

INAUGURAL-DISSERTATION

ZUR
ERLANGUNG DER DOKTORWÜRDE
DER
NATURWISSENSCHAFTLICH-MATHEMATISCHEN GESAMTFAKULTÄT
DER
RUPRECHT-KARLS-UNIVERSITÄT
HEIDELBERG

vorgelegt von
Dipl.-Inf. Andrey Vlasenko
aus Sevastopol am Schwarz Meer

Tag der mündlichen Prüfung: 16 Juli, 2010

Physics-Based Fluid Flow Restoration Method

1. Gutachter: **Prof. Dr. Christoph Schnörr**
Image and Pattern Analysis Group
Heidelberg Collaboratory for Image Processing
Universitt Heidelberg
2. Gutachter: **Prof. Dr. Bernd Jähne**
Digital Image Processing Group
Heidelberg Collaboratory for Image Processing
Universitt Heidelberg

Zusammenfassung

Experimentelle Methoden und bildgebende Messverfahren zur Geschwindigkeitsmessung wie zum Beispiel Particle Image Velocimetry (PIV, etwa: Geschwindigkeitsmessung basierend auf Partikelbilder) und Particle Tracking Velocimetry (PTV, etwa: Geschwindigkeitsmessung basierend auf Partikelverfolgung) spielen in der Erforschung von Strömungen in Fluiden eine große Rolle. Sie sind sowohl für die Forschung als auch für eine große Reihe industrieller Anwendungen gleichbedeutend wichtig. Dennoch wird oft die geschätzte Geschwindigkeit von Fluiden durch Störungen, diversen Verfälschungen und fehlenden Fragmente beeinflusst, welches eine physikalische Interpretation der Werte sehr schwierig macht.

In der vorliegenden Arbeit wird ein neuer Algorithmus zur Rekonstruktion von Geschwindigkeitsfeldern in Fluiden vorgestellt. Der Algorithmus akzeptiert als Eingabe eine große Reihe an beschädigten zwei- oder dreidimensionalen Vektorfelder und erlaubt fehlende Fragmente wiederherzustellen und das Rauschen auf einem physikalisch plausiblen Weg zu entfernen. Das Verfahren nutzt im wesentlichen die physikalischen Eigenschaften von nicht komprimierbaren Fluiden aus und hängt nicht von einem bestimmten Rausch-Modell ab. Es besteht aus vier relativ einfachen Vorschriften. Davon basieren drei auf den Grundprinzipien der Kontinuumsmechanik wie die Kontinuitätsgleichung, die Momentenausgleichgleichung, sowie Ergebnisse der Turbulenztheorie, grundsätzlich das bergewicht an Niederfrequenzen in spektralen Bänder von Fluiden. Ein Ergebnis dieser physikalisch ausgerichteten Lösung ist, dass der entwickelte Algorithmus für verschiedene praxisrelevante Fehler und Störungen robust und effizient funktioniert.

Ein weiterer Aspekt der entwickelten Methode ist, dass experimentelle Daten in vielen Fällen Vektoren enthalten, welche in einem dreidimensionalen Volumen zufällig aber dünnbesetzt verteilt sind. Diese tauchen aufgrund technischer Anforderungen und Restriktionen der angewendeten Messmethoden zur Geschwindigkeitsschätzung von Partikelfelder auf. Das hier vorgestellte Verfahren wurde dementsprechend um einen hochauflösenden Ansatz erweitert um mit solchen Daten zurecht zu kommen. Die Methode akzeptiert beliebig beschädigte dünnbesetzte Vektorfelder als Eingangsdatensatz und rekonstruiert die fehlenden Teile des Flusses auf einer physikalisch konsistenten Art. Der Hochauflösungsansatz führt zu einer Wiederherstellung des Datensatzes in Form eines hochaufgelösten Vektorfeldes. Alle bedeutenden Aussagen werden anhand numerischer Experimente mit turbulenten Flussgeschwindigkeitsfelder bestätigt.

Das hier entwickelte Verfahren basiert auf einem Variationsansatz. Es wird in der Ausarbeitung gezeigt, dass man in der vorgeschlagene Methode zur diskreten Darstellung anhand verschiedener numerischen Techniken übergehen kann, z.B. anhand der Finite-Differenzen-Methode oder der Finite-Elemente-Methode.

Die vom Rekonstruktionsalgorithmus gelieferten Ergebnisse rechtfertigen die Annahme dass das vorgeschlagene Verfahren zum Entrauschen und Hochauflösen von Vektorfelder mit jeder Art von Störungen zurecht kommt.

Abstract

Experimental methods and image velocimetry like Particle Image Velocimetry (PIV) and Particle Tracking Velocimetry (PTV) play an important role in the investigation of fluid flows. They are equally important both for scientific research and for a large range of industrial applications. It is known, however, that estimating the velocity of fluid flows is susceptible to noise, various types of corruption and missing fragments, making physical interpretation of observational data quite problematic.

A new restoration algorithm specialized for fluid velocity vector fields is presented in this thesis. It handles a wide variety of corrupted two- and three-dimensional vector fields and allows the recovery of missing fragments and the removal of noise in a physically plausible way. The approach exploits essentially the physical properties of incompressible fluid flows and does not rely on any particular noise model. It consists of four relatively simple computational steps. Three of them are based on the main principles of the continuum mechanics such as continuity equation, momentum balance equation and some results of turbulence theory, basically the predominance of the low-frequency spectral band in fluid flows. As a result of this physically principled approach, the developed algorithm performs well, robustly and efficiently for different types of noise and errors occurring in practical velocimetry.

Another issue of the developed method is that the experimental data contains in many cases sparse vectors randomly distributed over a three-dimensional volume. They appear due to technical requirements and restrictions applied to the particle tracing velocimetry methods. The reconstruction algorithm presented here is extended by a super-resolution approach in order to cope with such situations. The method handles any corrupted sparse vector field and reconstructs the missing parts of the flow in a physically principled way. The super-resolution approach leads to the recovery of the data sets in a form of high-resolution vector field. All major conclusions are validated via numerical experiments with turbulent fluid flows.

The developed method is based on a variational approach. It is shown in the thesis that the suggested method can be easily discretized using various numerical techniques, i.e. finite differences or finite element method.

The analysis of the results produced by the reconstruction algorithm justify the fact that the presented methods, i.e. the denoising procedure and the super-resolution approach, can deal with any type of noise and corruptions, and do not depend on the flow regime.

Acknowledgements

The development of this thesis during my PhD period at the University of Heidelberg was a phased process, resulted in several papers. During my work I had long and helpful discussions with my colleagues and other people who had great impact on the content of this thesis and whom I would like to acknowledge. First of all, I would like to thank my supervisor Prof. Christoph Schnörr. Primarily his ideas initiated the work described in this thesis. I also would like to express my gratitude for supervising my dissertation and for giving me the opportunity to work in his group. Moreover, I am grateful to Prof. Bernd Jähne for serving as an external referee of this thesis.

I want to thank Stefania Petra for her moral support and participation in the development of the current dissertation. I also want to express my gratitude to Florian Backer for his valuable advices and fruitful discussions, especially concerning Cross-correlation methods in PIV.

I would like to thank my IPA colleagues Jörg Kappes, Dirk Breitenreicher, Jan Lellmann, Stefan Schmidt, Rezaul Karim, Thomas Schuüle and Stefan Weber for their participation in my research.

Especially I want to thank my parents who gave me valuable advices concerning writing of my thesis.

Contents

1	Introduction	5
1.1	Motivation	5
1.2	Related Work	6
1.2.1	Image Velocimetry Techniques	6
1.2.2	Non-Hydrodynamical Methods	7
1.2.3	Hydrodynamical Methods	8
1.2.4	Super-Resolution Approaches	10
1.2.5	Drawbacks of Existing Methods	10
1.3	Contribution of the Present Work	11
1.4	Outline	12
2	Governing Equations of Fluid Dynamics	15
2.1	Introduction	15
2.2	Continuity Equation and Equation of Incompressibility	15
2.3	Momentum Balance (Euler) Equation	16
2.4	Navier-Stokes and Vorticity Transport Equations	17
2.5	Reynolds number	19
2.6	Dimensional Analysis	20
2.7	Turbulence Theory	23
2.8	Boundary Conditions	26
3	Basic Image Processing in Experimental Fluid Mechanics	29
3.1	Particle-Based Fluid Motion Recording Techniques and Background	29
3.2	Particle Image Velocimetry (PIV)	29
3.3	Particle Tracking Velocimetry	33
3.4	Two-Dimensional Variational Approach	34
3.5	Noise Origin and Data Post-processing	36
3.5.1	Data Post-Processing	40
3.5.2	Total Variation	42
3.5.3	Summary	43
4	The Restoration Approach for Incompressible Fluid Flow	45
4.1	Overview	45
4.2	Two Dimensional VTE-Based Recovery Approach	46
4.2.1	Solenoidal Projection	46
4.2.2	Lowpass Filtering	47

Contents

4.2.3	Vorticity Rectification	48
4.2.4	Velocity Restoration	51
4.2.5	Applicability	52
4.3	Three-Dimensional VTE-Based Recovery Approach	52
4.4	Super-Resolution Approach	54
4.4.1	Background and Overview	54
4.4.2	The Algorithm for Super-Resolution Approach	55
5	Error Analysis	57
5.1	The Systematic Error of the Approach and Error Reduction Criterion	58
5.2	Assumptions	63
5.3	Divergence Removal	64
5.4	Error Reduction	66
5.4.1	Error Reduction: Lowpass Filtering	66
5.4.2	Error Reduction: VTE	66
5.4.3	Velocity reconstruction	71
5.5	Conclusion	73
6	Discretization	79
6.1	Finite Element Method	79
6.1.1	Ritz-Galerkin Method	79
6.1.2	Conventional Finite Element Method	81
6.1.3	Mixed Finite Element Method. The Discretization of the First Step	84
6.1.4	Discretization of the Third and Fourth Steps.	85
6.1.5	Outline	86
6.2	Linear Solvers	86
6.2.1	Conjugate Gradient Method	86
6.2.2	Preconditioning	88
6.2.3	Pre-Conjugate Gradient Method	90
6.3	Finite Differences	92
6.4	The Algorithm for Total Variation	95
7	Applications of the VTE Approach and its Comparison with other Methods	97
7.1	General Description	97
7.2	Two Dimensional Case	98
7.2.1	Uniformly Distributed Noise	98
7.2.2	Gaussian White Noise and Missing Data	98
7.2.3	Low Noise Level	102
7.3	Comparison with other Methods	103
7.3.1	Comparison with Scalar Valued Total Variation Denoising	103
7.3.2	Comparison with Vector Valued Total Variation Denoising	106
7.3.3	Comparison with the Median and Mean Value Operator Denoising	109
7.4	Three Dimensional Case	113
7.4.1	Uniformly Distributed Noise	113

7.5	Super-resolution Approach: Results	120
7.5.1	Data Sets and Set-Up	120
7.5.2	Results and Comparison	121
7.6	Experiments with Real Data	129
7.7	Summary	132
8	Properties of Restoration Approach	133
8.1	Choice of Parameter Values	133
8.2	Vorticity Melting	135
8.3	Multi Scale	139
9	Conclusions and Future Work	145
9.1	Conclusions	145
9.2	Future Work	146
10	Appendix	149
10.1	Vector Derivatives	149
10.2	Quadratic Functionals	151
10.2.1	Unconstrained Quadratic Functionals	151
10.2.2	Sobolev Spaces	152
10.2.3	Quadratic Functionals and Constraints	153
10.3	Theorems	154

List of Symbols

A, B, G, D	- matrixes, representing parts of linear systems
D^h	- grid, associated with matrix D
D_m	- characteristic scale of error velocity
C^n	- subset of n-times differentiable functions with
C_0^n	- subset of n-times differentiable functions with zero boundary values
\mathbf{F}	- force
F	- typical force
\mathcal{F}	- Fourier transform
\mathcal{F}^{-1}	- inverse Fourier transform
H	- Hilbert space
L	-typical length
$L^p(\Omega)$	-is the space of p times locally integrable functions for set $\Omega \in \mathbb{R}^n$
$[L^2(\Omega)]^d$	- the space of square integrable vector fields defined on Ω
Q^h	- grid with mesh size h
R	- Reynolds number
S	- Schur complement
T	- typical time
U	- typical velocity
U^h	- grid with mesh size h
$d = 2, 3$	- space dimensions
\mathbf{d}	- corrupted velocity vector field
g_σ	- the Gaussian filter with variance σ
\mathbf{g}	- ground truth vector field
h	- grid step
p	- pressure
p	- auxiliary function used in examples, proofs etc.
q	- auxiliary function used in examples, proofs etc.
\mathbf{q}	- auxiliary function used in examples, proofs etc.
t	- time
u, v, w	- x,y,z-components of velocity \mathbf{v}
\mathbf{u}	- recovered velocity vector field
\mathbf{v}	- velocity
\mathbf{v}^I	- ideal solution; a component of \mathbf{d} which satisfies all relevant hydrodynamic equations
v_n	- component of velocity \mathbf{v} normal to $\partial\Omega$
v_t	- component of velocity \mathbf{v} tangential to $\partial\Omega$
\mathbf{x}	- coordinate vector
x, y, z or x_1, x_2, x_3	- particle coordinates

δ	- noise; a component of \mathbf{d} caused by corruptions
δ_g	- noise remaining after Gaussian filtering
δ_3, δ_3	- noise remaining after the third step in 2D and 3D cases respectively
$\delta_4,$	- noise remaining after the fourth step
Ω	- domain in \mathbb{R}^3
$\partial\Omega$	- boundary of Ω
α	- user parameter for the third step in VTE-based restoration approach
α_i	- coefficient for conjugate and pre-conjugate gradient method
β	- user parameter for the fourth step in VTE-based restoration approach
β_i	- coefficient for pre-conjugate gradient method
λ	- user parameter for the regularizer in Total Variation denoising approach
ν	- kinematic viscosity
ρ	- density
σ	- variance in Gaussian Filter
τ	- as single means point spread function
a^τ	- as superscripts means the iteration number of a
ψ	- Lagrange parameter
ϕ	- frequency
φ	- basis function
ω	- 2D vorticity
ω_x, ω_y	- components of 2D vorticity
$\boldsymbol{\omega}$	- 3D vorticity
$\omega_x, \omega_y, \omega_z$	- components of 3D vorticity
$\langle \cdot, \cdot \rangle_\Omega$	- inner product; for $\phi(x), \psi(x) \in [L^2(\Omega)]^d$, $\langle \phi, \psi \rangle_\Omega = \int_\Omega \sum_{i=1}^d \psi_i(x) \phi_i(x) dx$
$\ \cdot \ _2$	- $L_2(\Omega)$ -norm; $\ \psi \ _2 = \sqrt{\langle \psi, \psi \rangle_\Omega}$
“.”	- Euclidian inner product; for $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$ $\mathbf{a} \cdot \mathbf{b} = \sum_{i=1}^d a_i b_i$
$ \cdot $	- Euclidian norm; $ \mathbf{a} = \sqrt{\langle \mathbf{a}, \mathbf{a} \rangle}$

Within the current work we will adopt the following rules and definitions. All symbols written in bold correspond to vectors and non-bold symbols represent scalars. The only exception is the symbol of "curl"; we will use notation \mathbf{curl} if this operator is applied to 3-D vector fields and $curl$ if it is applied to 2D vector fields. If not specified, all functions mentioned later belong to the space $[L^2(\Omega)]^2$.

Definition 1. *Scalar directional derivative.* Let $p \in C^1(\Omega)$ be some scalar function. The directional derivative of p is defined as $\frac{\partial p}{\partial \mathbf{n}} = \nabla p \cdot \frac{\mathbf{n}}{|\mathbf{n}|}$.

Definition 2. *Vector directional derivative.* Let \mathbf{a} be a d -dimensional one time differentiable vector-function defined on Ω and \mathbf{n} be a normal to the boundary $\partial\Omega$. The directional derivative vector consists of the directional derivatives of the vector \mathbf{a} with respect to the direction normal \mathbf{n} like in $\frac{\partial \mathbf{a}}{\partial \mathbf{n}} = \left(\nabla a_1 \cdot \frac{\mathbf{n}}{|\mathbf{n}|}, \dots, \nabla a_d \cdot \frac{\mathbf{n}}{|\mathbf{n}|} \right)$.

Contents

Let b be some scalar value. In this thesis we will often encounter the situation when all components of $\frac{\partial \mathbf{a}}{\partial \mathbf{n}}$ should be set to b . For this case we will use the notation $\frac{\partial \mathbf{a}}{\partial \mathbf{n}} = b$.

1 Introduction

1.1 Motivation

Knowledge of velocity distributions in gaseous and liquid media plays a crucial role for experimental fluid mechanics. It is quite important in a large variety of industrial applications, as well. For instance, the theory describing hydrodynamic phenomena such as turbulence or boundary layer motions contains a lot of gaps [42, 48], undiscovered problems and ambiguities which can be only clarified by experimental methods. Obviously, such investigations imply conducting some measurements and subsequent theoretical analysis of the velocity distributions obtained experimentally. Understanding of the fluid dynamics is even more important for the construction of various apparatus whose functional ability depends on the level of knowledge implemented in their design (e.g. planes, cars, ships, etc). For example, the flight performance of an aircraft is closely related to the process of its design, i.e. how correctly and accurately this plane was computed and manufactured. Another example: the efficiency and proper work of an artificial heart implies the absence of stagnant zones of blood in pumping areas. Accurate measurements of the velocity distributions (gas, water, fuel, blood, whatever) allows to reveal some defects of the current constructions, which, in turn, can give a hint for further improvement of their efficiency.

A wide variety of velocimetry methods are available at the moment [8, 35, 46, 67, 68, 71, 72, 73, 74, 77, 86, 87, 89, 101, 105, 107, 108]. They allow to measure velocity fields with different accuracy and resolution. Image velocimetry becomes the most advanced and promising technique for investigations of fluid flows over last two decades [2]. Particle Image Velocimetry (PIV) and Particle Tracking Velocimetry (PTV) are two most popular experimental method which are used in gas and fluid experimental physics on a regular basis [68, 89]. With respect to other methods they become the most powerful and informative in terms of obtained information and accuracy. The idea of these methods is as follows. Special tiny luminous particles are seeded in the moving medium. The position of all particles is recorded by a camera at different moments of time. Analyzing the obtained images and comparing the positions of all the particles in space one can calculate their velocities [61].

Being the state-of-the-art, the image velocimetry, however, produces some unavoidable noise inherent to every experiment [68, 77, 99]. As a result of the experimental errors, the measured vector field sometimes looks incomplete, and some fragments can be completely out of understanding. For instance, according to [99], the number of errors related only to spurious vectors in an ideal case is of about 5 %.

Since the image velocimetry works with digital images, the particle displacements measured in experiments are often discrete and equal to integer numbers of pixels [3]. The

1 Introduction

particle displacement of 5 pixel contains up to 10 % of errors. So, here we arrive at the lower boundary of sensitivity of the image velocimetry method. On the other hand, in a large class of experiments the low seeding particles concentration is a necessary requirement for successful measurements [16, 46, 82, 105]. Note, however, that it is not always easy to provide a homogeneous distribution of particles; the density of the particle field is usually inhomogeneous containing a number of gaps. As a result, the measured vector fields become irregular with low spatial resolution. Summarizing, one can conclude that no matter what velocimetry technique is used, the estimated vector field contains a lot of gaps and errors which, in turn, makes their further analysis difficult.

The main topic of this thesis, is to develop a method that can reconstruct and restore in a physically plausible way any fluid flow velocity fields obtained by a velocimetry method. Though the investigation is mostly based on the scenarios occurring in image velocimetry, the method can work with any type of fluid velocity data obtained by any technique and does not rely on a particular model of corruption rather than on physical properties of a fluid flow. Therefore, instead of modeling a noise explicitly, which is difficult and very specific due to the diversity of errors that can occur, the method should return a vector field that is close to the input data and approximately satisfies the basic physical equations governing the flow. At the same time, the method should be robust to various types of estimation errors and computationally efficient so that to be applicable to large-scale three-dimensional problems.

1.2 Related Work

The imperfection of image velocimetry results in the presence of errors and corruptions of the output velocity data, as discussed above. A variety of methods that can reduce this noise and improve the output of the existing techniques have been recently developed. Some of these methods allow only posterior data processing and improvement. Another class of denoising methods can be used at a real time basis, i.e. the output field is improving directly during actual measurements. Conventionally, all methods can be classified as those that essentially exploit hydrodynamical properties of fluid, and the methods that are not based on hydrodynamic models.

1.2.1 Image Velocimetry Techniques

The “cross-correlation method” and the “brightness change constraint model” are two major procedures usually exploited by the PIV and PTV routines for computing of the particles displacements [25, 68, 89, 101]. The first one, the cross-correlation method, is based on the assumption that in a moving fluid all tracing particles located in a small finite area should experience almost similar displacements within a short time interval. The idea of the method can be illustrated by the following 2D experiment. Two photographs of fluid flow made within a short time span are partitioned into small segments, and similar segments on both photographs are compared. It is assumed that fluid particles in the first image should have similar shift in the same segment in the second photograph. This

shift is normally calculated by a special cross-correlation routine which gives the name to this method [68]. More complicated 3D fluid flows require special particle position reconstructive procedures based on tomography [23, 65, 25].

The second method is based on the light intensity balance equation. It is hypothesized that intensity of the light radiated from particles is invariant with respect to time. All temporal variations of the light intensity must be compensated by the advection related to particle motions. Comparing these variations at every point one can calculate the particle velocity using the light intensity balance equation [35].

1.2.2 Non-Hydrodynamical Methods

In application of “non-hydrodynamical” methods it is assumed that the errors occurring in velocimetry measurements satisfy some certain statistical laws. For instance, the authors of [99] developed an algorithm that defines and eliminates the error related to spurious vectors. This method is based on the assumption that the spurious vectors are just local nonphysical artificial perturbations which are distributed randomly. They can be distinguished from an actual signal by sharp gradients and dramatic deviation in the light intensity from the surrounding field characteristics.

Precision and robustness of velocity measurements depend basically on the ability of accurate definition of the displacements of particles from their initial positions within a certain temporal interval [3]. Since particle displacement is determined by an integer number of pixels, the accuracy of measurements is restricted by a pixel size. The authors of [103] have shown that with the use of either parabolic or Gaussian “three point peak fit” approximation this accuracy might be improved up to sub-pixel level. The interrogation window offset on the integer part of pixel displacement was suggested in [100], and the corresponding domain deformation [8, 77, 7] can reduce the remaining error at least three times.

An increase of accuracy and further refining of the output field can be achieved in two-step procedure: the rough displacements in a large correlation window are calculated first, and after that the whole domain is split into several smaller fragments in order to make some corrections [33]. Note, however, that this method has a serious limitation on a number of particles presenting in the domain. For a robust cross-correlation method the smallest number of particles presenting in the interrogation window, which allows the validity of measurements of 90 – 95%, should be not less than 10 [3].

A novel advance procedure reported in [46] inherits the advantages of both, cross-correlation method and the probability theory. As distinct from the cross-correlation procedure, the fluid motion in the interrogation area in this method is not assumed to be strictly parallel. The displacement of each particle in interrogation area can slightly deviate from some value chosen for the given interrogation window. As a result, the velocity field obtained using this procedure contains more details.

The in- and out-of-plane particle motion is a usual source of error in PIV measurements. The amount of this error can be reduced if these motions are taken into account. In [67] the authors showed the feasibility of an out-of-plane velocity estimation by analyzing images of particles within parallel laser sheets.

1 Introduction

The study of some physical phenomena like tracking of pollutants requires an implementation of PTV method [16]. Since the seeding density in PTV is quite low and usually inhomogeneous, the resultant velocity vector field in many cases is unacceptably sparse and irregular. The procedures discussed in [84, 85] shows, in particular, how the resolution of the output fields can be increased using only the interpolation technique. Note, however, although the density of particles is quite low, the errors caused by particle overlapping do exist and can spoil the result of tracking. Several methods have been developed in order to eliminate the error related to the overlapping (see, for instance, [9, 19, 87, 105]).

Regardless of the type of data obtained in measurements, from the statistical point of view the errors containing in the observational data have a lot of common characteristics. Therefore the denoising methods that are widely applied in signal processing [17, 70] can be exploited for vector field denoising, as well. In these methods it is assumed that the noise satisfies some particular properties. For instance, in [17, 70] the noise is supposed to have unbounded total variation, and the authors suggested an algorithm that removes such a noise.

1.2.3 Hydrodynamical Methods

The advantages of hydrodynamical methods are based on the idea of a reasonable combination of the image velocimetry techniques and variational approach which substantially uses the fluid flow properties. A large variety of such methods are available at the moment (see, for instance, [71, 72, 73, 74, 86, 107, 108] and references herein). All these methods exploit variational technique which runs as follows: a cost functional is constructed first. It consists of two parts, i.e. a data term which gives a link between input image data and the output velocity vector field, and a "regularizer", which penalizes the undesirable features of the output flow. The quality of the output result depends on the fact how correctly the 'regularizer' has been chosen.

In general, all regularizers can be classified into two categories. Into first one fall the regularizers which use the information from the previous data set in order to pre-estimate some possible flow patterns at the current step. Exploiting an actual discretization of full system of hydrodynamic equations one evaluates the vector field obtained from preceding computations towards time step in numerical fluid solver. At the final stage the computed "predicted" flow is used as a constraint for the objective functional and thereby stick the evaluating data on the current step to the vector fields obtained from the previous data set [74].

The second-class regularizers operate only with in-situ current data and act as a penalty term for the data processing. As distinct from the previous approach, these regularizers represent only either reduced versions of the Navier-Stokes system, or vorticity transport equation and can be applied only to some particular cases. Having such a restriction these regularizers, however, have two advantages. The first one is that their construction does not require any data from the previous steps. In other words, one can obtain a velocity vector field only from a pair of flow frames, while the "predictor" approach uses at least three of them. As a consequence, the second advantage is that even though the flow at previous step was incorrectly estimated, this error does not affect the data set at the next

steps.

A number of papers have been published to illustrate the efficiency of the denoising procedures based on such type of regularizers. For instance, the authors of [5] suggested a variational method which, as distinct from [99], uses the fluid dynamical constraints rather than a simple statistical approach. More specifically, it is based on the application of some dynamical constraints such as continuity, smoothness and matching to the original data.

Another example of physically justified approach is presented in [102] where it is suggested to reconstruct characteristics of fluid flows using a sequence of images. The method itself substantially exploits the principles of fluid mechanics. The algorithm is constrained in such a way to satisfy the law of conservation of mass, as well as physically motivated boundary conditions.

Another method discussed in [20] represents an image-based integrated version of the continuity equation. Its main idea is concentrated around the mass conservation law. The authors have shown that the intensity of the light reflected by seeded particles in some finite volume behaves in a similar way as the density of the fluid in the same volume. Mathematically, the model is formalized in terms of a regularization functional which preserves divergence and vorticity blobs of the motion.

The improved method presented in the paper [38] is based on the similar idea as [20] but contains an additional regularizer in the penalty term. It exploits the physical principle of the cascading of turbulent energy along the spectrum and penalizes the appearance of the structures which do not satisfy the principle of energy cascading.

An attempt to use a full system of hydrodynamic equations as a second-class regularizer was reported in [86]. The Navier-Stokes and continuity equations were included into the objective functional as two individual regularizers weighted with some constants. Since these weights were chosen artificially, there was no guarantee that the output velocity vector field will satisfy at least one of the equations.

Being more sophisticated in comparison with "nonphysical" models, the hydrodynamical methods are able to give better and more realistic output. Note, however, that these methods have some restrictions, as well. For instance, they can be applied only to some particular cases. In addition, they need extra time-consuming computational procedures to be incorporated into the model.

In a very recent paper [20] the authors suggested to use a second-order regularization scheme. It is well adapted to fluid motion because it can enforces the regions of homogeneous vorticity and/or divergence. However, according to [38], this method suffers from the lack of physical consistency with the structure of turbulence and fails to represent precisely the variety of spatial structures, particularly at small scales. As a result, the applicability of the mentioned algorithm is reduced to non-turbulent cases.

The method suggested in [38] could cope with this drawback. However, it is applicable only to experiments in which the coefficients of turbulent exchange can be set in advance. However, these coefficients are assumed to be constant in space and time in the whole domain, which implies constant characteristics of the turbulence in every particular point of the flow. Quite natural that this model is not applicable to the cases of turbulence with variable characteristics in space and time.

1.2.4 Super-Resolution Approaches

Since the PIV method operates with individual particle images, the density and velocity distributions of the output vector fields are reconstructed using the coordinates of the particle images [68, 80]. The usual requirement for a typical PTV experiment is low concentration of a tracer. However, in many cases it is not possible to arrange a procedure which provides a uniform seeding of particles. As a result, the output velocity vector field becomes sparse and nonuniform. It normally requires an additional postprocessing in which the resolution of vector data will be increased and mapped onto a regular grid. Moreover, the PTV velocimetry has two well-known defects [85]. Firstly, it usually tends to involve some mismatched vectors into the data. Secondly, since the velocity vectors are randomly distributed in space, the differential and integral values such as vorticity and stream function cannot be directly obtained by ordinary numerical differentiation or integration procedures.

Several interpolation routines aimed at reconstruction of sparse data have been recently elaborated [33, 47, 80, 84, 85]. Some of these methods like [84, 85] are heavily based on the application of the spline theory [92]. In [85] the authors developed a method that rearranges the discrete velocity vectors onto a regular grid by hierarchical interpolation processes from lower to higher order resolution. This method is sensitive to the quality of input data; being sparse the input velocity vector fields are supposed to be free of noise. The method described in [84] is less sensitive to noise. It can operate with data containing Gaussian noise although a substantial requirement is that the true velocity vector field should be generally very smooth.

A novel procedure which combines advantages of the PIV and PTV methods has been suggested in [47, 80]. Important is that this method can estimate an approximate particle displacement using conventional cross-correlation which allows more precise computations of displacement than in the case of tracking individual particles. Thus, in an ideal case, the finest possible resolution which can be achieved is one velocity vector per particle.

1.2.5 Drawbacks of Existing Methods

The presence of some noise and errors is an unavoidable attribute of any real measurement. The level of noise (errors-to-noise ratio) presented in output vector fields obtained by the PIV or PTV methods depends on many factors such as concentration of particles, strengths of shears (velocity gradients) in a flow, etc. Another source of noise is a systematic error arising from the approximations and assumptions taken during the development of the method.

The major drawback of non-hydrodynamical methods of denoising is their ability to introduce some nonphysical structures into the flow. For instance, the error of velocity estimations of the smallest resolved Taylor-Green vortices obtained by the state-of-the-art cross-correlation PIV method (an ideal case is considered: no noise, outliers, peak locking effect and perfect correlation in each region) exceeds 11%. The value of error inherent to an estimation of sub-pixel particle displacement is around of 0.1 px, so the displacements smaller than 1 px contain at least 10% of noise.

