensional example to clarify each step of the proposed methodology. Extension to two and three dimensional problems is relatively straightforward, and will be explained in Sections 6.3 and 6.4, covering a synthetic test case and experimental data, respectively. Consider the following one dimensional example:

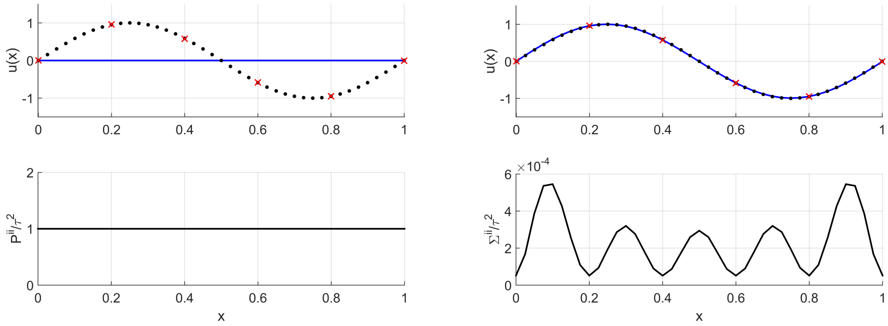
$$\frac{d^2p}{dx^2} = 2u\frac{du}{dx}, \quad \forall x \in (0,1), \text{ with } p(0) = 0, p(1) = \pi. \tag{6.15}$$

Eq.6.15 is similar to the Poisson equation Eq.6.1 in that the second derivative of the pressure is taken and the right hand side is nonlinear in the velocity $u$. The problem is completed with Dirichlet boundary conditions. If we take

$$u(x) = \sin(2\pi \cdot x),$$

it can be verified that the pressure is

$$p(x) = \pi \cdot x - \frac{1}{8\pi}\sin(4\pi \cdot x).$$



(a)

(b)

Figure 6.2: The one dimensional example. (a) True function evaluations (black dots), observations (red crosses), prior mean (blue line), prior variance scaled with $\tau^2$ (black line); (b) posterior mean (blue line), posterior variance scaled with prior variance $\tau^2$ (black line).

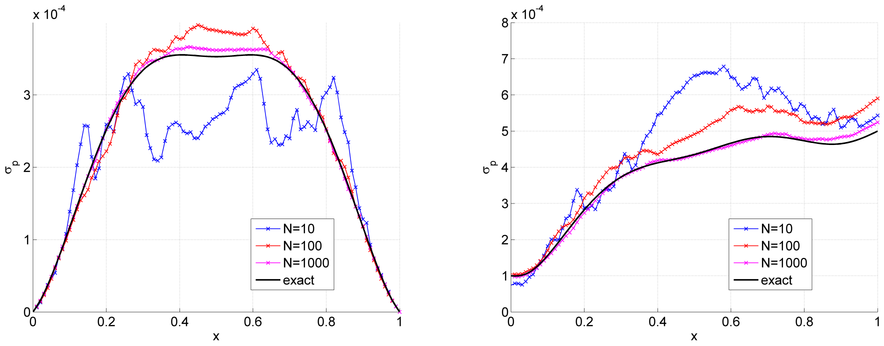
We take six uniformly distributed measurement locations ($m = 6$) but discretize the velocity at 41 points to illustrate the ability of GPR to interpolate the measurement points ($n = 41$).

For the prior mean, we assume the zero vector, so $\boldsymbol{\mu}_0 = \mathbf{0}$. The prior covariance matrix is set up according to Eq.6.3, where we have used the Wendland function from Eq.B.5 as the covariance function. The correlation length $\theta$ and the prior variance $\tau^2$ are found using a cross-validation approach (Viana et al, 2009) and are set to 2 and 0.48, respectively. The black dots in Figure 6.2(a) are the true function evaluations at the $n$ locations of interest. The thick blue line represents the prior mean. The bottom plot of Figure 6.2(a) shows the prior variance, scaled with $\tau^2$.

Measurement noise is included, with a variance of $\sigma^2 = 2.5 \cdot 10^{-5}$. It is assumed that the measurement uncertainty at the locations are uncorrelated to each other, therefore $R = \sigma^2 I$, where $I$ is the identity matrix. Since there are 6 uniformly distributed measurement locations and the state contains 41 uniformly distributed points, the observation operator $H \in \mathbb{R}^{6\times 41}$ is simply a Boolean matrix with $H^{ij} = 1$ at $j = 8i - 7$ ($i = 1, 2, ..., 6$) and $H^{ij} = 0$ everywhere else.

With all the relevant components defined, we can calculate the posterior mean and covariance with Eqs.6.5 and 6.6, respectively. Figure 6.2(b) shows the results. Starting with the mean (top plot): combining the prior and measurements we obtain a posterior that agrees well with the true function evaluations. Proceeding to the bottom plot, the posterior variance is strongly decreased with respect to the prior variance from the bottom plot of Figure 6.2(a). Also, we can clearly see a decrease in the variance at the measurement locations and an increase in between. Indeed, the measurements are informative and the larger the distance from them, the larger the uncertainty in the reconstruction. Note that the variance is not decreasing to zero at the measurement locations, since we included measurement noise with a variance of $\sigma^2 = 2.5 \cdot 10^{-5}$.

To illustrate how the exact propagation method presented in Section 6.2.2 compares



(a) (b)

Figure 6.3: The one dimensional example. Exact standard deviation of calculated pressure (black line), and Monte-Carlo results using $N$ realizations. (a) Perfect pressure measurements at boundaries, (b) assuming $\sigma_p = 1 \cdot 10^{-4}$ at $x = 0$ and $\sigma_p = 5 \cdot 10^{-4}$ at $x = 1$.

with Monte-Carlo simulations, we refer to Figures 6.3(a) and 6.3(b). They show how the random velocity error eventually propagates into the pressure. The difference with the two figures is due to the accuracy of the pressure measurements at the boundaries. In Figure 6.3(a), it is assumed that the pressure can be measured with perfect accuracy. In Figure 6.3(b), we have taken standard deviations of $1 \cdot 10^{-4}$ and $5 \cdot 10^{-4}$ for the pressure measurements at the left and right boundaries, respectively.

Finally, we shift our focus to Section 6.2.3 on the influence of discretization and interpolation errors. In the absence of measurement noise ($\sigma = 0$) and measurements available at all locations ($m = n$), the error reduces with a finer mesh (larger $n$ or decreasing mesh distance $h$), as illustrated by the red line in Figure 6.4. Since we used second order finite difference schemes to discretize the differential operators, we observe second order convergence. However, with measurements only available at the six locations ($m = 6$), GPR can be used to interpolate the measurements onto the finer mesh. The magenta line in Figure 6.4 illustrates the plateau that occurs once the interpolation error starts to dominate the discretization error, therefore hindering the initial monotonic decrease of the total error. With the blue line we have included the measurement noise: the initial decrease and the plateau are still present, however as expected we observe the influence of the noise and a larger final error.

In the above discussion we were able to calculate the error since we know the true pressure. In practice though, the true pressure is unknown and we aim to *estimate* it, which is of course the goal of the present work. The black line in Figure 6.4 represents the estimated error resulting from solving Eq.6.14 and shows that it is a good approximation of the true error, therefore illustrating the ability of the method to also include the interpolation error in the error estimate.

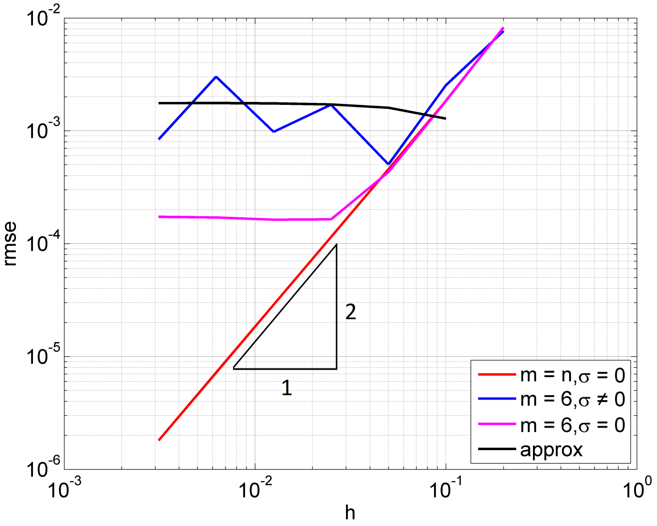


Figure 6.4: The one dimensional example. Root-mean-square error (rmse) of the reconstructed pressure as a function of mesh size $h = 1/(n-1)$. When m = n, $\sigma = 0$, the observations are available at all points, without measurement noise; m = 6, $\sigma \neq 0$ represents six equally spaced observations, with $\sigma^2 = 2.5 \cdot 10^{-5}$; m = 6, $\sigma = 0$ represents six equally spaced observations, without measurement noise; 'estimated' is the estimated error following from Eq.6.14 when $\sigma^2 = 2.5 \cdot 10^{-5}$.

## 6.3. NUMERICAL VERIFICATION

The proposed numerical procedure for uncertainty quantification of PIV-based pressure reconstruction is compared to a reference solution obtained using Monte-Carlo with 10,000 samples. The Monte-Carlo simulation employs exactly the same statistical models as our proposed approach, so that in this section we are verifying only the numerical procedure, and not the statistical modelling. The considered flow field is a two dimensional incompressible Lamb-Oseen vortex. In a polar reference system with radius $r$ and azimuthal angle $\theta$ the analytical expressions of azimuthal velocity $V_\theta$ and radial velocity $V_r$ are, respectively:

$$V_\theta = \frac{\Gamma}{2\pi r}\left(1 - \exp\left[-r^2/4\nu t\right]\right), \tag{6.16}$$

$$V_r = 0, \tag{6.17}$$

where $\Gamma$ is the vortex circulation, $\nu$ is the kinematic viscosity, and $t$ is time. A steady simulation is conducted with $t = 1$ s. The values of dynamic viscosity and density are set to $\mu = 2.4 \cdot 10^{-7}$ Pa · s and $\rho = 1.2$ kg/m$^3$, respectively, yielding a kinematic viscosity $\nu = 2.0 \cdot 10^{-7}$ m$^2$/s. The vortex circulation is $\Gamma = 0.02$ m$^2$/s. The analytical expression of the pressure field is obtained from integration of the Navier-Stokes equations in polar

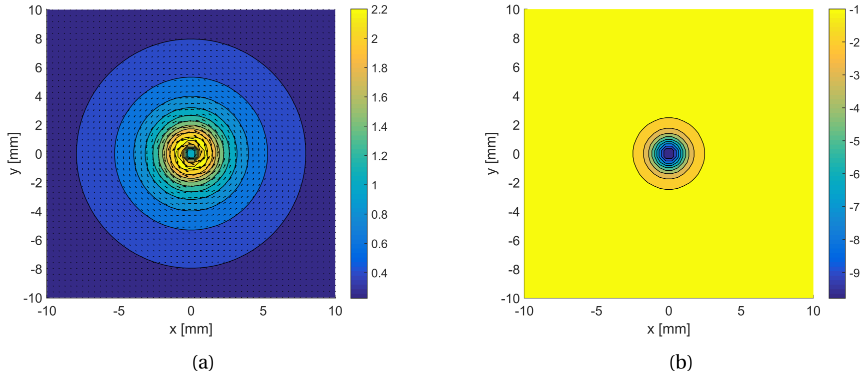


Figure 6.5: (a) Exact velocity field of the Lamb-Oseen vortex (in m/s); (b) exact pressure field (in Pa).

coordinates, yielding:

$$p = \frac{\rho\Gamma^2}{4\pi^2}\left[-\frac{1}{2r^2} + \frac{\exp\left[-r^2/4\nu t\right]}{r^2} - \frac{1}{2}\frac{\exp\left[-2r^2/4\nu t\right]}{r^2} + Ei\left(\frac{2r^2}{4\nu t}\right)\frac{1}{4\nu t} - Ei\left(\frac{r^2}{4\nu t}\right)\frac{1}{4\nu t}\right], \tag{6.18}$$

where $Ei$ is the exponential integral function. The square domain has a size of 20 mm, and the center of the vortex coincides with the center of the domain at $(x, y) = (0, 0)$. The domain is divided into 50 grid points in both horizontal and vertical directions, resulting in a grid spacing $d = 0.4$ mm. The analytical velocity and pressure field are illustrated in Figure 6.5; the pressure in the vortex core is $p_{\text{core}} = -9.78$ Pa. To simulate noise in the velocity measurements, zero-mean white Gaussian noise is added to the two velocity components, with a standard deviation ranging from 0% to 30% of the maximum velocity magnitude $V_{\text{mag}}$: $R = U^2 I$, where $U = \alpha V_{\text{mag}}$. $\alpha$ is between 0 and 0.30. The pressure field is reconstructed from the velocity data using the Poisson equation approach with pure Dirichlet boundary conditions. The pressure at the boundary of the measurement domain is computed via the incompressible flow momentum equation.

For all values of noise level, the standard uncertainty of the reconstructed pressure is evaluated with two approaches: (i) the Bayesian framework presented in Section 6.2; (ii) linear uncertainty propagation (Wieneke and Sciacchitano, 2015). In the latter approach, it is assumed that the two velocity components have the same uncertainty $U$ and that the velocity errors are uncorrelated in space, which is true for the current simulation, but not in typical PIV experiments due to the finite interrogation window size. The expression of the uncertainty of the pressure evaluated by the linear uncertainty propagation reads as:

$$U_p = U\frac{\rho d}{\sqrt{6}}\sqrt{\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2}. \tag{6.19}$$

The uncertainty results from both approaches are compared with the results obtained by Monte-Carlo simulations. The results at the center of the vortex are shown in Figure

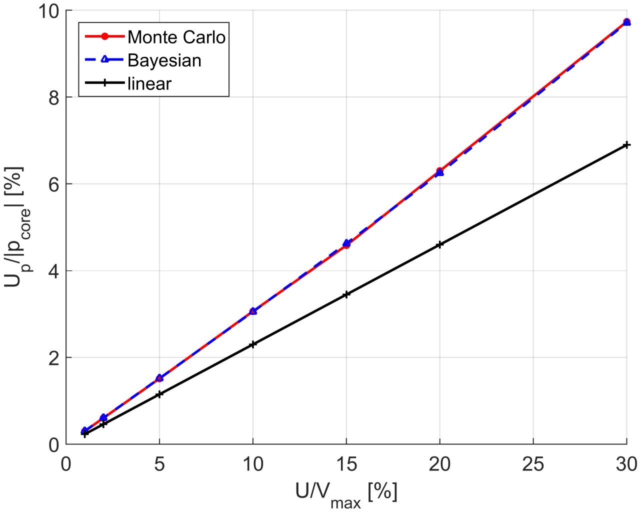
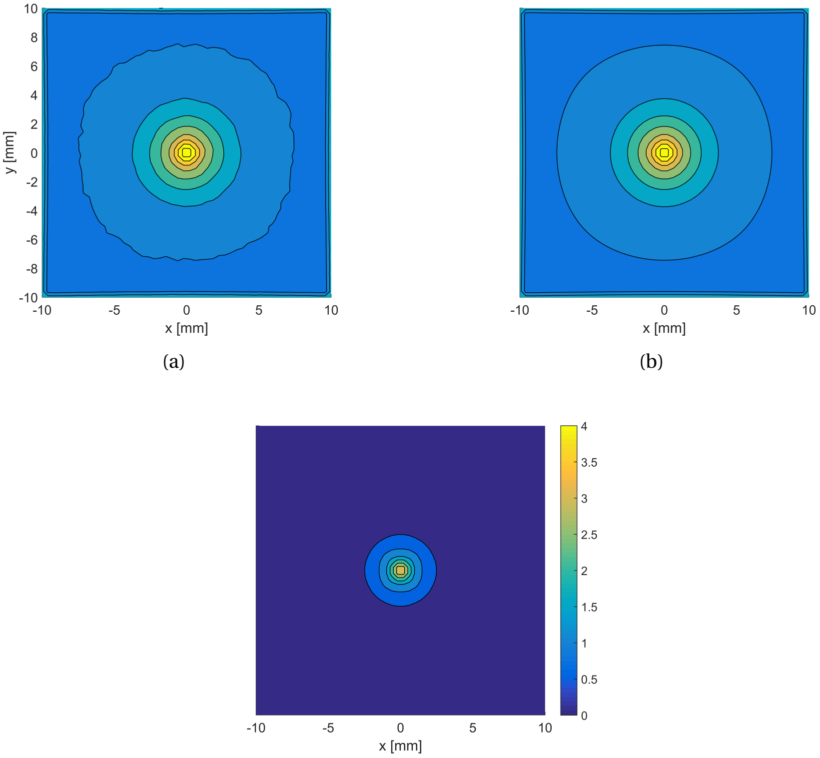


Figure 6.6: The Lamb-Oseen vortex. Comparison between the standard deviation of the pressure error from Monte-Carlo simulations, the Bayesian framework and linear uncertainty propagation, as a function of the uncertainty of the velocity.