The performance of hydrodynamical methods strongly depends on the "correctness" of the chosen regularizer. The regularizers suggested in the methods discussed above are suitable either for nearly laminar flows, or appear sensitive to errors remaining after the previous computations.

Some other methods operating with interpolation of scattered irregular data to a regular grid are extremely sensitive to the noise containing in the input vector field. In fact, the interpolation procedure suggests that the input field is noise free. Some combinations of variational approaches with the interpolation techniques might be a remedy for the problem. The question, however, remains what would be a better regularizer for such a procedure. For instance, first order regularization applied to interpolated data [84] is suitable only for the flows with small Reynolds numbers. Moreover, such a regularization removes small details related to a real flow. The weakest point of the hybrid PIV-PTV methods [47, 80] appears in the restriction applied to the density of particle seeding and strength of shears: for a successful measurements the concentration of tracers should be sparse enough to avoid particle overlapping and losses of correlation, while the flow itself should contain only small velocity gradients.

1.3 Contribution of the Present Work

This thesis covers the topic related to some areas of experimental particle image velocimetry. It also includes the discussion of various possible sources of the noise appearing during the conducting of observations and successive image processing. The traditional post-processing procedures of the noise removal are discussed in the concluding part of the thesis.

It is suggested here that the major contribution of the present work into the area of particle image velocimetry is in the development of a special method for reconstruction of observational vector fields. An advantage of this method is that it can reconstruct fluid flows obtained by *any* experimental measurement technique. It accepts a wide variety of corrupted two-dimensional (2D) and three-dimensional (3D) velocity vector fields as input data. Most importantly, this method allows to recover missing fragments and to remove noise in a physically plausible way using a fine resolution grid. The developed method essentially exploits physical properties of incompressible fluid and does not rely on any particular model of noise. Our approach comprises the following successive reconstructive steps:

- removal of the divergent component of the flow from the experimental data set;
- elimination of noise by the Gaussian low-pass filtering;
- enforcing of physically plausible flow structure in terms of vorticity satisfying the vorticity transport equation.

The developed algorithm performs well with different types of noise and estimation errors.

The discussed here method can be treated as an extension of some recently published and mentioned above works on data processing by incorporation of some hydrodynamical

1 Introduction

and super-resolution reconstructive principles. In comparison with the existing procedures the current method is noise free and can work with any types of corrupted fluid flows. Due to the computational complexity of the hydrodynamic equations in the vast majority of the previously developed methods e.g. [72, 102], one had to sacrifice many important properties of fluid flows by making some assumption on linearity, or incompressibility, etc. The current approach is free from many traditional assumptions and restrictions: it takes into account all hydrodynamical properties of fluid flows and in the same time it remains simple and efficient in computational sense.

The method was extended to a super-resolution approach which can restore sparse noisy data and rearrange them onto regular grids. In comparison with the reconstructive methods listed above, the introduced procedure returns hydrodynamically correct and noise-free velocity vector fields without undesirable loss of the details.

In addition, the convergence condition of this algorithm in case of low noise-to-signal level is proven in this thesis. It is derived that for a certain set of parameters the amount of noise containing in the output field will definitely be smaller than in the input data set. The reconstructive properties of the developed method are compared against the existed post-processing techniques. It is demonstrated that in comparison to other methods, the present algorithm reconstructs the velocity vector field without loss of flow details and without introduction of some additional nonphysical artifacts.

The numerical implementation of the reconstructive approach is discussed in detail. Two numerical techniques, i.e. the finite difference and finite element methods, are used for the numerical discretisation of the algorithm. It is discussed here as well how the reconstructive algorithm can be faster and computationally more efficient using preconditioning technique.

1.4 Outline

Chapter 2 gives a brief overview of the fundamental principles of fluid mechanics. Since the present work is heavily based on these principles, the aim of Chapter 3 is to give a short introduction to fluid mechanics. The main governing equations describing fluid motions and some basic statistical laws of turbulent fluid flows will be derived and explained.

Chapter 3 describes some basic elements of the particle velocimetry and the main possible sources of the noise and errors. This chapter continues the discussion over the velocity measuring technique started above. Here we give some additional reasonings motivated the present research. At the end of this chapter the main methods commonly used in the velocimetry data post-processing are discussed.

Chapter 4 is a central part of the current work. It provides the details on application of the physically-based restoration approach to two-, and three-dimensional fluid flows. All assumptions and physical reasonings are discussed here step by step in detail.

Chapter 5 provides an error reduction analysis for the developed algorithm. It is rigorously justified that in the case of low noise contamination the VTE-based restoration approach always provides the conditions for error reduction. It is demonstrated that the error reduction rate depends basically on a set of chosen adjustable parameters. The sensitivity of the method to these parameters is demonstrated using a series of examples.

Chapter 6 gives a brief overview of the methods used for the numerical implementation of the restoration algorithm discussed in Chapter 4. The discretization of the governing equations is conducted using two independent techniques: finite differences method and finite element method. This section briefly dwells on advantages and drawbacks of every method, but first of all it touches upon the foundation of the finite element and finite difference theory and discusses their application to every step of the algorithm. At the end some conclusions concerning the considered numerical procedures are formulated.

Chapter 7 demonstrates the performance of the developed procedure by considering a range of practically relevant scenarios. The demonstration includes the reconstruction of numerically simulated flows artificially damaged by adding a noise, and comparison of the reconstructed results with the ground truth. These experiments represent some typical situations which can occur in practice. All of them are characterized by the fact that the vector field is contaminated by the noise, and in addition, the information in some regions is missing. All examples are discussed in such a way to provide both visual impression on the quality of the restoration, and a quantitative estimates of the method efficiency. The results obtained using the developed procedure are compared also against some other methods. The chapter is finalized with the reconstruction of some real fluid flows observed in laboratory experiments.

Chapter 8 gives an overview of the properties of the developed algorithm and its dependence on the choice of parameter. In this chapter we discuss and illustrate how the wrong choice of parameters can affect the output fields.

Chapter 9 gives a summary of the presented work and discusses some possible improvements which can be done in future.

An auxiliary Chapter 10 contains some fundamental mathematical principles used in this work.

2 Governing Equations of Fluid Dynamics

2.1 Introduction

The current research is aimed an elaboration of a method that reconstructs a wide variety of corrupted two- and three dimensional (2D, 3D) fluid vector fields in a physically plausible way. The approach is based on the hydrodynamical properties of viscous fluid and enforces the corrupted data to be consistent with them.

This chapter gives a brief overview of the main principles of hydrodynamics that will be used in the construction of the method. As long as this section has an auxiliary function, we omit the derivation of some formulas or give them in short way and focus on the principles themselves instead of going into details. The additional information can be found in the original sources (see, for instance, [43, 53] and references herein).

Though gas and fluid represent different aggregate states, they satisfy the same continuum mechanics principles and are similar in terms of hydrodynamics. Moreover, being under the same conditions, the gas and liquid media quite often perform similar flow properties and satisfy the same laws. In the light of this fact we will use word 'fluid' when we speak either about gas or fluid. Finally, the fluid motion will be considered in a general 3D case.

2.2 Continuity Equation and Equation of Incompressibility

Our study concerns only the macroscopic motions in some volume Ω filled with gas or fluid. Saying "infinitesimal volume" we mean that it is infinitesimal in physical sense, so that it is negligible in comparison with Ω , but much larger than the distances between the molecules in fluid filling the volume. The same reasoning concerns the liquid point and liquid part.

Let us allocate any point in Ω and set a reference Cartesian co-ordinate system (x, y, z) there. The distributions of velocity $\mathbf{v}(x, y, z, t)$, pressure $p(x, y, z, t)$ and density $\rho(x, y, z, t)$ determine the physical state of a moving fluid (t is time).

The matter conservation law is one of the key principle of hydrodynamics. It states that the matter in a closed system remains constant over the time regardless the processes acting inside the system. In other words, the matter does not disappear to nowhere or does not appear from nowhere. In terms of formulas this principle takes the form of so called **continuity** equation

$$\frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \mathbf{v}) = 0 . \quad (2.1)$$

2 Governing Equations of Fluid Dynamics

In many practical cases the fluid flows relatively slow so that ρ is nearly constant. Let c denote the speed of sound according to [53]. All fluids which velocities satisfy the condition

$$|\mathbf{v}| \ll c \quad (2.2)$$

are called **incompressible**. For instance, the air moving with the speed up to 90 km/h can be treated as an incompressible fluid. In such a case the continuity equation (2.1) can be simplified to the *incompressibility* equation:

$$\operatorname{div}(\mathbf{v}) = 0 . \quad (2.3)$$

2.3 Momentum Balance (Euler) Equation

It is well known that motion of a real fluid attenuates in time provided no external forces are applied to it. This attenuation takes place due to the presence of internal friction inherent to every fluid. There is, however, a large class of fluid flows, so called *ideal fluid* flows, in which the friction effects appear to be negligible. For instance, the flows with large Reynolds numbers (see Sec. 2.7) or large scaled vortexes perform the motions with very small friction. In this section we will derive the momentum balance equations for an inviscid fluid which are widely applied to low friction cases.

Considering any volume V_0 located in Ω , we distinguish external and internal forces acting on V_0 . The internal forces \mathbf{F}_i appearing in fluid volume act on each particle separately. They can have different origin like the Archimedean force, viscous forces, etc. We assume here that the fluid has zero buoyancy and zero friction, so that we can set $\mathbf{F}_i = 0$.

External forces, \mathbf{F}_e , appear in action at the surface of V_0 . Obviously, they are proportional to the external pressure $-p$ integrated over the surface of V_0 , i.e.

$$\mathbf{F}_e = - \oint p d\mathbf{r} . \quad (2.4)$$

The vector $d\mathbf{r}$ is a small element of surface of V_0 . Its absolute value is equal to the area of this element and its direction coincides with the vector normal to it. Equation (2.4) can be rewritten in an equivalent form as follows:

$$\mathbf{F}_e = - \int \operatorname{grad}(p) dV . \quad (2.5)$$

It is clear that in the considered case the second Newton's law on the external force \mathbf{F}_e acting on the volume V_0 reads:

$$\rho dV \frac{d\mathbf{v}}{dt} = - \operatorname{grad}(p) dV . \quad (2.6)$$

As distinct from Sec. 2.2, the velocity \mathbf{v} in (2.6) is attributed to the moving volume dV but not to the fixed point in space. If so, the volume coordinates are functions of

2.4 Navier-Stokes and Vorticity Transport Equations

time, i.e. $\mathbf{v} = \mathbf{v}(x(t), y(t), z(t), t)$. Presenting the full derivative $d\mathbf{v}/dt$ in terms of partial derivatives we obtain:

$$\frac{d\mathbf{v}}{dt} = \frac{\partial\mathbf{v}}{\partial t} + \frac{\partial x}{\partial t} \frac{\partial\mathbf{v}}{\partial x} + \frac{\partial y}{\partial t} \frac{\partial\mathbf{v}}{\partial y} + \frac{\partial z}{\partial t} \frac{\partial\mathbf{v}}{\partial z} = (\mathbf{v} \cdot \nabla)\mathbf{v} . \quad (2.7)$$

Substituting (2.7) into (2.6) we finally obtain the momentum balance, or *Euler equations*:

$$\frac{\partial\mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} = -\frac{1}{\rho} \text{grad}(p) . \quad (2.8)$$

The actual rate of change of the velocity of a liquid particle at any point is defined by the derivative $\partial\mathbf{v}/\partial t$. It is equal to zero when all forces (both internal and external) are in a balance at every point and such flows are called “constant” or **stationary**. In the opposite case, i.e. when the residual force is nonzero and the velocity of fluid flow changes, such flows are called **non stationary**. The nonlinearity of a flow appears as an advection term $(\mathbf{v} \cdot \nabla)\mathbf{v}$. It represents the spatial characteristics of the velocity distribution. The fluid flows referred as *non linear flows* if the value of this term is comparable with other terms in (2.8), and one says that flow is *linear* if its impact is close to zero. Evidently, this term is most important in the areas with large spatial velocity gradients. Note also that this term appears from the decomposition of the full derivation $d\mathbf{v}/dt$ in (2.7) on temporal derivative and advection; the latter is not always dependent on time (in stationary fluid flows, for instance). The external forces acting on a liquid particle from the surrounding fluid is presented by the pressure gradient $\frac{1}{\rho} \text{grad}(p)$. All other external forces can be added to the right hand side of the Euler equations as follows:

$$\frac{\partial\mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} = -\frac{1}{\rho} \text{grad}(p) + \mathbf{F} . \quad (2.9)$$

The Euler equations do not change their canonic form (2.9) regardless the origin and type of the external force \mathbf{F} . However, to be more specific, the next section shows how the Euler equations transform into the Navier-Stokes equations when the viscosity is taken into account and the term \mathbf{F} is specified as shear stresses produced by the viscosity.

2.4 Navier-Stokes and Vorticity Transport Equations

Unlike the ideal fluid discussed above, the flow of real fluids is exposed to an extra frictional effect appearing between internal layers moving with different velocities (velocity shear stresses). Being derived for ideal fluids, the Euler equations must be modified according to this effect. In the updated form the momentum balance equations are called the **Navier-Stokes equations** [1] that include additional “viscous” terms. In general, the viscosity of real fluid flow depends on various parameters such as temperature, velocity, pressure etc. Although measurable, these dependencies are quite complicated, but they have a negligible impact on the flow in the vast majority of cases. This is the reason why the viscosity is usually assumed to be independent on these parameters.

2 Governing Equations of Fluid Dynamics

The Navier-Stokes equations were derived using the relationship between the strain rate tensor and the momentum flux density tensor [53]. We omit here some intermediate steps and give only the final form for the strain rate tensor:

$$\frac{\partial}{\partial t} \rho v_i = - \frac{\partial \Pi_{ik}}{\partial x_k} \quad i, k = x, y, z. \quad (2.10)$$

The strain rate tensor stands $\partial \rho v_i / \partial t$ and corresponds to the left hand side of the Euler equations (2.9). The tensor Π in the right hand side represents the momentum flux density, i.e. the amount of momentum crossing a small element of surface $d\mathbf{r}$ (see Sec. 2.3). It consists of reversible momentum flux $p\delta_{ik} + \rho v_i v_k$ (the mechanical movement of different parts of fluid driven by the differences in pressure) and irreversible momentum flux σ'_{ik} appearing due to the internal viscosity. Thus

$$\begin{aligned} \Pi &= p\delta_{ik} + \rho v_i v_k - \sigma'_{ik}, \\ \sigma'_{ik} &= \eta \left(\frac{\partial v_i}{\partial x_k} + \frac{\partial v_k}{\partial x_i} - \frac{2}{3} \delta_{ik} \frac{\partial v_l}{\partial x_l} \right) + \zeta \delta_{ik} \frac{\partial v_l}{\partial x_l}. \end{aligned} \quad (2.11)$$

Two new parameters in (2.11), i.e. η and ζ , are called “dynamic” and “second” viscosities, respectively. They represent an internal friction occurring in fluid. For the sake of simplicity we will use kinematic viscosity $\nu = \frac{\eta}{\rho}$ instead of dynamic viscosity η .

With the taken assumptions of incompressibility and constant viscosity, the equation (2.10) is simplified to the following:

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = - \frac{1}{\rho} \text{grad}(p) + \nu \Delta \mathbf{v} + \mathbf{F}_v. \quad (2.12)$$

Here the term \mathbf{F}_v represents so called ‘volume’ force, e.g. the Archimedean or Coriolis force. They act on each individual liquid particle in a whole volume and can be important in many geophysical applications. However, these two effects included into volume force \mathbf{F}_v (as well as some others participated in flow dynamics) do not introduce any substantial difficulty in the method development, which is why we assume them to be zero. They can be easily included into the model, if necessary.

Equation (2.12) contains the pressure function p which in many cases is difficult for calculation (mostly because of unknown boundary conditions at liquid and rigid boundaries). Taking into account that $\mathbf{curl}(\text{grad}(p)) = 0$ (see eq.(10.11)), one can exclude p by applying \mathbf{curl} operator to (2.12)

$$\mathbf{curl} \left(\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} \right) = \mathbf{curl} \left(- \frac{1}{\rho} \text{grad}(p) + \nu \Delta \mathbf{v} + \mathbf{F}_v \right). \quad (2.13)$$

Defining $\mathbf{curl}(\mathbf{v}) = \boldsymbol{\omega}$, we finally obtain the momentum balance equation in the following form without the pressure p :

$$\frac{\partial \boldsymbol{\omega}}{\partial t} + (\boldsymbol{\omega} \cdot \nabla) \mathbf{v} + (\mathbf{v} \cdot \nabla) \boldsymbol{\omega} = \nu \Delta \boldsymbol{\omega}. \quad (2.14)$$

This form of (2.12) is referred hereafter as the Vorticity Transport Equation (VTE), and the function ω is called “vorticity”. In 2D case the VTE equation has simpler form:

$$\frac{\partial \omega}{\partial t} + (\mathbf{v} \cdot \nabla) \omega = \nu \Delta \omega . \quad (2.15)$$

Note that in 2D case the vorticity is a scalar function, since in 2D case *curl* operator transforms vectors into scalars (see eq. (10.6) in Chap. 10).

2.5 Reynolds number

The complete system of the Navier-Stokes equations is quite complicated both for analytical and numerical analysis. Note, however, that in a large number of practical applications this system can be substantially simplified by performing a simple dimensional analysis. The understanding of physical processes allows one to omit sometimes a number of secondary terms from the system which do not give any principal contribution into the dynamics of fluid flow. The transition to dimensionless coordinates clearly shows which terms of the governing equations have infinitesimal contribution. Suppose some liquid volume moves over a distance of L within the time interval τ . We denote P and U as maximum pressure and speed value, respectively, occurred during that period of time. The new dimensionless variables \mathbf{v}^* , \mathbf{x}^* , t^* , p^* , ρ^* are given in a set of transition equations:

$$\begin{aligned} \mathbf{v} &= \mathbf{v}^* U \quad \text{where } \mathbf{v}^* \in [-1, 1] \times [-1, 1] \times [-1, 1] , \\ \mathbf{x} &= \mathbf{x}^* L \quad \text{where } \mathbf{x}^* \in [-1, 1] \times [-1, 1] \times [-1, 1] , \\ t &= t^* \tau \quad \text{where } \tau \in [0, 1] , \\ p &= p^* P \quad \text{where } p^* \in [0, 1] , \\ \rho &= \rho^* \Xi \quad \text{where } \rho^* \in [0, 1] . \end{aligned}$$

In dimensionless coordinates system (2.12) takes a new form:

$$\frac{U}{\tau} \frac{\partial \mathbf{v}^*}{\partial t^*} + \frac{U^2}{L} (\mathbf{v}^* \cdot \nabla^*) \mathbf{v}^* = - \frac{P}{L \Xi} \frac{1}{\rho} \text{grad}(p^*) + \nu \frac{U}{L^2} \Delta^* \mathbf{v}^* . \quad (2.16)$$

Since all dimensionless variables in (2.16) are of order of one, the real impact of every term on fluid dynamics is determined by the multipliers appeared in front of each term in (2.16). Eliminating their values one can bring the equations into a more simple form. Note, that instead of dealing with the multipliers it is much convenient to deal with dimensionless numbers combined from these multipliers. These numbers characterize the regimes of a flows, which we describe here shortly.

The introduced scales P , L , τ , U , Ξ are not independent quantities, in particular

$$\frac{L}{\tau} = U; \quad \frac{U^2}{L} = L^2 \tau^2 L = \frac{L}{\tau^2} = \frac{U}{\tau}; \quad \frac{P}{L} = \frac{\Xi U^2}{L} = \frac{\Xi U}{\tau} . \quad (2.17)$$

The substitution of (2.17) into (2.16) brings the Navier-Stokes equations to a new form:

2 Governing Equations of Fluid Dynamics

$$\frac{\partial \mathbf{v}^*}{\partial t^*} + (\mathbf{v}^* \cdot \nabla^*) \mathbf{v}^* = -\frac{1}{\rho^*} \text{grad}(p^*) + \frac{1}{Re} \Delta^* \mathbf{v}^* . \quad (2.18)$$

The nondimensional parameter in front of the viscous term is called the Reynolds number,

$$Re = \frac{UL\rho}{\eta} = \frac{UL}{\nu} . \quad (2.19)$$

It characterizes the influence of the viscosity on fluid dynamics. A special class of “laminar” fluid flows is generated at small values of the Reynolds number while large Reynolds numbers are associated with turbulent flows. In particular, the bigger Re is, the smaller impact of viscosity and the fluid flow is more turbulent.

Applying the same reasoning used for derivation of Re , we can obtain the other dimensionless combinations. The Strouhal number,

$$St = \frac{U\tau}{L} ,$$

characterizes the non stationarity of a flow. Obviously, zero value of St corresponds to stationary flow and increases while the flow becomes more unsteady.

2.6 Dimensional Analysis

In many practical cases some useful characteristics of physical processes can be obtain without solving the Euler or Navier-Stokes equations, but just using a relatively simple dimensional analysis. This method is based on the idea that the parameters that control fluid flows are normally constructed from quantities which have fixed dimensions, such as length, mass, time, temperature, etc. The combination of these quantities must be organized in such a way that a resultant physical relationship is a certain combination of series of nondimensional complexes. For instance, let us assume that the observed quantity a depends on several variables like b_1, b_2, \dots, b_n , where all quantities have fixed physical dimension. Obviously, only a few products of powers of the variables b_1, b_2, \dots, b_n can be arranged in order to give the right dimension of a . Based on this fact the dimensional analysis shows how a correct combination of b_1, b_2, \dots, b_n can be organized to give the right formula for the value a .

Thus, the dimensional analysis appears to be a powerful tool which allows to study physical properties of fluid flows considering only the dimensions of all dependent and independent variables. It helps to understand on which quantities given physical phenomenon depends on, which, in turn, allows one to gain a deeper insight into the problem. Let’s illustrate this idea with the help of the following simple example considering a steady viscous fluid flow streamlining a rectangular plate of length L . The task is to derive a formula which describes the drag force acting on the plate. Obviously, this force must depend on the typical value of the flow speed U , length scale L , dynamic viscosity ν and fluid density ρ . The dimensions of all mentioned functions, variables and parameters are

as follows: $F = [\text{mass} \times \text{length}/\text{time}^2]$, $U = [\text{length}/\text{time}]$, $\rho = [\text{mass}/\text{length}^3]$ and $\nu = [\text{length}^2/\text{time}]$. The combination of these quantities gives the following expression:

$$F = U^a L^b \rho^c \nu^d .$$

It is clear that the right hand side of this formula should have the dimension of force. If so, the powers a , b , c and d should be chosen in such a way to satisfy the following relation

$$[\text{mass} \times \text{length}/\text{time}^2] = [\text{length}/\text{time}]^a [L]^b [\text{mass}/\text{length}^3]^c [\text{length}^2/\text{time}]^d ,$$

which is equivalent to the following system of linear equations:

$$\begin{aligned} \text{mass: } 1 &= c, \\ \text{length: } 1 &= a + b - 3c + 2d, \\ \text{time: } -2 &= -a - d. \end{aligned}$$

These system has the solution: $c = 1$, $b = a$, $d = 2 - a$, so that

$$F = U^a L^a \rho \nu^{2-a}$$

or

$$F = \rho U^2 L^2 \left(\frac{UL}{\nu} \right)^n = \rho V = U^2 L^2 Re^n = \rho U^2 L^2 f(Re) , \quad (2.20)$$

where $n = 2 - a$ is an undefined coefficient which is usually determined from experiments. Re is the Reynolds number (see 2.5). Note, that we derived the formula for the drag force considering only the dimensional properties that should give the combination of dependent variables.

The dimensional analysis of many physical problems normally starts with the correct choice of *the basic or fundamental units*, i.e. the units from which all other physical values can be derived. The following simple example illustrates an importance of this statement. Let length L , mass M and time T are chosen as the fundamental units. The task is to derive the formula which represents the force F acting on some object. The dimension of force is $[\text{mass} \times \text{length}/\text{time}^2]$. Combining the dimensions L , M and T in such a way to find the complex which matches the dimension of force we can obtain the only combination which gives the dimension of force, and which is nothing else as the second Newton's law, i.e. $F = M(L/T^2)$. Here *the derived* value F , i.e. the force, was obtained just using the correct choice of the primary dimensional units without any experiments and theoretical developments. Note, however, that in the case when instead of length and time the gravitational acceleration (for instance) is used as a fundamental unit, the equation for the force will be still valid from the point of view of dimensions but the result will be absolutely wrong from physical point of view. This means that certain level of physical understanding and heuristic intuition is necessary in application of the dimensional analysis.

2 Governing Equations of Fluid Dynamics

Various systems of units were in use last century [43]. This circumstance introduced some inevitable difficulties in transition from one system to another. To facilitate an international collaboration it was agreed in earliest 60's that only the *International System of Units* (SI) is a preferable unit system for scientific and all other purposes. In this work the SI system is also used as a basis for dimensional analysis. The SI-system has the following fundamental units:

Unit	Dimension notation	
Meter	[m]	unit of length
Second	[s]	unit of time
Kilogram	[kg]	unit of mass
Kelvin	[K]	unit of absolute temperature
Ampere	[A]	unit of electricity current
Mol	[mol]	unit of substance
Candela	[cd]	unit of light intensity

Once the the fundamental units have been chosen, the dimensional analysis proceeds with the listing of the *independent variables*, i.e. the parameters that control the physical problem. This is one of the most important steps in dimensional analysis. If the number of the variables exceeds the number of variables involved into the problem, the final equation will contain some superfluous variables. And vice-versa, in case when the number of variables is not enough for the analysis, the incomplete equations can emerge which result in false inferences. According to [43], the dimensional analysis is based on the hypothesis that a solution of a problem does not depend on the choice of independent units only if the independent variables are correctly selected.

In many cases it is easier to work with *dimensionless quantities* combined from independent variables rather than dealing with independent variables themselves. In such cases the dimensional analysis is just reduced to the application of the Buckingham theorem [28] discussed below.

Buckingham Theorem (II theorem) Let some physical phenomenon depends on n physical variables. It is assumed that these variables can be presented in terms of k independent fundamental physical quantities. If so, the original expression representing this phenomenon is equivalent to an equation involving a set of $p = n - k$ dimensionless variables constructed from the original variables. In a more formal mathematical presentation the Buckingham theorem reads: *"Every dimensionally homogeneous equation can be transformed into an equation involving only dimensionless products"*.

Let us illustrate the idea of the II theorem using the considered above example on the drag force applied to a plate in a fluid flow. The independent variables in this experiment are: F, U, L, ν, ρ while the independent fundamental physical units are as follows: time, length, mass. Thus the number of involved dimensionless variables is $p = 5 - 3 = 2$.

Combining the dimensionless products in such a way that

$$P = \frac{F}{\rho U^2 L^2}, \quad Re = \frac{UL}{\nu},$$

we obtain the relation in the following form:

$$f(P, Re) = 0,$$

from which we can express the drag force explicitly:

$$F = \rho U^2 L^2 f(Re).$$

The comparison of this formula with (2.20) shows their similarity. In general, it makes no difference what method is used, i.e. the direct dimension analysis or the Buckingham theorem. Being applied correctly both give the same result. However, the choice of every particular method is the matter of convenience and personal preferences.

2.7 Turbulence Theory

This section provides some background introduction to the characteristics of turbulent fluid flows which is necessary for the further analysis.

The distinct difference between laminar and turbulent motions becomes evident if these two flow regimes are compared. In case of turbulent motion the main characteristics of a fluid flow, namely velocity and pressure, oscillate around some mean value. Such oscillations appear in a large number of energetic vortexes of different spatial scales and strong velocity gradients chaotically moving with respect to each other, whereas the laminar counterpart reveals rather smooth and regular flow patterns without sharp velocity gradients and oscillations. From the first glance, it is clear that to describe turbulent fluid flows it would, in principle, suffice to solve the Navier-Stokes equations (2.12). As long as these equations describe all hydrodynamic phenomena, from microscopic to global scales, the simple idea is to develop an efficient solver for (2.12) in order to predict the characteristics of turbulent fluid. However, the problem lies in the fact that the wide variety of physical processes must simultaneously be resolved at different spatial scales. The range of length scales which must be analyzed can cover several orders (extend from kilometers down to micrometers, for instance). If one would attempt to solve such a problem on a computer, the grid size of the model should be directed on the resolution of the smallest scales. Thus, the direct computation makes the investigation of the turbulent flow time consuming. The feasibility and reasonableness of this approach is highly questionable. There is, however, an alternative more efficient way of turbulent flow investigations which is based on the physical understanding of the energy cascading in turbulent fluid flow. This method is briefly discussed below.

The general structure of fluid motion is determined by the Reynolds number. Flows with small values of Re , mostly slow flows, appear to be laminar. The liquid particles flow along the streamlines arranged in parallel levels without any distortion between the

2 Governing Equations of Fluid Dynamics

layers. As Re grows the flow becomes less stable, and when the Reynolds number reaches the critical value Re_c , the pressure and velocity start to fluctuate. The fluctuations appear both in velocity values and its directions, so the motion seems completely chaotic [93] or *turbulent* [60].

Obviously, it is quite difficult to give a precise description of the turbulent behavior in terms of \mathbf{v} , \mathbf{x} and p . Some useful laws, however, are easily deduced by taking the mean values of these parameters. Since the changes of state of any physical system is convenient to describe in terms of energy, we start our analysis with the derivation of the relation between fluid flow energy losses and its turbulization. In doing so, let us consider some fluid volume of unit mass in such a flow. Note that the energy flux ϵ outgoing from this volume depends on average changes of velocity which take place within the volume. Hence the value ϵ should be a combination of velocity and length scales.

Let t^* be the longest pulsation's lifetime period, define \mathbf{v}^* as a time-averaged quantity of velocity over a time period $T \gg t^*$, then the difference $\mathbf{v}' = \mathbf{v} - \mathbf{v}^*$ represent the velocity pulsations. Define $\Delta v = \max |\mathbf{v}'|$ as the scale of the maximum fluctuation of speed. Combining these scales together and applying the Π theorem we deduce estimation of average losses of energy, ϵ , that dissipates by unit mass per unit of time, i.e.

$$\epsilon \sim \frac{(\Delta v)^3}{L} . \quad (2.21)$$

A remarkable fact is that energy dissipation in fluid occurs just due to viscosity which, however, is not included in (2.21). The explanation lies in a simple reasoning: the impact of viscosity is determined by a *local* Reynolds number combined from characteristic scales of local fluctuations of the current.

Let λ and Δv_λ are the fluctuation scales of length and velocity, respectively. It is clear that the growth of λ and Δv_λ increases the value of the local Reynolds number, $Re_\lambda = \lambda \Delta v_\lambda / \nu$. Consequently, the impact of viscosity reduces on large scales. The viscous effects become significant when $Re_\lambda \sim 1$. Such Reynolds numbers correspond to small scaled fluctuations called *microscale of turbulence*. Thus we conclude that the viscosity becomes insignificant on large scales where the turbulent flow behaves as an ideal fluid. As λ becomes smaller, the impact of viscosity grows, and the energy losses become more remarkable. So, the energy is 'transferred' from large- to small-scale fluctuations where it finally dissipates due to viscosity.