6.6: the pressure error increases linearly from 0.3% to 10% of $p_{\text{core}}$ as the uncertainty of the velocity is increased. The Bayesian framework and linear uncertainty propagation reproduce the linear behavior of the uncertainty. However, the linear approach underestimates the uncertainty by about 30% with respect to the Monte-Carlo results, whereas the agreement between the Bayesian framework and Monte-Carlo results is excellent. For illustration purposes, we show the comparison between the standard deviation computed with the different methods. Figure 6.7(a) illustrates the standard deviation of the pressure error obtained via Monte-Carlo simulations for the case where the standard deviation of the added noise is 15% of the velocity magnitude; the pressure error is below 1% outside the vortex core and increases up to 4% in the vortex center. The Bayesian framework for uncertainty propagation (Figure 6.7(b)) shows excellent agreement with the Monte-Carlo results. The linear error propagation however (Figure 6.7(c)) significantly underestimates the uncertainty.

## 6.4. APPLICATION TO EXPERIMENTAL DATA

The experimental test case considered is the fully developed turbulent boundary layer over a flat plate in air, described in Appendix A.2.2. This experiment is particularly useful for the present chapter since microphone measurements were conducted *simultaneously* with the tomo-PIV measurements. Specifically, a pinhole microphone located at the wall. According to Pröbsting et al (2013), the uncertainty of the microphone does not exceed 500 $\mu$Pa, which is approximately 0.05% of the measured pressure fluctuations. The microphone measurements can therefore be used as a reliable ground truth, enabling the possibility to assess the usefulness of the proposed uncertainty propagation



(c)

Figure 6.7: The Lamb-Oseen vortex: comparison between the standard deviation of the pressure error from Monte-Carlo simulations (a), the Bayesian framework (b) and linear uncertainty propagation (c). The results are normalized with $|p_{\text{core}}|$ and in percentage. The noise level on the velocity: 15% of the maximum velocity magnitude.

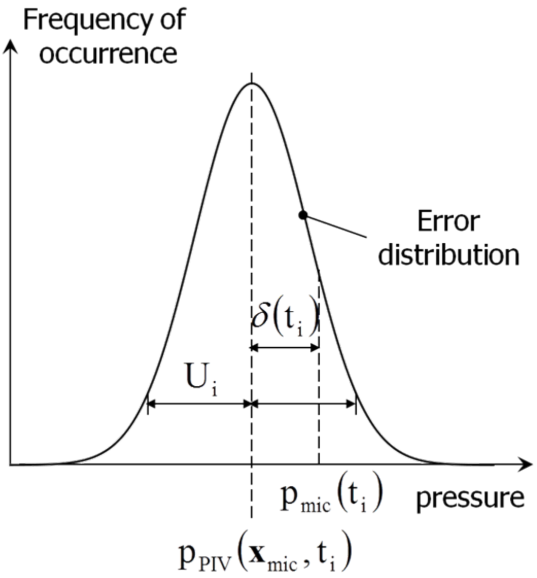


Figure 6.8: Schematic representation of the PIV-based pressure $p_{\text{PIV}}$, 'true' pressure from the microphone $p_{\text{mic}}$ and error $\delta$, and estimated uncertainty $U$.

method. The assessment is done in the following way: the instantaneous error at time instant $t_i$ is defined as the difference between the pressure from the microphone $p_{\text{mic}}$ and the PIV-based pressure $p_{\text{PIV}}$, evaluated at the microphone location $\mathbf{x}_{\text{mic}}$:

$$\delta(t_i) = p_{\text{PIV}}(\mathbf{x}_{\text{mic}}, t_i) - p_{\text{mic}}(t_i). \tag{6.20}$$

With the uncertainty propagation method described in the present chapter, the standard deviation of the PIV-based pressure at location $\mathbf{x}_j$ and time instant $t_i$ follows from the pressure covariance matrix, given by Eq.6.14:

$$\hat{\delta}(\mathbf{x}_j, t_i) = \sqrt{\Sigma_i^{jj}(\mathbf{p})}. \tag{6.21}$$

$\Sigma_i^{jj}$ represents the $j$-th diagonal term of the covariance matrix at time instant $t_i$. When evaluated at the microphone location, we will denote Eq.6.21 as $\hat{\delta}_{\text{mic}}$. With the expanded uncertainty $U = k\hat{\delta}_{\text{mic}}$, a certain percentage of $|\boldsymbol{\delta}|$ should be smaller than $U$. Timmins et al (2012) referred to this as the 'uncertainty effectiveness'. Assuming a Gaussian distribution, a 68.3% confidence level is obtained when $k = 1$ and 95% when $k = 1.96$ (Coleman and Steele, 2009). Figure 6.8 illustrates the definitions we have introduced.

To carry out the uncertainty propagation, we start out with identifying the prior. Two cases are distinguished:

(i) we completely rely on the PIV measurement and take no prior physical knowledge into account. Therefore, the posterior mean and covariance are simply the PIV field (Eq.6.8) and its uncertainty (Eq.6.10), respectively;

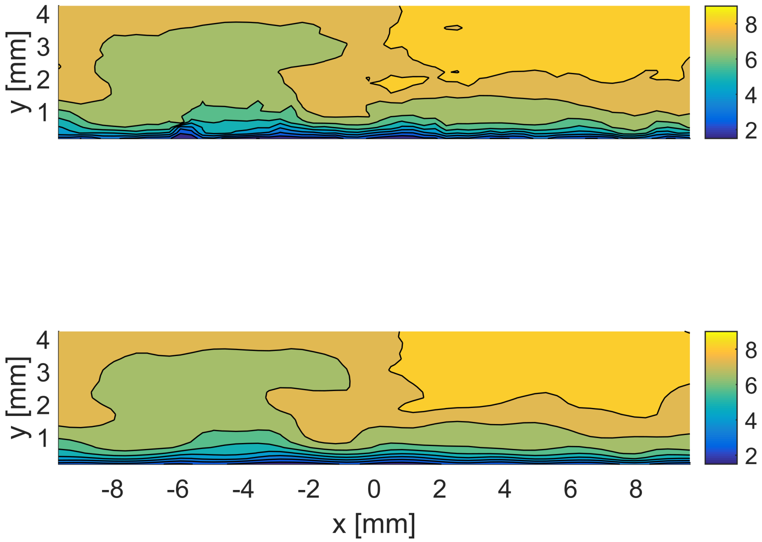


Figure 6.9: The experimental boundary layer flow. Streamwise velocity in [m/s] before (top) and after (bottom) applying the solenoidal filter

(ii) realizing the flow is incompressible, we enforce the divergence-free constraint by applying the solenoidal filter SGPR. The solenoidal filter is linear, therefore the posterior distribution will remain Gaussian distributed.

Proceeding to the measurement uncertainty, the uncertainty of the PIV field is expressed through the observation error covariance matrix $R$. A number of methods are nowadays available to calculate *a posteriori* the uncertainty from PIV (Sciacchitano et al, 2015). However, since the particle images were not available at the moment of the calculation, we have used a different method. Considering that the flow velocity should be divergence-free, we use the measured spurious velocity divergence as an estimator for the uncertainty in the velocity field. This has already been observed by others (Scarano and Poelma, 2009; Violato, 2013). In the present chapter, we use the following approach: we apply the solenoidal filter to the PIV data $\mathbf{u}_{\mathrm{PIV}}$, rendering an *analytically* divergence-free velocity field $\mathbf{u}_{\mathrm{sol}}$. Figure 6.9 shows the streamwise velocity at the center plane before (top) and after (bottom) applying the solenoidal filter at one time instant. The difference between $\mathbf{u}_{\mathrm{PIV}}$ and $\mathbf{u}_{\mathrm{sol}}$ is an approximation of the true error. With 1500 samples, the variance of the error in time is taken as an approximation for the variance of the measurement uncertainty, i.e. the diagonal terms in the observation error covariance matrix $R$. Figure 6.10 shows pdf's of $\mathbf{u}_{\mathrm{PIV}} - \mathbf{u}_{\mathrm{sol}}$ for the 1500 samples at two locations, namely the microphone location (bottom) and the upper side of the measurement domain directly

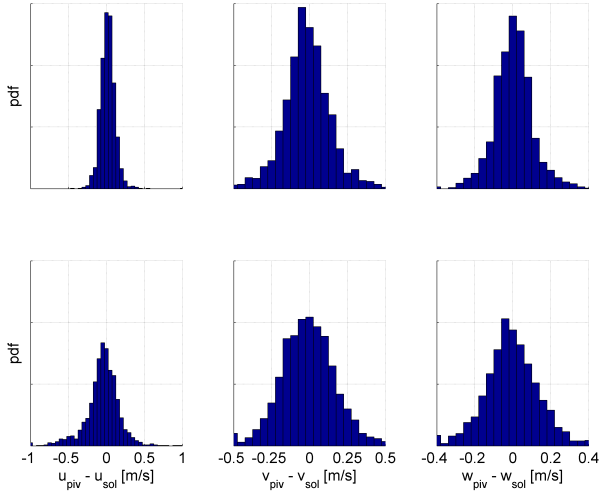


Figure 6.10: Pdf's of $\mathbf{u}_{\mathrm{PIV}} - \mathbf{u}_{\mathrm{sol}}$ for the 1500 samples, evaluated at the microphone location (bottom) and the upper side of the measurement domain directly above the microphone (top).

above the microphone (top). The pdf's look approximately Gaussian, so our assumption of a Gaussian distribution for the PIV measurement uncertainty seems acceptable. Comparing the top plots with their bottom counterparts, notice the increase in variance of the bottom plots, indicating a larger measurement uncertainty closer to the wall. To investigate this further, Figure 6.11 shows the standard uncertainty of each velocity component at the center plane. We clearly see a spatially varying measurement uncertainty for each velocity component. In particular, we observe that the uncertainty increases closer to the wall. The PIV data were processed with a 32 × 16 × 32 interrogation window with 75% overlap. Due to the use of overlapping windows, the measurement error will be spatially correlated. Mathematically, the off-diagonal terms of the observation error covariance matrix $R$ are unequal to zero. Wieneke and Sciacchitano (2015) investigated how the spatial correlation of the measurement uncertainty depends on the overlap and interrogation window sizes using Monte-Carlo simulations. From their analysis, they found correlation values for different grid spacings. Rather than directly inserting them into $R$, we have approximated these values by fitting the Wendland function of order 2 (see Eq.B.5) through them. This ensures that $R$ will be numerically positive definite, thereby the matrix can be inverted.

The boundary conditions we impose are the same as applied by Pröbsting et al (2013), namely Dirichlet at the top boundary and Neumann at all other boundaries. We use both the Eulerian and the Lagrangian approaches to evaluate the material acceleration. According to Violato et al (2011), the Lagrangian approach should result in a more accurate reconstructed pressure field, since the boundary layer flow is a convection dominated type of flow. Whether this is indeed true should not only follow from comparison with the microphone measurement, but is also expected to be represented by a

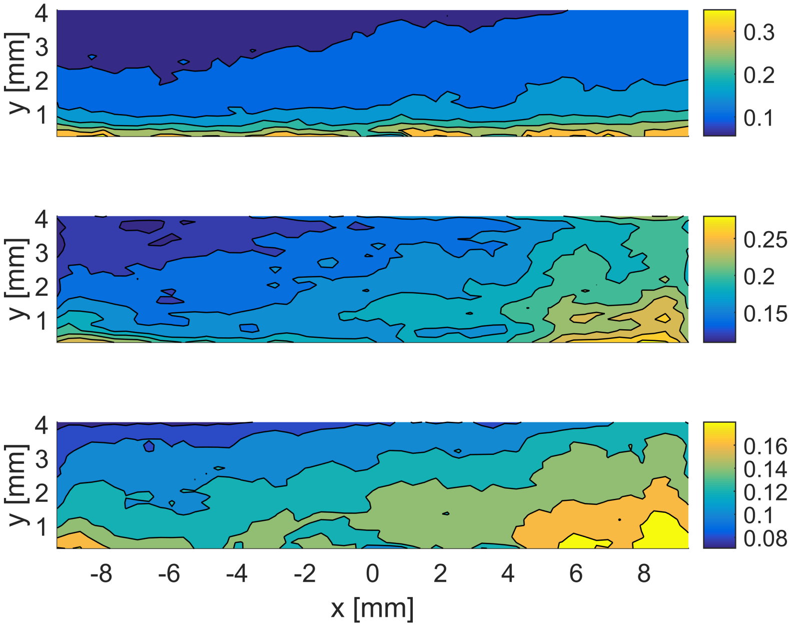


Figure 6.11: Standard uncertainty in [m/s] for the streamwise (top), wall normal (middle) and spanwise (bottom) velocity components.

smaller uncertainty in the reconstructed pressure. To propagate the uncertainties, we carry out Monte-Carlo simulations using 100 realizations. This is the number suggested by Houtekamer and Mitchell (1998) in the context of the Ensemble Kalman Filter.