To find the relation between the microscales characteristics of turbulence and average losses of energy ϵ one can apply the same reasoning as we used for the derivation of equation (2.21). In the present case it leads to the formula:

$$\Delta v_\lambda \sim (\epsilon \lambda)^{1/3} . \quad (2.22)$$

In literature this expression is also known as the Kolmogorov-Obukhov's law. Its substitution into (2.21) gives a new expression

$$\frac{\Delta v_\lambda}{\Delta v} \sim \left(\frac{\lambda}{L} \right)^{1/3} , \quad (2.23)$$

which establishes a bridge between large- and small-scales of turbulence. It also follows from the expression (2.23) that any two turbulent flows are always similar in statistical sense up to some scaling parameter which appears in scales of length and velocity. Rewriting the local Reynolds number Re_λ in terms of equation (2.23) one can finally obtain the relation between local and standard Reynolds numbers:

$$Re_\lambda = \frac{\lambda \Delta v_\lambda}{\nu} = \frac{\lambda^{4/3} \Delta v}{\nu L^{1/3}} \sim Re \left(\frac{\lambda}{L} \right)^{1/3}. \quad (2.24)$$

Since the viscosity effects are significant for the motions at small scales with $Re_\lambda \sim 1$, equating (2.24) to 1, we obtain

$$\lambda_0 \sim L/Re^{3/4}. \quad (2.25)$$

While λ becomes smaller, the viscosity effects are increased, and starting from $\lambda = \lambda_0$ their impact becomes so strong that fluctuations with scales $\lambda < \lambda_0$ can not appear at all. Thus, λ_0 is the smallest microscale of fluctuations which can be found in turbulent flow. This scale is usually referred as the Kolmogorov's scale. The idea that there is a clear dependence between an impact of viscosity and the scale of turbulence λ leads to the conclusion that the principal part of energy of fluid flow should concentrate in a large scaled (low frequency) fluctuations. In terms of formulas this idea has a very clear presentation. Let us introduce the wave number of fluctuations as $k \sim \lambda^{-1}$. Substituting it into (2.21), we immediately derive that

$$\Delta v_k \sim \left(\frac{\epsilon}{k} \right)^{1/3}. \quad (2.26)$$

The same expression in terms of energy reads:

$$E(k) \sim \left(\frac{\epsilon}{k} \right)^{2/3}. \quad (2.27)$$

This is a spectral presentation of the Kolmogorov-Obukhov's law (2.22). In the context of the present research the most important conclusions from this theory is that the velocity and energy spectra of turbulent flows are located in a low-frequency band, as it is shown in Fig. 2.1. The last circumstance has fundamental implications for the method developed here. They are discussed in the rest of the present study in more detail.

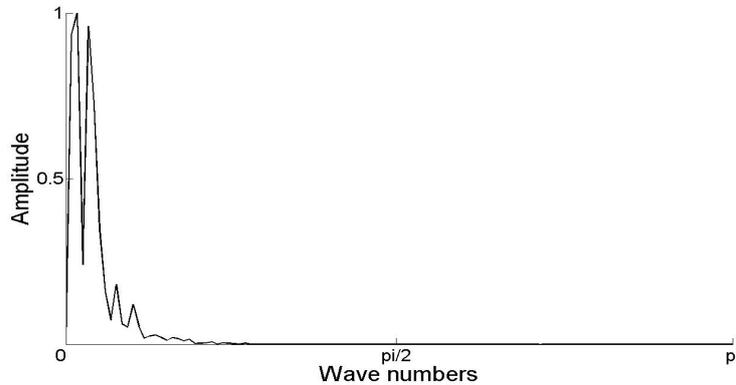


Figure 2.1: Typical distribution of the energy spectrum calculated for a turbulent fluid flow.

2.8 Boundary Conditions

Solution of hydrodynamic equations requires setting of some concrete boundary conditions (BC). A large variety of boundary conditions is considered in theory for usual PDEs, although in the case of hydrodynamic problems there are only a few of them. This disproportion accounts for a complexity of turbulent fluid flows. In many cases formulation of accurate and justified BCs is not obvious. Let us consider the most typical boundary conditions commonly used in fluid mechanics. They are as follows.

Solid wall The boundary is assumed to be rigid and fluid-proof, which means that the shape of wall remains always constant, and no liquid particles can pass through it. Due to the effects of viscosity, fluid particles 'stick' to the wall. As a result, the boundary conditions on the rigid wall $\partial\Omega$ read:

$$\mathbf{v}|_{\partial\Omega} = 0 . \quad (2.28)$$

As a consequence of the 'non-slip' condition, all velocity components degrade fast close to the boundary. In a boundary layer near solid walls the velocity gradients become quite high so that the vorticity reaches its maximum right on the wall. Thus, one can say that solid surfaces generate vorticity.

Free surface The fluid motions often develop in contact with gaseous or liquid media. In such situations the boundary conditions for a moving fluid have to be set on the interface separating this fluid flow from other media. The sea surface is a good example of such a boundary (Fig. 2.2 left). The position of this surface is commonly not fixed, although it can be fluid-proof. Quite natural BCs for such surfaces would be vanishing of the normal component of the velocity vector on the boundary, while the tangential remains nonzero. Correct formulation of such BCs requires the knowledge of the surface topography at every moment of time which is not always available. However, in some cases when surface deformations are infinitesimally small with respect to the scales of the considering phenomena, one can neglect all surface perturbations and consider the free surface as a rigid boundary although without sticking of fluid particles to it.

Symmetry (zero fluxes) Some fluid flows possess the property of symmetry. For instance, the motion in convective cells [29, 69] given in Fig. 2.2(right) can be classified as a symmetrical fluid flow. The concept 'symmetry' in such fluid flow means that the velocities of particles moving equidistantly from the symmetry line are equal. Hence, the normal component of velocity v_n on the symmetry line ∂S should be equal to zero. Then the corresponding BCs for the vorticity and velocity on the symmetry line read:

$$v_n|_{\partial S} = 0 , \quad (2.29)$$

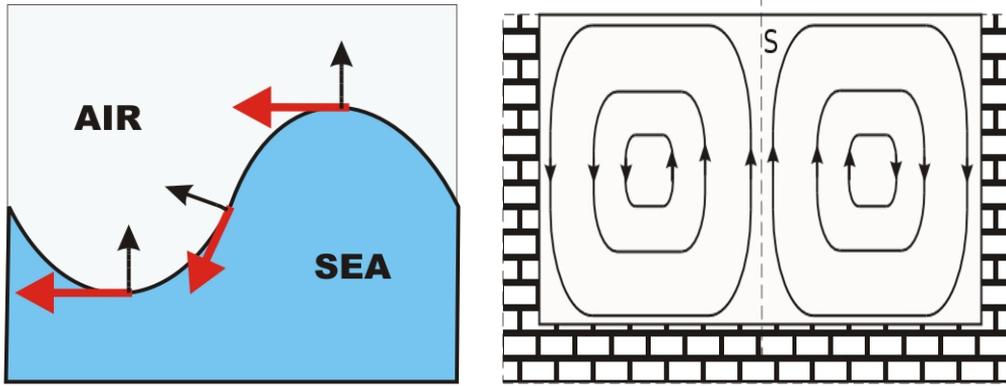


Figure 2.2: **Left:** The velocity distribution on the free sea surface (red arrows). The normals are denoted by black arrows. **Right:** The trajectory profiles of a two-cell convective flow in a rectangular tank with free surface. Fluid motions in both cells is symmetrical with respect to the symmetry line (dashed).

Steady flow at infinity Let us consider the hydrodynamic phenomena developing far away from some natural boundaries like solid walls or free surfaces. Such an example could be the fluid flow around a submarine moving in the deep ocean. Since the distribution of velocity is unknown, it is difficult to formulate the correct BCs on the fluid boundary surrounding the submarine. However, assuming that the considered motion takes place in the infinitely large volume we can state that all perturbations and oscillations occurring in the near-field of the object must vanish at the infinity due to ordinary viscosity and radial energy spreading. As a result, the fluid flow should become constant or zero everywhere in the far-field. Mathematically such a condition can be formulated as follows:

$$\begin{aligned} \mathbf{v}(\infty) &= \text{const} , \\ \boldsymbol{\omega}(\infty) &= 0 . \end{aligned} \tag{2.30}$$

Free boundaries Let O is some volume filled with fluid. Assume that the studied hydrodynamical phenomenon takes place in a smaller volume Ω of a fixed form which is an inner part of O . The boundary $\partial\Omega$ of Ω is fixed but not fluid-proof, so the fluid can freely move in or out of Ω . Suppose that the velocity and vorticity distribution in O is unknown. We want to formulate such boundary conditions for velocity and vorticity on $\partial\Omega$ with acceptable accuracy which approximate the BCs on $\partial\Omega$ for all possible types of flows occurring in O .

Let \mathbf{n} be a unit vector normal to $\partial\Omega$ and $\mathbf{x}_0 \in \partial\Omega$ and let $\Delta\mathbf{x}$ be an infinitesimal increment of \mathbf{x}_0 in the direction collinear to \mathbf{n} . Consider the first two terms of the \mathbf{v} Taylor series near the point $(\mathbf{x}_0 + \Delta\mathbf{x})$:

$$\mathbf{v}(\mathbf{x}_0 + \Delta\mathbf{x}) = \mathbf{v}(\mathbf{x}_0) + \frac{\partial\mathbf{v}(\mathbf{x}_0)}{\partial\mathbf{n}}\Delta\mathbf{x} + O(\Delta\mathbf{x}^2).$$

2 Governing Equations of Fluid Dynamics

Suppose that $O(\Delta\mathbf{x}^2)$ has weak impact on the flow in comparison with two first terms, so that the behavior of \mathbf{v} near $\partial\Omega$ becomes close to linear and we can formulate the general boundary conditions for the unknown flow as follows:

$$\begin{aligned}\frac{\partial\mathbf{v}(\mathbf{x}_0)}{\partial\mathbf{n}} &= \frac{\mathbf{v}(\mathbf{x}_0 + \Delta\mathbf{x}) - \mathbf{v}(\mathbf{x}_0)}{\Delta\mathbf{x}}, \\ \frac{\partial\boldsymbol{\omega}(\mathbf{x}_0)}{\partial\mathbf{n}} &= 0.\end{aligned}\tag{2.31}$$

The 'correctness' of such boundary conditions depends on how accurate the linear expansion approximates the true velocity distribution on $\partial\Omega$. For instance, linear approximation fits well to laminar flows where all functions are nearly linear. A turbulent flow consists of a large number of vortexes of various size for which this approximation is less accurate.

Let us estimate an impact of such an 'inaccuracy' of linear approximation using the following reasoning. According to Sec. 2.7, the characteristic scale of velocity pulsations relates to its length scale l as $\Delta v \sim l^{1/3}$. Since the propagation distance of a single pulsation is of the order of its characteristic scale l , we can assume that small-scale pulsations start to vanish when the distance to $\partial\Omega$ is less than l . The large-scale pulsations still remain significant since the velocity frequencies are small and they nearly fit the linear approximation. Thus, we arrive at a conclusion that the impact of 'inaccuracy' appears only at the distances of l related to small-scale pulsations.

3 Basic Image Processing in Experimental Fluid Mechanics

This chapter outlines some general principles related to the basis of the particle velocimetry measuring technique. The main target of this review is to identify major possible sources of noise and errors appearing in the course of measurements and data processing. Most important, it focuses on the characteristics of some noise-removal post-processing methods and highlights their strengths and weaknesses. For more comprehensive analysis of the particle image velocimetry methods we refer the reader to the background material [40, 46, 49, 61, 63, 66, 68, 98].

3.1 Particle-Based Fluid Motion Recording Techniques and Background

The starting point of the denoising method developed here is the acquisition of some original vector fields obtained experimentally. These vector fields are used as input data sets for conducting the denoising procedure. The most popular measuring techniques used nowadays on a regular basis in experimental fluid dynamics are the particle-based registration methods. In this section two versions of this method are briefly outlined, and only some major ideas relevant to the present work are discussed.

First of all let's consider a typical experimental set-up. Imagine some restricted volume filled with fluid or gas (for the simplicity the terms fluid and gas are treated hereafter equally as 'fluid', as it was done in Chap. 2). In practice for visualization of the fluid flows some special tiny luminous particles [61] are seeded in the medium, see Fig. 3.1. The fluid is moving in the volume together with these seeded particles. The size of particles and density of seeding are chosen in such a way that they follow the flow and almost identically copy the trajectories of fluid motion.

High-rate laser pulses are used in the experimental domain for visualization of the particle motions. The positions of particles in space is recorded with a high-speed digital camera (or several cameras) which is synchronized with the laser pulses. Analyzing the images and comparing the positions of the particles obtained at two subsequent moments of time one can estimate the velocity of the particles (see Fig. 3.1).

3.2 Particle Image Velocimetry (PIV)

Particle Image Velocimetry (PIV) is a challenging area of experimental hydrodynamics which is often used in many industrial applications. This section gives some introductory

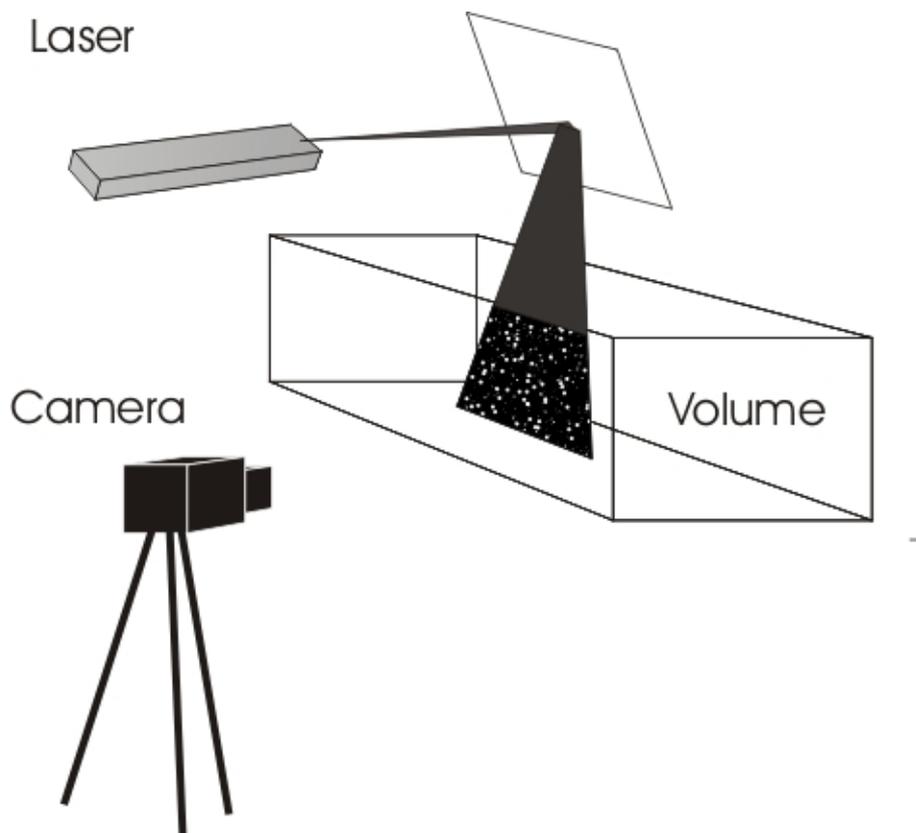


Figure 3.1: A principle scheme of the particle-based visualization set up. Given some experimental volume filled with moving fluid. A laser pulse reflected from a mirror alights the fluorescent seeding distributed in this volume. A high-speed camera synchronized with the laser pulses records the position of these seedings.

information on these principles and indicates some possible reasons and sources of errors in the PIV output fields. Let's focus only on the fundamental ideas of the method omitting the details (much more specific information can be found in special literature).

The general idea of the PIV method is as follows. Suppose two images of particles are taken at moments of time t_0 and $t_0 + t$. The objective is to determine the displacements of the same particles displayed in these two frames. To this end, the PIV records are split into a set of small sub-domains, interrogation windows, such that all particles in these areas have almost the same displacement within the time interval $[t_0, t_0 + t]$. Obviously, the fulfillment of this condition requires a high concentration of seeding in the volume. In other words, the method works well when a group of particles located in the first interrogation window appears in the second image in the same interrogation window and almost in the same positions, shifted a little bit at a some distance, \mathbf{d} . Thus, instead of working with the displacements of every single particle (which is quite problematic and questionable in terms of accuracy in the case of high particle concentration) the PIV task is reduced to the computation of an average displacement of the group of particles.

For the simplicity of presentation only two-dimensional fluid flows are considered here. In other words, it is assumed that all particle displacements take place in a laser sheet oriented parallel to XY -plane within a layer with thickness ΔZ . The motion is recorded by a single camera which is oriented along OZ direction perpendicular to the XY -plane. The camera's magnification lens is assumed to be ideal, i.e. it has absolutely flat focal plane with the same magnification factor M for all points on this plane. The capital letters \mathbf{X}_i , \mathbf{D}_i hereafter denote the real coordinates and displacement of particles in the experimental volume, respectively, and the small letters $\mathbf{x}_i, \mathbf{y}_i$ are the appropriate coordinates of the corresponding particle i in the image. The relation between $\mathbf{X}_i, \mathbf{D}_i$ and $\mathbf{x}_i, \mathbf{d}_i$ is given by simple linear formulas: $\mathbf{x}_i = M\mathbf{X}_i$, $\mathbf{d}_i = M\mathbf{D}_i$. In addition, it is assumed that the laser sheet is relatively thin so that the particle size has no visible effect on the image.

Let us define an interrogation volume as a part of three-dimensional experimental domain which is projected by the camera onto the image into a two-dimensional interrogation window. Let's assume also that N particles have been imaged within the some interrogation window. The state of this ensemble at a given moment of time t is described by the vector:

$$\Gamma = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \cdot \\ \cdot \\ \mathbf{X}_N \end{bmatrix} \quad \text{where } \mathbf{X}_i = (x_i, y_i, z_i)^T . \quad (3.1)$$

The image intensity field of the first exposure can be expressed by the formula:

$$I = I(\mathbf{x}, \Gamma) = \tau(\mathbf{x}) * \sum_{i=1}^N V_0(\mathbf{X}_i) \delta(\mathbf{x} - \mathbf{x}_i) , \quad (3.2)$$

where τ is a point spread function which is assumed to be a Gaussian-type function with

3 Basic Image Processing in Experimental Fluid Mechanics

respect to coordinates, i.e.

$$\tau = \frac{8\tau_0}{\pi d_\tau^2} \exp\left(-\frac{8\|\mathbf{x}\|_2^2}{d_\tau^2}\right) .$$

Values τ_0 and d_τ denote the integral intensity of the point spread function and particle image diameter, respectively. Values $V_0(X_i)$ and $\delta(\mathbf{x} - \mathbf{x}_i)$ in (3.2) are the transfer function giving the light intensity of the image of an individual particle i , and the delta function, respectively. The function $V_0(X_i)$ describes the shape, extension and location of an actual interrogation volume. According to [68] a simplified version of V_0 reads:

$$V_0 = V_0(X, Y, Z) = W_0(X, Y)I_0(Z) ,$$

where $I_0(Z)$ is the intensity profile of a laser sheet in Z -direction, and $W_0(X, Y)$ is the interrogation window function. If the laser beam has a top-hat intensity distribution, then V_0 can be approximated as a rectangular box:

$$I_0(Z) = \begin{cases} I_Z & \text{if } |Z - Z_0| \leq \Delta Z_0/2 , \\ 0 & \text{elsewhere ,} \end{cases} \quad (3.3)$$

and

$$W_0(X, Y) = \begin{cases} W_{X,Y} & \text{if } |X - X_0| \leq \Delta X_0/2 \text{ and } |Y - Y_0| \leq \Delta Y_0/2 , \\ 0 & \text{elsewhere.} \end{cases} \quad (3.4)$$

Let's assume that particle images do not overlap. If so, since $\tau(\mathbf{x} - \mathbf{x}_i) = \tau(\mathbf{x}) * \delta(\mathbf{x} - \mathbf{x}_i)$, equation (3.2) appears in a new form:

$$I = I(\mathbf{x}, \Gamma) = \sum_{i=1}^N V_0(X_i) \tau(\mathbf{x} - \mathbf{x}_i) . \quad (3.5)$$

Expression (3.5) is the image intensity field for a group of particles located in the interrogation volume V_0 at a moment of time t_0 . Let's assume now that this group of particles has been shifted by the time moment $t_0 + t$ at a distance \mathbf{d} . If so, the previous expression is transformed into:

$$I_t = I_t(\mathbf{x}, \Gamma) = \sum_{i=1}^N V_0(X_i + \mathbf{D}) \tau(\mathbf{x} - \mathbf{x}_i - \mathbf{d}) . \quad (3.6)$$

In the case when the light sheets and window characteristics at t_0 and $t_0 + t$ are identical, the cross-correlation function of two interrogation areas can be written as follows:

$$R_{II}(\mathbf{s}, \Gamma, \mathbf{D}) = \frac{1}{a_I} \sum_{i,j} V_0(\mathbf{X}_i) V_{0t}(\mathbf{X}_j + \mathbf{D}) \int_{a_I} \tau(\mathbf{x} - \mathbf{x}_i) \tau(\mathbf{x} - \mathbf{x}_i + \mathbf{s} - \mathbf{d}) d\mathbf{x} , \quad (3.7)$$

where \mathbf{s} is a separation vector in the correlation plane which provides the correspondence of τ at a different moments of time; a_I is the window area, and \mathbf{D} is a constant displacement

of all particles inside the interrogation volume. According to [68] this expression can be reduced as follows:

$$R_{II}(\mathbf{s}, \Gamma, \mathbf{D}) = \sum_{i \neq j} V_0(\mathbf{X}_i) V_0(\mathbf{X}_j + \mathbf{D}) R_\tau(\mathbf{x} - \mathbf{x}_i + \mathbf{s} - \mathbf{d}) + R_\tau(\mathbf{s} - \mathbf{d}) \sum_{i=1}^N V_0(\mathbf{X}_i) V_0(\mathbf{X}_i + \mathbf{D}), \quad (3.8)$$

where

$$R_\tau(\mathbf{s}) = \exp\left(\frac{-8|\mathbf{s}|^2}{(\sqrt{2}d_\tau)^2}\right) \frac{1}{a_I} \int_{a_I} \tau\left(\mathbf{x} - \mathbf{x}_i + \frac{\mathbf{s}}{2}\right) d\mathbf{x}.$$

One can split the correlation into three parts:

$$R_{II}(\mathbf{s}, \Gamma, \mathbf{D}) = R_C(\Gamma, \mathbf{D}) + R_F(\mathbf{s}, \Gamma, \mathbf{D}) + R_D(\mathbf{s}, \Gamma, \mathbf{D}),$$

where

$$R_C(\Gamma, \mathbf{D}) = R_\tau(0) \sum_{i,j=1}^N V_0(\mathbf{X}_i) V_0(\mathbf{X}_j + \mathbf{D}),$$

is the convolution of the mean intensities of I and I_t , and

$$R_F(\mathbf{s}, \Gamma, \mathbf{D}) = R_\tau(\mathbf{x} - \mathbf{x}_i + \mathbf{s} - \mathbf{d}) \sum_{i \neq j} V_0(\mathbf{X}_i) V_0(\mathbf{X}_j + \mathbf{D}),$$

is the fluctuating noise component resulted from the $i \neq j$ terms. The term $R_D(\mathbf{s}, \Gamma, \mathbf{D})$ represents the component of the cross-correlation function that corresponds to the correlation of images of particles obtained from the first exposure with the identical particles obtained from the second exposure ($i = j$ terms):

$$R_D(\mathbf{s}, \Gamma, \mathbf{D}) = R_\tau(\mathbf{s} - \mathbf{d}) \sum_{i=1}^N V_0(\mathbf{X}_i) V_0(\mathbf{X}_i + \mathbf{D}). \quad (3.9)$$

It is clear that the displacement correlation function reaches its maximum when $\mathbf{s} = \mathbf{d}$. The location of this maximum gives an average in-plane displacement and thus average u and v components of the velocity vector inside the flow.

If the tested flow is relatively slow, the average velocity vector field coincides with the real velocity vector field almost perfectly. However, with the increase of the flow speed this similarity between the averaged estimates and real vector field is violated. Moreover, inevitable restrictions of the measuring technique and imperfection of the experimental set-up, as well as the assumptions taken in the data processing, - all this leads to a situation when the recorded vector field contains certain amount of noise (the more detailed discussion of this issue, i.e. the noise origin, is included in Sec. 3.5).

3.3 Particle Tracking Velocimetry

As distinct from the cross-correlation PIV method which operates with groups of particles, the Particle Tracking Velocimetry (PTV) method controls the motion of every single particle separately. It is assumed in PTV that a particle displacement between two sequent

camera shots is negligibly small in comparison with the typical distance between particles in the entire volume. If so, one can easily identify the very same particles in two images. After that the task of finding of the particle displacement becomes trivial. Of course, the method works well only in case of 'smooth' motions, i.e. when particles change their motion smoothly, without rapid changes of their trajectory between two camera shots. It is clear that this assumption excludes the cases of 'spurious' particle images appearing in different places of sequent frames.

Conventionally, the PTV procedure can be split into two steps: at the first stage one tries to identify every single particle in both images, while at the second stage the correspondence between the same particles in two frames is determined. The particle detection can be performed in two ways applying region growing approach or particle mask correlation [63]. In the region growing approach the local intensity maxima are detected in the photograph plane. Then the pixels neighboring to each maximum are checked whether their intensity function is above a certain threshold; if this is the case, they are added to the individual regions. At the final stage the centroids of the individual regions are computed. The particle mask correlation uses the Gaussian particle mask (size of ca. 3x3 px) which is entered on all pixels. The peaks of this mask indicate the position of particles.

After the positions of all particles have been identified, the procedure determines the correspondence between the same particles on sequent frames. Two methods, i.e. the cross-correlation technique for individual particles [46] and nearest-neighbor search [49], are commonly used for this purpose. The questions related to the noise appearance in the PTV method are discussed in Sec. 3.5 in greater detail.

3.4 Two-Dimensional Variational Approach

A fundamental assumption for this section is that an intensity function (usually referred as a gray value) is conserved over time. Let $I(\mathbf{x}_0, t_0)$ is the light intensity recorded at the some part of fluid with coordinates (\mathbf{x}_0, t_0) , and let $I(\mathbf{x}_0 + \Delta\mathbf{x}, t_0 + \Delta t)$ is the similar value at the same point but recorded at a new moment of time $t_0 + \Delta t$ moved to a position $\mathbf{x}_0 + \Delta\mathbf{x}$. If so, the following condition must be satisfied:

$$I(\mathbf{x}_0, t_0) = I(\mathbf{x}_0 + \Delta\mathbf{x}, t_0 + \Delta t) . \quad (3.10)$$

This equation implies that the luminous seeding trajectories copy exactly the stream-functions of fluid motion so that the gray value behaves as incompressible fluid.

Let's expand the right hand of (3.10) into a Taylor series around the point (\mathbf{x}_0, t_0) . Suppose that two moments of time t_0 and $t_0 + \Delta t$ are close enough so that the velocity of particles within the interval $[t_0, t_0 + \Delta t]$ can be treated constant. The basic contribution in $I(\mathbf{x}_0 + \Delta\mathbf{x}, t_0 + \Delta t)$ expansion gives two first terms of the Taylor series. Taking into account that $\Delta\mathbf{x} = \mathbf{v}\Delta t$, one can write that

$$\begin{aligned} I(\mathbf{x}_0 + \Delta\mathbf{x}, t_0 + \Delta t) \approx & I(\mathbf{x}_0, t_0) + \partial_x I(\mathbf{x}_0, t_0) \mathbf{v} \Delta t + \partial_y I(\mathbf{x}_0, t_0) \mathbf{v} \Delta t + \partial_t I(\mathbf{x}_0, t_0) \Delta t = \\ & I(\mathbf{x}_0, t) + \nabla I(\mathbf{x}_0, t_0) \cdot \mathbf{v} \Delta t + \partial_t I(\mathbf{x}_0, t_0) \Delta t , \end{aligned} \quad (3.11)$$

where ∂_x and ∂_y mean the partial derivatives over each component of $\mathbf{x} = (x, y)^T$. Substitution (3.11) into (3.10) leads to:

$$\partial_t I(\mathbf{x}_0, t_0) + \nabla I(\mathbf{x}_0, t_0) \cdot \mathbf{v} = 0 . \quad (3.12)$$

This expression is called *the Brightness Change Constraint Equation* (BCCE) which defines the relation between the grey values and fluid velocities (for more details see [35]). Note, however, that (3.12) is not well posed because any vector field with components $\nabla I(\mathbf{x}_0, t_0) \cdot \mathbf{v} = -\partial_t I(\mathbf{x}_0, t_0)$ satisfies this equation. Since the velocities of fluid flows in most cases are smooth functions, let's require the smoothness of \mathbf{v} . Therefore, the aforementioned equation should be converted into a minimization problem which can be expressed as follows:

$$J = G + \lambda Q , \quad (3.13)$$

where

$$G = \int_{\Omega} (\partial_t I(\mathbf{x}_0, t_0) + \nabla I(\mathbf{x}_0, t_0) \cdot \mathbf{v})^2 d\mathbf{x} .$$

Here λ is a Lagrange multiplier or adjustable parameter, and Q is a functional representing constraints called 'regularizer', which penalizes discontinuities, ramps and other undesirable features of the output vector field.

Quality of the output result depends on the fact how accurately and correctly the 'regularizer' has been chosen. In general, there are two types of regularizers: those that rely on some particular models of noise, and those that exploit the physical properties of the flow. Because of the complexity of the hydrodynamical equations, the methods based on the second class of regularizers are usually more complicated in sense of numerical implementation, although according to [38] they give more realistic output.

The simplest first-class regularizer was suggested by Horn and Schunck [40]. For two-dimensional flows ($\mathbf{v} = \mathbf{v}(u, v)^T$) it reads:

$$Q = \int_{\Omega} (|\nabla u|^2 + |\nabla v|^2) d\mathbf{x} .$$

This regularizer penalizes strong gradients. High-frequency flow oscillations are usually treated by this regularizer as a noise. As a result of that, it can truncate or over-smooth some real physical structures which really present in the flow.

A higher-order *div - curl* regularization as that given in [107, 108], i.e.

$$Q = \int_{\Omega} (|\nabla \text{curl}(\mathbf{v})|^2 + |\nabla \text{div}(\mathbf{v})|^2) d\mathbf{x} ,$$

makes the output vector field less smoothed. It preserves the real flow fragments that are reduced by the Horn and Schunck regularization, but, on the other hand, it is more sensitive to noise and can create some nonphysical structures. The regularizers that take into account physical properties of fluid flow are given in [72, 73, 74]. Their disadvantage is that they are valid in a restricted number of cases, and in addition they are sensitive to noise. For instance, regularizer given in [72] is based on the hydrodynamical equation

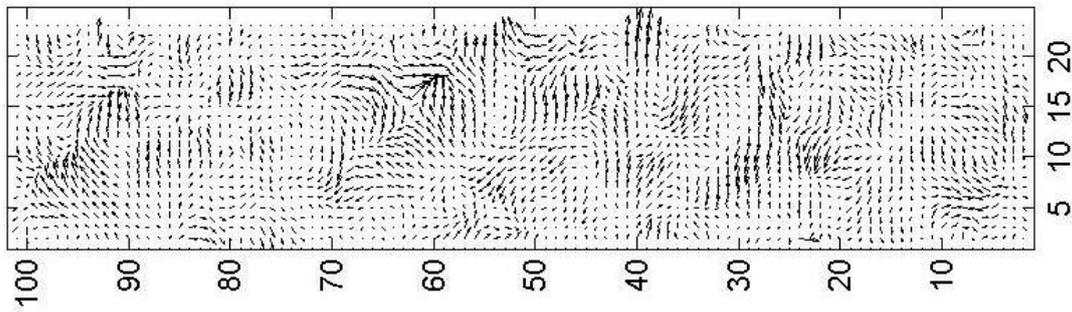


Figure 3.2: Corrupted and noisy velocity vector field recorded in laboratory experiments with a boundary layer.

which is satisfied only in the cases of nearly laminar flows. The methods based on the regularizers suggested in [74, 73] appear to be sensitive to errors remaining after the previous computations which are used for flow prediction. Moreover, the applied zero Neumann boundary conditions in these methods restrict their implementation to the cases when the investigated fluid flow is surrounded by solid walls.

3.5 Noise Origin and Data Post-processing

Theoretical issues on the fundamentals of the PIV-method discussed above are relevant to this work for better understanding of all possible sources of the noise appeared in the course of data recording and processing. Hypothesis, limitations and assumptions taken in the methodology inevitably affect an accuracy of the method. Awareness of all the details is necessary for greater insight into a problem of signal denoising.

Obviously, any experimental data set contains definite level of noise/errors which can't be removed without application of some specific post-processing procedures. These errors have different origin. In general, one can distinguish two most typical groups of errors: the first one appears from imperfection of measuring technique; the second type of errors arises from incorrect assumptions and simplifications taken in theoretical background of the experimental methodology. Such kind of errors are called *systematic errors*, and they peculiar to every measurement. The most commonly encountered noise appearing in the output data relates to spurious vectors, losses of correlation, biases, sparsity and non-homogeneity of the data set. Figure 3.2 represents an example of a corrupted and noisy vector field obtained in experimental boundary layer measurements [81]. Some possible sources of errors usually encountered in image velocimetry technique have already been mentioned in the introductory section Chap. 1 although without going in depth. This section gives more detailed explanation of the noise origin and its possible consequence for the final output data set.

Particle overlapping and non-homogeneous seeding density are two major reasons that usually cause great errors in PIV applications. In the case of a very high seeding concentration it is quite probable that the particles located closer to the frontal side of a laser sheet might overlap with particles situated behind them. This circumstance can lead to a situation when some particle images which are not presented in the first image appear in the second one. The consequences of the appearance of 'redundant' particles can result in:

- losses of correlation or wrong velocity estimation in the regions where they appear for cross-correlation computation technique (left panel in Fig. 3.3);
- local violation of the constancy contrast principle and appearance of spurious vectors resulted from BCCE-based computation procedure;
- over- or underestimation of local vectors in the course of the computing procedure.

It is clear that this undesirable effect of particle overlapping can be mitigated only through the reduction of seeding density, which, however, can lead to an opposite effect, i.e. to sparsity of the output data.

Another problem of the PIV method comes from possible inhomogeneity of seeding concentration. If this is the case, i.e. some parts of experimental domain have low particle density, the velocity vector field recovered from these areas usually has lower accuracy in comparison with other parts. For instance, it is known from [3] that for an accurate and robust application of the cross-correlation method the minimum number of particles presenting in the interrogation window should be not less than 10 in order to provide the validity of measurements at the level of 90 – 95%. If this is not true, i.e. due to various reasons the image particle concentration in some interrogation windows is much less than 10, the fluid velocity recovered from such areas will be not enough accurate or even wrong at all.

Errors related to the assumptions concerning the characteristics of fluid flow and appropriate choice of the window size are the most traditional errors which inevitably appear in the output field. They affect the cross-correlation techniques and make the velocity recovery worse. As it was mentioned above, the conventional cross correlation method is based on the assumption that all particles within the interrogation volume have equal velocity, which in practice is not always true. Such an assumption leads to the fact that the velocity variations within one interrogation volume are completely neglected, which, in turn, results in biases and sharp ramps in the output vector fields (right panel in Fig. 3.3). The modern cross-correlation methods based on the interpolation of the velocity gradients within the interrogation window or window adaption techniques reduce the impact of such errors. The discussion about these methods and the related errors is given below.

Thus, the correct definition of the size of interrogation window is highly important for the robust recovery of the fluid flow velocity: larger number of particles included in the window provides better conditions for an accurate application of the cross-correlation

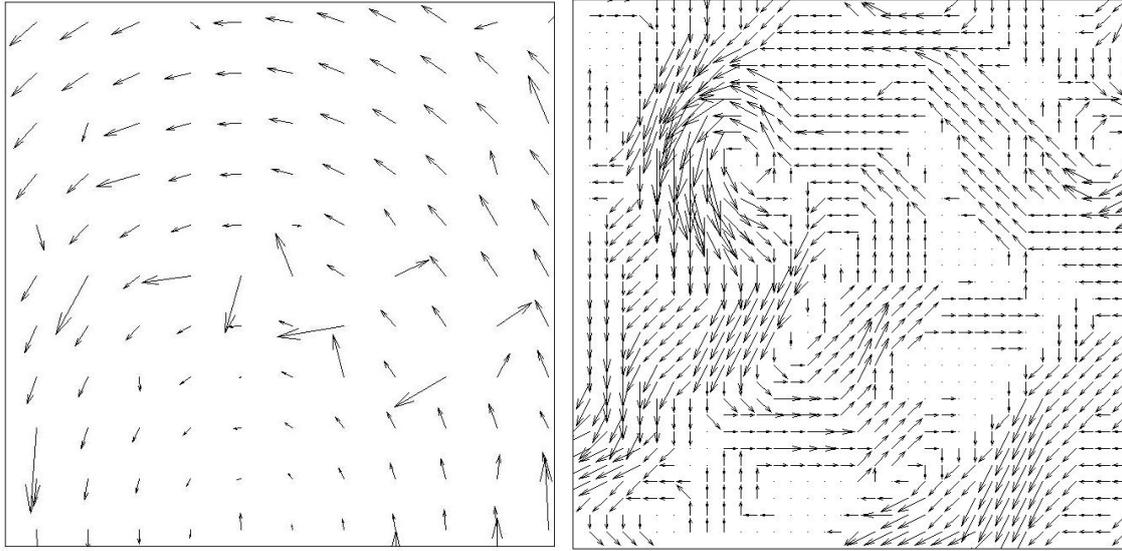


Figure 3.3: **Left panel:** A simulated field spurious vectors (outliers) and wrong velocity estimates occurring due to local losses of correlation. **Right panel:** A simulated sample of biases and ramps in the velocity vector field arising from assumption of 'constant velocity' within the interrogation window.

method. It is clear that such conditions are better for larger windows. On the other hand, big size of the interrogation window makes an output field less detailed. Really, the PIV method is a discrete procedure in which a quantum of measurements (resolution) is defined by the size of the window. In other words, the resolution of the resulting output velocity field is not finer than the scale of the interrogation volume. If so, according to the Shannon–Whittaker sampling theorem (10.3.3) (see Appendix), the scale of the smallest flow structures resolved by the method is restricted by two sizes of the interrogation volume. Due to such a restriction some important sub-grid characteristics of the flow can be lost.

The scale of interrogation window imposes a limitation on spatial resolution of the method. Reduction of the interrogation volume increases the resolution of the output data and reduces the biases, but at the same time increases the probability of errors related to the wrong correlation estimates, as it was discussed above. This accounts for the amount of particles containing in the volume: the smaller volume size, the lesser particles are involved in correlation process, and consequently, the lower probability of a successful correlation.

In order to combat the problem and satisfy however these controversial demands, the most advanced PIV-procedures for cross-correlation velocity computations perform some kind of window deformations [7, 8, 77] which can reduce the remaining error at least three times. For instance, the authors of [77] developed their own original method which interpolates the velocity gradients within the interrogation windows using linear func-

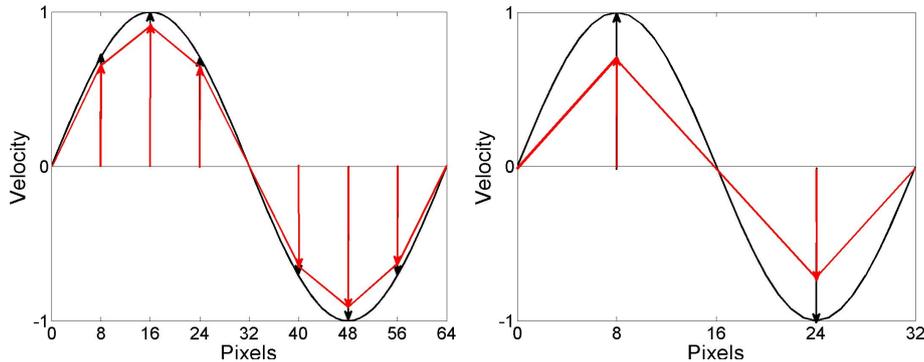


Figure 3.4: The velocity hodographs of 64 pixel sized (left) and 32 pixel sized (right) Green-Taylor vortices. The ground truth velocity profile is indicated by black color, measured velocity profile is shown in red.

tions. However, even with such an improvement it is not possible to remove all the errors mentioned above. Most significantly, these errors affect the resolution of the small-scale structures. As a confirmation of that, two simulated velocity hodographs of 64 pixel sized and 32 pixel sized the Green-Taylor vortices are presented in Fig. 3.4. The ground truth profile is presented in black, the red profile simulates the cross-correlation velocity estimates obtained with 16x16 px interrogation window and 50% window overlapping (the spatial spacing of the method is one velocity vector per every 8 px).

The velocity gradients within the interrogation window were approximated by linear functions. As it can be seen from Fig. 3.4, the ground truth velocity profile differs significantly from the 'measured' one. Obviously, this discrepancy arises from the mismatch of the ground truth sinusoidal profile to its linear approximation. The difference increases as the scale of vortex becomes smaller (compare, for instance, left and right panels in Fig. 3.4). A graph depicted in Fig. 3.5 shows the dependence between the vortex scale and the magnitude of average relative error.

The peak locking is an error arising from an integer presentation of fluid motion, i.e. the position of a particle (correlated peak) is defined with accuracy of a half-of-pixel. However, this 'uncertainty' in peak location can be nearly removed by using the interrogation window offset [100]. The dependence between an absolute deviation of the correlation peak location and particle displacement was found experimentally: the absolute deviation is constant if the particle displacement is larger than one pixel, and linearly decreases to zero proportionally if the particle displacement is less than one pixel. Thus, computing rough displacement and shifting the interrogation window in the case when the residual particle displacement is less than one pixel, one can recalculate it with smaller absolute error. In doing so iteratively, one can reduce in some cases the uncertainty down to 0.01 px.

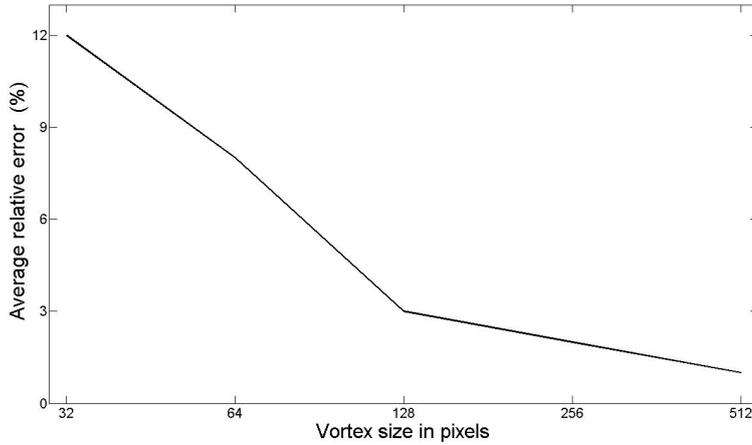


Figure 3.5: A graph of average relative error which arises due to the incorrectness of the velocity approximation.

In- and out-of-plane motions. This sort of errors appears in output fields due to the fact that the laser beam used in the PIV measurements has a finite thickness. If so, some particles can leave the tested area or, quite opposite, can get into the recorded volume during the time span between two camera shots. This remark concerns the boundaries of the interrogation volumes, as well, i.e. the particles can 'migrate' to or from the neighboring interrogation volumes. This process is called in-out-of-plane particle motions. As a result of that, the cross-correlation velocity estimations loose the pairs of particles in both images with obvious consequence for the velocity estimates, i.e. appearance of the same type of noise as in the particle overlapping case.

Shutter blind delay and fluctuations of laser beam can make one of the images in a short sequence over- or underexposed. As long as the principle of the constancy of mean intensity is one of the fundamental assumptions taken both for the cross-correlation method and the BCCE-based velocity procedures, the image pairs containing badly exposed frames are not suitable for further processing provided, of course, that no special post-processing operation has been done.

3.5.1 Data Post-Processing

As we have already discussed above, there are many reasons for appearance of noise in the velocity records. Quite naturally that the obtained velocimetry data require an application of some post-processing procedure to remove the noise from the records. According to [54] the operators used for the data post-processing can be classified in two categories: (1) global operators which use the information of the whole flow field for the validation, and (2) local operators which use just the neighbors to a certain vector for its validation. Some

of these procedures commonly used in such a post-processing are briefly outlined below.

Global histogram operator is the operator of the first kind, i.e. it is applied for removal of the vectors that do not satisfy some certain requirements in length and direction [68]. These requirements are usually obtained from some a priori information about the flow. Although this operator might seemed as coarse, it can be quite helpful at preliminary steps before applying the operators of the second kind [31, 54].

Let $\mathbf{v}_{i,j}$ is the velocity value at a grid point (i, j) . The idea of this method is based on a statement that the difference between two neighboring velocity vectors should be smaller than a certain positive threshold value ϵ_t :

$$|\mathbf{v}_{i,j} - \mathbf{v}_{i\pm 1, j\pm 1}| < \epsilon_t .$$

If a testing vector does not satisfy this criterion, it will be excluded from the data.

The obtained velocity vector field in this method is divided into several small sub-domains, or cells, so that one can construct a histogram of the velocity vectors included in these cells taken as bins. The procedure is as follows. All velocity vectors included in every cell are summed up, and a mean velocity value within the cell is obtained for an averaged histogram. The obtained ensemble of averaged velocities is analyzed in order to choose the value (or values) for the threshold (or several thresholds, if necessary). After all, the velocity vectors are checked to fulfill the threshold criteria, and those vectors which do not satisfy the condition are substituted for mean value.

Median Filter operator is the second class of operators. It is commonly used for the removal of outliers in the velocity data caused by erroneous estimations [58, 99] or due to multiphase containment of a flow [78]. It implies that all neighboring velocity vectors are ordered linearly with respect to the magnitude of the velocity vector, or their components, where the central value in this order is the median value \mathbf{v}_{med} . The inspection procedure is similar to that mentioned above, i.e. the velocity vector $\mathbf{v}_{i,j}$ is checked whether it is above the 'median' threshold or not, i.e:

$$|\mathbf{v}_{med} - \mathbf{v}_{i,j}| < \epsilon_t .$$

If $\mathbf{v}_{i,j}$ does not satisfy this criterion, it is substituted by the corresponding median value.

Dynamic mean-value operator performs good validation for noise-containing vector fields in presence of a few outliers and large motion characteristic scales [54]. The idea of this method is based on the assumption that velocity vector field is assumed to be continuous so that the difference between two neighboring vectors should be relatively small. All vectors that do not satisfy this criterion are substituted by the corresponding mean values. In a 2D case the average value is calculated for $N=8$ (or more) neighbors:

$$\mathbf{a}_{i,j} = \frac{1}{N} \sum_{n=1}^N \mathbf{v}_n .$$

3 Basic Image Processing in Experimental Fluid Mechanics

The averaged magnitude of the vector difference between the average vector and its 8 neighbors is also calculated as follows:

$$s_{i,j}^2 = \frac{1}{N} \sum_{n=1}^N |\mathbf{a}_{i,j} - \mathbf{v}_n|^2 .$$

The data validation criterion reads:

$$|\mathbf{a}_{i,j} - \mathbf{v}_{i,j}| < \epsilon_t ,$$

where $\epsilon_t = C_1 + C_2 s_{i,j}$ with constants C_1, C_2 . Note that as the number of outliers increases, according to [54] the local prediction given by the neighbours (some of which may be outliers) becomes increasingly affected by outliers.

3.5.2 Total Variation

Total variational approach is one of the-state-of-the-art methods for recovering piecewise smooth signal. The idea of the method is to penalize large gradients in the output data [70]. At the same time, the noisy signal usually possesses fast randomly changing characteristics with large amount of sharp ramps, jumps and spikes. If this is true, excluding this fast randomly changing fragments from a given data set one can obtain a 'clean' signal which is a piecewise smooth function. Under such an assumption on the model of noise and clean signal one can consider them as two functions with bounded and unbounded variations, respectively. The authors of [17, 70] suggested to take such a condition as a basis for their variation method which, in fact, is a minimization of a functional with respect to u :

$$J = \frac{1}{2} \|u - d\|_2^2 + \lambda \sup_{\boldsymbol{\xi}} \int_{\Omega} u(x) \operatorname{div}(\boldsymbol{\xi}(x)) dx, \quad \boldsymbol{\xi}(x) \in C_c^1(\Omega, \mathbb{R}^2), \quad |\boldsymbol{\xi}(x)| \leq 1 \quad \forall x \in \Omega . \quad (3.14)$$

Here d is the given data and λ is an adjustable parameter. If u is sufficiently regular, this expression is simplified to

$$\min_u \{J(u)\} = \frac{1}{2} \|u - d\|_2^2 + \lambda \int_{\Omega} |\nabla u(x)| dx .$$

According to [17], the minimizer of this functional is given by the formula

$$u = d - \pi_{\lambda}(d) ,$$

in which $\pi_{\lambda}(d)$ is an orthogonal projection of d onto a set $\{\operatorname{div}(\boldsymbol{\xi}) : |\boldsymbol{\xi}(x)| < 1 \quad \forall x \in \Omega\}$. Computation of $\pi_{\lambda}(d)$ leads to solving of the following problem:

$$\min \|\lambda \operatorname{div}(\boldsymbol{\xi}) - d\|^2 : |\boldsymbol{\xi}(x)|^2 \leq 1, \quad \forall x \in \Omega .$$

Note that the method described above represents the scalar version of the TV-denoising. A similar algorithm for the vector valued data is given in [56].

3.5.3 Summary

Three methods of data post-processing discussed above are widely used in a particle-based velocimetry [68]. The TV approach described in the last part of this section is commonly used for data denoising and can be applied to the vector data, as well [56]. The advantage of these methods is that they are relatively simple and, as a consequence, computationally efficient, which make these procedures very attractive for practical use. However, the disadvantage of these post-processing methods is that their applicability is restricted by the denoising of piecewise smooth signals.

As it was mentioned above, the median filter easily removes single outliers but it has worse performance if data contain homogeneously distributed noise. The situation is quite opposite in the case of the mean value filter. In comparison with these three procedures the denoising capability of the Total Variation Based Noise Removal Algorithm is much wider and area of application is richer because of a more general model of noise taken for its design. For instance, it fairly good removes single-point corruptions (outliers) and the noise distributed in the data. At the same time its nonphysical background is a great disadvantage which, as it will be shown in Sec. 7.3.1, makes it ineffective for the reconstruction of complex fluid flows.

Thus, the basic conclusion from this section is that a more advanced computation method should be developed and applied for a successful vector field reconstruction. As long as this work deals with fluid flows which obey the laws of continuum mechanics, the theoretical background of this new method should be based on similar principles, i.e. the momentum balance equations and the continuity equation to make it applicable to a very general case with wide range of data corruptions. The elaboration of this algorithm is a central point of this dissertation. The details of this method are discussed in chapters 4 and 7.

4 The Restoration Approach for Incompressible Fluid Flow

This chapter is a central part of the present work. Here we formulate all basic statements and assumptions that form the restoration procedure for incompressible fluid flows based on the Vorticity Transport Equation (VTE). The discussion here starts with the simplest 2D model of the VTE algorithm, then it proceeds with the extension of the method to a 3D case and finally this chapter ends with the report of the VTE based super-resolution approach. All theoretical issues with their evolution from a simplified two-dimensional flows to a general 3D case have been reported in a series of papers [94, 95, 96, 97]. Here they are collated and summarized in order to illustrate the generality of the approach.

The restoration algorithm itself consists of four relatively independent steps. For the simplicity it is reasonable to arrange four subsections for the method description (a separate subsection for every particular step). Before we start, let's first give a general overview of the procedure.

4.1 Overview

Concepts of noise and ideal solution. We start the discussion of the 2D model with the definition of the concepts of noise and ideal solution. To this end the principal idea of the current method formulated as an essence of the whole approach. It reads:

Definition 3 (Definition of noise). *No matter what is the origin of the noise, its presence in the vector fields makes them inconsistent with the main hydrodynamic principles, i.e. momentum balance equation (2.14) and continuity equation (2.3).*

If so, we can assume that every corrupted input data set \mathbf{d} consists of two parts, viz. divergent free "ideal solution", \mathbf{v}^I , which satisfies all hydrodynamic equations ((2.3),(2.14)) (the VTE equation, in particular) and the remaining "noisy" part, δ . In other words, we can split \mathbf{d} into two parts as follows:

$$\mathbf{d} = \mathbf{v}^I + \delta .$$

Note that the concept "noise" has not been explicitly defined yet. We consider it in a rather broad way as non-physical data set.

Our task is to extract a vector field \mathbf{u} from \mathbf{d} which will be close to \mathbf{v}^I , i.e.

$$\|\mathbf{u} - \mathbf{v}^I\|_2 < \|\mathbf{d} - \mathbf{v}^I\|_2 . \tag{4.1}$$

The suggested method which meets these criteria (for proof see Chap. 5) comprises following four steps:

4 The Restoration Approach for Incompressible Fluid Flow

1. Removal of the divergence from \mathbf{d} by projection onto the linear subspace of incompressible vector fields, i.e. $\{\mathbf{v} \mid \nabla \cdot \mathbf{v} = 0\}$;
2. Removal of the noise by the Gaussian low-pass filtering so that the spectrum of the resulting vector field \mathbf{v}_g satisfies the Kolmogorov's theory of turbulence;
3. Computation of the vorticity $\boldsymbol{\omega}_v = \nabla \times \mathbf{v}_g$ and enforcing of a physically plausible flow structure in terms of a vorticity field $\boldsymbol{\omega}$ satisfying the VTE (2.14);
4. Recovering of an incompressible velocity field \mathbf{u} from the vorticity field $\boldsymbol{\omega}$.

Let us consider these steps one by one paying attention to all necessary details.

4.2 Two Dimensional VTE-Based Recovery Approach

4.2.1 Solenoidal Projection

The procedure which is described in this chapter is aimed at removal of the divergent component of the noise from the given vector field. Note that this step is justified only for fluid flows which are supposed to be incompressible.

The method starts with the removal of the divergence from the given data set. It is not clear in advance, however, why the *div*-removal procedure should be completed first followed by the steps of the Gaussian filtering and computation of the vorticity, but not vice-versa. There are two basic reasons for such a choice. The first reason is that after the removal of the divergence all successive steps are 'divergent free'. Once the divergence has been removed from the input field, it will never appear again during the further recovering (the proof is given in Chap. 5). The second reason comes from the fact that the result of the vorticity rectification (3-rd step) depends on the contaminants of the divergence presented in the input vector field. If the data set still contains the divergence, the vorticity field will be restored as if the flow is compressible. However, as it will be shown in Chap. 5, the particular order of steps 1 and 2, i.e. the solenoidal projection and the Gaussian filtering, is not important.

Let $\Omega \subset \mathbb{R}^2$ be a simply-connected domain and $[L^2(\Omega)]^2 = L^2(\Omega) \times L^2(\Omega)$; denote by $H(\text{div}0; \Omega) \subset [L^2(\Omega)]^2$ the subspace of divergence-free vector fields, and denote by $H_0^1(\Omega) \subset L^2(\Omega)$ the subspace of functions with square-integrable first-order derivatives and vanishing boundary values. Further define $\nabla H_0^1(\Omega) \subset [L^2(\Omega)]^2$ as a subspace of nonzero potentials, such as $\nabla H_0^1(\Omega) := \{\mathbf{grad}(\psi) \in [L^2(\Omega)]^2 \mid \mathbf{grad}(\psi) \neq 0, \psi \in H_0^1(\Omega)\}$. Then the following orthogonal decomposition of the vector fields holds [21]:

$$[L^2(\Omega)]^2 = \nabla H_0^1(\Omega) \oplus H(\text{div}0; \Omega) . \quad (4.2)$$

In other words, the given divergent data \mathbf{d} (we assume by default that $\mathbf{d} \in [L^2(\Omega)]^2$) can be presented as a sum of solenoidal and irrotational vector fields. The task is to exclude the irrotational component from \mathbf{d} . To this end the following minimization problem is considered

$$\min \|\mathbf{d} - \mathbf{v}\|_2^2 \quad \text{s.t.} \quad \text{div}(\mathbf{v}) = 0 . \quad (4.3)$$

It is equivalent to a dual problem

$$\max_{\psi} \min_{\mathbf{v}} \left\{ \frac{1}{2} \|\mathbf{d} - \mathbf{v}\|_2^2 - \langle \mathit{div}(\mathbf{v}), \psi \rangle_{\Omega} \right\} , \quad (4.4)$$

where ψ is a Lagrange variable. Applying the integration-by-parts to the second term in eq. (4.4) one obtains: $\langle \mathit{div}(\mathbf{v}), \psi \rangle_{\Omega} = -\langle \mathbf{v}, \mathbf{grad}(\psi) \rangle_{\Omega}$ and thus

$$\max_{\psi} \min_{\mathbf{v}} \left\{ \frac{1}{2} \|\mathbf{d} - \mathbf{v}\|_2^2 + \langle \mathbf{v}, \mathbf{grad}(\psi) \rangle_{\Omega} \right\} . \quad (4.5)$$

The min-max theorem (see [22]) allows us to swap min and max and to obtain the equivalent saddle-point problem:

$$\min_{\mathbf{v}} \max_{\psi} \left\{ \frac{1}{2} \|\mathbf{d} - \mathbf{v}\|_2^2 + \langle \mathbf{v}, \mathbf{grad}(\psi) \rangle_{\Omega} \right\} . \quad (4.6)$$

Note that the second term in (4.6) is minimal only if \mathbf{v} is solenoidal. According to the Saddle Point Theorem (see in Quadratic Functionals theorem (10.2.4) in Appendix) the problem (4.6) has a unique solution. The calculus of the first variation gives

$$\langle \boldsymbol{\eta}, \mathbf{v} - \mathbf{d} \rangle_{\Omega} + \langle \boldsymbol{\eta}, \mathbf{grad}(\psi) \rangle_{\Omega} = 0 , \quad (4.7a)$$

$$\langle \boldsymbol{\eta}, \mathit{div}(\mathbf{v}) \rangle_{\Omega} = 0 , \quad (4.7b)$$

where $\boldsymbol{\eta}$ is any function. The corresponding to (4.7a)-(4.7b) system of the Euler-Lagrange equations reads:

$$\mathbf{d} = \mathbf{grad}(\psi) + \mathbf{v} , \quad (4.8a)$$

$$\mathit{div}(\mathbf{v}) = 0 . \quad (4.8b)$$

It represents the decomposition of \mathbf{d} on solenoidal \mathbf{v} and irrotational $\mathbf{grad}(\psi)$ components. Computing the solution of (4.6) one instantaneously obtains the divergent-free component \mathbf{v} and the potential ψ .

Equations (4.7a)-(4.7b) and (4.8a)-(4.8b), represent the vector field decomposition on solenoidal and irrotational components. The numerical implementation of this procedure will be presented in Chap. 6.

Having computed \mathbf{v} , one can continue the removal of noise by application of a lowpass filtering.

4.2.2 Lowpass Filtering

An incompressible turbulent fluid flow can be considered as a superposition of eddies of various scales (see Fig. 4.3). Let λ is the size of a vortex, ϵ is the average dissipation energy, and $k \propto 1/\lambda$ is the corresponding wave number. Theory [50] suggests that the energy spectrum of homogeneous turbulence obeys the law $v_{\lambda}^2 \sim (\epsilon/k)^{2/3}$ with the most significant part of energy concentrated at small wavenumbers k . If the data set is corrupted by some noise produced either by measurements or data processing, the spectral characteristics of

“clean” and “polluted” data sets will be different, especially at larger wavenumbers k . Figure 4.1 depicts such a typical spectrum by red line. For the comparison reason the spectrum of a flow \mathbf{d} corrupted by “white” noise is also shown by black line.

Figure 4.1 suggests (blue line) that a considerable amount of noise can be easily removed by a lowpass filtering. While sophisticated filtering schemes employing a multiscale expansion of the flow [32] are conceivable, we will show below that traditional Gaussian lowpass filtering in conjunction with the other three steps of our overall approach (cf. Sec. 4.1) works well. We only have to choose a conservative cutoff frequency that is large enough so that not to damage physically significant structures of the flow.

Denoting the impulse response and its Fourier transform by

$$g_\sigma(\mathbf{x}) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{1}{2\sigma^2}|\mathbf{x}|^2\right), \quad (4.9a)$$

$$\hat{g}_\sigma(\mathbf{w}) = \exp\left(-\frac{\sigma^2}{2}|\phi|^2\right), \quad (4.9b)$$

the scale parameter σ is chosen as follows: The smallest vortex size that can be resolved on a computational grid has a size of about 3 pixels, corresponding to the angular wavenumber $2\pi/3$. We empirically choose $\sigma = 1.34$ in (4.9b) to lower the amplitude spectrum at this point by the factor $1/50$.

Figure 4.1 shows the amplitude spectrum of the smoothed flow

$$\mathbf{v}_g = g_\sigma * \mathbf{v}, \quad (4.10)$$

as a blue line ($*$ denotes convolution). This result indicates that our choice $\sigma = 1.34$ is conservative and it does not affect the “true” spectrum. On the other hand, despite having effectively removed noise (see Fig. 4.2), it is also obvious that a significant non-physical noise component still remains (compare Fig. 4.2 and Fig. 4.3).