Following Pröbsting et al (2013), Figure 6.12 shows the pressure time series for a subset of the data. The red lines in the plots are the microphone signal. The results are normalized with the free stream dynamic pressure $q_\infty \approx 60$ Pa. The fluctuations of the microphone signal are in the order of 2% of the free stream dynamic pressure, which is around 1.2 Pa. The black lines in Figure 6.12 are the PIV-based pressure signals. The results represent four cases. From top to bottom: without applying the prior knowledge of divergence-free velocity fields through the solenoidal filter and using the Eulerian approach; without the prior knowledge using the Lagrangian approach; including the solenoidal prior and using the Eulerian approach; including the solenoidal prior and using the Lagrangian approach. From the figure we can already deduce the observations made by Violato et al (2011), namely that the Lagrangian approach results in more accurate pressure reconstruction than the Eulerian approach, and the observations made in Chapter 3 that using the solenoidal prior improves the pressure reconstruction.

In addition to the PIV-based pressure signals we have also plotted the estimated uncertainty. The bounds of the gray region are $p_{\text{PIV}}(\mathbf{x}_{\text{mic}}, t_i) \pm k\hat{\delta}_{\text{mic}}(t_i)$, where we have taken for the coverage factor $k = 1$, so an 68.3% confidence level for a Gaussian distribution. We have not plotted the 95% confidence level ($k = 1.96$) since it would obscure the figures. Already from the plots we can observe that the bounds indeed become tighter when switching from the Eulerian to the Lagrangian approach and when including the solenoidal prior.

Table 6.1 quantifies the above observations. First of all, it shows the standard deviation (std) of the error $\delta$ between the microphone and PIV-based reconstruction. The errors range from 1.2 Pa without the solenoidal prior and the Eulerian approach to approximately 0.6 Pa when using the solenoidal prior together with the Lagrangian approach. The focus of the present work is how the estimated uncertainty compares with this. The rightmost column of Table 6.1 shows the standard deviation of the approximated error $\hat{\delta}_{\text{mic}}$, averaged over the 1500 fields in time. First of all, we notice the excellent agreement between the true and estimated errors. In addition, we observe that switching from the Eulerian to the Lagrangian approach indeed results in a decrease of the estimated uncertainty. The uncertainty is further decreased when including the prior knowledge of

Table 6.1: Standard deviation of the true error, $\text{std}(\delta)$ and the standard deviation of the approximated error averaged over the 1500 fields in time, $\hat{\delta}_{\text{mic,avg}}$. Distinctions made between using no prior knowledge in terms of divergence-free (solenoidal) velocity fields and including this knowledge, and using the Eulerian approach and Lagrangian approach. The results are normalized with the free stream dynamic pressure $q_\infty \approx 60$ Pa.

| | $\text{std}(\delta)/q_\infty$ | $\hat{\delta}_{\text{mic,avg}}/q_\infty$ |
|---|---|---|
| no solenoidal prior (Eulerian) | 0.022 | 0.021 |
| no solenoidal prior (Lagrangian) | 0.015 | 0.013 |
| solenoidal prior (Eulerian) | 0.012 | 0.015 |
| solenoidal prior (Lagrangian) | 0.011 | 0.010 |

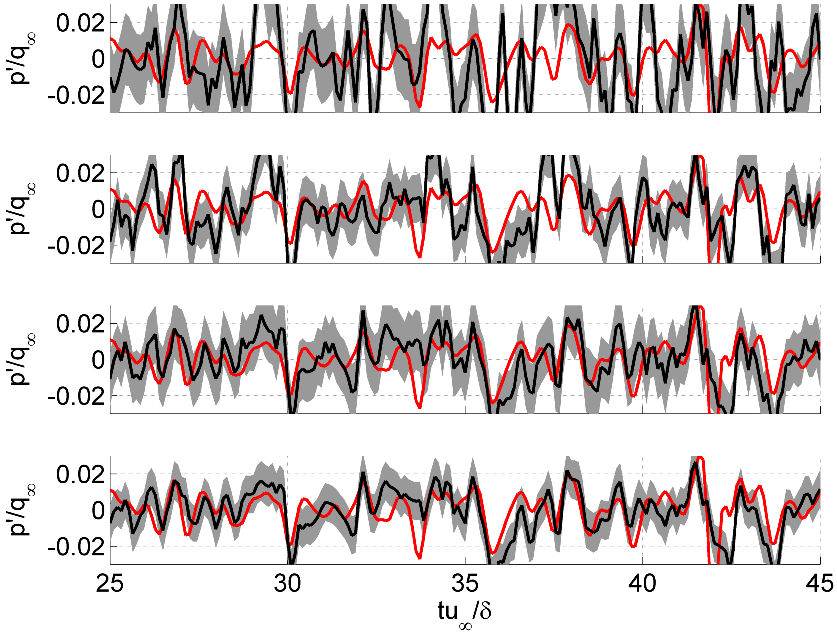


Figure 6.12: Pressure time series obtained with the microphone (red) and PIV (black). From top to bottom: without applying the prior knowledge of divergence-free velocity fields through the solenoidal filter and using the Eulerian approach; without the prior knowledge using the Lagrangian approach; including the solenoidal prior and using the Eulerian approach; including the solenoidal prior and using the Lagrangian approach. $\delta$ is the boundary layer thickness, $u_\infty$ is the free stream velocity, $q_\infty$ is the free stream dynamic pressure and $p'$ the pressure fluctuation. The gray regions represent the estimated uncertainty, where the bounds are $p_{\text{PIV}}\left(\mathbf{x}_{\text{mic}}, t_i\right) \pm k\hat{\delta}_{\text{mic}}\left(t_i\right)$.

divergence-free velocity fields. Table 6.2 shows the results for the uncertainty effectiveness for the 68.3% and 95% confidence levels. In conclusion, the approximated error

Table 6.2: The uncertainty effectiveness.

| | Uncertainty effectiveness (%) | Theoretical value of uncertainty effectiveness (%) |
|---|---|---|
| no solenoidal prior (Eulerian) | 67 ; 93 | 68.3 ; 95 |
| no solenoidal prior (Lagrangian) | 61 ; 90 | 68.3 ; 95 |
| solenoidal prior (Eulerian) | 81 ; 98 | 68.3 ; 95 |
| solenoidal prior (Lagrangian) | 68 ; 94 | 68.3 ; 95 |

that follows from the uncertainty propagation method is indeed useful, given that the uncertainty effectiveness is close to the theoretical value. We used coverage factors corresponding to a Gaussian distribution. Though the velocity uncertainties we started out with were Gaussian, due to the nonlinearity of the Navier-Stokes equations the pressure need not be Gaussian distributed anymore. A Gaussian distribution is known to have skewness 0 and a kurtosis of 3. Since we used Monte-Carlo simulations to propagate the uncertainties, we have the full probability distributions of the pressure. The skewness and kurtosis of the velocity at the microphone location were 0.034 and 3.02, respectively, since we used a *finite* number of random realizations, i.e. 100. Table 6.3 shows the resulting values for the pressure distributions, extracted from the position of the microphone in the PIV field. There is indeed a departure from the Gaussian observable, though not considerable. Therefore, it is acceptable to use the Gaussian coverage factors.

Table 6.3: Skewness and kurtosis of the velocity and pressure distribution, extracted from the microphone position in the PIV field.

| | skewness velocity | skewness pressure | kurtosis velocity | kurtosis pressure |
|---|---|---|---|---|
| no solenoidal prior (Eulerian) | 0.034 | -0.20 | 3.02 | 2.93 |
| no solenoidal prior (Lagrangian) | 0.034 | -0.024 | 3.02 | 2.44 |
| solenoidal prior (Eulerian) | 0.034 | -0.15 | 3.02 | 2.59 |
| solenoidal prior (Lagrangian) | 0.034 | 0.15 | 3.02 | 2.89 |

To appreciate how the PIV measurement uncertainty propagates to the calculated pressure field, we refer to Figure 6.13. It shows the standard uncertainty of the reconstructed pressure field as a function of the distance from the wall where the microphone is located. Notice the decreasing uncertainty as we move away from the wall and compare with Figure 6.11, which shows the standard uncertainty of the PIV measurement. Similarly, we notice a decrease in the standard uncertainty with increasing distance from the wall.

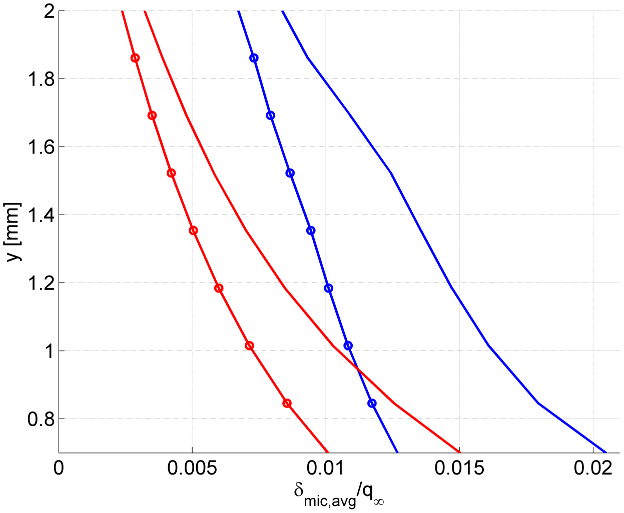


Figure 6.13: Standard uncertainty of the reconstructed pressure field as a function of the distance from the wall. Excluding (blue) and including (red) solenoidal filter, using the Eulerian (solid) and Lagrangian (circles) approaches.

## 6.5. CONCLUSION

This chapter proposed a methodology for the *a posteriori* quantification of the uncertainty of pressure data retrieved from PIV measurements. The approach relies upon the Bayesian framework, where the posterior distribution (probability distribution of the true velocity, given the PIV measurements) is obtained from the prior distribution (prior knowledge of the velocity, e.g. within a certain bound or divergence-free) and the distribution representing the PIV measurement uncertainty. Once the covariance matrix of the velocity is known, it is propagated through the Poisson equation. Provided the velocity uncertainty is Gaussian and the Eulerian approach is used, the uncertainty can be calculated exactly. Otherwise, Monte-Carlo simulations can be used. Numerical assessment of the method on a steady Lamb-Oseen vortex showed excellent agreement of the propagated uncertainty with Monte-Carlo simulations, while linear uncertainty propagation underestimates the uncertainty of the pressure by up to 30%. The method was finally applied to an experimental test case of a turbulent boundary layer in air, obtained using time-resolved tomographic PIV. The pressure reconstructed from tomographic PIV data was compared to the surface measurement conducted by a microphone to determine the actual error of the former. The PIV uncertainty was quantified by applying a solenoidal filter to remove spurious divergence. In addition, spatial correlation of the PIV uncertainty was included. It was found that close to the wall, the velocity uncertainty increases, most likely due to the increased velocity gradient closer to the wall. This was observed to be propagated into the pressure field as well. The comparison between actual and approximated error showed the effectiveness of the proposed method for uncertainty quantification of pressure data from tomographic PIV experiments. The

Lagrangian approach resulted in more accurate reconstructed pressure fields than the Eulerian approach, reducing the errors from approximately 2% of the free stream dynamic pressure to approximately 1.5%. Also, enforcing the divergence-free constraint was found to result in a more accurate reconstructed pressure field, eventually reducing the errors to 1%. Both observations also followed from the uncertainty quantification through a decrease in the estimated uncertainty. In addition, the estimated errors were in excellent agreement with the actual error of the pressure.

# 7

# CONCLUSIONS AND RECOMMENDATIONS

*In this chapter we give conclusions regarding the physics-based postprocessing methods developed in the present thesis and our recommendations for future research in this field. We subdivide this chapter according to the postprocessing methods that were developed: Section 7.1 describes SGPR for the filtering of tomographic PIV data (Chapter 3); Section 7.2 describes SGPR applied to increase the spatial resolution of sparse 3D-PTV data (Chapter 4); Section 7.3 describes the spatio-temporal method applied to increase the spatial resolution of time-resolved sparse 3D-PTV data (Chapter 5); Section 7.4 describes the* a posteriori *uncertainty quantification method for PIV-based pressure fields (Chapter 6).*

## 7.1. SOLENOIDAL FILTERING OF TOMOGRAPHIC PIV DATA

For volumetric velocity measurements of incompressible flows, spurious velocity divergence arises due to measurement noise and finite resolution. To remove this spurious divergence from tomographic PIV measurements, we investigated the use of an analytically solenoidal filter based on Gaussian process regression (SGPR). Inspired by a number of recently proposed *a posteriori* PIV measurement uncertainty methods, we formulated the filter from a Bayesian perspective to allow natural inclusion of measurement uncertainty. We used two synthetic test cases to assess the accuracy of the filtered velocity and derived vorticity and pressure fields and compared with conventional (non-physics-based) filtering and two other recently proposed solenoidal filters that have been applied to volumetric velocity measurements as well. The advantage of the other solenoidal filters is that they do not require a tuning parameter. SGPR on the other hand does, but we found that as long as the user makes use of his *a priori* physical knowledge of the field of interest, the results are quite insensitive to the exact value chosen. This demonstrated that the method is robust to its tuning parameters. The first experimental data set showed that by including the measurement uncertainty, SGPR is able to reconstruct a solenoidal velocity field that more faithfully follows the measurements. The second experimental data set showed that all three solenoidal filters improve the reconstructed pressure signal, independent of which scheme is used to evaluate flow acceleration. Though we eventually found that similar accuracies could be achieved by applying a least-squares regression of the fluid parcel's velocity using a first-order polynomial basis that includes nine snapshots, applying the solenoidal filters is preferred. The least squares regression is a Lagrangian approach that advects measurements forward and backward, therefore cropping of the measurement results. The more snapshots are used, the more cropping occurs. The solenoidal filters on the other hand only carry out spatial filtering, thereby leaving the measurement volume unchanged while at the same time they do not need time-resolved measurements.

### 7.1.1. RECOMMENDATIONS

The component $HPH'$ of the SGPR gain matrix given in Eq.3.14 is formulated in its most general form, allowing different covariance functions to be used for the different directions of the vector potential. Also, from Eq.2.14, the radial distance required by the covariance function can be scaled differently in each coordinate direction by using different correlation lengths. In the present work, we used a single covariance function and a single correlation length for all coordinate directions. This allowed us to choose *a priori* a correlation length, therefore avoiding a potentially costly optimisation. Future work could aim to investigate to what extent the results from SGPR can be further improved by using anisotropic correlation lengths, for example to account for flow anisotropy. This would complicate an *a priori* choice of the correlation lengths, therefore calling for an optimisation routine to be used for this based on for example maximum likelihood estimation. To avoid excessive computational time, one could follow the approach by de Baar et al (2013) who enabled fast estimation of these parameters by carrying out the optimisation in the frequency-domain. They applied it to Gaussian process regression, so modification to solenoidal Gaussian process regression would be our recommendation for future work.

## 7.2. SOLENOIDAL INTERPOLATION OF SPARSE 3D-PTV DATA

Having formulated SGPR for the solenoidal filtering of tomographic PIV data, we subsequently investigated whether enforcing the divergence free constraint can improve the interpolation of velocity (and derived) fields obtained from sparse 3D-PTV data of incompressible flows. For this type of data, choosing an appropriate correlation length *a priori* proved to be more difficult since, coming from sparse data, the underlying velocity field is more difficult to identify. We therefore proposed a methodology inspired by procedures common in the machine learning community to find the optimum tuning parameters. We found that incorporating the constraint analytically using SGPR results in more accurate interpolations than applying a discrete solenoidal filter to a linearly interpolated field or a field obtained using GPR. In addition, we found that SGPR consistently returns the most accurate velocity, vorticity and pressure fields. The improvement is most pronounced for the sparsest data sets. These observations were confirmed using both synthetic test cases and an experimental data set.

### 7.2.1. RECOMMENDATIONS

To obtain the pressure field from sparse velocity measurements, it is customary to first interpolate the velocity field onto a regular grid, allowing the pressure field to subsequently be obtained with finite-differencing. Recently though, increased attention has been paid to extracting pressure fields directly from the sparse measurements, bypassing conversion to the regular grid. We refer to Neeteson and Rival (2015) and the references therein. Having confirmed that SGPR can increase the spatial resolution of the measurements, future work could focus on combining SGPR with these methods to further enhance the derived pressure field.

A second recommendation is to investigate an *irrotational* version of Gaussian process regression (Scheuerer and Schlather, 2012), that could be denoted *irrotational Gaussian process regression* (IGPR). Starting with the Navier-Stokes equations (see Eq.1.2) and neglecting the viscous term, we get:

$$\nabla p = -\rho \frac{D\mathbf{u}}{Dt}, \tag{7.1}$$

where $D\mathbf{u}/Dt$ is the material derivative of the velocity field, which can easily be obtained by taking the derivative along the particle tracks. Barring the constant density and the minus sign, Eq.7.1 states that the material acceleration is equal to the gradient of the pressure field, which is a scalar. From vector calculus it therefore follows that the material acceleration is irrotational. In the same way that SGPR improved the spatial resolution of solenoidal velocity fields, IGPR could improve the spatial resolution of the irrotational material acceleration. This could subsequently result in an improvement of the estimated pressure field.

## 7.3. SPATIO-TEMPORAL INTERPOLATION OF SPARSE 3D-PTV DATA

We carried out a proof-of-concept study to investigate how incorporating the temporal information of particle tracks in 3D-PTV, and adding the momentum conservation equation as prior knowledge, can improve the reconstructed velocity and vorticity fields. Specifically, we used the vorticity-velocity formulation of the momentum conservation equation and the vortex-in-cell (VIC) method to solve it. The data assimilation method we used is similar to 4DVAR, therefore the problem reduces to an optimisation problem. The cost function to be minimized is a least-squares sum of the discrepancy between the time-varying measurements and the simulated field. The optimisation variables are the initial velocity field and the time-varying boundary conditions. To calculate gradients efficiently, we exploited the reverse mode of differentiation, which we hand coded. The assimilation algorithm was applied to an experimental test case of a circular jet in water. We compared it with SGPR, which interpolates the snapshots independent from each other, therefore not taking the temporal information into account. We found clear improvements in the reconstructed field when incorporating temporal information.

### 7.3.1. RECOMMENDATIONS

Considering that the investigation of the spatio-temporal interpolation was a proof-of-concept, we can propose a number of recommendations for future work. First of all, to make the method applicable for more general flow cases, we need to include the viscosity term in the vorticity transport equation. This will enable the method to be used for wall-bounded flows, as well as allow integrating over longer time domains, i.e. assimilate longer particle tracks. Including the viscosity term will have implications for the VIC solver, which needs to be modified to accommodate it, and subsequently the adjoint code as well. When including the viscous term, the inviscid vorticity transport equation is not time-reversible anymore. Consequently, the full state needs to be stored when running the adjoint code. Checkpointing was already mentioned as a possible solution, so one would need to investigate how to explicitly implement this or an alternative.

Another recommendation, which will again have computational implications, is to augment the state with the full unknown velocity field. Mathematically, we then acknowledge that the model is not perfect. For the inviscid vorticity transport equation. In the field of 4DVAR, this modification is known as *weak constraint* 4DVAR, as opposed to the *strong constraint* 4DVAR we have used.

## 7.4. *A posteriori* UNCERTAINTY QUANTIFICATION OF PIV-BASED PRESSURE

We proposed a methodology for the *a posteriori* quantification of the uncertainty of pressure data retrieved from PIV measurements. The approach relies upon the Bayesian framework to combine the PIV measurement uncertainty with prior knowledge, like the divergence free constraint. Once the posterior covariance matrix of the velocity is known, we propagated it through the Poisson equation. Provided the velocity uncertainty is Gaussian and the Eulerian approach is used, the uncertainty can be calculated exactly. Otherwise, Monte-Carlo simulations can be used. Numerical assessment of the method on a steady Lamb-Oseen vortex showed excellent agreement of the propagated uncertainty with Monte-Carlo simulations, while linear uncertainty propagation underestimates the uncertainty of the pressure by up to 30%. The method was finally applied to an experimental test case of a turbulent boundary layer in air, obtained using time-resolved tomographic PIV. The pressure reconstructed from tomographic PIV data was compared to the surface measurement conducted by a microphone to determine the actual error of the former. The PIV uncertainty was quantified by applying a solenoidal filter to remove spurious divergence. In addition, spatial correlation of the PIV uncertainty was included. It was found that close to the wall, the velocity uncertainty increases. This was observed to be propagated into the pressure field as well. The comparison between actual and approximated error showed the effectiveness of the proposed method for uncertainty quantification of pressure data from tomographic PIV experiments: for a 95% confidence level, 93% of the data points fell within the estimated uncertainty bound with the Eulerian approach, and 90% with the Lagrangian approach. When using the prior knowledge that the velocity field should be divergence free, these values were 98% with the Eulerian approach and 94% with the Lagrangian approach. The Lagrangian approach resulted in more accurate reconstructed pressure fields than the Eulerian approach. Also, enforcing the divergence free constraint was found to result in a more accurate reconstructed pressure field. Both observations also followed from the uncertainty quantification, through a decrease in the estimated uncertainty.

### 7.4.1. RECOMMENDATIONS

The uncertainty due to the application of a 2D model to a 3D flow has not been tackled. However, as it stands, the method can be applied to it. Practically, the unmeasured velocity components are specified through the prior but they will not be updated by the measurements. The challenge therefore lies in the hands of the experimentalist to come up with a proper estimation of the prior. To verify the method for this, one can take a time-resolved tomographic PIV experiment, choose a plane to represent the planar PIV experiment and carry out the uncertainty quantification method. The estimated mean pressure together with its accompanying confidence intervals can then be compared with the pressure obtained using the full volumetric data.

Another recommendation is to propagate the uncertainty one step further: to forces. In industrial wind tunnel applications, force measurements are of course very common so an augmented version of the proposed method may become very useful indeed.

# A

# TEST CASES

*In this appendix, we present the test cases used in Chapters 3 to 6 to assess the physics-based postrpocessing methods developed in the present thesis. We make a distinction between synthetic test cases, discussed in Section A.1, and experimental test cases, discussed in Section A.2. The test cases were chosen on the basis that they represent incompressible flow, their prior use in the PIV literature, the ability to properly assess the data assimilation methods and that they challenge the methods.*

## A.1. SYNTHETIC TEST CASES

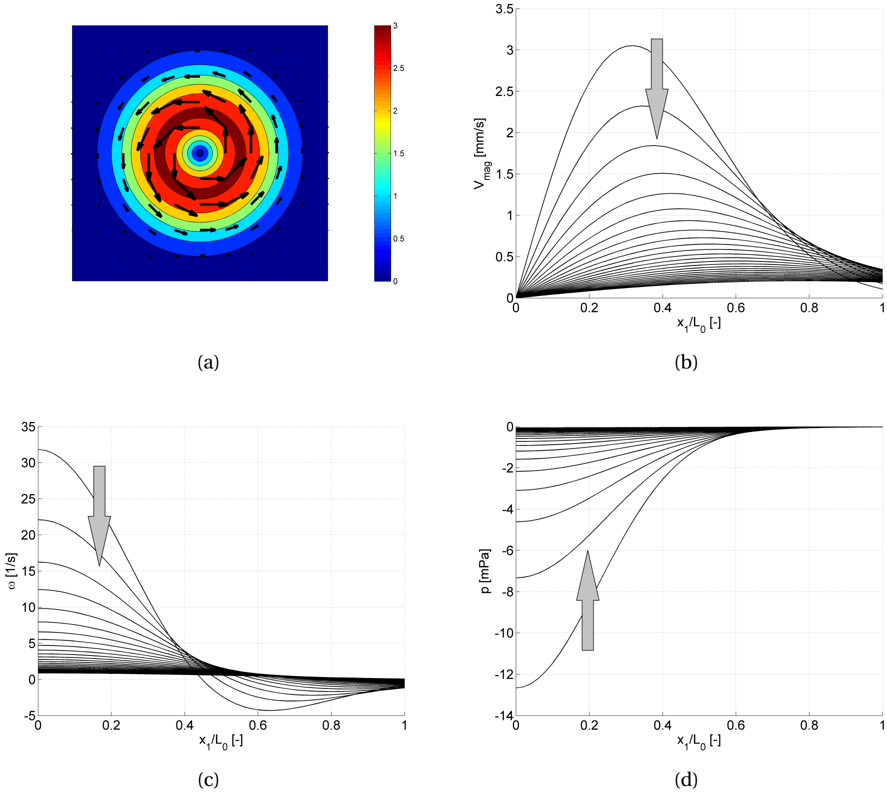
Assessing the synthetic test cases is straightforward since the exact velocity and derived fields are available. In addition, they are exactly divergence-free. The first test cases also exactly satisfies the Navier-Stokes equations, whereas the second test case is based on the linearised vorticity equation for incompressible flows. The data was generated point wise, meaning that a velocity measurement at a specified point is obtained by evaluating the analytical velocity field at that particular point. Gaussian noise is added to the evaluated velocity field to represent measurement noise. In Chapter 3, we generated correlated noise to model the filtering effect of the interrogation windows. More details can be found in that chapter.

### A.1.1. TAYLOR-VORTEX

The Taylor-vortex is a single decaying counter clock wise rotating vortex flow defined in 2D (Panton, 2013). The velocity field is solenoidal and in addition satisfies the Navier-Stokes equations. The tangential velocity is defined as:

$$u_\theta = \frac{H}{8\pi}\frac{r}{\nu t^2}\exp\left(-\frac{r^2}{4\nu t}\right), \tag{A.1}$$

where $H$ is the total angular momentum of the vortex, $r$ is the radial distance from the center of the vortex, $\nu$ is the kinematic viscosity and $t$ is time. The radial velocity is zero.



(a) (b)

(c) (d)

Figure A.1: Taylor-vortex: (a) velocity magnitude and velocity vectors at $t = 0.05$ s; (b) velocity magnitude, (c) vorticity and (d) pressure of all samples at $x_2 = 0$ m, $0 \le x_1/L_0 \le 1$. Arrow directed towards increasing time.

The vorticity is given by:

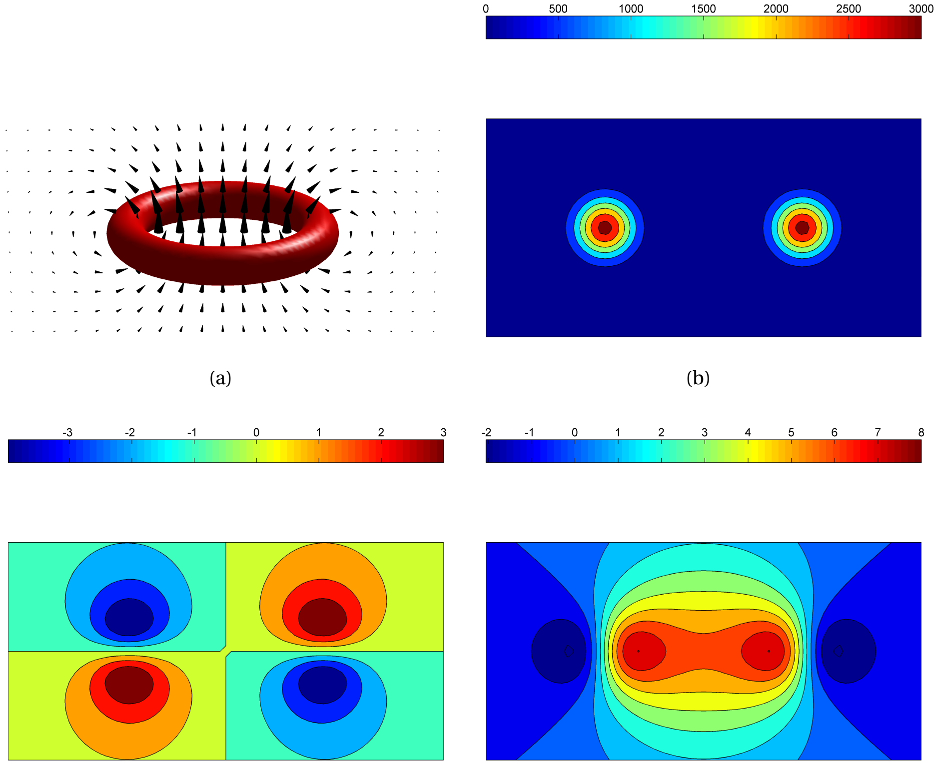
$$\omega = -\frac{H\left(r^2 - 4\nu t\right)}{16\pi\nu^2 t^3} \exp\left(-\frac{r^2}{4\nu t}\right). \tag{A.2}$$

The pressure is given by:

$$p = -\frac{\rho H^2}{64\pi^2 \nu t^3} \exp\left(-\frac{r^2}{2\nu t}\right) + p_\infty, \tag{A.3}$$

where $\rho$ is the constant fluid density and $p_\infty$ is the pressure at an infinite distance from the vortex center.

For the present problem, we take $H = 1\times10^{-6}$ m$^2$, $\nu = 1\times10^{-6}$ m$^2$ s$^{-1}$, $\rho = 1000$ kg m$^{-3}$. The domain size is $-L_0 \le x_i \le L_0$, $i = 1,2$, where $L_0 = 1\times10^{-3}$ m. Samples are taken between $t = 0.05$ s and $t = 0.30$ s with a sampling frequency of 100 Hz. These same settings were also used by Charonko et al (2010). They used this test case to investigate the effects of a number of factors on PIV-based pressure reconstruction, namely the noise level



(a) (b)

(c) (d)

Figure A.2: Vortex ring: (a) isosurface of vorticity magnitude and velocity field at the symmetry plane; (b) vorticity magnitude, (c) velocity in the horizontal direction and (d) velocity in the vertical direction, evaluated at the symmetry plane.

of the velocity measurements, the pressure integration algorithm used, the spatial and temporal resolution and out-of-plane velocities. Figure A.1 shows the velocity, vorticity and pressure fields.