The way for further improvements of the velocity estimate becomes clear if one takes into account that \mathbf{v}_g in (4.10) does not satisfy the equations governing the dynamics of incompressible fluid. Some possible consequences will be considered in two subsequent subsections that complement the overall approach.

4.2.3 Vorticity Rectification

In order to enforce physical consistency of the flow (4.10) computed in the previous step, we compute its vorticity, i.e.

$$\omega_g := \nabla \times \mathbf{v}_g, \quad (4.11)$$

and enforce consistency with the VTE (2.15) by minimizing the functional

$$\min_{\omega} \left\{ \|\omega - \omega_g\|_2^2 + \alpha \left(\nu \|\nabla \times \omega\|_2^2 + 2\langle e(\mathbf{v}_g), \omega \rangle_\Omega \right) \right\}. \quad (4.12)$$

Here $e(\mathbf{v}_g) = \frac{\partial \omega_g}{\partial t} + (\mathbf{v}_g \cdot \nabla) \omega_g$ is used as an abbreviation of the left hand side of (2.15). Notice that ω_g and $e(\mathbf{v}_g)$ are evaluated using the flow (4.10) computed at the previous

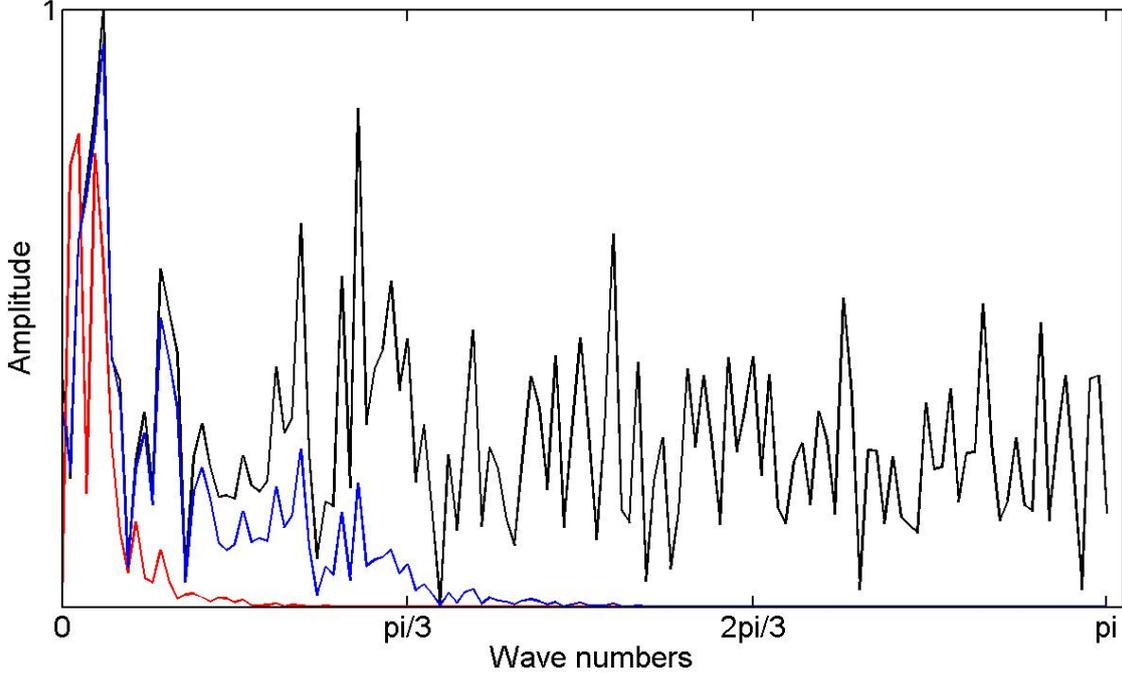


Figure 4.1: The velocity spectra of an original turbulent fluid flow (red line), the flow corrupted by noise (black line), and the flow after solenoidal projection (4.3) and the Gaussian lowpass filtering (4.10) (blue line). Although noise has been effectively removed, a significant non-physical component of the fluid flow estimate remains – cf. Fig. 4.2 and Fig. 4.3.

step. Coefficient α is a user parameter which has the dimension of $[time]$. It provides the balance and the dimension consistency between the first and the second terms. The proper choice of α will be discussed in Chap. 8.

The rationale behind formula (4.12) becomes apparent when inspecting the corresponding Euler-Lagrange equation

$$\omega - \alpha\nu\Delta\omega = \omega_g - \alpha e(\mathbf{v}_g) . \quad (4.13)$$

This is just a linear diffusion equation to be solved for the restored vorticity field ω . The viscosity coefficient ν acts here as a natural smoothing parameter. This equation can be rearranged as follows

$$\omega = \omega_g - \alpha(e(\mathbf{v}_g) - \nu\Delta\omega) , \quad (4.14)$$

which shows that ω corresponds to the ω_g corrected by the residual of the VTE (2.14).

Note that for setting up the functional (4.12) we deliberately omitted the temporal derivative $\frac{\partial\omega}{\partial t}$ in (2.14). This was done in accordance with the assumption that the temporal derivative is small in comparison with all other terms which are retained in the equation. This assumption is well justified for quasi-stationary flows, and is valid as well

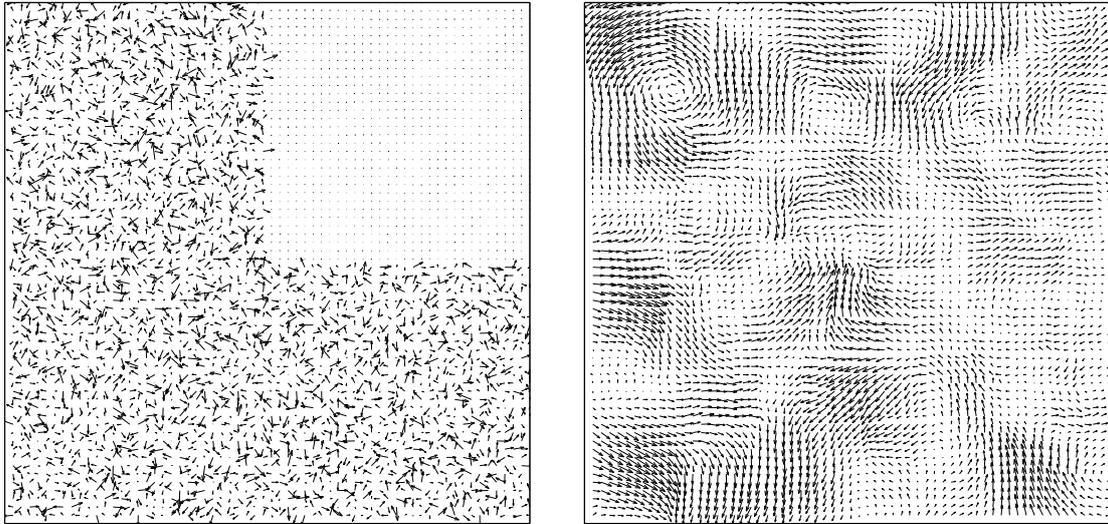


Figure 4.2: **Left panel:** Noisy input field. The original data field without noise is shown in the upper right quadrant to illustrate the signal-to-noise ratio. **Right panel:** The data field after filtering using the Gaussian filter. Although noise has been effectively removed, a significant non-physical component of the fluid flow estimate still remains – cf. Fig. 4.3.

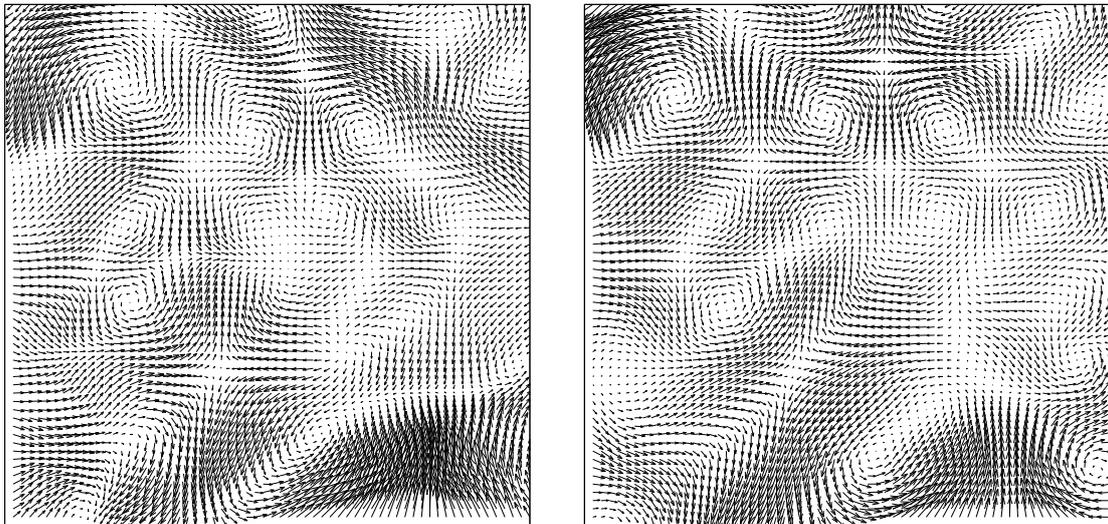


Figure 4.3: **Left panel:** The restored flow corresponding to Fig. 4.2 but after the complete cycle of four computational steps of the overall approach. **Right panel:** Ground truth vector field.

in many other typical cases when fluid flows contain basically large-scale vortexes. As

a result of this assumption, we need a single fluid flow estimate at a single moment of time. The experiments in Sec. 4.2.4 will illustrate that this simplification appears to be reasonable.

Another important issue concerns boundary conditions. In general, there are two types of boundary conditions corresponding to "liquid" and "rigid" boundaries. In the case of solid wall or any other rigid contour all velocity components are equal to zero at this boundary. The situation is less evident in the case of liquid boundary when there is no obstacle preventing penetration of the fluid through it. If the liquid boundary of the calculational domain coincides with the streamline (e.g. at the borders of convective cells), one can set there 'slip conditions', i.e. all velocity component normal to the contour should be equal to zero.

The problem is less obvious in the case of liquid boundaries that do not coincide with any streamline so that the flow characteristics on this contour are unknown. The *natural* boundary conditions

$$\frac{\partial \omega}{\partial \mathbf{n}} = 0 \quad (4.15)$$

are usually taken for the analysis of the variational problem (4.12) in this case (here \mathbf{n} is a unit vector normal to the boundary $\partial\Omega$). It is clear that this condition is not justified in proper extent and gives only a rough approximation for some real scenarios and can lead to the loss of small-scale details of the flow. However, in some cases this is the only way to complete the formulation of a boundary value problem.

4.2.4 Velocity Restoration

The final step of the procedure is a conversion of the restored vorticity field ω back to the velocity field \mathbf{u} . This task is accomplished by solving the following minimization problem:

$$\min_{\mathbf{u}} \left\{ \|\mathbf{u} - \mathbf{v}_g\|_2^2 + \beta \|\nabla \times \mathbf{u} - \omega\|_2^2 \right\}, \quad (4.16)$$

with \mathbf{v}_g and ω computed at the previous steps (4.10) and (4.12), respectively. Here β is a dimensional user parameter which defines an impact of the second term in (4.16) (for more details see Chap. 8). Using the Green's formula (see eq. (10.25)) system (4.16) can be transformed to the constrained variational system as follows:

$$\langle \mathbf{u}, \boldsymbol{\phi} \rangle_{\Omega} + \beta \langle \nabla \times \mathbf{u}, \nabla \times \boldsymbol{\phi} \rangle_{\Omega} = \langle \beta \nabla \times \omega + \mathbf{v}_g, \boldsymbol{\phi} \rangle_{\Omega}, \quad \forall \boldsymbol{\phi} \in C_0^1(\Omega). \quad (4.17)$$

The result of this procedure is a velocity field \mathbf{u} that is physically plausible due to the consistency with the vorticity transport equation enforced in the previous step. Finally, we arrive at the equation

$$\mathbf{u} - \Delta \mathbf{u} = \text{curl}(\omega) - \mathbf{v}_g, \quad (4.18)$$

which defines the denoised velocity vector field. Note that for case of liquid boundaries we accept the linear behaviour of a flow on $\partial\Omega$, hence $\Delta \mathbf{u}$ and $\text{curl}(\omega)$ vanishes there, and $\mathbf{u}|_{\partial\Omega} = \mathbf{v}_g$ (for more details and other types of B.Cs see Sec. 2.8).

4.2.5 Applicability

It is clear that all physical processes develop in three dimensions. In some cases one of the velocity components is equal to zero or it is infinitesimally small comparing with the other two so that can be excluded from the analysis. As a result, a 2D approach is well justified and can be used for the analysis. Note, however, that pure 2D flows occur quite seldom in real practice. Fortunately, there is quite a large class of 3D fluid motions, so called 'flat flows', which are 'nearly' two-dimensional. This class of motions is characterized by the fact that one of the velocity components is small in comparison with two others and in many practical cases can be neglected. In this respect one can mention here all meso- and synoptic-scale motions developing in the ocean and atmosphere (large scale oceanic currents, atmospheric cyclones etc.) as examples of such nearly 2D-flows.

Two-dimensional velocimetry methods are widely used in operational meteorology and oceanography for investigation of such motions on a regular basis: the images obtained from satellites are analyzed in order to reconstruct the velocity distribution of flows or currents [36, 37, 64]. However, due to meteorological perturbations of the atmosphere, inhomogeneous cloud structure and fast propagation of the camera carrier (about ten kilometers per second) the resulting output velocity vector fields contain a number of corruptions and need to be processed and recovered.

The fluid flows which can be classified as flat flows occur on smaller scales, as well. For instance, the structure of the wake developing at small Reynolds numbers behind a cylinder can represent such type of flows. They are usually studied using two-dimensional PIV methods [106]. The errors occurring in the course of such laboratory measurements are typical for the PIV procedures (details are discussed in Sec. 3.5).

The discussed above 2D approach can be successfully applied to the vector fields obtained from such velocimetry measurements without any substantial losses of their physical structure. However, if all three components of the fluid flow are of the same order, one should refrain from the application of the 2D assumption and use the 3D version of the algorithm, instead.

4.3 Three-Dimensional VTE-Based Recovery Approach

The denoising procedure presented above can be easily generalized to a three-dimensional case. The structure of the algorithm and mathematical relationships remain exactly the same as they were in 2D case. The only difference between them appears in the presentation of a curl operator, the Gaussian filter function g_σ and the VTE presentation. In the 3D case the curl of any vector $\mathbf{v} = (u, v, w)^t$ is given by expression (10.5). If so, the 3D vorticity $\boldsymbol{\omega}$ reads: $\boldsymbol{\omega} = \mathbf{curl}(\mathbf{v}) = (\partial w/\partial y - \partial v/\partial z, \partial u/\partial z - \partial w/\partial x, \partial v/\partial x - \partial u/\partial y)$. It is easy to show that in three dimensions all relationships between *div*, **grad** and *curl* remain the same as they are in the 2D case (the details are included in Appendix 10). According to (2.14) a new nonlinear term appears in the left hand side of the 3D VTE which was not included in the 2D VTE. Hence, the vector $\mathbf{e}(\boldsymbol{\omega})$ in the 3D case reads:

$$\mathbf{e}(\mathbf{v}) = \frac{\partial}{\partial t}\boldsymbol{\omega} + (\mathbf{v} \cdot \nabla)\boldsymbol{\omega} + (\boldsymbol{\omega} \cdot \nabla)\mathbf{v}. \quad (4.19)$$

4.3 Three-Dimensional VTE-Based Recovery Approach

One can use this definition for the abbreviation of the 3D VTE, i.e.

$$\mathbf{e}(\mathbf{v}) = \nu \Delta \boldsymbol{\omega} . \quad (4.20)$$

A new multiplier appears in the 3D Gaussian filter function as well, i.e.

$$g_\sigma(\mathbf{x}) = \frac{1}{(2\pi\sigma^2)^{3/2}} \exp\left(-\frac{1}{2\sigma^2}\|\mathbf{x}\|^2\right) . \quad (4.21a)$$

Formal substitution of the 2D functions \mathbf{v} , $\boldsymbol{\omega}$, g_σ , $\mathbf{e}(\mathbf{v})$ with their 3D representations in the corresponding formulas in Sec. 4.2 converts the 2D VTE reconstructive approach into the 3D version.

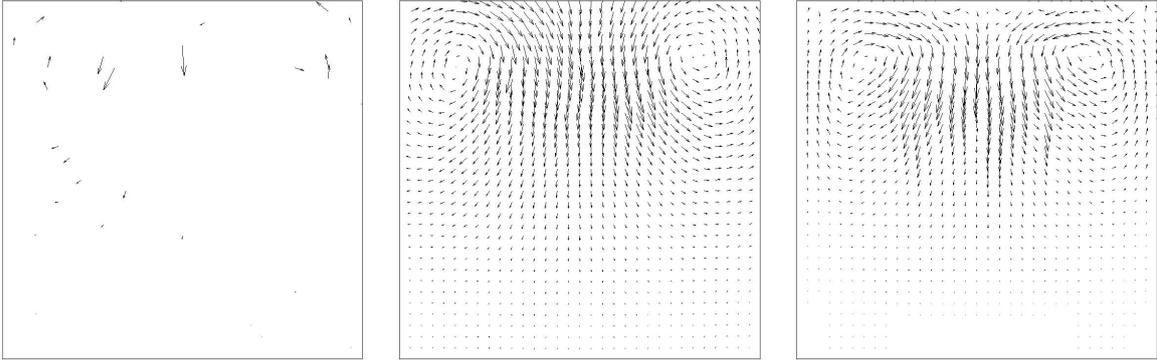


Figure 4.4: The 2D slices of 3D vector fields. **Left:** Sparse vector field. **Middle:** Restored vector field. **Right:** Ground truth

4.4 Super-Resolution Approach

4.4.1 Background and Overview

Numerous errors like 'overlapping particles' appearing in the velocimetry output files usually occur due to a high density of seeding. In the light of this fact an idea of conducting experiments with the low particle concentration sounds quite reasonable. A number of modifications of the PTV method based on this idea, i.e. low-density seeding, have been developed recently [41, 46, 89] and gave good and reliable results. On the other hand, in many cases this sparse-seeding methodology can be a great disadvantage because of the very same reason, i.e. a lot of details and flows fragments can be missing in the areas with extremely low particle concentration (it is not possible to provide a homogeneous density of particles everywhere). This reasoning is a strong motivation for the development of the procedure that could increase/restore the field resolution using the hydrodynamical properties of the fluid.

To address this issue, a special resolution-reconstructive approach is presented in this chapter. This method is a combination of the multi-resolution technique discussed in [104] and the modified version of the denoising algorithm developed here (see: Chap. 4) adopted to the reconstruction of the extremely sparse velocity vector fields. Some examples of such initially sparse fields and their restored versions are given in Fig. 4.4.

Before starting the discussion over the details and great efficiency of the super-resolution approach, let's illustrate first the basic ideas of the method. To this end we apply the VTE-based restoration algorithm discussed in Sec. 4 to a vector field with 50% of sparsity (Fig. 4.5 left panel). Worthwhile mentioning that the algorithm treats the gaps in the input data as a conventional "noise" that does not satisfy any hydrodynamical principle. If so, the algorithm plausibly fills the gaps and returns a flow estimate with twice higher spatial resolution. Comparing middle and right panels of Fig. 4.5 we see that the missing data were almost perfectly reconstructed with minor losses in details. However, if the

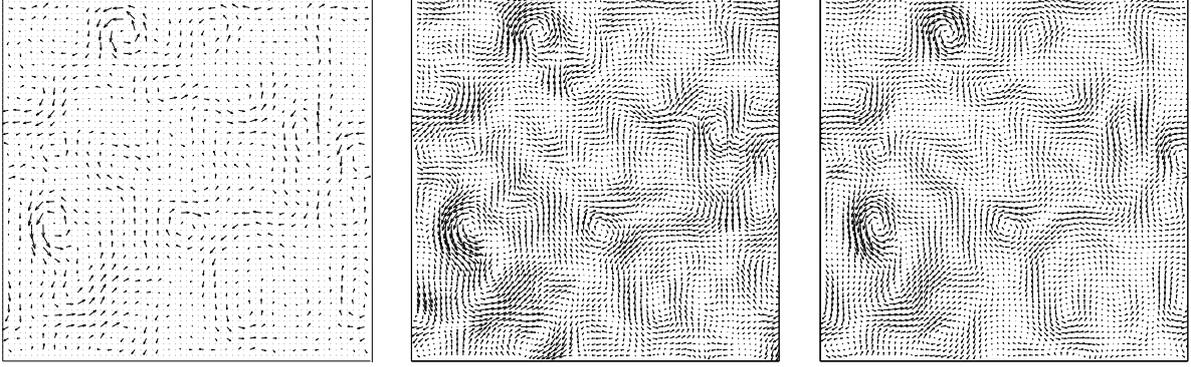


Figure 4.5: **Left:** The data with losses of density. **Middle:** Restored data. **Right:** Ground truth.

sparsity exceeds the level of 50%, the quality of reconstruction becomes increasingly worse with the increase of sparsity. Thus, an important conclusion from this discussion is that the vector field will be restored with a good quality if its sparsity does not exceed 50%.

4.4.2 The Algorithm for Super-Resolution Approach

Suppose a *sparse* vector field \mathbf{d} is taken as an input data set (cf. left panel of Fig. 4.4). Because of the most of entries in this field are zero, the restoration algorithm operates in a mode of from-coarse-to-fine resolution iterative application (see: Chap. 4). In other words, the sequence of steps is as follows:

- First, we construct a sequence of dyadic voxel grids and corresponding representations of the data

$$\mathbf{d}^{2^L h}, \mathbf{d}^{2^{(L-1)} h}, \dots, \mathbf{d}^h,$$

where \mathbf{d}^h corresponds to the data \mathbf{d} discretized at the finest grid with mesh size h . As it is specified below, the fine-to-coarse transfer $\mathbf{d}^{2^k h} \rightarrow \mathbf{d}^{2^{k+1} h}$ involves a local averaging operation decreasing the portion of nonzero entries. The coarsest level L is chosen in such a way that this portion is less than a half of all grid position.

- Coarse-to-fine transfer $\mathbf{d}^{2^k h} \rightarrow \mathbf{d}^{2^{k-1} h}$ by tri-linear interpolation defines a prolongation operator P (cf. [104]) and, in turn, a restriction operator R for the fine-to-coarse transfer through the procedure

$$\langle \mathbf{u}^h, P\mathbf{v}^{2h} \rangle_{\Omega, h} = \langle R\mathbf{u}^h, \mathbf{v}^{2h} \rangle_{\Omega, 2h}, \quad \forall \mathbf{u}^h, \forall \mathbf{v}^{2h},$$

with $\langle \cdot, \cdot \rangle_{\Omega, h}$, $\langle \cdot, \cdot \rangle_{\Omega, 2h}$ denoting the inner products at the corresponding levels.

- At each level the restoration algorithm (see: Chap. 4) is iteratively applied until the termination criterion is satisfied. The result is transferred to the next finer grid as input data for the next iteration.

4 The Restoration Approach for Incompressible Fluid Flow

Fine-to-coarse grid mapping. Let U^h be an initial grid with mesh size h . The mapping data from U^h onto two times coarser grid U^{2h} reads:

$$U_{i,j,k}^{2h} = C(U_{i,j,k}^h + U_{i+1,j,k}^h + U_{i-1,j,k}^h + U_{i,j+1,k}^h + U_{i,j-1,k}^h + U_{i,j,k+1}^h + U_{i,j,k-1}^h). \quad (4.22)$$

Equation (4.22) maps the mean value taken from neighboring values U^h onto U^{2h} . The averaging coefficient C which excludes the influence of zero elements, is calculated as follows

$$C = \frac{1}{1 + \sum_{i,j,k=1}^2 g_{i,j,k}}, \quad (4.23)$$

where function $g_{i,j,k}$ is

$$g_{i,j,k} = \begin{cases} g_{i,j,k} = 1 & \text{if } U_{i,j,k}^h \neq 0, \\ g_{i,j,k} = 0 & \text{otherwise.} \end{cases}$$

Termination criterion. The reconstruction procedure of the sparse data set runs in several iterations. At every step the sparse vector field is updated with the new values obtained from the current iteration. This procedure continues until the termination criteria is satisfied.

The routine 'Termination' analyzes the energy changes in the vector fields computed before and after each iteration. The kinetic energy is calculated as follows:

$$E(U^\tau) = \sum_{i,j,k=1}^{l,m,n} (U_{i,j,k}^\tau)^2. \quad (4.24)$$

Here U^τ is the $(l \times m \times n)$ data matrix updated at iteration τ . For the successful termination of the procedure only the convergence of $E(U^\tau)$ to some constant is required, i.e.

$$|E(U^2) - E(U^1)| > |E(U^3) - E(U^2)| > \dots > |E(U^{\tau+1}) - E(U^\tau)| = \epsilon, \quad (4.25)$$

where ϵ is a required accuracy parameter. Expression (4.25) represents the termination criterion: the restoration routine stops when the difference of energy between two successive iterations decreases below a necessary negligibly small level. If (4.25) is not valid and the convergence condition is not satisfied, the restoration procedure is stopped.

5 Error Analysis

The development of any computational procedure implies conducting its error analysis. By the error reduction one should understand the situation when the discrepancy between an input data set and the original 'ground truth' (in sense of some concrete norm) is getting smaller after every single application of the algorithm.

We present here the error analysis of the developed algorithm considering five relatively independent steps. In order to make the reading easier we briefly outline every step as follows:

- We draft section 5.1 '**The Systematic Error of the Approach and Error Reduction Criterion**' by defining the classes of vector fields of noise, ground truth and ideal solutions. Here we estimate the systematic error of the approach and formulate the error reduction criterion;
- In section 5.2 '**Assumptions**' we summarize the main assumptions and simplifications used for the error reduction analysis;
- In section 5.3 '**Divergence Removal**' we show that the divergent component of the noise is removed from the processed data set by application of the VTE-based reconstruction approach;
- In section 5.4 '**Error Reduction**' we illustrate how the remaining divergent-free error reduces after applying of the four steps of the reconstruction approach. This section includes one general statement and some preparatory lemmas which are necessary for the proof of the basic theorem;
- The convergence analysis is concluded in subsection 5.5 where we summarize the obtained results.

In error analysis we consider separately four basic steps of the VTE-restoration approach. All these steps are discussed in Sec. 4 in detail. Here we just briefly outline them as follows:

Step 1: Solenoidal projection. The decomposition of the input data field \mathbf{d} onto divergent $\nabla\phi$ and divergent-free components \mathbf{v} , i.e.

$$\mathbf{v} = \mathbf{d} - \nabla\phi \in \mathbf{V}_{sol} . \quad (5.1)$$

Step 2: Gaussian filtering. Spectra cleaning is performed by application of the Gaussian filter g_σ :

$$\mathbf{v}_g = g_\sigma * \mathbf{v} . \quad (5.2)$$

Step 3: Vorticity rectification. This step implies an application of the procedure

$$\boldsymbol{\omega} - \alpha\nu\Delta\boldsymbol{\omega} = \boldsymbol{\omega}_g - \alpha\mathbf{e}(\mathbf{v}_g), \quad \boldsymbol{\omega}_g = \nabla \times \mathbf{v}_g , \quad (5.3)$$

with boundary conditions

$$\frac{\partial\boldsymbol{\omega}}{\partial\mathbf{n}} = 0 . \quad (5.4)$$

This step enforces the physical structures in the flow in terms of the VTE. Here $\boldsymbol{\omega}$ is the vorticity, $\mathbf{e}(\mathbf{v}_g)$ is the left hand side of the VTE (for the details see Subsec. 4.2.3).

Step 4: Velocity reconstruction. This step converts the vorticity cleaned at the previous step into velocity by application of the procedure:

$$\mathbf{u} - \beta\Delta\mathbf{u} = \mathbf{v}_g - \beta\nabla \times \boldsymbol{\omega} \quad (5.5)$$

with boundary conditions:

$$\frac{\partial\mathbf{u}}{\partial\mathbf{n}} = 0. \quad (5.6)$$

The vector field \mathbf{u} is the cleaned output. Note that in the description of denoising procedure we use more general B.C than eq. (5.6). The latter ones were chosen for a sake of simplification of error analysis (for details see Sec. 5.2).

5.1 The Systematic Error of the Approach and Error Reduction Criterion

In section 4.1 we hypothesized existence of an additive model of errors. In other words, we assumed that an experimental data set \mathbf{d} is a superposition of two components, $\mathbf{d} = \mathbf{v}^I + \boldsymbol{\delta}$, where the first one, \mathbf{v}^I , satisfies the continuum mechanics equations (2.3) and (2.14) and the second component, $\boldsymbol{\delta}$, is an additive perturbations that does not satisfy the hydrodynamics equations. In this section we discuss an effect which this assumption can have on the final result of the denoising. In addition we give an approximate estimation of the noise-to-signal ratio above which the data reconstruction procedure fails. Since the denoising algorithm is based on the VTE which hardly can be solved analytically in a very general case, we restrict our analysis considering the case of potential flows for which $((\mathbf{v} \cdot \nabla)\mathbf{v} = 0)$. In a more general case we rely on the results of numerical runs. As a starting point of the analysis let us introduce some necessary definitions.

Definition 4 (Class of ideal velocity data). *Define S_I as a class of all linear velocity vector fields satisfying the VTE and the continuity equation.*

5.1 The Systematic Error of the Approach and Error Reduction Criterion

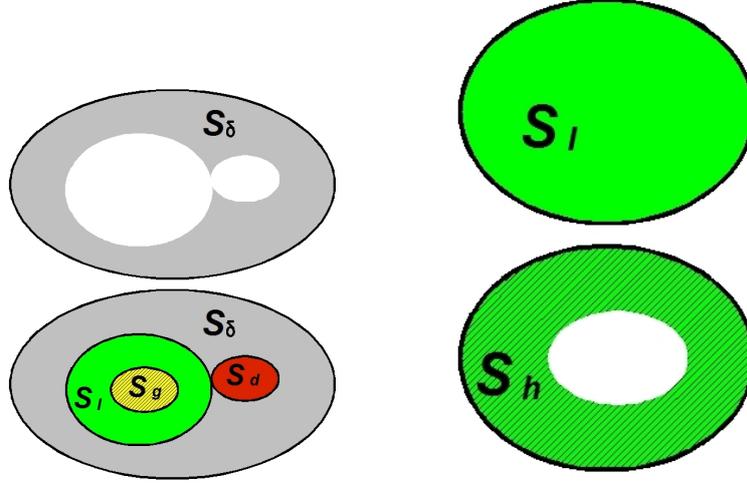


Figure 5.1: The Venn-diagrams of sets $S_d, S_I, S_g \subset S_I, S_\delta, S_h = S_I \setminus S_g$.