## A.1.2. VORTEX RING

The vortex ring has become a popular test case to numerically assess reconstruction methods in tomographic PIV since its use in Elsinga et al (2006). However, the displacement field, applied to artificial particles, does not result in a divergence-free velocity field. Therefore, we have used a different model proposed by Kaplanski et al (2009). They derived analytical expressions for the velocity and vorticity fields by linearizing the vorticity equation for incompressible flows. The axial ($u_a$) and radial ($u_r$) velocities are defined as:

$$u_a = \pi\theta^2 \int_0^\infty \mu F(\mu,\eta)\, J_1(\theta\mu)\, J_0(\sigma\mu)\, d\mu, \tag{A.4}$$

$$u_r = -\pi\theta^2 \int_0^{\infty} \mu\tilde{F}(\mu,\eta)\, J_1(\theta\mu)\, J_1(\sigma\mu)\, d\mu, \tag{A.5}$$

where $J_0$ and $J_1$ are Bessel functions. The functions $F(\mu,\eta)$ and $\tilde{F}(\mu,\eta)$ are:

$$F(\mu,\eta) = \exp(\eta\mu)\,\mathrm{erfc}\left(\frac{\mu+\eta}{\sqrt{2}}\right) + \exp(-\eta\mu)\,\mathrm{erfc}\left(\frac{\mu-\eta}{\sqrt{2}}\right), \tag{A.6}$$

$$\tilde{F}(\mu,\eta) = \exp(\eta\mu)\,\mathrm{erfc}\left(\frac{\mu+\eta}{\sqrt{2}}\right) - \exp(-\eta\mu)\,\mathrm{erfc}\left(\frac{\mu-\eta}{\sqrt{2}}\right). \tag{A.7}$$

The parameters $\sigma$, $\eta$ and $\theta$ are:

$$\sigma = \frac{r}{l}, \qquad \eta = \frac{x_1}{l}, \qquad \theta = \frac{R_0}{l}.$$

The vorticity is given by:

$$\omega = \frac{\Gamma_0 R_0}{\sqrt{2\pi}\, l^3} \exp\left[-\frac{1}{2}\left(\sigma^2+\eta^2+\theta^2\right)\right] I_1(\sigma\theta), \tag{A.8}$$

where $I_1$ is the modified Bessel function. Contrary to the Taylor-vortex, we do not have an analytical expression for the pressure field. Also, the field is not time-resolved. To obtain a vortex ring with a similar shape to the experimental test case in Section A.2.1, we have chosen the following parameters: $R_0 = 0.0067$, $l = 0.0031$, $\Gamma_0 = 4\pi R_0$. The domain size is $-2.5R_0 \leq x_1 \leq 2.5R_0$ and $-5R_0 \leq x_2, x_3 \leq 5R_0$. The coordinate direction 1 is taken as the axial direction. Figure A.2 shows the velocity and vorticity fields of the vortex ring.

## A.2. EXPERIMENTAL TEST CASES

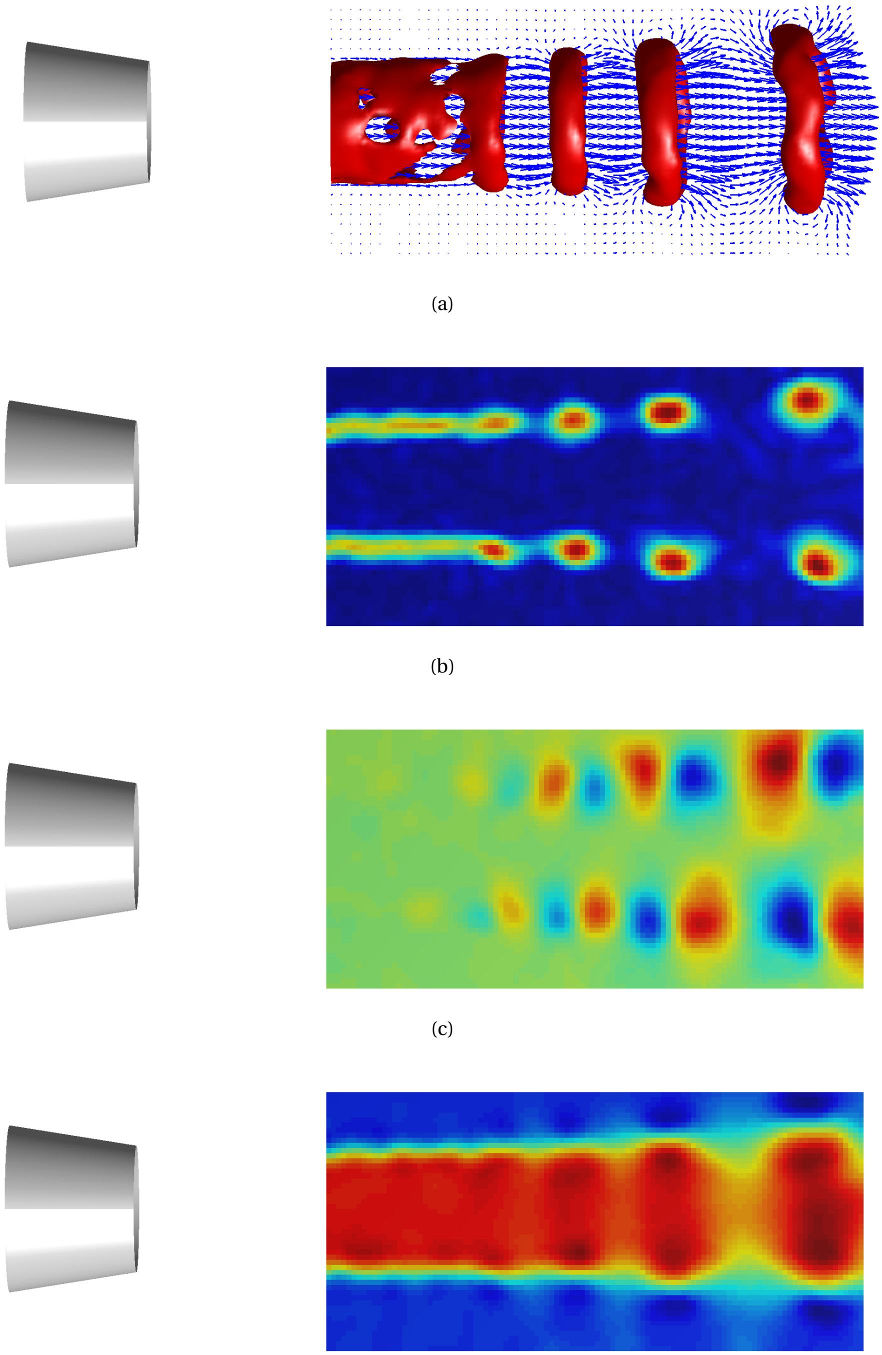
Both experimental data sets were obtained using time-resolved tomographic PIV. Assessing the methods is however not as straightforward as it was for the synthetic test cases. This is because it is more difficult to obtain a reliable ground truth. For the first experimental test case, we exploit a recently proposed *a posteriori* PIV uncertainty quantification method to obtain an estimate of the measurement uncertainty. For the second experimental test case, we exploit the availability of simultaneous microphone measurements that were carried out, enabling comparison of the PIV-based pressure with the microphone signal.

### A.2.1. CIRCULAR JET IN WATER

The experiment of the circular jet in water was carried out by Violato and Scarano (2011) (see Figure A.3). The jet velocity at the nozzle exit is 0.5 m/s, so incompressible flow is an excellent approximation. The nozzle diameter is 10 mm, giving a Reynolds number based on this diameter of approximately 5,000. The jet periodically sheds vortex rings due to the growth of Kelvin-Helmholtz instabilities, at a frequency of approximately 30 Hz. This can clearly be seen from Figure A.3(a). Since the acquisition frequency of the measurements is 1 kHz, this large scale vortex shedding is captured with good temporal resolution. The measurement volume is approximately 42 mm in the jet direction and 20 mm in the cross directions. The dataset consists of 289,328 vectors and is defined on a regular grid, with 107 vectors in the jet direction and 52 vectors in the cross directions. Further downstream, the jet becomes turbulent, however in the volume considered the flow is laminar. The high shear and rotation occurring within the interface between the jet core and the surrounding stagnant fluid and the strong streamlines curvature and acceleration resulting from the vortex shedding has made the circular jet in water a popular test case, both for testing PIV reconstruction methods (Lynch and Scarano, 2013; Novara, 2013; Sciacchitano et al, 2012b; Wieneke, 2013) and data assimilation methods (Scarano and Moore, 2012; Schneiders et al, 2014; Sciacchitano et al, 2012a).

### A.2.2. TURBULENT FLAT PLATE BOUNDARY LAYER IN AIR

An experiment of the fully developed turbulent boundary layer over a flat plate in air was carried out by Pröbsting et al (2013). The free stream velocity is 10 m/s, so incompressible flow is again an excellent approximation. Figure A.4(a) shows the instantaneous velocity component in the free stream direction at the center plane of the measurement volume. The boundary layer thickness $\delta$ is 9.4 mm. The Reynolds number based on the boundary layer thickness is $Re_\delta \approx 6,240$. The acquisition frequency of the measurements is 10 kHz and 1500 samples were obtained during the measurements. The measurement volume spans 19.7 mm in the stream wise (x) direction, 4.2 mm in the wall-normal (y) direction and 41.3 mm in the span wise (z) direction. The dataset consists of 185,484 vectors and is defined on a regular grid, with 58 vectors in stream wise, 26 vectors in the wall-normal and 123 vectors in the span wise direction. A microphone was located on the wall to measure the pressure fluctuations. The microphone measurements were carried out simultaneously with the PIV measurements, allowing comparison of the microphone based pressure signal with the PIV-based pressure signal. This test case is therefore very suitable to assess the *a posteriori* uncertainty quantification method for PIV-based pressure, which is the subject of Chapter 6. Figure A.4(b) shows the stream wise velocity fluctuations from the PIV data (top), evaluated at the bottom of the measurement volume at the microphone location and the pressure signal from the microphone (bottom).



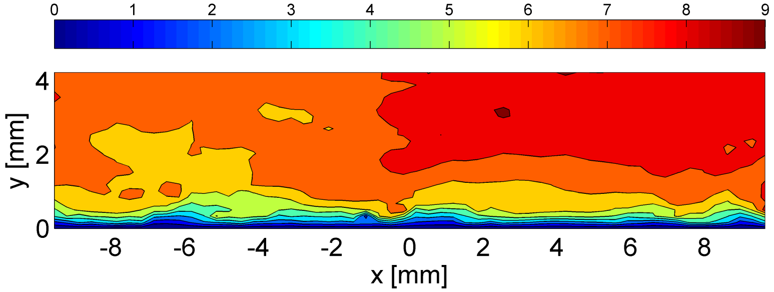
(a)

(b)

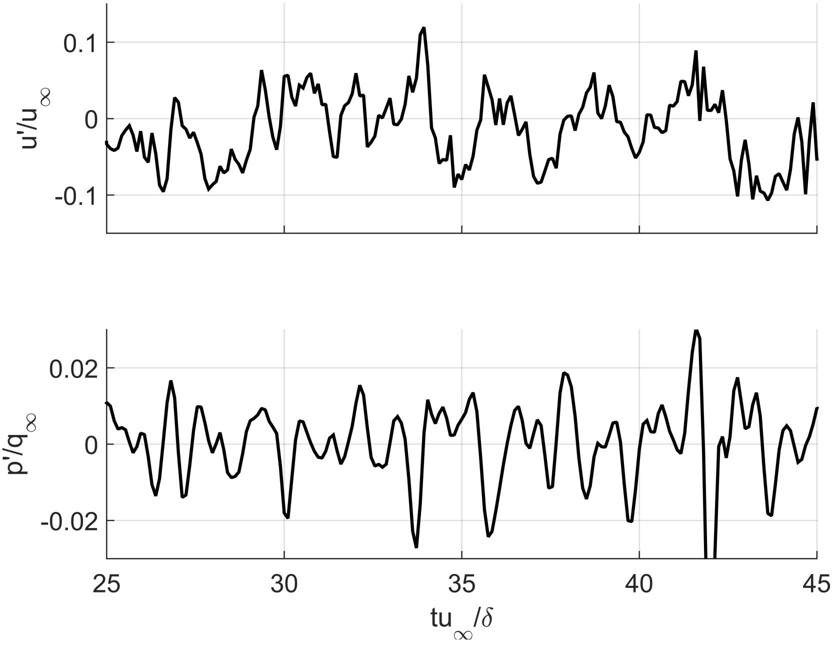
(c)

(d)

Figure A.3: Circular jet in water: (a) velocity vector slice in the symmetry plane and isosurfaces of vorticity magnitude at $\omega = 220$ 1/s; (b) vorticity magnitude, (c) cross flow velocity and (d) stream wise velocity at the symmetry plane.



(a)



(b)

Figure A.4: Turbulent flat plate boundary layer in air: (a) stream wise instantaneous velocity component at the center plane of the measurement volume; (b) stream wise velocity fluctuations from the PIV data (top), evaluated at the bottom of the measurement volume at the microphone location and pressure signal from the microphone (bottom). $\delta$ is the boundary layer thickness, $u_\infty$ is the free stream velocity, $q_\infty$ is the free stream dynamic pressure, $u'$ is the free stream velocity fluctuation and $p'$ the pressure fluctuation.

# B

# WENDLAND FUNCTIONS AND THEIR SECOND PARTIAL DERIVATIVES

*This appendix gives the Wendland functions* $\phi_{3,k}$ *with orders* $k = 1,2,3,4$ *and smoothness* $C^{2k}$*, together with analytical expressions of their second partial derivatives as required by the gain matrix of SGPR (see Eq.3.14).*

Placing the covariance function at $\left(x^i, y^i, z^i\right)$, the scaled radial distance with a point at $\left(x, y, z\right)$ is defined as:

$$r = \sqrt{\left(\frac{x-x^i}{\gamma_1}\right)^2 + \left(\frac{y-y^i}{\gamma_2}\right)^2 + \left(\frac{z-z^i}{\gamma_3}\right)^2}, \tag{B.1}$$

where $\gamma_l$, $l = 1,2,3$ is the correlation length in the $x$-,$y$- and $z$-directions, respectively. In the following sections, the term $(1-r)_+$ means that $(1-r)_+ = 0$ for $r \geq 1$. Also, we will show the expressions for $\partial^2\phi/\partial x^2$ and $\partial^2\phi/\partial x\partial y$. Of course, the derivatives with respect to the other coordinate directions can easily be obtained from them. Figure B.1 illustrates the four Wendland functions.

## B.1. ORDER 1

$$\phi_{3,1} = (1-r)_+^4 (4r+1) \tag{B.2}$$

$$\frac{\partial^2\phi_{3,1}}{\partial x^2} = \frac{20(r-1)_+^2}{r\gamma_1^2}\left(r^2 - r + \frac{3\left(x-x^i\right)^2}{\gamma_1^2}\right) \tag{B.3}$$

$$\frac{\partial^2\phi_{3,1}}{\partial x\partial y} = \frac{60(r-1)_+^2\left(x-x^i\right)\left(y-y^i\right)}{r\gamma_1^2\gamma_2^2} \tag{B.4}$$

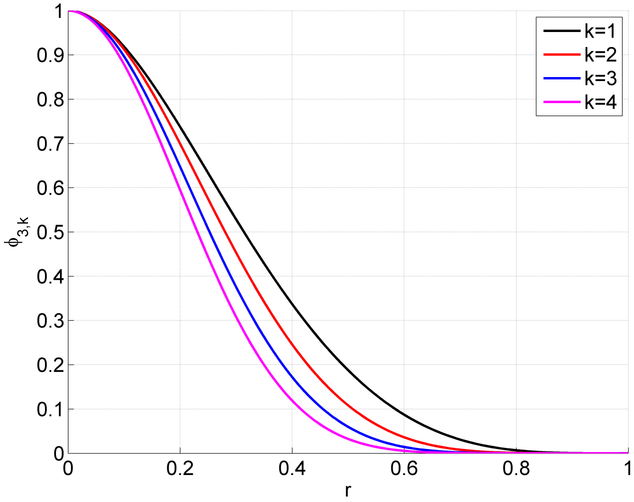


Figure B.1: The Wendland functions $\phi_{3,k}$ with orders $k = 1,2,3,4$.