Definition 5 (Classes of measured and ground truth velocity data). *Define S_d as a class of all velocity vector fields measured in fluid velocimetry experiments. Define also S_d as a class of ground truth flows, and S_g , that corresponds only to the true velocity distributions which take place in fluid flow during the measurements.*

The uniqueness of ground truth flows. We consider here the measurements conducted in a motionless coordinate system. Hence, each element in S_d corresponds to only one and unique element from S_g , but not vice versa. Such an ambiguous dependence is explained by a simple fact that the velocity measurements of the same flow can be performed by different independent devices, and, as the result, might differ from each other while the ground truth always remains unique.

Definition 6 (Classes of hydrodynamic and non-hydrodynamic noise.). *Define a class of non-hydrodynamical noise S_δ as a set of some random vector fields which do not satisfy the hydrodynamical equations (2.14)-(2.3) and do not represent actual measurements. They are usually generated by the imperfectness of measuring technique or post-processing procedures. Define also $S_h = S_I \setminus S_g$ as a class of hydrodynamical noise. The latter is a set of all vector fields that satisfy (2.14) and (2.3) but are not actual fluid flows.*

The Venn-diagrams given in Fig. 5.1 show schematically the relationships between the introduced classes. The grey oval given at the top of the left figure represents the class S_δ . Note that white areas inside this oval do not belong to S_δ ; they are filled with other sets (see the bottom part of the same figure). Note that each measuring data set contains noise and does not satisfy the hydrodynamic equations. As a result, $S_I \cap S_d = \emptyset$. Important is that although both classes, S_d and S_δ , do not satisfy (2.3) and (2.14), their intersection is empty. By this fact we want to highlight the difference between these two classes: the

vector fields from S_δ do not correspond to any actual measurement, while the vector fields in S_d do represent some actual measurements. Similarly, $S_h \cap S_g = \emptyset$, the green oval in the upper part of the right figure represents S_I and the dashed area of the same green oval in the bottom part of the figure represents S_h . However, $S_h \subset S_I$ and $S_g \subset S_I$ since the vector fields from these sets satisfy (2.3), (2.14).

Decomposition of noise. Due to a large variety of independent noise sources appearing in fluid velocimetry measurements (see Chap. 3.5) the noise generated in output data can be considered as a random vector field Θ [27]. We assume here that this vector field can be considered as a sum of two vector fields, $\mathbf{h} \in S_h$ and $\delta \in S_\delta$, for the most of possible cases of noise. In normally and homogeneously distributed Gaussian noises the statistics obtained from multiple experiments on the detection of the amount of hydrodynamical component such an assumption is well justified.

The experiments are conducted as follows: the VTE-restoration approach is applied to the conventional normally distributed Gaussian noise Θ . The extracted from the noise vector field \mathbf{h} is considered as a hydrodynamical part of Θ (some examples of these vector fields are given in Fig. 5.2). After that the norms of \mathbf{h} and Θ are computed and compared against each other. Being repeated one thousand times this procedure can give the statistics over the containment of the hydrodynamical component of the noise in Θ . Similar procedure is performed for homogeneously distributed noise. The statistics for both experiments is presented by the histograms in Fig. 5.3. We have performed 1000 experiments and in each experiment the noise contained some hydrodynamical part. If so, applying the law of big numbers we obtain that the probability that Θ does not contain hydrodynamical part is less than 0.001.

Linear Case. Let $\mathbf{v} \in S_I$ and $\mathbf{h} \in S_h$ such that $(\mathbf{h}\nabla)\mathbf{h} = (\mathbf{v}\nabla)\mathbf{v} = 0$; then $\mathbf{v} + \mathbf{h} \in S_I$. This assumption is obviously true since the VTE in this case is linear. According to the adopted additive model of the noise, the measured data can be considered as a sum of ground truth, hydrodynamical and non-hydrodynamical components of the noise: $\mathbf{d} = \mathbf{g} + \Theta = \mathbf{g} + \mathbf{h} + \delta$. Since $\mathbf{g}, \mathbf{h} \in S_I$, their sum also belongs to S_I . So, with these results we arrived at an important conclusion:

Conclusion 1 (Systematic error). *If the input vector field \mathbf{d} is corrupted by noise, the output vector field \mathbf{u} restored by the VTE-algorithm inevitably contains some part of the systematic error $\mathbf{h} \in S_h$, as well.*

Nonlinear General Case. As it was mentioned in the beginning of this section, the nonlinear VTE can hardly be solved analytically in a very general case, except of probably restricted number of specific cases. Therefore, instead of conducting a thorough analysis of a systematic error in a general nonlinear case, we restrict our efforts by only the investigation of the results of some numerical runs. An example of such run is described below.

5.1 The Systematic Error of the Approach and Error Reduction Criterion

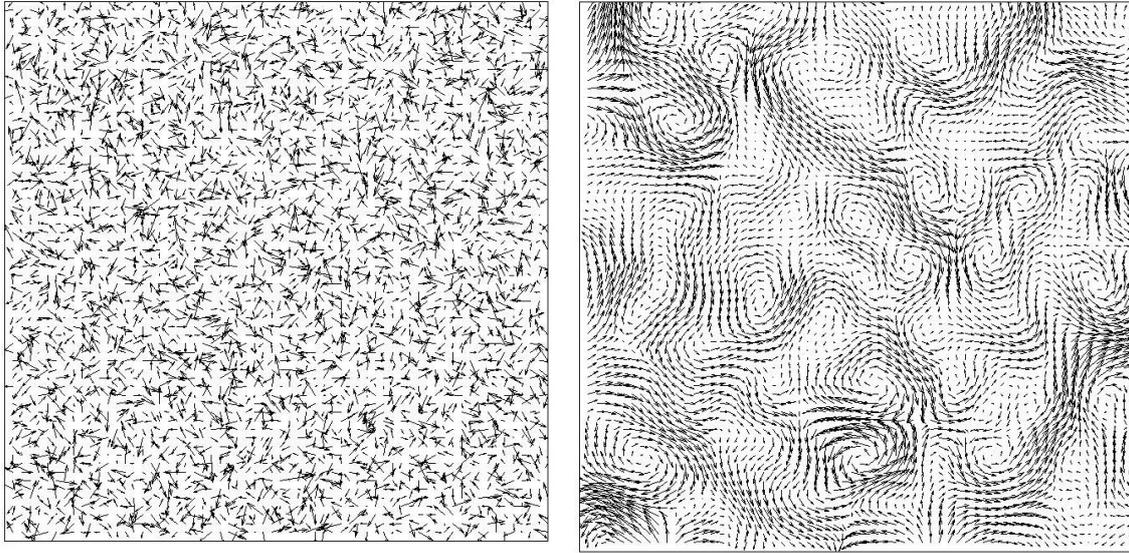


Figure 5.2: **Left:** The conventional Gaussian Noise. **Right:** The hydrodynamical component extracted from the Gaussian noise.

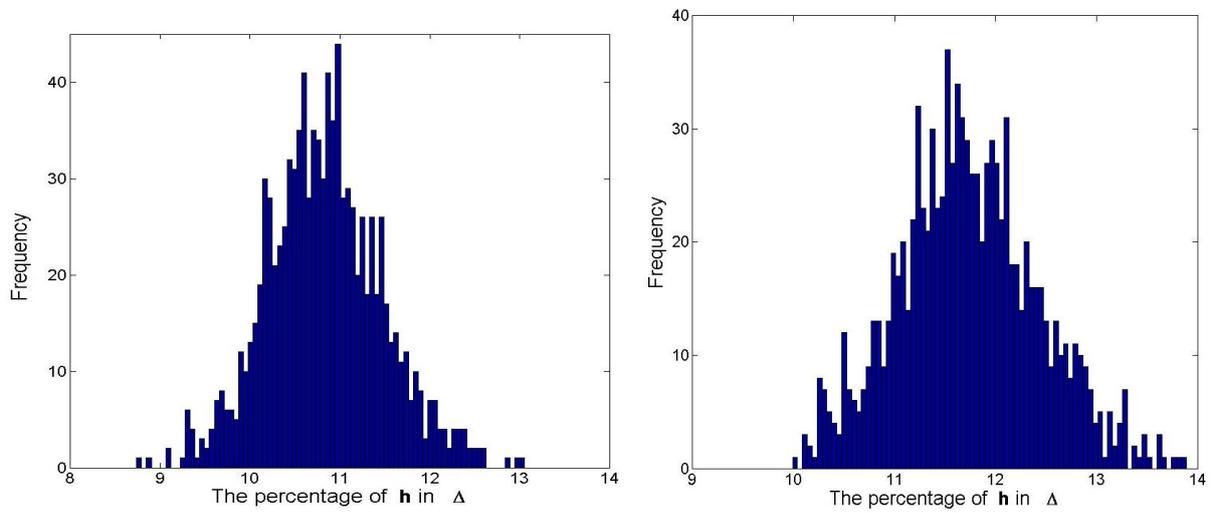


Figure 5.3: The histograms for the amount of hydrodynamical noise in normally distributed Gaussian noise (left) and in homogeneously distributed Gaussian noise (right).

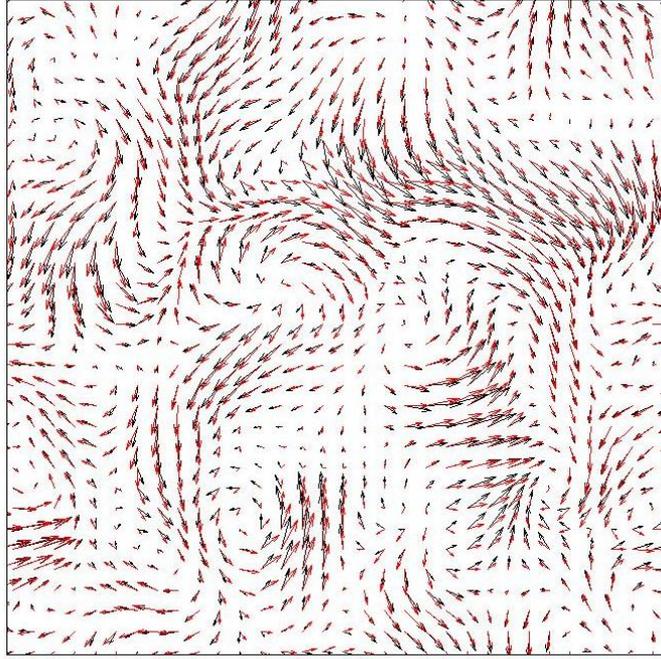


Figure 5.4: The initial vector field \mathbf{v}^I (black arrows) and the resulted vector field $\mathbf{v}_r = \mathbf{v}^I - \mathbf{v}^1$ (red arrows).

A sum of two turbulent vector fields, \mathbf{v}^1 and \mathbf{v}^2 , was set as an input data into the VTE-based algorithm. After that the vector field \mathbf{v}^2 was subtracted from the output \mathbf{v}^I and compared against \mathbf{v}^1 . Figure 5.4 represents a visual example of this comparison. It illustrates the resulted vector field $\mathbf{v}_r = \mathbf{v}^I - \mathbf{v}^2$ (red arrows) and \mathbf{v}^1 (black arrows). As one can see from this figure, both vector fields nearly coincide in the whole domain except of a few areas where the strong nonlinearity is observed. The overall discrepancy between these vector fields is less than 10%. Suppose that in this case $\mathbf{v}^1 = \mathbf{g} \in S_g$ and $\mathbf{v}^2 = \mathbf{h} \in S_h$, then \mathbf{h} in \mathbf{v}^I almost preserves except of the areas with developed strong nonlinearity. Thus, the estimations of the noise-to-signal ratio obtained for the linear case is also valid.

Thus, the output vector field \mathbf{v}^I is a sum of \mathbf{g} and \mathbf{h} . According to the histograms given in Fig. 5.3, the maximum amount of the hydrodynamical component in the noise is not higher than 13%. If the noise-to-signal ratio is in the range between 8 and 12, the specific weight of the hydrodynamical noise and the ground truth flow in \mathbf{v}^I will be approximately the same. Hence we arrive at the following conclusion:

Conclusion 2 (The amount of systematic error.). *The VTE-based restoration approach can definitely recover all corrupted data fields efficiently in case when the noise-to-signal ratio is less than 8.*

Remark 1. *An impact of the Gaussian filter with the default values of $\sigma = 1.34$ on the vector fields satisfying the Kolmogorov's spectra is of about 2 – 3%, therefore we assume hereafter that the changes in spectrum of \mathbf{h} after applying the second step of algorithm are negligible. Hence, if the measured vector field contains only the hydrodynamical component of noise, then the output $\mathbf{u} = \mathbf{g} - \mathbf{h} = \mathbf{d} \in S_I$ and if $\boldsymbol{\delta} = \mathbf{h} = 0$ then $\mathbf{u} = \mathbf{g}$.*

Since the systematic error can not be removed from the field, our goal is to show that the remaining in \mathbf{d} non-hydrodynamical component $\boldsymbol{\delta}$ reduces after the implementation of the VTE-based approach. According to the definition of ideal solution $\mathbf{g} + \mathbf{h} = \mathbf{v}^I \in S_I$, and we can formulate the error reduction criterion:

Definition 7 (Error reduction criterion). *The amount of noise $\boldsymbol{\delta} \in S_{\boldsymbol{\delta}}$ containing in the processed vector field reduces if for some constant $C \in (0, 1)$ the following inequality is valid:*

$$\|\mathbf{u} - \mathbf{v}^I\|_2 \leq C \|\mathbf{d} - \mathbf{v}^I\|_2 ,$$

or alternatively

$$\|\boldsymbol{\delta}_a\|_2 < C \|\boldsymbol{\delta}_b\|_2 ,$$

where $\boldsymbol{\delta}_a$ and $\boldsymbol{\delta}_b$ are the non-hydrodynamical parts of the noise containing in \mathbf{u} and \mathbf{a} , respectively.

5.2 Assumptions

For a more transparent mathematical representation we perform the error reduction analysis considering a rectangular domains Ω : $\Omega = [0, L] \times [0, M] \times [0, N]$. Note, however, that all major results are valid for a more general case of arbitrary volumes. We also assume the case when the first and the second terms in equation (4.12) have an equal impact and set $\alpha = 1$.

Symmetry boundary conditions. Another assumption concerns the boundary conditions. Since the fluid motion is analyzed *inside* the volume Ω , we are free to make some reasonable hypothesis about the fluid motion outside Ω . One can assume that the velocity at the boundary does not change in the line normal of the boundary $\partial\Omega$. The consistency of this assumptions with the hydrodynamical principles was already discussed in Sec. 2.8. Having such boundary conditions and freedom in hypothesizing about the fluid motion outside Ω , we apply here a **symmetry principle** traditionally used in many applied computational problems, the analysis of convective motions [29, 69], for instance. In this experiment the fluid motion in a large volume V is split into a large number of rectangular cells Ω . It is assumed that the fluid motion is similar in every elementary cell and is **symmetrically** reproduced from cell to cell across the whole volume. The physical meaning of the symmetry principle is often associated with the mathematically formulated zero fluxes boundary condition. Thus, the fluid flow within a finite rectangular area with zero flux boundary condition can be treated as a fragment of infinitely replicated symmetrical motion. We will often refer to this fact in this chapter.

Noise model. We restrict the error analysis by the cases when noise is an isotropic and fast-oscillating function with the noise-to-signal ratio much less than one. In practical aspect, the results which will be obtained below in the error analysis remain the same for the noise-to-signal ratios larger than one, although an accurate mathematical proof of its convergence is not obvious in a more general case.

The VTE-based denoising algorithm introduces some changes in values $\mathbf{v}, \omega, \delta$, and these corrections occur at every step. Let us use hereafter the rule: if any object has been modified by the Gaussian filtering (second step), it is marked by a subscript g ; all modifications at 1-st, 3-rd or 4-th are denoted by an appropriate subscripts, i.e. 1, 3 or 4, respectively. For instance, δ_g means the changes of δ at the second step, and ω_3 defines the vorticity obtained after the third step.

A single equation for steps 3 and 4. Note that steps 3 and 4 are represented by similar PDEs, namely eq. (5.3) and (5.5). That means that all relations and conclusions obtained for (5.3) are valid for (5.5), as well. If so, for the simplicity reason instead of working with every equation separately, let us operate with a similar abstract equation which has the same form as (5.3) or (5.5).

Let \mathbf{r} and \mathbf{q} are two vector functions defined on Ω such that \mathbf{q} is twice weakly differentiable and \mathbf{r} is finite. Let ν be a positive constant such that $\nu > 0$. Instead of working with (5.3) and (5.5) we consider below the following abstract problem:

$$\begin{aligned} \mathbf{q} - \nu \Delta \mathbf{q} &= \mathbf{r} , \\ \left. \frac{\partial \mathbf{q}}{\partial \mathbf{n}} \right|_{\partial \Omega} &= 0 . \end{aligned} \tag{5.7}$$

5.3 Divergence Removal

At the first step the solenoidal projection removes the divergent part of the noise from δ . Let's prove that the remaining steps, e.g. eqs. (5.2)-(5.5), do not introduce any extra divergence into the data set. To this end we will prove that the output from the second step is divergent free.

Let $\hat{\mathbf{v}}^I$ be the Fourier transform of some vector field \mathbf{v}^I . The operator *div* in the Fourier space is reduced to a simple scalar product of the frequency \mathbf{w} and the corresponding vector. Thus, in the Fourier domain the continuity equation (2.3) reads

$$\langle i\mathbf{w}, \hat{\mathbf{v}}^I \rangle_{\Omega} = 0 . \tag{5.8}$$

If so, the first two steps of the algorithm, i.e. equations (5.1) and (5.2), can be presented in the Fourier space as a scalar product: $\langle i\mathbf{w}, \hat{\mathbf{v}}_g^I \rangle_{\Omega}$.

In the light of the symmetry assumption introduced in Sect. 5.2, and according to [11], the convolution for multiple areas is a commutative operator, and the last expression can be rearranged as follows

$$\langle i\mathbf{w}, \hat{\mathbf{v}}_g \rangle_{\Omega} = \langle i\mathbf{w}, \hat{g}_{\sigma}(\mathbf{w}) \hat{\mathbf{v}} \rangle_{\Omega} = \hat{g}_{\sigma}(\mathbf{w}) \langle i\mathbf{w}, \hat{\mathbf{v}} \rangle_{\Omega} .$$

This term coincides with the left-hand-side of the continuity equation multiplied by $\hat{g}_\sigma(\mathbf{w})$ and must be equal to zero according to (5.8). Hence, we arrive at an important conclusion: within the studied domain the low-pass filtering does not introduce any divergence into the field. As a result, the third step (5.3) starts with partly cleaned flow.

Let us prove now that steps 3 and 4 do not introduce any divergence, as well. Consider the following lemma.

Lemma 5.3.1. *Assume $\text{div}(\mathbf{r}) = 0$ and \mathbf{q} is three times differentiable function. Then the problem (5.7) has only a trivial solution.*

Proof. Assume $\mathbf{r} = 0$; the homogeneous part of (5.7) is equivalent to

$$\min_{\mathbf{q}} \left\{ \|\mathbf{q}\|_2^2 + \nu \left\| \sum_{i=1}^d \nabla q_i \right\|_2^2 \right\}, \quad \text{where } \mathbf{q}_p = (q_1, \dots, q_d),$$

which has only a trivial minimizer, i.e. $\mathbf{q} = 0$. Hence the homogeneous part of (5.7) has only a trivial solution.

According to the hypothesis $\text{div}(\mathbf{r}) = 0$; then applying the operator div to the problem (5.7) it is reduced to the following:

$$\begin{aligned} \text{div}(\mathbf{q} - \nu \Delta \mathbf{q}) &= 0, \\ \frac{\partial \mathbf{q}}{\partial \mathbf{n}} \Big|_{\partial \Omega} &= 0. \end{aligned} \tag{5.9}$$

Introducing a new variable $\text{div}(\mathbf{q}) = p$ and using the properties $\text{div}(\Delta \mathbf{q}_p) = \Delta \text{div}(\mathbf{q}_p)$ (for details see appendix, formula (10.14)) and $\text{div} \left(\frac{\partial \mathbf{q}_p}{\partial \mathbf{n}} \right) = \frac{\partial \text{div}(\mathbf{q}_p)}{\partial \mathbf{n}}$ one obtains again the homogeneous problem, viz.

$$p - \nu \Delta p = 0, \quad \frac{\partial p}{\partial \mathbf{n}} \Big|_{\partial \Omega} = 0.$$

Hence, $p = 0$ and $\text{div}(\mathbf{q}) = 0$. □

Remark 2. *Note that equations (5.3) and (5.5) have the same form as equation (5.7). Moreover, their right-hand-sides are, in fact, divergent-free functions, $\mathbf{e}(\vec{v}_g)$, $\text{curl}(\boldsymbol{\omega}_g)$ and $\text{curl}(\boldsymbol{\omega})$ (see for details in appendix equation (10.8)).*

Conclusion 3. *According to Sect. 5.3.1, the basic outcome from the performed analysis is that all steps of the procedure, i.e. equations (5.1) - (5.6), produce only divergent-free solutions. Hence, the noise component which contains nonzero divergence is completely removed from the field during the application of Steps 1 - 4. Note, however, that this procedure does not affect the divergence free part of the noise which still exists and contaminates the vector field.*

5.4 Error Reduction

According to the chosen notations, δ_1 is the divergent-free component of the error δ remaining after the application of the first step. Our task within this section is to demonstrate that the remaining part of the noise is reduced step-by-step by application of the Gaussian filtering and the VTE-based filtering.

5.4.1 Error Reduction: Lowpass Filtering

Let $\delta_g = g * \delta_1$ be the error remaining after the Gaussian filtering. According to Young's inequality (see theorem (10.3.4) in Appendix) the following equation is valid:

$$\|\delta_1 * g\|_r \leq C_{p,q} \|\delta_1\|_q \|g\|_p, \quad \text{where } \frac{1}{l} + \frac{1}{p} = \frac{1}{r} + 1 \quad \text{and} \quad C_{p,q} < 1.$$

Let $r = 2$, $p = 1$, and $q = 2$. Since $\|g\|_1 = 1$ and $C_{p,q} < 1$, we arrive at

$$\|\delta_1 * g\|_2 < \|\delta_1\|_2.$$

Hence $\|\delta_g\|_2 < \|\delta_1\|_2$.

5.4.2 Error Reduction: VTE

Following the rule for notations adopted in Sec. 5.2, let us introduce some new definitions used below: $\omega^I = \mathbf{curl}(\mathbf{v}^I)$ is an ideal solution in terms of vorticity; $\omega_g = \mathbf{curl}(\mathbf{v}_g)$ is a vorticity of \mathbf{v}_g ; δ_3 is the error remaining after the third step, and $\mathbf{f}(\mathbf{v}_g) = \omega_g - \mathbf{e}(\mathbf{v}_g)$ is the notation for the RHS of (5.3).

Since we assumed a negligible impact of the Gaussian filtering on the elements S_I (see remark 1) and since \mathbf{v}^I is the solution of steps 1,3 and 4, the substitution of $\mathbf{v} = \mathbf{v}^I + \delta$ into (5.1), (5.3) and (5.5) results in a cancelation of the terms containing only \mathbf{v}^I . The remaining parts in these equations should be attributed to errors.

We derive an expression for the error appearing at the third step. Thereto substitute the velocity vector field obtained after the second step $\mathbf{v}_g = \mathbf{v}^I + \delta_g$ into (5.3) and single out from its right-hand-side the parts containing only \mathbf{v}^I . As a result we obtain

$$\omega^I + \delta_3 - \nu \Delta \omega^I - \nu \Delta \delta_3 = \mathbf{f}(\mathbf{v}^I) + \mathbf{f}(\mathbf{v}^I, \delta_g).$$

Here we used the rearranged RHS of (5.3) where

$$\mathbf{f}(\mathbf{v}^I) = \omega^I - (\omega^I \cdot \nabla) \mathbf{v}^I - (\mathbf{v}^I \cdot \nabla) \omega^I, \quad (5.10)$$

and

$$\mathbf{f}(\mathbf{v}^I, \delta_g) = \mathbf{curl}(\delta_g) - \mathbf{curl}((\delta_g \cdot \nabla) \delta_g) - \mathbf{curl}((\delta_g \cdot \nabla) \mathbf{v}^I) - \mathbf{curl}((\mathbf{v}^I \cdot \nabla) \delta_g). \quad (5.11)$$

Note that although the third step enforces the consistency of the output data with VTE, the denoised result still contains some noise. The incomplete noise removal of the third

step is explained by the fact that instead of Lagrangian multiplier we set the constant parameter α in the regularizer at the third step. As a result the regularizer which enforces the consistency with VTE does not have proper impact on the functional. Let δ_3 be an error remaining after the third step, and $\omega = \omega^I + \delta_3$ be the denoised vorticity at this stage. After substitution of ω and $f(\mathbf{v}_g)$ into (5.3) the third step reads

$$\omega^I + \delta_3 - \nu \Delta \omega^I - \nu \Delta \delta_3 = \mathbf{f}(\mathbf{v}^I) + \mathbf{f}(\mathbf{v}^I, \delta_3) .$$

Since \mathbf{v}^I satisfies the VTE by the definition, it should satisfy also the third step: $\omega^I - \nu \Delta \omega^I = \mathbf{f}(\mathbf{v}^I)$. Using this equality we can simplify this equation to the following

$$\delta_3 - \nu \Delta \delta_3 = \mathbf{f}(\mathbf{v}^I, \delta_3) . \quad (5.12)$$

This is a general expression for the error δ_3 remaining after three steps of the denoising procedure. For its solution we need some theoretical background which will be formulated below in the form of some supplementary theorems. We start with the Plancherel's theorem which is formulated here without proof (for more details we refer the reader to the original source [24]).

Theorem 5.4.1 (Plancherel's Theorem). *Let $u \in L^2(\mathbb{R})$ and \hat{u} and u are it's the Fourier and inverse images, respectively. Then $\hat{u}, u \in L^2(\mathbb{R}^d)$, and*

$$\|u\|_2 = \|\hat{u}\|_2 = \|u\|_2 . \quad (5.13)$$

Proposition 1. *Consider equation (5.7). If $\hat{\mathbf{q}}$ and $\hat{\mathbf{r}}$ are the Fourier images of the solution \mathbf{q} and the right hand side \mathbf{r} , respectively, then the following expression is valid [24]:*

$$\hat{\mathbf{q}} = \frac{\hat{\mathbf{r}}}{1 + \nu |\phi|^2} , \quad (5.14)$$

where ϕ is a frequency in the Fourier domain.

Let's investigate the impact of each single term in \mathbf{f} on the final solution of (5.12). The equation (5.7) is linear, hence its solution \mathbf{q} can be presented as a sum of the solutions \mathbf{q}_1 and \mathbf{q}_2 of two similar linear equations:

$$\begin{aligned} \mathbf{q}_1 - \nu \Delta \mathbf{q}_1 &= \mathbf{r}_1, & \mathbf{q}_2 - \nu \Delta \mathbf{q}_2 &= \mathbf{r}_2 , \\ \left. \frac{\partial \mathbf{q}_1}{\partial \mathbf{n}} \right|_{\partial \Omega} &= 0 & \left. \frac{\partial \mathbf{q}_2}{\partial \mathbf{n}} \right|_{\partial \Omega} &= 0 \end{aligned} \quad (5.15)$$

where $\mathbf{r} = \mathbf{r}_1 + \mathbf{r}_2$.

Theorem 5.4.2. *Assume that the Fourier image of the right hand side \mathbf{r}_1 in (5.15) has a low band spectrum distribution with its maximum at zero frequency and vanishing level to the right of some frequency ϕ_f . Let $\|\mathbf{r}_2\|_2 = m$ and $\|\mathbf{r}_1\|_2 = M$. If $\frac{M}{(1+\nu|\phi_f|^2)} > m$ and $C = \frac{M}{m(1+\nu|\phi_f|^2)}$, then the following estimate is valid: $\|\mathbf{q}_1\|_2 > C\|\mathbf{q}_2\|_2$.*

5 Error Analysis

Proof. According to (5.14)

$$\|\hat{\mathbf{q}}_2\|_2 = \left\| \frac{\hat{\mathbf{r}}_2}{1 + \nu|\phi|^2} \right\|_2 < \|\hat{\mathbf{r}}_2\|_2.$$

Using the Placherel's theorem (5.4.2) which states that $\|\hat{\mathbf{q}}_2\|_2 = \|\mathbf{q}_2\|_2$ and $\|\hat{\mathbf{r}}_2\|_2 = \|\mathbf{r}_2\|_2$, we arrive at:

$$\|\mathbf{q}_2\|_2 < m. \quad (5.16)$$

Let us obtain a similar relation for $\|\mathbf{q}_1\|_2$. Since $\hat{\mathbf{r}}_1$ vanishes at frequency $|\phi_f|$, the following estimation is valid:

$$\|\mathbf{q}_1\|_2 = \|\hat{\mathbf{q}}_1\|_2 = \left\| \frac{\hat{\mathbf{r}}_1}{1 + \nu|\phi|^2} \right\|_2 > \frac{\|\hat{\mathbf{r}}_1\|_2}{1 + \nu|\phi_f|^2} = \frac{\|\mathbf{r}_1\|_2}{1 + \nu|\phi_f|^2} = \frac{M}{1 + \nu|\phi_f|^2},$$

or

$$\|\mathbf{q}_1\|_2 > \frac{M}{1 + \nu a^2}. \quad (5.17)$$

Dividing (5.16) by (5.17) we obtain

$$\frac{\|\mathbf{q}_2\|_2}{\|\mathbf{q}_1\|_2} < \frac{m(1 + \nu|\phi_f|^2)}{M},$$

or

$$\|\mathbf{q}_1\|_2 > C\|\mathbf{q}_2\|_2.$$

□

Simplification of equation (5.12). In the reminder of this section, we discuss upper bounds of the RHS of (5.12). Because this is a highly nonlinear term, the discussion will be informal, and we do not claim to derive rigorous mathematical results. Yet, our insights will be mathematically confirmed later on.

Equation (5.12) can hardly be solved directly in order to find δ_3 . The main reason for that is the complexity of its right hand side, i.e. $\mathbf{f}(\mathbf{v}^I, \delta_g)$ and the presence of nonlinear terms. However, it can be significantly simplified in cases when it is possible to exclude some terms from the analysis which have negligible impact. The formulated above theorem 5.4.2 gives a clue for such simplifications. It follows from eq. (5.11) that

$$\mathbf{f}(v^I, \delta_g) = \mathbf{curl}(\delta_g) - \mathbf{curl}((\delta_g \cdot \nabla)\delta_g) - \mathbf{curl}((\delta_g \cdot \nabla)\mathbf{v}^I) - \mathbf{curl}((\mathbf{v}^I \cdot \nabla)\delta_g). \quad (5.18)$$

Let us estimate the norm of the first term in (5.18). In the beginning of this section we have assumed that δ is isotropic and homogeneously distributed function. The Gaussian filtering does not disturb these properties, so δ_g is isotropic and homogeneous, as well. Suppose that Ω is sufficiently large so that we can estimate δ_g and its derivatives with their mean values. For instance

$$\|\mathbf{curl}(\delta_g)\|_2^2 = \langle \mathbf{curl}(\delta_g), \mathbf{curl}(\delta_g) \rangle_\Omega \approx \overline{\langle \mathbf{curl}(\delta_g), \mathbf{curl}(\delta_g) \rangle}_\Omega,$$

should be valid. Here $\overline{\mathbf{curl}(\boldsymbol{\delta}_g)}$ is the mean value of $|\mathbf{curl}(\boldsymbol{\delta}_g)|$. Let D_m be an average amplitude of oscillation of $\boldsymbol{\delta}_g$ and L_m be the average distance for such oscillation. Then we estimate $|\overline{\mathbf{curl}(\boldsymbol{\delta}_g)}|$ using D_m/L_m , and for the norm estimation we can write

$$\|\mathbf{curl}(\boldsymbol{\delta}_g)\|_2 \approx \frac{D_m}{L_m} .$$

Applying similar reasoning we estimate the second term in (5.18) as follows

$$\|\mathbf{curl}((\boldsymbol{\delta}_g \cdot \nabla)\boldsymbol{\delta}_g)\|_2 \approx \frac{D_m^2}{L_m^2} .$$

Let U be the velocity mean value and L be the scale for the average velocity change. Then applying the same reasoning as above we obtain the norm for the third term, i.e.

$$\|\mathbf{curl}((\boldsymbol{\delta}_g \cdot \nabla)\mathbf{v}^I)\|_2 \approx \frac{UD_m}{L_m L} ,$$

and for the fourth term,

$$\|\mathbf{curl}((\mathbf{v}^I \cdot \nabla)\boldsymbol{\delta}_g)\|_2 \approx \frac{UD}{L_m^2} .$$

With the aim of establishing the relationships between the terms in (5.18) let us write the estimations for norms obtained above in one line (the order of this sequence corresponds to the order of the terms in (5.18))

$$\frac{D_m}{L_m} \left| \frac{D_m^2}{L_m^2} \right| \frac{UD_m}{L_m L} \left| \frac{UD_m}{L_m^2} \right| .$$

Since the dimensional parameter α was set to one its value was omitted from the consideration but not its dimension. Thus, one should keep in mind that the last three terms in the sequence above are implicitly multiplied by the α 's dimension which is $[time]$, and the whole expression has the dimension of $[1/time]$. Dividing the sequence by $\frac{D_m}{L_m}$ one obtains the following series

$$1 \left| \frac{D_m}{L_m} \right| \frac{U}{L} \left| \frac{U}{L_m} \right| . \quad (5.19)$$

Let us finalize the reasoning concerning the terms in (5.19) in order to determine the term in RHS (5.18) which has the strongest impact on the solution of (5.12). In Sec. 5.2 we assumed that $\boldsymbol{\delta}_g$ is a fast oscillating function (in comparison with \mathbf{v}^I). This means that $L \gg L_m$. We have hypothesized as well that the error scale is much smaller than the velocity scale, i.e. $D_m \ll U$. Hence, the norm of the fourth term in (5.19) is much greater than the norms of other terms. According to the Kolmogorov-Obuhov law (see Sec. 2.7), the velocity included in the fourth term must have a low-band spectrum distribution. If so, in the Fourier domain the whole fourth term should have a low-band spectrum distribution, as well, with its value vanishing at the Kolmogorov's frequency which is much lower than π .

5 Error Analysis

Summarizing the results from the previous paragraph and applying theorem 5.4.2 one can identify some terms of (5.18) with negligible contribution. To this end, we combine the first three terms in (5.18) in one and estimate its norm as the largest value which is characteristic to one of them. In terms of theorem 5.4.2 let us define this norm as m , and the norm of the fourth term by M . Since $M \gg m$ and the value $|\phi_f|$ for the fourth term is much smaller than π , the constant $C \gg 1$. Consequently, according to theorem 5.4.2, the fourth term in (5.18) has the strongest impact on the solution of (5.12). Hence, without substantial losses of accuracy, one can simplify $\mathbf{f}(\mathbf{v}^I, \delta_g)$ to the following form:

$$\mathbf{f}(\mathbf{v}^I, \delta_g) = -\mathbf{curl}((\mathbf{v}^I \cdot \nabla)\delta_g) = -(\mathbf{v}^I \cdot \nabla)\mathbf{curl}(\delta_g) - (\mathbf{curl}(\mathbf{v}^I) \cdot \nabla) \times \delta_g. \quad (5.20)$$

Again, estimating norms of the terms in the right hand side with their average values we obtain

$$\frac{UD_m}{L_m^2} \mid \frac{UD_m}{LL_m},$$

or

$$\frac{1}{L_m} \mid \frac{1}{L}.$$

Thus, the term $-(\mathbf{v}^I \cdot \nabla)\mathbf{curl}(\delta_g)$ has the strongest impact on the solution of (5.12) which in its simplified version reads:

$$\delta_3 - \nu \Delta \delta_3 = -(\mathbf{v}^I \cdot \nabla)\mathbf{curl}(\delta_g). \quad (5.21)$$

At this point we can obtain some results in a 2D-case for which $\mathbf{v}^I = \mathbf{v}^I(u, v)$ and $\delta_g = \delta_g(\delta_{gx}, \delta_{gy})$. In our further analysis we will use the following obvious relations:

$$\begin{aligned} -(\mathbf{v}^I \cdot \nabla)\mathbf{curl}(\delta_g) &= \left(u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y} \right) \left(\frac{\delta_{gx}}{\partial y} - \frac{\delta_{gy}}{\partial x} \right) = \\ &= u \frac{\partial^2 \delta_{gx}}{\partial x \partial y} + v \frac{\partial^2 \delta_{gx}}{\partial y^2} - u \frac{\partial^2 \delta_{gy}}{\partial x^2} - v \frac{\partial^2 \delta_{gy}}{\partial x \partial y}. \end{aligned} \quad (5.22)$$

The function δ_g obeys the continuity equation. In other words, it is divergent-free which means that

$$\frac{\delta_{gx}}{\partial x} = -\frac{\delta_{gy}}{\partial y}. \quad (5.23)$$

The last circumstance immediately leads to the following simplification

$$-(\mathbf{v}^I \cdot \nabla)\mathbf{curl}(\delta_g) = (v - u) \left(\frac{\partial^2 \delta_{gx}}{\partial x^2} + \frac{\partial^2 \delta_{gy}}{\partial y^2} \right). \quad (5.24)$$

As a result, we arrive at the equation

$$\delta_3 - \nu \Delta \delta_3 = (u - v) \left(\frac{\partial^2 \delta_{gx}}{\partial x^2} + \frac{\partial^2 \delta_{gy}}{\partial y^2} \right). \quad (5.25)$$

Using (5.14) and formulas (10.26), (10.27) we obtain

$$\hat{\delta}_3 = \left(\hat{\delta}_{gx} + \hat{\delta}_{gy} \right) * \frac{(\hat{u} - \hat{v}) |\phi|^2}{1 + \nu |\phi|^2}. \quad (5.26)$$

This equation gives an exact estimation for δ_3 in the $2D$ case when $U \gg Dm$. However, the direct use of (5.26) will make the further analysis very complicated. Instead, we will use the estimation of (5.26). Thereto instead of the term $(\hat{u} - \hat{v})$ in eq. (5.26) let us substitute $Ub = 2 \max(|u|, |v|)$, then we obtain the following inequality:

$$|\hat{\delta}_3| \leq \left| Ub \frac{(\hat{\delta}_{gx} + \hat{\delta}_{gy}) |\phi|^2}{1 + \nu |\phi|^2} \right|. \quad (5.27)$$

5.4.3 Velocity reconstruction

So far we discussed the performance of the denoising algorithm at stages of solenoidal projection (Step 1), Gaussian filtering (Step 2), and vorticity rectification (Step 3). With the considered above theorems we could show the conditions when an error is reduced in the course of passing of these three stages and, as a result, the solution converges to the original real field. In this sub-section we consider the last step, i.e. the velocity reconstruction, and estimate an efficiency of this procedure for the error diminishing.

Let δ_3 be an error remaining after a single application of the first three steps. Note that equation (5.5) presenting the fourth step is similar to equation (5.12) for δ_3 . If so, applying the same reasoning as we did before, we derive the following boundary value problem for δ_4 :

$$\begin{aligned} \delta_4 - \beta \Delta \delta_4 &= \delta_g - \beta \text{curl}(\delta_3), \\ \frac{\partial \delta_4}{\partial n} \Big|_{\partial \Omega} &= 0. \end{aligned} \quad (5.28)$$

Similarly to what was done in the previous subsection by using equation (5.14) and the Fourier differentiation rule (10.26), we obtain here the expression for the Fourier images for δ_4 as it is shown below:

$$\begin{aligned} \hat{\delta}_{4x} &= \frac{\hat{\delta}_{gx} - \beta \hat{\delta}_3 \phi_y}{1 + \beta |\phi|^2}, \\ \hat{\delta}_{4y} &= \frac{\hat{\delta}_{gy} + \beta \hat{\delta}_3 \phi_x}{1 + \beta |\phi|^2}. \end{aligned} \quad (5.29)$$

A routine substitution of (5.26) in (5.29) gives the exact formula for δ_4 . Expression of δ_g via δ gives us the desirable relation between δ and δ_4 . It is very complicated for the convergence analysis due to the complexity of the *exact* formulas for δ_3 and δ_4 . However, some results for the error estimation at the fourth step might be obtained by considering

5 Error Analysis

the extreme values of (5.29). Thereto let us at first mark out the set of parameters at which the inequality $|\hat{\delta}_4| < |\hat{\delta}_g|$ takes place and the regions where the relation $|\hat{\delta}_4| \geq |\hat{\delta}_g|$ is possible.

As long as the values of $\hat{\delta}_{4y}$ and $\hat{\delta}_{4x}$ are of the same order, it is sufficient to consider only one of them for the estimation of the error δ_4 . Moreover, since $\hat{\delta}_3$ and $\hat{\delta}_g$ have isotropic spectra with respect to the directions in the Fourier space (in the beginning we assumed that error is isotropic in frequency space), $\hat{\delta}_4$ has also isotropic spectrum. Hence, instead of the analysis of $\hat{\delta}_{4y}$ and $\hat{\delta}_{4x}$ in whole frequency domain, we can restrict the analysis considering them in some definite amplitude-frequency slice. To be more specific, let us consider only positive values of δ_{4y} distributed in $(\phi_x, \phi_y) \in [0, \infty) \times [0, \infty)$.

As it follows from (5.29), $\delta_{4y} \geq |\delta_{4x}|$ when $\delta_{gy}, \delta_{gx}, \delta_3$ are positive. Let us find the interval of possible increase of $\hat{\delta}_{4y}$. For that we introduce a new variable $\eta = \frac{Ub|\phi|^2}{1+\nu|\phi|^2}$. Then using the second expression in (5.29) we can derive an estimation for the maximum values of $\hat{\delta}_{4y}$ which is

$$\hat{\delta}_{4y} \leq \hat{\delta}_{gy} \frac{(1 + 2\beta\eta\phi_x)}{1 + \beta|\phi|^2}. \quad (5.30)$$

The right hand side of this expression corresponds to the maximum possible value of $|\hat{\delta}_{4y}|$. Obviously, it will increase when the numerator is greater than the denominator, which means that

$$(1 + 2\beta\eta\phi_x) > 1 + \beta|\phi|^2.$$

Since $|\phi|^2 = \phi_x^2 + \phi_y^2 \geq \phi_x^2$, then for a given ϕ_x the right hand side of the expression above has the minimum values if $|\phi| = \phi_x$. That means that for $|\phi| = \phi_x$ we obtain

$$\phi_x(1 + 2\beta\eta\phi_x) > 0.$$

From this inequality it follows that an interval of possible error increase is not larger than $(0, 2\eta)$.

Now let us estimate the maximum value of δ_{4y} which it reaches within $(0, 2\eta)$. According to (5.30), it should be smaller than the maximum of the ratio

$$\frac{(1 + 2\beta\eta\phi_x)}{1 + \beta\phi_x^2}.$$

Taking the derivative of the ratio and equating it to zero, we obtain that it reaches its maximum at $\phi_m = \frac{\sqrt{1+4\beta\eta^2}}{2\beta\eta}$. Substituting ϕ_m into the right hand side of (5.30) we obtain

$$|\hat{\delta}_{4y}| \leq \frac{2\beta\eta^2}{\sqrt{1 + 4\beta\eta^2} - 1}. \quad (5.31)$$

Thus, if $\eta = 1$, which means no error reduction at the second step, the error amplification within interval $(0, 2\eta)$ is not greater than 1.62. At the same time if $\eta = 0.1$, the error amplification is around 1.001. Analyzing the results presented above one can conclude that the error partially transfers from high to low frequencies.

Making a short summary one can conclude that the behavior of the error $\hat{\delta}_4$ (its increase or decrease) depends on the fragment of of the spectra where it is analyzed. However, it is not clear in advance weather its norm δ_4 is reduced after the fourth step. Since the fourth step operates with the data containing some errors from the second and the third steps, in order to make the relation between errors more transparent, let us compare the norms of errors from steps 2 and 4. Thereto, assume $\hat{\delta}_g$ is a stepwise function such that it is equal to some constant $\hat{\delta}_g(\phi_c) = C$ in the circle with radius $[0, \phi_c]$, and zero elsewhere. If so, the estimation norm of $\hat{\delta}_4$ reads ¹

$$\|\hat{\delta}_4\|_2 \leq C \sqrt{\pi \left(\frac{1 + \beta(2\phi_c - \eta)\eta}{1 + \phi_c^2\beta} + \frac{(1 + \beta\eta^2) \arctan(\phi_c\sqrt{\beta})}{\phi_c\sqrt{\beta}} \right)}, \quad (5.32)$$

and the norm of $\hat{\delta}_4$ is

$$\|\hat{\delta}_g\|_2 = C \sqrt{\pi\phi_c}. \quad (5.33)$$

Then, the ratio $\|\hat{\delta}_4\|_2/\|\hat{\delta}_g\|_2$ can be estimated as follows

$$\frac{\|\hat{\delta}_4\|_2}{\|\hat{\delta}_g\|_2} \leq Q = \sqrt{\frac{1 + \beta(2\phi_c - \eta)\eta}{\phi_c(1 + \phi_c^2\beta)} + \frac{(1 + \beta\eta^2) \arctan(\phi_c\sqrt{\beta})}{\phi_c^2\sqrt{\beta}}}. \quad (5.34)$$

The value Q in (5.34) might be greater or smaller than 1 depending on the values of parameters η, ϕ_c, β . The surface plotted in the left panel in Figure 5.5 corresponds to the dependence $Q(\eta, \phi_c)$ with $\beta = 1$. Two areas colored in blue and red depicted to the right mark out the regions of parameters (η, ϕ_c) where Q is bigger or smaller than 1. The red area corresponds to values of $Q \geq 1$, while the blue area covers the set of $(\eta, |\phi|)$ where $Q < 1$.

Note, that $Q \geq \|\hat{\delta}_4\|_2/\|\hat{\delta}_g\|_2$, so the size of the red area might be bigger than the real size of the set of (η, ϕ_c) when the ratio $\|\hat{\delta}_4\|_2/\|\hat{\delta}_g\|_2 \geq 1$. However, the blue part of this surface marks the area where $Q < 1$ and hence for the set of (η, ϕ_c) corresponding to blue area the ratio $\|\hat{\delta}_4\|_2/\|\hat{\delta}_g\|_2$ is definitely less than one, and, consequently, the algorithm converges.

5.5 Conclusion

Basic conclusions from this chapter concern the functions δ_g that are homogeneous with respect to the direction of the background flow. For such functions and viscosity values comparable to 1 the error reduction tends to U/ν with the increase of $|\phi|$ in (5.27). In other words, for a high-frequency part of noise the following estimation of error reduction takes place

$$\delta_3 \sim R|\delta_g|. \quad (5.35)$$

¹This norm was computed using the Wolfram Mathematica software.

5 Error Analysis

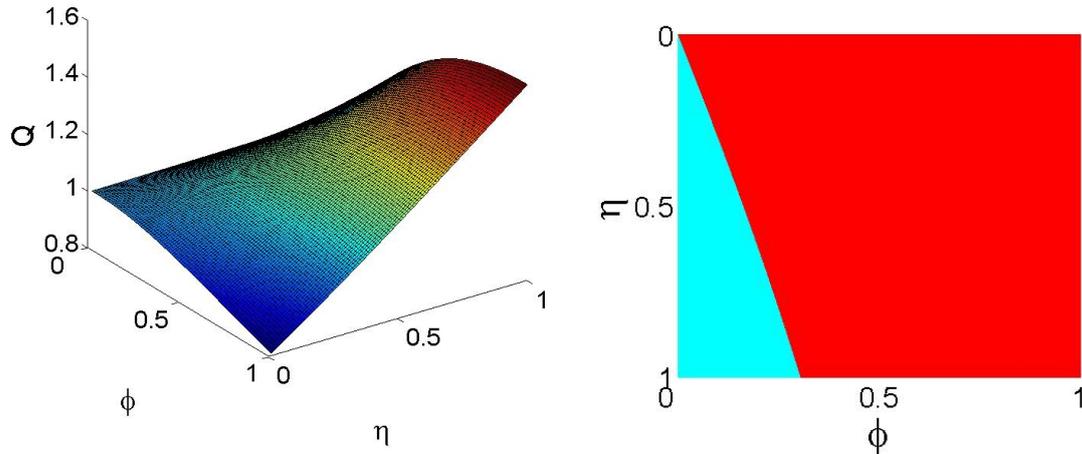


Figure 5.5: The error reduction plots. **Left:** The surface graph of Q depending on η and $|\phi|$. **Right:** Two marked areas correspond to values $Q \geq 1$ (red) $Q < 1$ (blue).

Though the form of equation (5.12) implies that its right-hand-side should have a decreasing at high-frequencies band spectrum, the form of the solution for δ_3 is independent on frequency; it is just a re-scaled version of δ_g for an appropriate value of ν . Three graphs shown in Fig. 5.6 confirm this idea. Note that the 're-scaled' by factor 5 error and the remaining after denoising procedure error to the right of $\pi/4$ coincide nearly perfectly. Figure 5.7 depicts four spectra obtained for different numbers of ν . It is clear that even in a low-frequency band the spectrum of δ_3 is just a re-scaled version of δ_g almost fully copying the shape of the initial errors.

However, too large viscosity values result in damping of the whole output spectrum together with error spectrum, see Fig. 5.8. Just as an opposite case if the viscosity value was set too close to zero, then according to (5.27) the high frequencies of δ_3 will be amplified at the third step, the example of such amplification is given on Fig. 5.9. The correct viscosity values makes the shape of the output vorticity spectrum closer to the ground truth vorticity spectrum. Figure 5.10 justifies this statement.

It is important to note also that the developed method has some restrictions on the error values. According to the results obtained in this section, the reduction of error can be guaranteed in the case when the denoising at the third step is sufficient enough; otherwise the method efficiency is not obvious (for more details see Fig. 5.5).

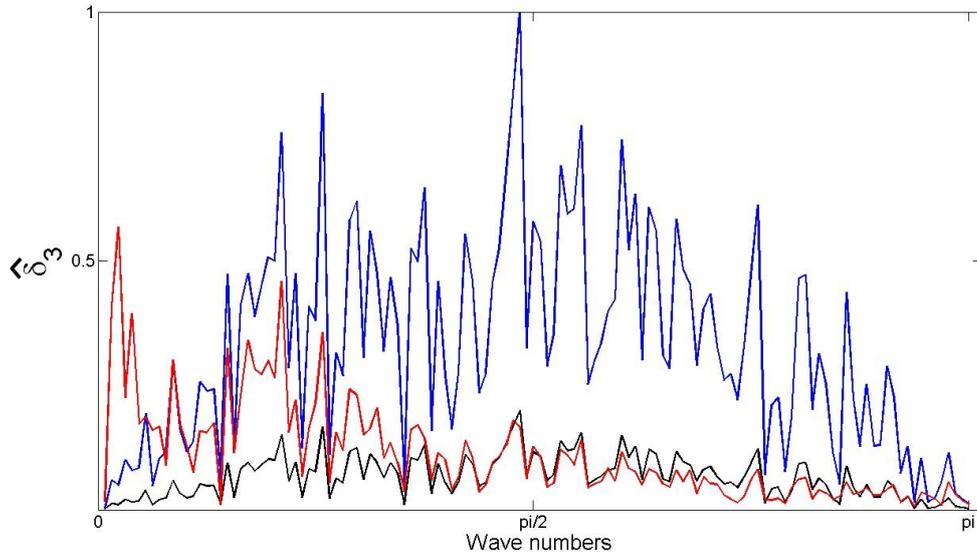


Figure 5.6: The error decay plot. The value δ reduces in proportion to the growth of ν . Three graphs represent the spectra of δ_3 and $\text{curl}(\delta_g)$. **Blue:** $\text{curl}(\delta_g)$ without denoising. **Black:** δ_3 remaining in ω after denoising with $\nu = 1$. **Red:** The re-scaled δ_3 by a factor 5.

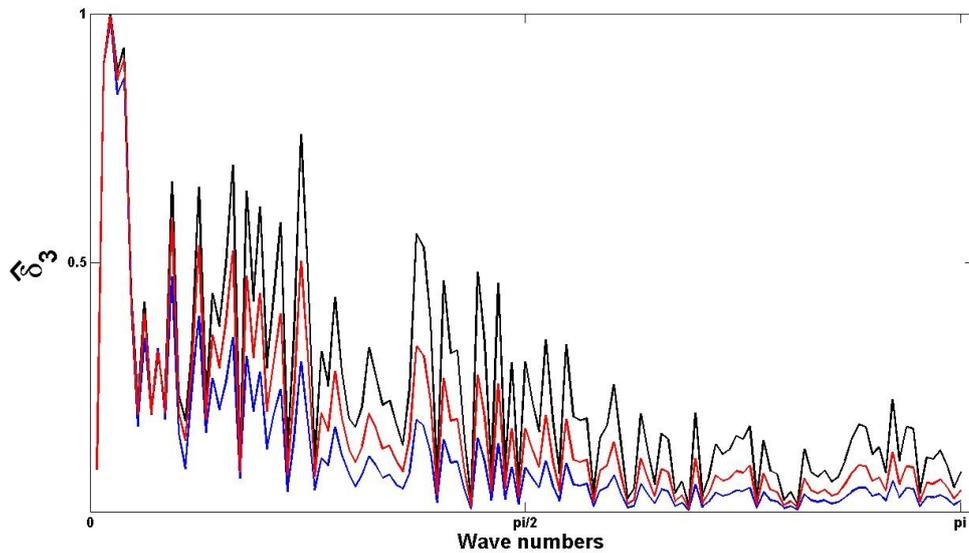


Figure 5.7: The error decay plot. The parameter δ is reduced in proportion to the growth of ν . Three graphs represent the spectra of δ_3 , remaining in ω after denoising with different values of ν : **Blue:** $\nu = 4$. **Red:** $\nu = 2$. **Black:** $\nu = 1$.

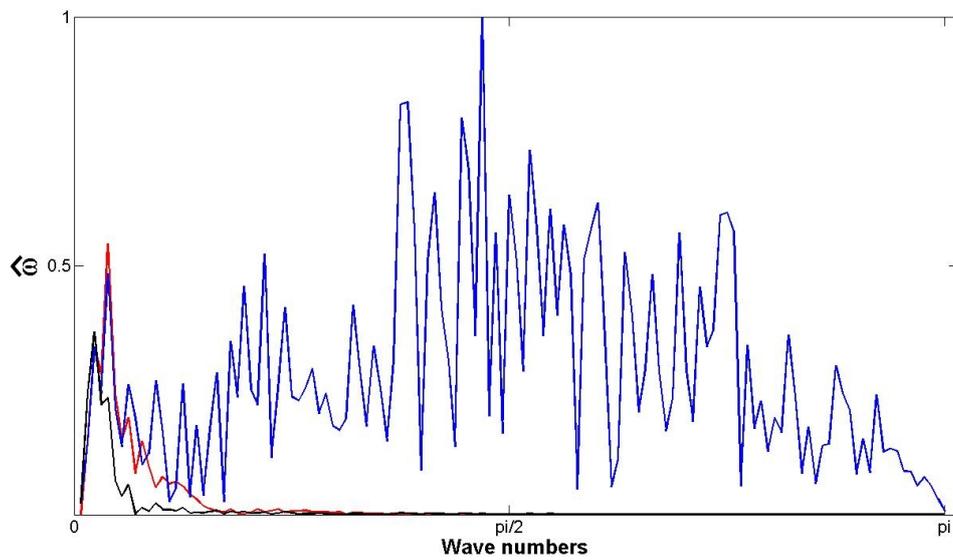


Figure 5.8: Three spectra graphs of vorticity. **Blue:** Noisy signal. **Red:** Ground truth signal. **Black:** Denoised signal. The viscosity was taken $\nu = 100$.

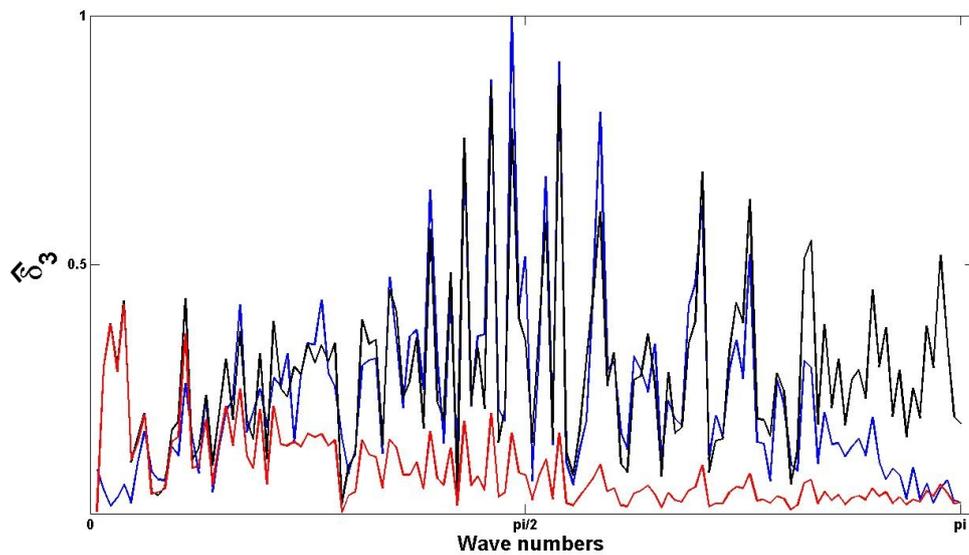


Figure 5.9: Three spectra graphs of error after denoising at the third step. **Blue:** Noisy signal. **Red:** Denoised signal with $\nu = 1$. **Black:** Denoised signal with $\nu = 0$.

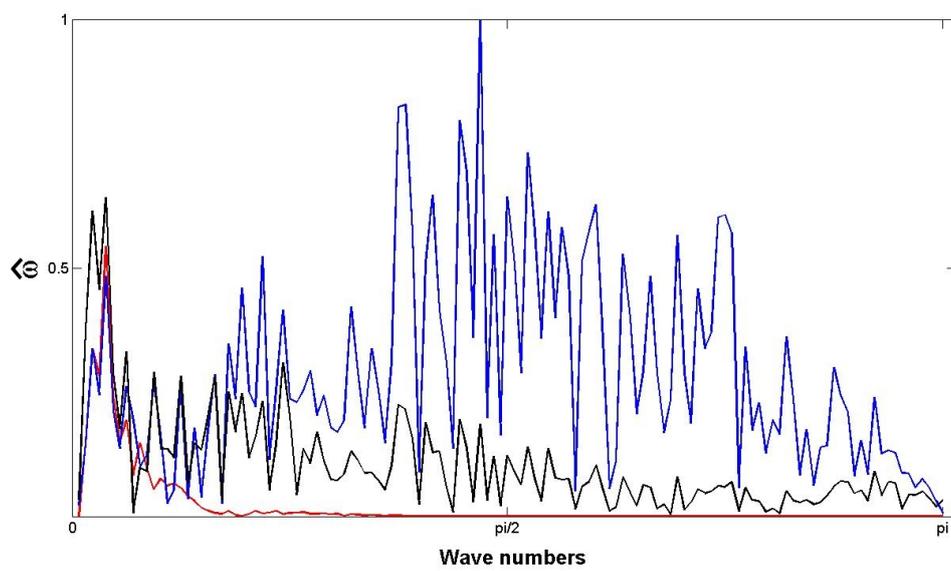


Figure 5.10: Three spectra graphs of vorticity. **Blue:** Noisy signal. **Red:** Ground truth signal. **Black:** Denoised signal. The value of ν corresponds to the viscosity of the ground truth flow.

6 Discretization

In this chapter we give a brief overview of the computational methods used for the numerical implementation of the restoration algorithm described in Sec. 4. The discretization of the governing equations can be done using two independent techniques: finite difference method (FDM) and finite element method (FEM). In this section we dwell briefly on advantages and drawbacks of every, but first of all we touch upon the basics of the finite element and finite difference theories and discuss their applicability to every step of the algorithm. At the end we will formulate some conclusions concerning the discussed numerical procedures. Since we focus only on a routine application of the FEM and FDM to a particular problem and do not intend to develop any new theoretical issues of these methods, the scope of this chapter will be restricted by the discussion on the main statements derived in the FEM and FDM theories which will be given here without proof. For more details we refer the reader to the background sources (handbooks [6, 4, 10, 39] and [51]).

6.1 Finite Element Method

As a starting point of this section we will introduce some notations. Let

$$a : H \times H \rightarrow \mathbb{R}$$

be a symmetric bilinear form, and

$$\mathbf{f} : H \rightarrow \mathbb{R}$$

be a linear functional. Let V be a closed convex set in H . Consider the following continuous minimization problem

$$J(\mathbf{v}) := a(\mathbf{v}, \mathbf{v}) - \langle \mathbf{f}, \mathbf{v} \rangle_{\Omega} \rightarrow \min_{\mathbf{v}}, \quad (6.1)$$

where $\mathbf{v} = (v_1, \dots, v_d)^T$.

6.1.1 Ritz-Galerkin Method

The Ritz-Galerkin method underlies the FEM. In a very general context it can be formulated as follows: for the energy functional J find a discrete minimizer \mathbf{u}_h defined in a finite dimension space $V^h \in V$ with the discretization parameter h , such that at $h \rightarrow 0$ the minimizer \mathbf{u}_h converges to continuous minimizer for J .

Theorem 6.1.1. *Given a minimization problem:*

$$\min_{\mathbf{u} \in V^h} J(\mathbf{u}) . \quad (6.2)$$

It has a unique minimum \mathbf{u}_h if

$$a(\mathbf{u}_h, \boldsymbol{\eta}) = \langle \mathbf{f}, \boldsymbol{\eta} \rangle_{\Omega} \quad \forall \boldsymbol{\eta} \in V^h , \quad (6.3)$$

where $a(\mathbf{u}_h, \boldsymbol{\eta})$ is a H -elliptic bilinear form.