## B.2. ORDER 2

$$\phi_{3,2} = (1-r)_+^6 \left(\frac{35}{3} r^2 + 6r + 1\right) \tag{B.5}$$

$$\frac{\partial^2 \phi_{3,2}}{\partial x^2} = \frac{56\,(r-1)_+^4}{3\gamma_1^2} \left(5r^2 - 4r - 1 + 30\left(\frac{x-x^i}{\gamma_1}\right)^2\right) \tag{B.6}$$

$$\frac{\partial^2 \phi_{3,2}}{\partial x \partial y} = \frac{560\,(r-1)_+^4}{\gamma_1 \gamma_2} \left(\frac{x-x^i}{\gamma_1}\right)\left(\frac{y-y^i}{\gamma_2}\right) \tag{B.7}$$

## B.3. ORDER 3

$$\phi_{3,3} = (1-r)_+^8 \left(32r^3 + 25r^2 + 8r + 1\right) \tag{B.8}$$

$$\frac{\partial^2 \phi_{3,3}}{\partial x^2} = \frac{22\,(r-1)_+^6}{\gamma_1^2} \left((r-1)\left(16r^2 + 7r + 1\right) + (144r + 24)\left(\frac{x-x^i}{\gamma_1}\right)^2\right) \tag{B.9}$$

$$\frac{\partial^2 \phi_{3,3}}{\partial x \partial y} = \frac{528\,(r-1)_+^6\,(1+6r)}{\gamma_1 \gamma_2} \left(\frac{x-x^i}{\gamma_1}\right)\left(\frac{y-y^i}{\gamma_2}\right) \tag{B.10}$$

## B.4. ORDER 4

$$\phi_{3,4} = (1-r)_+^{10} \left(\frac{429}{5} r^4 + 90r^3 + 42r^2 + 10r + 1\right) \tag{B.11}$$

$$\frac{\partial^2\phi_{3,4}}{\partial x^2} = \frac{26\,(r-1)_+^8}{5\gamma_1^2}\{\frac{(r-1)\left(231r^3+159r^2+45r+5\right)}{\gamma_1^2}+\ldots \\ 132\left(21r^2+8r+1\right)\left(\frac{x-x^i}{\gamma_1}\right)^2\} \tag{B.12}$$

$$\frac{\partial^2\phi_{3,4}}{\partial x\partial y} = \frac{3432\,(r-1)_+^8}{5\gamma_1\gamma_2}\left(\left(21r^2+8r+1\right)\left(\frac{x-x^i}{\gamma_1}\right)\left(\frac{y-y^i}{\gamma_2}\right)\right) \tag{B.13}$$

# C

# EFFICIENT ADJOINT-BASED COST FUNCTION GRADIENT CALCULATION

*This appendix explains how to efficiently calculate the gradient of the cost function J from Eq.5.20. To this end, we have hand-coded the adjoint of the flow diagram from Figure 5.2. Also, we have avoided the simple yet inefficient approach of setting up matrices and taking their transpose, which is the essence of the adjoint operator.*

The flow diagram of the adjoint is illustrated by Figure C.1. As we can see from the diagram, the flow of variables is reversed and we start at the bottom now. Also notice that all variables $(\cdot)$ have been replaced by their adjoint counterparts $\delta(\cdot)^*$. Starting at the bottom now, we set

$$\delta J^* = 1.$$

In the original code,

$$J = \sum_{l=1}^{N_t} J^l,$$

see also Eq.5.8. This can also be written as:

$$J = \mathbf{1}'\mathbf{J}, \tag{C.1}$$

where $\mathbf{1} \in \mathbb{R}^{N_t}$ is a vector composed of all ones and $\mathbf{J} \in \mathbb{R}^{N_t}$ is a vector with components $J_i = J^i$, $i = 1, ..., N_t$. The adjoint is obtained by replacing the order of the variables in Eq.C.2 and taking the transpose of $\mathbf{1}'$:

$$\delta\mathbf{J}^* = \mathbf{1}\delta J^*. \tag{C.2}$$

Since we set $\delta J^* = 1$, this value flows unchanged into its components, i.e. :

$$\delta J^{*l} = 1, \tag{C.3}$$

$l = 1, ..., N_t$. In the following sections, we explain the adjoint blocks from Figure C.1.

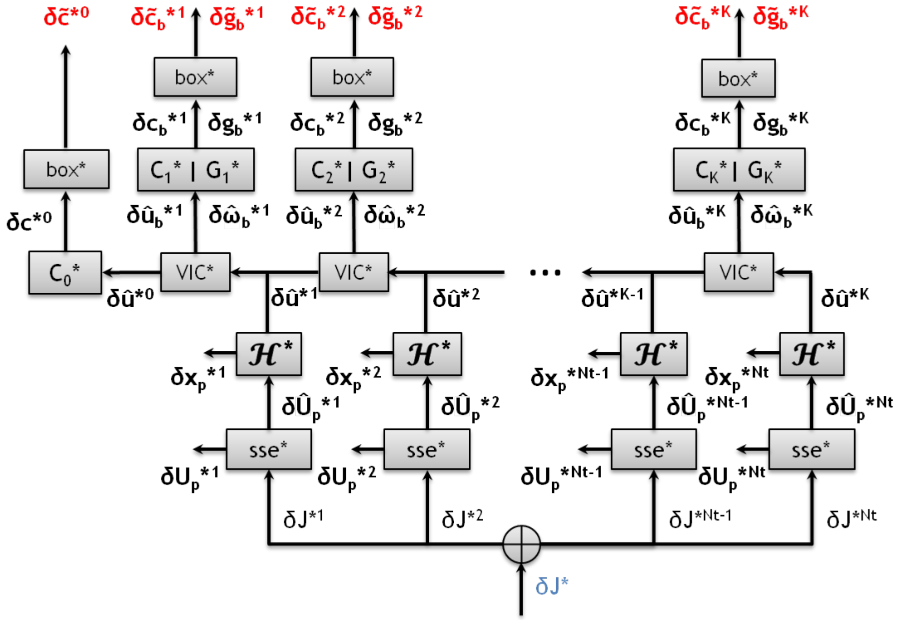


Figure C.1: Adjoint of the flow diagram from Figure 5.2.

## C.1. SSE$^*$

The ‘sse$^*$’ block calculates the sum of squared errors. We have:

$$\delta\hat{\mathbf{U}}^{*l}_{p_j} = -\frac{2}{\sigma^{l\,2}_{j}}\left(\mathbf{U}^{l}_{p_j} - \hat{\mathbf{U}}^{l}_{p_j}\right), \tag{C.4}$$

Eq.C.4 follows from Eqs.5.8-5.10 and Eq.C.3. We do not calculate $\delta\mathbf{U}^{*l}_{p_j}$, i.e. the other term that comes out of the ‘sse’ block, since it does not contribute to the top variables of the flow diagram in Figure C.1, which form the components of the gradient $\nabla J$ we want to calculate.

## C.2. $\mathcal{H}^*$

The adjoint operator of $\mathcal{H}$ (see also Eq.5.11) that applies trilinear interpolation of the velocity from the regular grid to a particle position is given by:

$$\delta\hat{\mathbf{u}}^{*l}_{g_i} = \sum_{j\in L}\frac{V^{l}_{ij}}{V}\delta\hat{\mathbf{U}}^{*l}_{p_j}. \tag{C.5}$$

To explain the components in Eq.C.5, we use Figure C.2. The black dot is the point on the regular grid $\mathbf{x}_{g_i}$ for which we want to calculate the adjoint velocity $\delta\hat{\mathbf{u}}^{*l}_{g_i}$. We take the eight cells surrounding this grid point and look for the particle locations (red dots)

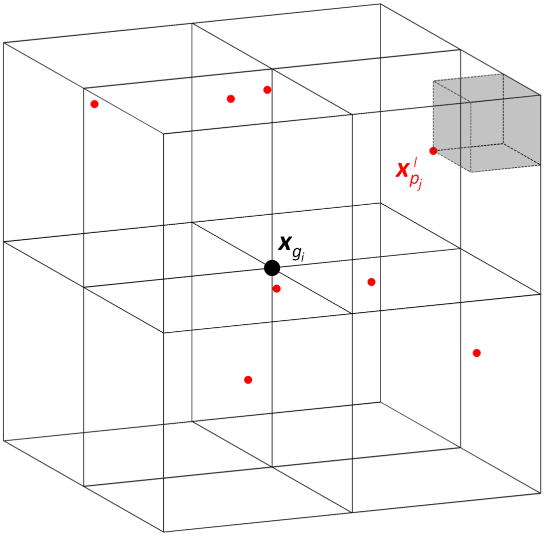


Figure C.2: Workings of the adjoint operator of trilinear interpolation.

they contain, forming a set $L$. Figure C.2 explains for the point $\mathbf{x}_{p_j}^{l}$ what the volume $V_{ij}$ represents, namely the volume formed between $\mathbf{x}_{p_j}^{l}$ and the point opposite $\mathbf{x}_{g_i}$ across the diagonal of the cell. If the eight surrounding cells would not contain any particles, we would have $\delta\hat{\mathbf{u}}_{g_i}^{*l} = 0$.

We could have followed the simple yet inefficient approach of constructing the matrix that represents the operator $\mathcal{H}$ and then form its transpose. Instead, using Eq.C.5 is much more efficient and in addition illustrates the physical meaning of the adjoint. Where the forward tangent-linear represents how disturbances from the grid points propagate into the interpolation at a particle location, the adjoint represents the particle locations that a grid point has affected.

## C.3. VIC*

This section explains the adjoint code of the VIC method. Like Chapter 5, we distinguish the four steps *blob advection, vortex stretching, vorticity interpolation* and *velocity inversion*. However, we now traverse the components in reverse order and replace the variables by their adjoint counterparts, as shown by Figure C.3.

### C.3.1. VELOCITY INVERSION

The original velocity inversion step was expressed by Eq.5.27. The tangent-linear version (see Eq.5.40) was simple since the Laplace and curl operators are linear. To explain the adjoint, we split the step into two sub-steps, starting with the Poisson equation and followed by the curl operation.

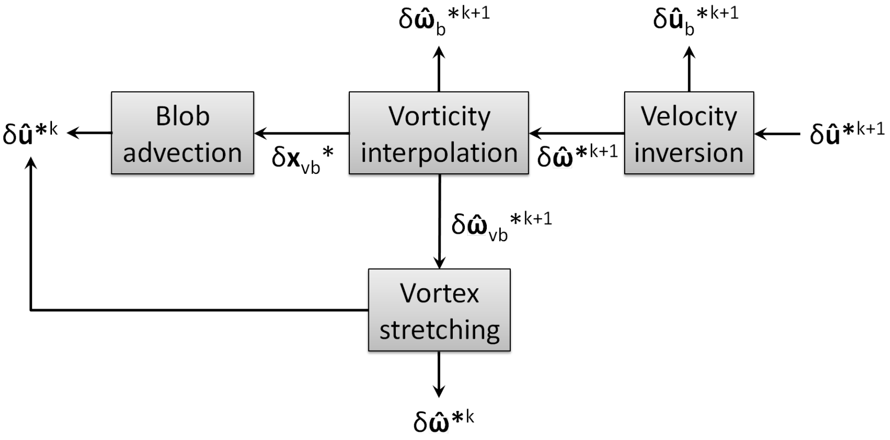


Figure C.3: One time integration step of the *adjoint* VIC solver.

### POISSON EQUATION

The tangent-linear of the Poisson equation can be expressed as:

$$A\delta\hat{\mathbf{u}}_{\text{int}}^{k+1} = \delta\hat{\mathbf{f}}^{k+1}, \tag{C.6}$$

where $A$ is the discrete Laplacian operator, $\delta\hat{\mathbf{u}}_{\text{int}}^{k+1}$ the disturbance of the internal points of the new velocity field and $\delta\hat{\mathbf{f}}^{k+1}$ the disturbance of the contributions from the curl of the vorticity field and the velocity boundary condition:

$$\begin{aligned}\delta\hat{\mathbf{f}}^{k+1} &= \delta\hat{\mathbf{g}}^{k+1} + \delta\hat{\mathbf{h}}^{k+1}\\ &= I\delta\hat{\mathbf{g}}^{k+1} + I\delta\hat{\mathbf{h}}^{k+1}\\ &= \begin{bmatrix} I & I \end{bmatrix}\begin{pmatrix}\delta\hat{\mathbf{g}}^{k+1}\\ \delta\hat{\mathbf{h}}^{k+1}\end{pmatrix},\end{aligned} \tag{C.7}$$

where $I$ is the identity matrix, $\delta\hat{\mathbf{g}}^{k+1}$ is the discretization of $-\nabla\times\boldsymbol{\omega}$ at the internal grid points and $\delta\hat{\mathbf{h}}^{k+1}$ is the contribution from the velocity boundary condition. The adjoint counterpart of Eq.C.6 is:

$$A\delta\hat{\mathbf{f}}^{*k+1} = \delta\hat{\mathbf{u}}_{\text{int}}^{*k+1}, \tag{C.8}$$

where we have used the property that the Laplacian operator is self-adjoint, i.e. $A = A'$;. Solving Eq.C.8, we obtain $\delta\hat{\mathbf{f}}^{*k+1}$. The adjoint of Eq.C.7 is:

$$\begin{pmatrix}\delta\hat{\mathbf{g}}^{*k+1}\\ \delta\hat{\mathbf{h}}^{*k+1}\end{pmatrix} = \begin{bmatrix} I\\ I \end{bmatrix}\delta\hat{\mathbf{f}}^{*k+1}, \tag{C.9}$$

so

$$\delta\hat{\mathbf{g}}^{*k+1} = \delta\hat{\mathbf{h}}^{*k+1} = \delta\hat{\mathbf{f}}^{*k+1}$$

Finally, the adjoint of the velocity boundary condition $\delta\hat{\mathbf{u}}_b^{*k+1}$ is obtained by adding the boundary points from $\delta\hat{\mathbf{u}}^{*k+1}$, denoted $\delta\hat{\mathbf{u}}_{\text{bnd}}^{*k+1}$ and $\delta\hat{\mathbf{h}}^{*k+1}$.

### CURL

To explain how $\delta\hat{\boldsymbol{\omega}}^{*k+1}$ is obtained from $\delta\hat{\mathbf{g}}^{*k+1}$, we investigate the continuous counterpart first to enable a clear interpretation. Consider

$$\mathbf{g} = -\nabla \times \boldsymbol{\omega}, \tag{C.10}$$

which we write for the present purposes as

$$\begin{pmatrix} g_x \\ g_y \\ g_z \end{pmatrix} = -\begin{bmatrix} 0 & -\frac{\partial\cdot}{\partial z} & \frac{\partial\cdot}{\partial y} \\ \frac{\partial\cdot}{\partial z} & 0 & -\frac{\partial\cdot}{\partial x} \\ -\frac{\partial\cdot}{\partial y} & \frac{\partial\cdot}{\partial x} & 0 \end{bmatrix} \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix}. \tag{C.11}$$

Taking the adjoint of Eq.C.11,

$$\begin{pmatrix} \delta\omega_x^* \\ \delta\omega_y^* \\ \delta\omega_z^* \end{pmatrix} = -\begin{bmatrix} 0 & \frac{\partial\cdot^*}{\partial z} & -\frac{\partial\cdot^*}{\partial y} \\ -\frac{\partial\cdot^*}{\partial z} & 0 & \frac{\partial\cdot^*}{\partial x} \\ \frac{\partial\cdot^*}{\partial y} & -\frac{\partial\cdot^*}{\partial x} & 0 \end{bmatrix} \begin{pmatrix} \delta g_x^* \\ \delta g_y^* \\ \delta g_z^* \end{pmatrix}. \tag{C.12}$$

Since the first derivative operator is skew-symmetric (e.g. $\partial\cdot^* /\partial x = -\partial\cdot /\partial x$), Eq.C.12 becomes

$$\begin{pmatrix} \delta\omega_x^* \\ \delta\omega_y^* \\ \delta\omega_z^* \end{pmatrix} = -\begin{bmatrix} 0 & -\frac{\partial\cdot}{\partial z} & \frac{\partial\cdot}{\partial y} \\ \frac{\partial\cdot}{\partial z} & 0 & -\frac{\partial\cdot}{\partial x} \\ -\frac{\partial\cdot}{\partial y} & \frac{\partial\cdot}{\partial x} & 0 \end{bmatrix} \begin{pmatrix} \delta g_x^* \\ \delta g_y^* \\ \delta g_z^* \end{pmatrix}. \tag{C.13}$$

## C.3.2. VORTICITY INTERPOLATION

Referring back to the flow diagram in Figure C.3, we see that the adjoint variable $\delta\hat{\boldsymbol{\omega}}^{*k+1}$ goes in and the three adjoint variables $\delta\hat{\boldsymbol{\omega}}_b^{*k+1}$, $\delta\hat{\boldsymbol{\omega}}_{vb}^{*k+1}$, and $\delta\mathbf{x}_{vb}^*$ come out. $\delta\hat{\boldsymbol{\omega}}_b^{*k+1}$ is obtained by simply extracting the boundary nodes from $\delta\hat{\boldsymbol{\omega}}^{*k+1}$. This is because in the forward mode, the vorticity boundary condition is overlaid *after* interpolating the vorticity from the vortex blobs to the regular grid. Proceeding to $\delta\hat{\boldsymbol{\omega}}_{vb}^{*k+1}$, from the forward code explained in Section 5.4.3, we know that the new vorticity field on the regular grid is obtained by interpolating the vorticity field formed by the vortex blobs, see also Eq.5.23. Similar to the adjoint of trilinear interpolation, discussed in Section C.2, the adjoint of the vorticity at the vortex blobs is obtained by interpolating the adjoint of the vorticity field from the regular grid:

$$\delta\hat{\boldsymbol{\omega}}_{vb,j}^{*k+1} = \frac{1}{V}\sum_{i=1}^{n} \delta\hat{\boldsymbol{\omega}}_i^{*k+1} \prod_{l=1}^{3} \zeta_l\left(\frac{x_{l,vb,j} - x_{l,g,i}}{\Delta x_l}\right), \tag{C.14}$$

Finally, we provide the expression for $\delta\mathbf{x}_{vb}^*$:

$$\delta x_{vb,j}^* = \hat{\omega}_{x,vb,j} \sum_{i=1}^{n} \delta\hat{\omega}_{x,i}^{*k+1} \frac{\delta\zeta_{1,ji}}{\delta x} \zeta_{2,ji}\zeta_{3,ji}. \tag{C.15}$$

Similar expressions are obtained for $\delta y_{vb,j}^*$ and $\delta z_{vb,j}^*$.

### C.3.3. VORTEX STRETCHING

Two variables flow out of the vortex stretching block, namely $\delta\hat{\boldsymbol{\omega}}^{*k}$ and $\delta\hat{\mathbf{u}}^{*k}$. Following the tangent-linear vortex stretching term from Eq.5.22, it can be shown that we get:

$$\delta\hat{\boldsymbol{\omega}}^{*k} = \delta\hat{\boldsymbol{\omega}}_{vb}^{*k+1} + \Delta t \nabla\hat{\mathbf{u}}^{k}\delta\hat{\boldsymbol{\omega}}_{vb}^{*k+1}, \tag{C.16}$$

$$\nabla\delta\hat{\mathbf{u}}^{*k} = \Delta t\hat{\boldsymbol{\omega}}^{k}\left(\delta\hat{\boldsymbol{\omega}}_{vb}^{*k+1}\right)'. \tag{C.17}$$

The adjoint of the velocity can be obtained from Eq.C.17. We have for the velocity in the $i$-direction:

$$\delta u_i^* = -\nabla\cdot\left(\nabla\delta u_i^*\right). \tag{C.18}$$

### C.3.4. BLOB ADVECTION

Finally, the adjoint of the velocity is obtained by adding Eq.C.18 to $\Delta t \cdot \delta\mathbf{x}_{vb}^*$.

## C.4. $C_k^*, G_k^*$

The matrices $C_k$ and $G_k$ are square symmetric matrices and therefore self-adjoint. Therefore

$$\delta\mathbf{c}^{*0} = C_0\delta\hat{\mathbf{u}}^{*0},$$
$$\delta\mathbf{c}_b^{*k} = C_k\delta\hat{\mathbf{u}}_b^{*k},$$
$$\delta\mathbf{g}_b^{*k} = G_k\delta\hat{\boldsymbol{\omega}}_b^{*k},$$

## C.5. BOX$^*$

The original expression for the constraint handling comes from Eq.5.13. Its adjoint is:

$$\delta\tilde{\phi}^* = \frac{1}{2}\left(\phi_{ub} - \phi_{lb}\right)\cos\tilde{\phi}\delta\phi^*. \tag{C.19}$$

Using Eq.C.19, we can obtain $\delta\tilde{\mathbf{c}}^{*0}$ from $\delta\mathbf{c}^{*0}$, $\delta\tilde{\mathbf{c}}_b^{*k}$ from $\delta\mathbf{c}_b^{*k}$ and $\delta\tilde{\mathbf{g}}_b^{*k}$ from $\delta\mathbf{g}_b^{*k}$. These are the components of the gradient of $J$.

# BIBLIOGRAPHY

Agüí JC, Jimenez J (1987) On the performance of particle tracking. J Fluid Mech 185(1):447–468

Ammar GS, Gragg WB (1988) Superfast solution of real positive definite Toeplitz systems. SIAM J Matrix Anal & Appl 9(1):61–76

Anderson BD, Moore JB (2012) Optimal filtering. Courier Corporation

Anjyo K, Lewis J (2011) RBF interpolation and Gaussian process regression through an RKHS formulation. J Math-for-Ind 3(6):63–71

Azijli I, Dwight RP (2015) Solenoidal filtering of volumetric velocity measurements using Gaussian process regression. Exp Fluids 56(11)

Azijli I, Sciacchitano A, Ragni D, Palha A, Dwight RP (2016) *A posteriori* uncertainty quantification of PIV-based pressure data. Exp Fluids 57(5)

de Baar JHS (2014) Stochastic surrogates for measurements and computer models of fluids. PhD thesis, Delft University of Technology

de Baar JHS, Dwight RP, Bijl H (2013) Speeding up Kriging through fast estimation of the hyperparameters in the frequency-domain. Comput Geosci 54:99–106

de Baar JHS, Percin M, Dwight RP, van Oudheusden BW, Bijl H (2014) Kriging regression of PIV data using a local error estimate. Exp Fluids 55(1):1–13

Battle G, Federbush P (1993) Divergence-free vector wavelets. Mich Math J 40(1):81–195

Bauer F, Lukas MA (2011) Comparing parameter choice methods for regularization of ill-posed problems. Math Comput Simulat 81(9):1795–1841

Baur T, Köngeter J (1999) PIV with high temporal resolution for the determination of local pressure reductions from coherent turbulence phenomena. In: 3rd Int Workshop on PIV

Bayes T (1763) An essay towards solving a problem in the doctrine of chances. Philos T Roy Soc 53:370–418

Bohrnstedt GW, Goldberger AS (1969) On the exact covariance of products of random variables. J Am Stat Assoc 64(328):1439–1442

Brown D, Alexander N (1991) The analysis of the variance and covariance of products. Biometrics pp 429–444

Brücker C (1995) Digital-particle-image-velocimetry (DPIV) in a scanning light-sheet: 3D starting flow around a short cylinder. Exp Fluids 19(4):255–263

Busch J, Giese D, Wissmann L, Kozerke S (2013) Reconstruction of divergence-free velocity fields from cine 3D phase-contrast flow measurements. Magnet Reson Med 69(1):200–210

Capizzano SS (2002) Matrix algebra preconditioners for multilevel Toeplitz matrices are not superlinear. Linear Algebra Appl 343:303–319

Casa LDC, Krueger PS (2013) Radial basis function interpolation of unstructured, three-dimensional, volumetric particle tracking velocimetry data. Meas Sci Technol 24(6):065,304

Chan RH, Jin XQ (2007) An introduction to iterative Toeplitz solvers, vol 5. SIAM

Charonko J, Vlachos P (2013) Estimation of uncertainty bounds for individual particle image velocimetry measurements from cross-correlation peak ratio. Meas Sci Technol 24(6):065,301

Charonko JJ, King CV, Smith BL, Vlachos PP (2010) Assessment of pressure field calculations from particle image velocimetry measurements. Meas Sci Technol 21(10)

Chernih A, Sloan I, Womersley R (2014) Wendland functions with increasing smoothness converge to a Gaussian. Adv Comput Math 40(1):185–200

Chorin AJ, Marsden JE, Marsden JE (1990) A mathematical introduction to fluid mechanics, vol 3. Springer

Coleman HW, Steele WG (2009) Experimentation, validation, and uncertainty analysis for engineers. John Wiley & Sons

Cottet GH, Koumoutsakos PD (2000) Vortex methods: theory and practice. Cambridge university press

Cottet GH, Poncet P (2004) Advances in direct numerical simulations of 3D wall-bounded flows by vortex-in-cell methods. J Comput Phys 193(1):136–158

Cressie N (2015) Statistics for spatial data. John Wiley & Sons

Davis GJ, Morris MD (1997) Six factors which affect the condition number of matrices associated with Kriging. Math Geol 29(5):669–683

Deriaz E, Perrier V (2006) Divergence-free and curl-free wavelets in two dimensions and three dimensions: application to turbulent flows. J Turbul (7)

Elkins C, Markl M, Pelc N, Eaton J (2003) 4D magnetic resonance velocimetry for mean velocity measurements in complex turbulent flows. Exp Fluids 34(4):494–503

Elsinga G, Scarano F, Wieneke B, van Oudheusden B (2006) Tomographic particle image velocimetry. Exp Fluids 41(6):933–947

Evensen G (1994) Sequential data assimilation with a non-linear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. J Geophys Res 99(5)

Evensen G (2003) The ensemble Kalman filter: Theoretical formulation and practical implementation. Ocean Dyn 53(4):343–367

Evensen G (2009) Data assimilation: the ensemble Kalman filter. Springer Science & Business Media

Fasshauer GE (2011) Positive definite kernels: past, present and future. DRNA 4:21–63

Fornberg B, Wright G (2004) Stable computation of multiquadric interpolants for all values of the shape parameter. Comput Math Appl 48(5):853–867

Forrester A, Sobester A, Keane A (2008) Engineering design via surrogate modelling: a practical guide. John Wiley & Sons

Ghaemi S, Ragni D, Scarano F (2012) PIV-based pressure fluctuations in the turbulent boundary layer. Exp Fluids 53(6):1823–1840

Gibbs BP (2011) Advanced Kalman filtering, least-squares and modeling: a practical handbook. John Wiley & Sons

Giering R, Kaminski T (1998) Recipes for adjoint code construction. ACM Trans Math Softw (TOMS) 24(4):437–474

Griewank A, Walther A (2000) Algorithm 799: revolve: an implementation of checkpointing for the reverse or adjoint mode of computational differentiation. ACM Trans Math Softw (TOMS) 26(1):19–45

Gunes H, Sirisup S, Karniadakis GE (2006) Gappy data: To Krig or not to Krig? J Comput Phys 212(1):358–382

Gurka R, Liberzon A, Hefetz D, Rubinstein D, Shavit U (1999) Computation of pressure distribution using PIV velocity data. In: 3rd Int Workshop on PIV

Haigermoser C (2009) Application of an acoustic analogy to PIV data from rectangular cavity flows. Exp Fluids 47(1):145–157

Heitz D, Mémin E, Schnörr C (2010) Variational fluid flow measurements from image sequences: synopsis and perspectives. Exp Fluids 48(3):369–393

Hinsch K (2002) Holographic particle image velocimetry. Meas Sci Technol 13(7):R61–R72

Houtekamer PL, Mitchell HL (1998) Data assimilation using an ensemble Kalman filter technique. Mon Weather Rev 126(3):796–811

Inggs MR, Lord RT (1996) Interpolating satellite derived wind field data using Ordinary Kriging, with application to the nadir gap. IEEE Trans Geosci Remote Sens 34(1):250–256

Jeon YJ, Chatellier L, David L (2014) Fluid trajectory evaluation based on an ensemble-averaged cross-correlation in time-resolved PIV. Exp Fluids 55(7):1–16

Julier SJ, Uhlmann JK (1997) New extension of the Kalman filter to nonlinear systems. In: AeroSense'97, International Society for Optics and Photonics, pp 182–193

Kailath T, Kung SY, Morf M (1979) Displacement ranks of a matrix. B Am Math Soc 1(5):769–773

Kalman RE (1960) A new approach to linear filtering and prediction problems. Journal Fluid Eng 82(1):35–45

Kaplanski F, Sazhin SS, Fukumoto Y, Heikal SBM (2009) A generalized vortex ring model. J Fluid Mech 622:233–258

de Kat R, van Oudheusden BW (2010) Instantaneous planar pressure from PIV: analytic and experimental test-cases. In: Proceedings of the 15th international symposium on applications of laser techniques to fluid mechanics, Lisbon, Portugal

Ko J, Kurdila AJ, Rediniotis OK (2000) Divergence-free bases and multiresolution methods for reduced-order flow modeling. AIAA J 38(12):2219–2232

Koschatzky V, Moore P, Westerweel J, Scarano F, Boersma B (2011) High speed PIV applied to aerodynamic noise investigation. Exp Fluids 50(4):863–876

Krige D (1951) A statistical approach to some basic mine valuation problems on the witwatersrand

Lee SL, Huntbatch A, Yang GZ (2008) Contractile analysis with Kriging based on MR myocardial velocity imaging. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2008, Springer, pp 892–899

Liburdy JA, Young EF (1992) Processing of three-dimensional particle tracking velocimetry data. Opt Laser Eng 17(3):209–227

Liu DC, Nocedal J (1989) On the limited memory BFGS method for large scale optimization. Math Program 45(1-3):503–528

Liu X, Katz J (2006) Instantaneous pressure and material acceleration measurements using a four-exposure PIV system. Exp Fluids 41(2):227–240

Lowitzsch S (2005) A density theorem for matrix-valued radial basis functions. Numer Algorithms 39(1-3):253–256

Lynch KP (2015) Advances in time-resolved tomographic PIV. PhD thesis, Delft University of Technology

Lynch KP, Scarano F (2013) A high-order time-accurate interrogation method for time-resolved PIV. Meas Sci Technol 24(3):035,305

Lynch KP, Scarano F (2015) An efficient and accurate approach to MTE-MART for time-resolved tomographic PIV. Exp Fluids 56(3):1–16

Ma Z, Chew W, Jiang L (2013) A novel fast solver for Poisson's equation with Neumann boundary condition. Progr in Electromagnetics Res 136:195–209

Maas H, Gruen A, Papantoniou D (1993) Particle tracking velocimetry in three-dimensional flows. Exp Fluids 15(2):133–146

Malik NA, Dracos T (1995) Interpolation schemes for three-dimensional velocity fields from scattered data using Taylor expansions. J Comput Phys 119(2):231–243

Matheron G (1971) The theory of regionalized variables and its applications, vol 5. École national supérieure des mines

Metropolis N, Ulam S (1949) The monte carlo method. J Am Stat Assoc 44(247):335–341

Miller RN, Ghil M, Gauthiez F (1994) Advanced data assimilation in strongly nonlinear dynamical systems. J Atmos Sci 51(8):1037–1056

Monaghan J (1985) Extrapolating B splines for interpolation. J Comput Phys 60(2):253–262

Murai Y, Nakada T, Suzuki T, Yamamoto F (2007) Particle tracking velocimetry applied to estimate the pressure field around a Savonius turbine. Meas Sci Technol 18(8):2491

Narcowich FJ, Ward JD (1994) Generalized Hermite interpolation via matrix-valued conditionally positive definite functions. Math Comput 63(208):661–687

Neeteson NJ, Rival DE (2015) Pressure-field extraction on unstructured flow data using a Voronoi tessellation-based networking algorithm: a proof-of-principle study. Exp Fluids 56(2):1–13

Ng MK, Pan J (2010) Approximate inverse circulant-plus-diagonal preconditioners for Toeplitz-plus-diagonal matrices. SIAM J Sci Comput 32(3):1442–1464

Novara M (2013) Advances in tomographic PIV. PhD thesis, Delft University of Technology

Novara M, Batenburg KJ, Scarano F (2010) Motion tracking-enhanced MART for tomographic PIV. Meas Sci Technol 21(3):035,401

Novara M, Schanz D, Kähler CJ, Schröder A (2015) Shake-The-Box for multi-pulse tomographic systems: towards high seeding density particle tracking in high speed flows. In: 11th Int Symp on PIV - PIV15

van Oudheusden B (2013) PIV-based pressure measurement. Meas Sci Technol 24(3):1–32

Paige CC, Saunders MA (1975) Solution of sparse indefinite systems of linear equations. SIAM J Numer Anal 12(4):617–629

Paige CC, Saunders MA (1982) Lsqr: An algorithm for sparse linear equations and sparse least squares. ACM T Math Software 8(1):43–71

Panton R (2013) Incompressible flow. John Wiley & Sons

Pröbsting S, Scarano F, Bernardini M, Pirozzoli S (2013) On the estimation of wall pressure coherence using time-resolved tomographic PIV. Exp Fluids 54(7):1–15

Raffel M, Willert C, Kompenhans J (1998) Particle Image Velocimetry: A Practical Guide; with 24 Tables. Springer

Ragni D, Van Oudheusden B, Scarano F (2011) Non-intrusive aerodynamic loads analysis of an aircraft propeller blade. Exp Fluids 51(2):361–371

Raiola M, Discetti S, Ianiro A (2015) On PIV random error minimization with optimal POD-based low-order reconstruction. Exp Fluids 56(4):1–15

Rasmussen CE (2006) Gaussian processes for machine learning

Reichle RH, McLaughlin DB, Entekhabi D (2002) Hydrologic data assimilation with the ensemble Kalman filter. Mon Weather Rev 130(1):103–114

Robinson AR, Lermusiaux PF (2000) Overview of data assimilation. Harvard reports in physical/interdisciplinary ocean science 62:1–13

Sadati M, Luap C, Kröger M, Öttinger HC (2011) Hard vs soft constraints in the full field reconstruction of incompressible flow kinematics from noisy scattered velocimetry data. J Rheol 55(6):1187–1203

Scarano F (2013) Tomographic PIV: principles and practice. Meas Sci Technol 24(1):1–28

Scarano F, Moore P (2012) An advection-based model to increase the temporal resolution of PIV time series. Exp Fluids 52(4):919–933

Scarano F, Poelma C (2009) Three-dimensional vorticity patterns of cylinder wakes. Exp Fluids 47(1):69–83

Scarano F, Ghaemi S, Caridi G, Bosbach J, Dierksheide U, Sciacchitano A (2015) On the use of helium-filled soap bubbles for large-scale tomographic PIV in wind tunnel experiments. Exp Fluids 56(2)

Schaback R (1995) Error estimates and condition numbers for radial basis function interpolation. Adv Comput Math 3(3):251–264

Schanz D, Schröder A, Gesemann S, Michaelis D, Wieneke B (2013) ‘Shake The Box’: A highly efficient and accurate Tomographic Particle Tracking Velocimetry (TOMO-PTV) method using prediction of particle positions. In: 10th Int Symp on PIV

Schanz D, Schröder A, Gesemann S (2014) ‘Shake The Box’ - a 4D PTV algorithm: accurate and ghostless reconstruction of lagrangian tracks in densely seeded flows. In: 17th International Symposium on Applications of Laser Techniques to Fluid Mechanics

Scheuerer M, Schlather M (2012) Covariance models for divergence-free and curl-free random vector fields. Stoch Models 28(3):433–451

Schiavazzi D, Coletti F, Iaccarino G, Eaton JK (2014) A matching pursuit approach to solenoidal filtering of three-dimensional velocity measurements. J Comput Phys 263(0):206–221

Schneiders JFG, Dwight RP, Scarano F (2014) Time-supersampling of 3D-PIV measurements with vortex-in-cell simulation. Exp Fluids 55(3):1–15

Schneiders JFG, Azijli I, Scarano F, Dwight RP (2015a) Pouring time into space. In: 11th Int Symp on PIV - PIV15

Schneiders JFG, Dwight RP, Scarano F (2015b) Tomographic PIV noise reduction by simulating repeated measurements. In: 11th Int Symp on PIV - PIV15

Schräder D, Wendland H (2011) A high-order, analytically divergence-free discretization method for Darcy's problem. Math Comput 80(273):263–277

Sciacchitano A (2015) *A posteriori* uncertainty quantification for tomographic PIV data. In: 11th Int Symp on PIV - PIV15

Sciacchitano A, Dwight RP, Scarano F (2012a) Navier-Stokes simulations in gappy PIV data. Exp Fluids 53(5):1421–1435

Sciacchitano A, Scarano F, Wieneke B (2012b) Multi-frame pyramid correlation for time-resolved PIV. Exp Fluids 53(4):1087–1105

Sciacchitano A, Wieneke B, Scarano F (2013) PIV uncertainty quantification by image matching. Meas Sci Technol 24(4):045,302

Sciacchitano A, Neal DR, Smith BL, Warner SO, Vlachos PP, Wieneke B, Scarano F (2015) Collaborative framework for piv uncertainty quantification: comparative assessment of methods. Meas Sci Technol 26(7):074,004

Segel L (2007) Mathematics Applied to Continuum Mechanics. SIAM

de Silva CM, Philip J, Marusic I (2013) Minimization of divergence error in volumetric velocity measurements and implications for turbulence statistics. Exp Fluids 54(7):1–17

Song SM, Napel S, Glover GH, Pelc NJ (1993) Noise reduction in three-dimensional phase-contrast MR velocity measurementsl. J Magn Reson Im 3(4):587–596

Suthaharan S (2015) Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning, vol 36. Springer

Suzuki T, Ji H, Yamamoto F (2009) Unsteady PTV velocity field past an airfoil solved with DNS: Part 1. algorithm of hybrid simulation and hybrid velocity field at Re$\approx 10^3$. Exp Fluids 47(6):957–976

Talagrand O (1997) Assimilation of observations, an introduction. J Meteorol Soc Jpn 2 75:81–99

Talagrand O, Courtier P (1987) Variational assimilation of meteorological observations with the adjoint vorticity equation. I: Theory. Q J Roy Meteor Soc 113(478):1311–1328

Timmins B, Wilson B, Smith B, Vlachos P (2012) A method for automatic estimation of instantaneous local uncertainty in particle image velocimetry measurements. Exp Fluids 53(4):1133–1147

Tropea C, Yarin AL, Foss JF (2007) Springer handbook of experimental fluid mechanics, vol 1. Springer Science & Business Media

Urban K (1996) Using divergence free wavelets for the numerical solution of the Stokes problem. Citeseer

Vennell R, Beatson R (2009) A divergence-free spatial interpolator for large sparse velocity data sets. J Geophys Res-Oceans 114(C10)

Venturi D, Karniadakis GE (2004) Gappy data and reconstruction procedures for flow past a cylinder. J Fluid Mech 519:315–336

Viana FA, Haftka RT, Steffen Jr V (2009) Multiple surrogates: how cross-validation errors can help us to obtain the best predictor. Struct Multidiscip O 39(4):439–457

Violato D (2013) 3D flow organization and dynamics in subsonic jets: aeroacoustic source analysis by tomographic PIV. PhD thesis, Delft University of Technology

Violato D, Scarano F (2011) Three-dimensional evolution of flow structures in transitional circular and chevron jets. Phys Fluids 23(12):124,104

Violato D, Moore P, Scarano F (2011) Lagrangian and Eulerian pressure field evaluation of rod-airfoil flow from time-resolved tomographic PIV. Exp Fluids 50(4):1057–1070

Wendland H (2005) Scattered data approximation, vol 17. Cambridge University Press Cambridge

Wendland H (2009) Divergence-free kernel methods for approximating the Stokes problem. SIAM J Numer Anal 47(4):3158–3179

Wieneke B (2013) Iterative reconstruction of volumetric particle distribution. Meas Sci Technol 24(2):024,008

Wieneke B, Prevost R (2014) DIC uncertainty estimation from statistical analysis of correlation values. In: Advancement of Optical Methods in Experimental Mechanics, Volume 3, pp 125–136

Wieneke B, Sciacchitano A (2015) PIV uncertainty propagation. In: 11th Int Symp on PIV - PIV15

Wikle C, Berliner M (2007) A Bayesian tutorial for data assimilation. Physica D 230(1):1–16

Worth N (2012) Measurement of three-dimensional coherent fluid structure in high Reynolds number turbulent boundary layers. PhD thesis, University of Cambridge

Wright SJ, Nocedal J (1999) Numerical optimization, vol 2. Springer New York

Yang GZ, Kilner PJ, Firmin DN, Underwood SR, Burger P, Longmore DB (1993) 3D cine velocity reconstruction using the method of convex projections. In: Computers in Cardiology 1993, Proceedings., IEEE, pp 361–364

Zhang J, Tao B, Katz J (1997) Turbulent flow measurement in a square duct with hybrid holographic PIV. Exp Fluids 23(5):373–381

Zhong J, Weng JJ, Huang TS (1992) Vector field interpolation using robust statistics. In: Robotics-DL tentative, International Society for Optics and Photonics, pp 58–67

# LIST OF PUBLICATIONS

**Journal papers**

- **I. Azijli**, A. Sciacchitano, D. Ragni, A. Palha and R.P. Dwight, *A posteriori uncertainty quantification of PIV-based pressure data*, Exp Fluids 57:5 (2016).
- **I. Azijli** and R.P. Dwight, *Solenoidal filtering of volumetric velocity measurements using Gaussian process regression*, Exp Fluids 56:198 (2015).

**Conference papers**

- **I. Azijli**, A. Sciacchitano, D. Ragni, A. Palha, and R.P. Dwight, *A posteriori uncertainty quantification of PIV-derived pressure fields*, 11th International Symposium on Particle Image Velocimetry, September 14-16, 2015, Santa Barbara, CA, USA.
- J.F.G. Schneiders, **I. Azijli**, F. Scarano, and R.P. Dwight, *Pouring time into space*, 11th International Symposium on Particle Image Velocimetry, September 14-16, 2015, Santa Barbara, CA, USA.
- **I. Azijli** and R.P. Dwight, *Vibro-acoustic properties of a gas-enclosed disk under high-pressure, high-density conditions: an experimental and numerical parametric study*, International Conference on Structural Engineering Dynamics, ICEDyn 2015, June 22-24, 2015, Lagos, Algarve, Portugal.
- **I. Azijli**, R.P. Dwight and H. Bijl, *A Bayesian approach to physics-based reconstruction of incompressible flows*, Numerical Mathematics and Advanced Applications – ENUMATH 2013, Lecture Notes in Computational Science and Engineering 103, pp.529-536 (2015).

**Book chapters**

- R.P. Dwight, J.H.S. de Baar and **I. Azijli**, *Surrogate modeling and adaptive sampling techniques*, VKI lecture notes: Introduction to optimization and multidisciplinary design in aeronautics and turbomachinery, von Karman Institute, Rhode-St-Genese, Belgium, May 7-11, 2012.

# ABOUT THE AUTHOR

Iliass Azijli was born on August 17, 1987 in Amsterdam, The Netherlands. Growing up, he was always interested in cars and planes; drawing them, constructing them and then breaking them. At elementary school, he already knew that he wanted to study Aerospace Engineering at TU Delft. When he was at high school, he had already narrowed it down to Aerodynamics. Specifically, his dream was to design fighter planes. By the time he had reached the final year of high school though, reality set in…he came to realize the death and destruction that these machines bring with them. Fortunately, a documentary came along that steered him into the fast lane: *The Secret Life of Formula One*. He couldn't believe what he was seeing. Upside down fighter jets bringing joy and excitement to cheering crowds, a sport where the true heroes are the engineers.

When he finally started the Aerospace Engineering BSc program at TU Delft in 2005, he was all fired up. In 2008, he graduated with distinction and then started the Aerodynamics MSc program at the same university. In 2009, he got his first taste of Formula One when he started an internship at the Aerodynamics Department of the Renault F1 Team. His expectations didn't let him down, because he followed it with an internship at their CFD Department. It was during this time that he became interested in the field of data assimilation, because he literally could see the car being developed both in the wind tunnel and CFD. How do we combine the strengths of both worlds?

In 2011, he graduated with distinction and started his PhD at the Aerodynamics Group of the Aerospace Engineering Faculty at TU Delft under the supervision of Richard Dwight and Hester Bijl. The research topic was improving the understanding of vibro-acoustics in dense gas applications. In the meantime though, he found some time to investigate the use of data assimilation methods to enhance experiments in fluids, which is the subject of the present thesis.

As of December 2015, Iliass joined the Mercedes AMG F1 Team, hoping to contribute to their winning ways.