Proof. For $\mathbf{u}_h, \boldsymbol{\eta} \in V^h$ and $\epsilon \in \mathbb{R}$ one can write

$$\begin{aligned} J(\mathbf{u}_h + \epsilon \boldsymbol{\eta}) &= a(\mathbf{u}_h + \epsilon \boldsymbol{\eta}, \mathbf{u}_h + \epsilon \boldsymbol{\eta}) - \langle \mathbf{f}, \mathbf{u}_h + \epsilon \boldsymbol{\eta} \rangle_{\Omega} = \\ &= J(\mathbf{u}_h) + 2\epsilon \left(a(\mathbf{u}_h, \boldsymbol{\eta}) - \langle \mathbf{f}, \boldsymbol{\eta} \rangle_{\Omega} \right) + \epsilon^2 a(\boldsymbol{\eta}, \boldsymbol{\eta}) . \end{aligned}$$

Using condition (6.3) one can obtain that for $\forall \epsilon > 0$

$$J(\mathbf{u}_h + \epsilon \boldsymbol{\eta}) = J(\mathbf{u}_h) + \epsilon^2 a(\boldsymbol{\eta}, \boldsymbol{\eta}) > J(\mathbf{u}_h) .$$

Thus, \mathbf{u}_h is a unique minimizer for (6.2). □

Having proved the uniqueness of minimizer (6.2), let's derive the explicit expressions for \mathbf{u}_h . Let $\varphi_1, \dots, \varphi_N$ be a set of basis functions of V^h . Then

$$\mathbf{u}_h = \sum_{i=1}^N \mathbf{x}_i \varphi_i , \quad (6.4)$$

where $\mathbf{x}_i = (x_{1i}, \dots, x_{di})^T$ are unknown coefficients. Since $\boldsymbol{\eta}$ is any function in V^h , let's take $\boldsymbol{\eta} = \varphi_j \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon}$ is a d -dimensional unit vector. With this remark the latter expression takes the form

$$a(\mathbf{u}_h, \varphi_j \boldsymbol{\epsilon}) = \langle \mathbf{f}, \varphi_j \boldsymbol{\epsilon} \rangle_{\Omega} .$$

After substitution of the expression (6.4) for \mathbf{u}_h , one can write

$$\sum_{i=1}^N a(\varphi_i \mathbf{x}_i, \varphi_j \boldsymbol{\epsilon}) = \langle \mathbf{f}, \varphi_j \boldsymbol{\epsilon} \rangle_{\Omega} \quad j = \overline{1, N} ,$$

or alternatively

$$\sum_{i=1}^N a(\varphi_i \boldsymbol{\epsilon}, \varphi_j \boldsymbol{\epsilon})(\mathbf{x}_i, \boldsymbol{\epsilon}) = \langle \mathbf{f}, \varphi_j \boldsymbol{\epsilon} \rangle_{\Omega} \quad j = \overline{1, N} .$$

This formulae can be rewritten as a system of linear algebraic equations (SLAE) as follows:

$$AX = F. \quad (6.5)$$

Here $A_{ij} = a(\mathbf{e}\varphi_i, \mathbf{e}\varphi_j)$, $F_i = \langle \varphi_j, \mathbf{f}_i \rangle$ with \mathbf{f}_i meaning the value of \mathbf{f} at a node i , and $X_i = \mathbf{x}_i$. Note that the matrix in (6.5) is positive definite. Denote it as \mathfrak{A} and consider the following expression:

$$\mathbf{x}\mathfrak{A}\mathbf{x} = a\left(\sum_i \mathbf{x}_i\varphi_i, \sum_j \mathbf{x}_j\varphi_j\right) = a(\mathbf{u}_h, \mathbf{u}_h) \geq \gamma\|\mathbf{u}_h\|_2^2,$$

where γ is some positive constant. We have just described the basic points of the Ritz-Galerkin method for solving positive definite problem (6.3). Before starting the discussion of the FEM itself, some important properties should be clarified.

Theorem 6.1.2 (Stability). *Regardless of the chosen subspace V^h , the solution of (6.3) satisfies the condition:*

$$\|\mathbf{u}_h\|_2 \leq \gamma^{-1}\|\mathbf{f}\|_2,$$

where γ is some positive constant.

Proof. Suppose \mathbf{u}_h is a solution of (6.3), and let $\mathbf{v} = \mathbf{u}_h$. Then

$$\gamma\|\mathbf{u}_h\|_2^2 \leq a(\mathbf{u}_h, \mathbf{u}_h) = \langle \mathbf{f}, \mathbf{u}_h \rangle_\Omega \leq \|\mathbf{f}\|_2\|\mathbf{u}_h\|_2.$$

□

Lemma 6.1.3 (Céa's lemma [45]). *Let H be a Hilbert space equipped with a norm $\|\cdot\|$. Let $a(\mathbf{v}, \mathbf{v})$ be a H -elliptic bilinear form (see. Chap. 10.2) so that for two non-negative constants C and γ the following inequalities are valid:*

$$a(\mathbf{u}, \mathbf{v}) \leq C\|\mathbf{u}\| \cdot \|\mathbf{v}\| \quad \forall \mathbf{u}, \mathbf{v} \in H,$$

$$a(\mathbf{u}, \mathbf{u}) \geq \gamma\|\mathbf{u}\|^2 \quad \forall \mathbf{u} \in H.$$

Let $V^h \subset H$ be a finite-dimensional subspace, and let $\mathbf{u}_h \in V^h$ and $\mathbf{u} \in H$ are the numerical and exact solutions of variational problem (6.1), respectively. Then

$$\|\mathbf{u} - \mathbf{u}_h\| \leq \frac{C}{\gamma} \inf_{\mathbf{v}_h \in V^h} \|\mathbf{u} - \mathbf{v}_h\| \quad \forall \mathbf{v}_h \in V^h.$$

6.1.2 Conventional Finite Element Method

The FEM is a particular case of the Ritz-Galerkin method. Let us illustrate the major principles of the FEM considering the problem in a rectangular domain Ω which is exposed to a regular quadratization.

Domain segmentation. Define in Ω a square grid Q^h with spatial step h . Let a pair of indexes (i, j) corresponds to the grid node of Q^h . Associate with the symbol Q^h the segmentation of Ω on a series of square sub-domains Ω_{ij} with horizontal scale h so that the number of each sub-domain coincides with the number of the grid node located in the left-bottom corner of Ω_{ij} (grey element in the left panel of Fig. 6.1 represents such a sub-domain). If so, $\Omega = \bigcup_{i,j=1}^{N \times M} \Omega_{ij}$ and $\Omega_{ij} \cap \Omega_{kt} = \emptyset$ is valid for all $(i, j) \neq (k, t)$. Define also a domain O_{lm} with the centre at the node $(l, m) = (i + 1, j + 1)$ as a combination of four neighbouring sub-domains Ω_{ij} : $O_{lm} = \bigcup_{i,j}^{i+1,j+1} \Omega_{ij}$. An example of such a domain is shown in the left panel of Fig. 6.1 in green (the nodes (l, m) corresponding to O_{lm} are presented by red circles). Note that the nodes (l, m) are internal nodes of the grid Q^h . They are located *inside* the domain Ω and do not lie at the boundary $\partial\Omega$. We allocate a pair of indexes (r, s) for the same nodes.

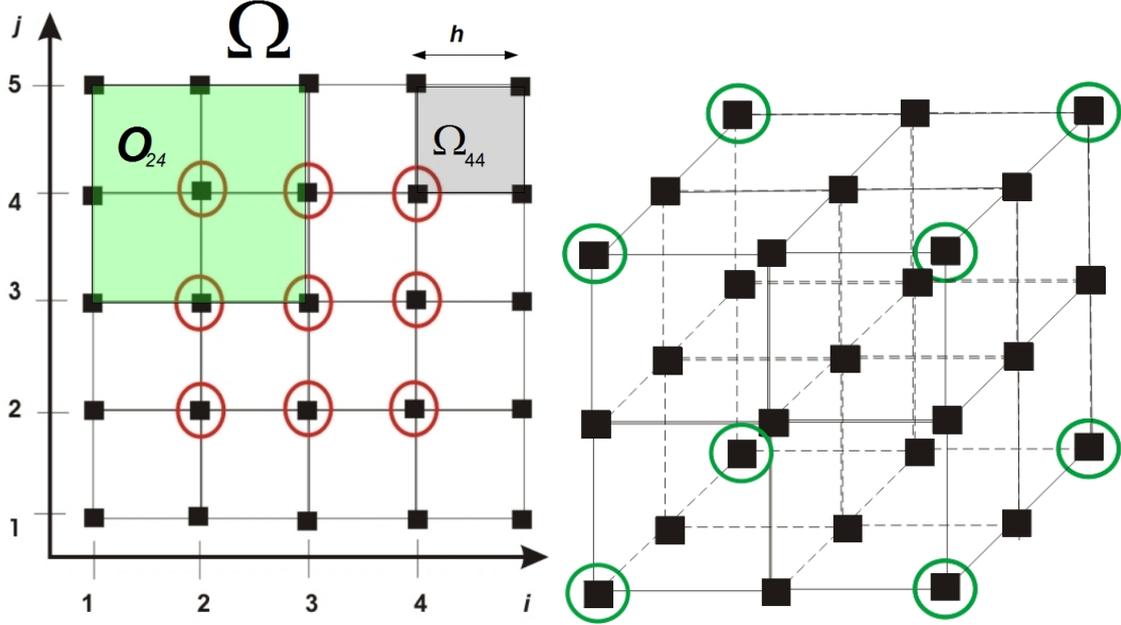


Figure 6.1: **Left:** The discretization of Ω on a square grid Q^h : the black squares correspond to the nodes numbered by indexes (i, j) ; the red circles correspond to the nodes numbered by indexes (l, m) . **Right:** A sketch of quadratization of a rectangular area for 3D Taylor-Hood FEM: the velocity basis function is defined in all nodes (squares) while the Lagrange variable basis function is defined only in odd-numbered nodes (circles).

Semi-orthogonal basis functions. Let us define a class of semi-orthogonal basis functions $\varphi_{lm}(x_{ij})$ such that they have a compact support only on O_{lm} , and, in addition, $\max\{\varphi_{lm}(x)\} = \varphi_{lm}(x_{lm}) = 1$. Then for $\forall \varphi_{lm}(x)$ the following conditions are valid:

$$\langle \varphi_{lm}, \varphi_{rs} \rangle_{\Omega} = \begin{cases} C_1, & \text{if } l = r \text{ and } m = s \\ C_2, & \text{if } |l - r| = 1 \text{ and } m = s \text{ or } |m - s| = 1 \text{ and } l = r \\ C_3, & \text{if } |l - r| = 1 \text{ and } |m - s| = 1 \\ 0, & \text{if } |l - r| > 1 \text{ or } |m - s| > 1 \end{cases} \quad (6.6)$$

in which $C_1 > C_2 > C_3$. The values of constants C_1, C_2 and C_3 depend on a concrete form of φ_{lm} .

Discretization. Similar to what was done in the previous chapter, let us express \mathbf{u}_h in terms of the basis functions, i.e.

$$\mathbf{u}_h = \sum_{l,m=1}^{N,M} \mathbf{u}_{lm} \varphi_{lm} \quad (6.7)$$

and substitute this series into bilinear form (6.1). This procedure leads to a system of linear equations similar to (6.5).

Boundary conditions. The FEM can be applied to problems with both essential boundary conditions and natural boundary conditions. If the problem implies the natural boundary conditions, then computations are conducted in the whole domain Ω including the nodes at the boundary $\partial\Omega$. In the opposite case the boundary should be excluded from the analysis, and the computations should be performed only in the inner part of the domain Ω which does not contain boundary nodes.

Summary. Note that the described above method can be extended to a general 3D case in which the shape of the area Ω not necessary must be rectangular. Moreover, the form of the basis elements should not necessary be square, and the grid can be irregular, as well. All mathematical reasoning for these extended cases remain the same as given above.

6.1.3 Mixed Finite Element Method. The Discretization of the First Step

3D case. The discretization similar to that presented in Sec. 4.2.1 was discussed in [72]. In this section we just repeat concisely some major points of this analysis and accommodate them to the considered here case. As it was shown in Sec. 4.2.1, the projection of \mathbf{d} on a set of divergent-free functions can be presented as a following constrained minimization problem:

$$\min \|\mathbf{d} - \mathbf{v}\|_2^2 \quad \text{s.t.} \quad \text{div}(\mathbf{v}) = 0 . \quad (6.8)$$

According to (4.7a) and (4.7b) its weak formulation can be presented as follows

$$\begin{aligned} \langle \boldsymbol{\eta}, \mathbf{v} - \mathbf{d} \rangle_{\Omega} + \langle \boldsymbol{\eta}, \mathbf{grad}(\psi) \rangle_{\Omega} &= 0 , \\ \langle \boldsymbol{\eta}, \text{div}(\mathbf{v}) \rangle_{\Omega} &= 0 , \end{aligned}$$

or, alternatively,

$$\begin{aligned} \langle \boldsymbol{\eta}, \mathbf{v} \rangle_{\Omega} + \langle \boldsymbol{\eta}, \mathbf{grad}(\psi) \rangle_{\Omega} &= \langle \boldsymbol{\eta}, \mathbf{d} \rangle_{\Omega} , \\ \langle \boldsymbol{\eta}, \text{div}(\mathbf{v}) \rangle_{\Omega} &= 0 , \end{aligned} \quad (6.10)$$

where $\boldsymbol{\eta}$ is an arbitrary function.

Similar to the previous section let's define the basis functions for \mathbf{v} and ψ . According to [13], the numerical solution of discretized problem (6.10) is stable only if the basis functions for the Lagrange multiplier and \mathbf{v} are polynomial functions of different order. Define two grids, Q^h and Q^{2h} , consisting of quadratic meshes with side sizes h and $2h$, respectively. For the discretization of the velocity the second-order polynomial defined in the Q^h grids nodes are taken, i.e.

$$\varphi_v(x, y, z) = a_v x^2 + b_v y^2 + c_v z^2 + d_v xy + e_v xz + f_v yz + g_v x + h_v y + k_v z + l_v . \quad (6.11)$$

For the Lagrange multiplier a first order polynomial defined in nodes of Q^{2h} is used, viz.

$$\zeta(x, y, z) = a_{\psi} x + b_{\psi} y + c_{\psi} z + d_{\psi} . \quad (6.12)$$

The constants $a_v, b_v, c_v, d_v, e_v, f_v, g_v, h_v, k_v, l_v$ and $a_{\psi}, b_{\psi}, c_{\psi}, d_{\psi}$ are the polynomial coefficients. A sketch given in the right panel of Fig. 6.1 illustrates an arrangement of the individual basis functions.

After substitution of $\mathbf{v} = \mathbf{V}_{lmn} \psi_{lmn}$ and the Lagrange variable $\psi = \Psi_{rst} \psi_{rst}$ into the corresponding expression for the weak formulation (6.10) we arrived at two expressions:

$$\mathbf{V}_{lmn} \langle \varphi_{ijk}, \varphi_{lmn} \rangle_{\Omega} + \Psi_{rst} \langle \varphi_{ijk}, \mathbf{grad}(\zeta_{rst}) \rangle_{\Omega} = \langle \varphi_{ijk}, \mathbf{d}(\mathbf{x}) \rangle_{\Omega} ,$$

$$\langle \zeta_{rst}, \text{div}(\mathbf{V}_{lmn} \varphi_{lmn}) \rangle_{\Omega} = 0 , \quad \forall \varphi_{lmn}, \varphi_{ijk} \in Q^h \quad \text{and} \quad \forall \zeta_{rst} \in Q^{2h} .$$

The variational task is simplified to a SLAE:

$$\begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} \mathbf{V} \\ \Psi \end{bmatrix} = \begin{bmatrix} \mathbf{d} \\ 0 \end{bmatrix} . \quad (6.13)$$

Here the blocks A , B^T and B are matrices composed by the elements $\langle \varphi_{ijk}, \varphi_{lmn} \rangle_\Omega$, $\langle \varphi_{lmn}, \mathbf{grad}(\zeta_{rst}) \rangle_\Omega$ and $\langle \zeta_{rst}, \mathit{div}(\varphi_{lmn}) \rangle_\Omega$, respectively, while \mathbf{V} and Ψ contain the unknown constants \mathbf{V}_{lmn} and Ψ_{rst} stacked in a column. Solving this SLAE using e.g CG method (see Sec. 6.2.1) one obtains the numerical solution of (4.7a)-(4.7b).

2D case. The mathematical reasoning used here for the discretization of the first step of the VTE-based reconstructive approach in a 2D case is exactly the same as given above for the 3D case.

6.1.4 Discretization of the Third and Fourth Steps.

Discretization of the third step in 3D case. The third step is given by the following minimization problem:

$$\min_{\boldsymbol{\omega}} \left\{ \|\boldsymbol{\omega} - \boldsymbol{\omega}_g\|_2^2 + \alpha \left(\nu \|\nabla \times \boldsymbol{\omega}\|_2^2 + 2\langle e(\mathbf{v}_g), \boldsymbol{\omega} \rangle_\Omega \right) \right\}. \quad (6.14)$$

The corresponding weak form reads

$$\langle \boldsymbol{\eta}, \boldsymbol{\omega} \rangle_\Omega - \alpha \nu \langle \boldsymbol{\eta}, \Delta \boldsymbol{\omega} \rangle_\Omega = \langle \boldsymbol{\eta}, \mathbf{f}_3 \rangle_\Omega, \quad \mathbf{f}_3 = \boldsymbol{\omega}_g - \alpha e(\mathbf{v}_g), \quad (6.15)$$

where $\boldsymbol{\eta} \in C_0^\infty$ is arbitrary function. Repeating the procedure applied in Sec. (6.1.2), we define the unknown functions as follows: $\boldsymbol{\omega} = \mathbf{W}_{lmn} \varphi_{lmn}$ and $\boldsymbol{\eta} = \mathbf{E}_{ijk} \varphi_{ijk}$, where the basis functions φ were taken similiary to (6.11). Substituting $\boldsymbol{\omega}$ and $\boldsymbol{\eta}$ into (6.15) we obtain

$$\mathbf{W}_{lmn} \langle \varphi_{ijk}, \varphi_{lmn} \rangle_\Omega - \alpha \nu \mathbf{W}_{lmn} \langle \varphi_{ijk}, \Delta \varphi_{lmn} \rangle_\Omega = \langle \varphi_{lmn}, \mathbf{f}_3 \rangle_\Omega, \quad \text{where } \mathbf{f}_3 = \boldsymbol{\omega}_g - \alpha e(\mathbf{v}_g). \quad (6.16)$$

As it was done in Sec. 6.1.3, equation (6.16) is reduced to a SLAE:

$$AX = F.$$

Here the matrix A , column F , and the right hand side F consist of the elements $\langle \varphi_{ijk}, \varphi_{lmn} \rangle_\Omega$, \mathbf{W}_{lmn} and $\langle \varphi_{lmn}, \mathbf{f}_3 \rangle_\Omega$, respectively. Solving this system we obtain the numerical solution of (6.14).

Discretization of the fourth step. At the last step the following minimization procedure must be executed:

$$\min_{\mathbf{u}} \left\{ \|\mathbf{u} - \mathbf{v}_g\|_2^2 + \beta \|\nabla \times \mathbf{u} - \boldsymbol{\omega}\|_2^2 \right\}. \quad (6.17)$$

Its weak form reads:

$$\langle \boldsymbol{\eta}, \mathbf{u} \rangle_\Omega - \beta \langle \boldsymbol{\eta}, \Delta \mathbf{u} \rangle_\Omega = \langle \boldsymbol{\eta}, \mathbf{f}_4 \rangle_\Omega, \quad \text{where } \mathbf{f}_4 = \mathbf{v}_g + \beta \nabla \times \mathbf{v}_g, \quad (6.18)$$

where $\boldsymbol{\eta}$ is arbitrary function. Since equations (6.17) and (6.18) are similar to (6.14) and (6.15), for the discretization of the fourth step one has just to repeat the discretization method described in the previous paragraph.

2D case for steps 3 and 4. The mathematical procedure applied to discretize the third and fourth steps of the VTE reconstructive approach in a 2D case is exactly the same as given above for three dimensions, and can be easily simplified to the case of two independent variables.

6.1.5 Outline

As a summary of this section we should outline some remarkable advantages that gives the Finite Element Method for solving some mathematical problems:

- The direct numerical solution of variational problems requires application of complicated procedures which takes a lot of computation time. The FEM reduces variational problems to a system of linear equations which are simpler for computations and require less CPU time;
- Some numerical methods (like FDM) are sensitive to the shape of a boundary of the area in which the problem is computed. The FEM is independent on this restriction provided that a proper triangularization or quadratization has been chosen.

The reduction of the variational problem to a system of linear equations allows one to solve them using conventional linear solvers which are briefly outlined below.

6.2 Linear Solvers

6.2.1 Conjugate Gradient Method

Problem formulation. The conjugate gradient method (CGM) is aimed to solve a SLAE. The typical problem for CGM sounds: given a SLAE

$$A\mathbf{x} = \mathbf{b} , \tag{6.19}$$

where matrix A has a full rank. The task is to find a solution \mathbf{x} .

Direct conjugate gradient method. It is essential for the conjugate gradient method that the matrix A in (6.19) is symmetric, positive, definite and real (to be specific, let's assume it has size $n \times n$). Another assumption is that \mathbf{x}_s is a solution of (6.19). We define two nonzero conjugate vectors \mathbf{u} and \mathbf{v} (conjugate with respect to A) if they satisfy the following condition:

$$\mathbf{u}^T A\mathbf{v} = 0.$$

Using the properties of A (i.e. it is a symmetric and positive definite matrix), we can rewrite this expression in terms of scalar product as follows:

$$\langle A\mathbf{u}, \mathbf{v} \rangle = \langle \mathbf{u}, A\mathbf{v} \rangle = \mathbf{u}^T A\mathbf{v} = 0.$$

This formula, in fact, represents the orthogonality of two vectors. In other words, two vectors are conjugate if they are orthogonal with respect to this product. If so, one can

construct a basis containing a sequence of conjugate vectors \mathbf{e}_i . As a result, the solution \mathbf{x}_s can be presented as an expansion:

$$\mathbf{x}_s = \sum_{i=1}^n \alpha_i \mathbf{e}_i. \quad (6.20)$$

Substituting this series in (6.19) we can derive an expression for the coefficients α_i , i.e

$$\mathbf{b} = A\mathbf{x}_s = \sum_{i=1}^n \alpha_i A\mathbf{e}_i. \quad (6.21)$$

Multiplying the latter by \mathbf{e}_j and taking into account that $\mathbf{e}_i^T A\mathbf{e}_j \neq 0$ only if $i = j$, we finally obtain:

$$\mathbf{e}_i \mathbf{b} = \alpha_i \mathbf{e}_i^T A\mathbf{e}_i. \quad (6.22)$$

Thus, we arrived at the following equation for the calculation of the coefficients α_i :

$$\alpha_i = \frac{\mathbf{e}_i^T \mathbf{b}}{\mathbf{e}_i^T A\mathbf{e}_i}. \quad (6.23)$$

Summarising, one can formulate an algorithm for solving equation (6.19). It consists of two steps: at first step a sequence of conjugate vectors should be found, and after that the coefficients α_i are calculated.

The computational algorithm for conjugate gradient method. In practice the direct conjugate method is not used on a regular basis because it is time consuming in calculations especially for large systems. Therefore, in many practical cases instead of direct computations it is more convenient to use an alternative version of the method which is targeted to find an approximate solutions of (6.19). This method starts with the initial guess that $\mathbf{x} = \mathbf{x}_0 = 0$ in (6.19); it is updated at each iteration until \mathbf{x}_a became close enough to \mathbf{x}_s . The difference between the exact solution \mathbf{x}_s and approximated solution \mathbf{x}_a is evaluated with a quadratic function:

$$G = \frac{1}{2} \mathbf{x}^T A\mathbf{x} - \mathbf{b}^T \mathbf{x}.$$

This function is getting smaller as \mathbf{x}_a approaches \mathbf{x}_s , and reaches its minimum at $\mathbf{x}_a = \mathbf{x}_s$. This circumstance suggests that the best choice for the initialization of calculations is to take the basis vector \mathbf{e}_1 as a gradient of G at $\mathbf{x} = \mathbf{x}_0$, which equals $A\mathbf{x}_0 - \mathbf{b}$. Since $\mathbf{x}_0 = 0$, this means that we take $\mathbf{e}_1 = -\mathbf{b}$. The other vectors in the basis will be conjugate to the gradient.

Let $\mathbf{r}_i = \mathbf{b} - A\mathbf{x}^k$ is the residual vector at the i -th iteration, and let \mathbf{e}_i is the uncomplete basis of conjugate vectors. If one subtracts from \mathbf{r}_i the expansion of \mathbf{r}_i in \mathbf{e}_i , the resultant vector will be conjugate to \mathbf{e}_i (provided it is nonzero), so one can write

$$\mathbf{e}_{i+1} = \mathbf{r}_i - \sum_{k \leq i} \frac{\mathbf{e}_k^T A\mathbf{r}_i}{\mathbf{e}_k^T A\mathbf{e}_k} \mathbf{e}_k. \quad (6.24)$$

After all simplifications the iterative conjugate gradient method appears in a compact algorithmic form as follows:

Conjugate Gradient Method

 $\mathbf{r}_0 := \mathbf{b} - A\mathbf{x}_0$
 $\mathbf{e}_0 := \mathbf{r}_0$
 $i := 0$
while $i < N$
 $\alpha_i := \frac{\mathbf{r}_i^T \mathbf{r}_i}{\mathbf{r}_i^T A \mathbf{r}_i}$
 $\mathbf{x}_{i+1} := \mathbf{x}_i + \alpha_i \mathbf{e}_i$
 $\mathbf{r}_{i+1} := \mathbf{r}_i - \alpha_i A \mathbf{e}_i$
If \mathbf{r}_{i+1} is too small **then exit the loop**
 $\beta_i := \frac{\mathbf{r}_{i+1}^T \mathbf{r}_{i+1}}{\mathbf{r}_i^T \mathbf{r}_i}$
 $\mathbf{e}_{i+1} := \mathbf{r}_{i+1} + \beta_i \mathbf{e}_i$
 $i := i+1$
end while

6.2.2 Preconditioning

After the optimization considered above the task has been reduced to the solution of a SLAE. At this step we encounter another calculational problem: how to solve this system with minimum mathematical operations? A valuable tool for that could be a *preconditioning* procedure which is discussed below.

The term "preconditioning" refers to the transformation of a linear system $\mathcal{A}u = b$ into another system which iterative solution has better convergence rate. A preconditioner is a matrix P that allows such a transformation. Basically, preconditioning improves the spectral properties of a system, and this circumstance results in the increase of the convergence rate of a solver.

In this section we focus only on a block-diagonal preconditioner which is relevant to SLAE. For any additional information on preconditioning we refer the reader to the original source [10].

Problem formulation. Given a system of linear equations

$$\mathcal{A}u = b, \tag{6.25}$$

where \mathcal{A} is a non-singular matrix of the following form:

$$\mathcal{A} = \begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix}. \tag{6.26}$$

The task is to find a preconditioner matrix P for (6.25) which after its multiplication by equation (6.25) increases substantially the rate of convergence of the iterative algorithm.

Block diagonal preconditioners. An evident choice for an invertible saddle point matrix \mathcal{A} is a block diagonal preconditioning. In such a case the preconditioner is given by the matrix

$$P = \begin{bmatrix} A & 0 \\ 0 & -S \end{bmatrix}, \quad (6.27)$$

in which $S = -BA^{-1}B^T$ is a Schur complement. Let I is an identity matrix. Multiplying \mathcal{A} by P^{-1} one can obtain the following matrix

$$P^{-1}\mathcal{A} = \mathcal{M} = \begin{bmatrix} I & A^{-1}B^T \\ -S^{-1}B & 0 \end{bmatrix}. \quad (6.28)$$

Since we assumed that \mathcal{A} is invertible, the matrix \mathcal{M} is nonsingular and, as pointed out in [52] and [62], satisfies the following characteristic equation

$$(\mathcal{M} - I) \left(\mathcal{M} - \frac{1}{2} (1 + \sqrt{5}) I \right) \left(\mathcal{M} - \frac{1}{2} (1 - \sqrt{5}) I \right) = 0. \quad (6.29)$$

Hence, matrix \mathcal{M} can be transformed to a diagonal form which has only three distinct eigenvalues, namely 1 , $\frac{1}{2}(1 - \sqrt{5})$, and $\frac{1}{2}(1 + \sqrt{5})$. As a consequence, it has only three linear independent eigenvectors corresponding to these eigenvalues. Therefore, for each initial residual r_0 the standard solving procedure like the CGM, being applied to the preconditioned system with matrix \mathcal{M} , will terminate after at most 3 steps.

At first glance this fact looks pretty promising for improving the algorithm efficiency. However, according to [10] the formation of the preconditioned system $Mu = P^{-1}b$ out of the given saddle point system $\mathcal{A}u = b$ using the block diagonal preconditioner (6.27) is a computationally expensive, as a direct computing of the inverse matrix \mathcal{A}^{-1} . Thus, it is reasonable to replace the exact preconditioner (6.27) by its approximation

$$\hat{P} = \begin{bmatrix} \hat{A} & 0 \\ 0 & \hat{S} \end{bmatrix}, \quad (6.30)$$

where $\hat{A} = \text{diag}(A)$, and $\hat{S} = B\hat{A}^{-1}B^T$ are approximations of A and S , respectively. So, multiplying both sides of (6.25) by \hat{P}^{-1} we obtain a system $\hat{P}^{-1}\mathcal{A} = \hat{P}^{-1}$ which is similar to (6.25) but according to [10] has higher convergence rate.

Note that for the high convergence rate of $\hat{P}^{-1}\mathcal{A} = \hat{P}^{-1}$ the matrix A should be diagonal dominant, which is the case of the solenoidal projection at step one (Fig. 6.2). Tables 6.1-6.2 demonstrate the convergence rates of CGM with and without preconditioning (special version of CGM with preconditioning will be discussed in the next chapter). As one can see from these tables, the preconditioning substantially increases the average convergence rate (more than 20 times).

6 Discretization

Table 6.1: The residuals and the corresponding number of iterations applied in CGM for computation of (6.13) without preconditioning.

residual	9.710^{-2}	$9.5 \cdot 10^{-3}$	$1 \cdot 10^{-4}$	$1 \cdot 10^{-5}$	$1 \cdot 10^{-6}$	$9.9 \cdot 10^{-7}$
iterations	38	132	288	471	631	754

Table 6.2: The residuals and the corresponding number of iterations applied in CGM for computation of (6.13) with preconditioning.

residual	8.510^{-2}	$0.5 \cdot 10^{-3}$	$0.5 \cdot 10^{-4}$	$0.51 \cdot 10^{-5}$	$0.19 \cdot 10^{-6}$	$0.37 \cdot 10^{-7}$
iterations	4	8	11	14	18	21

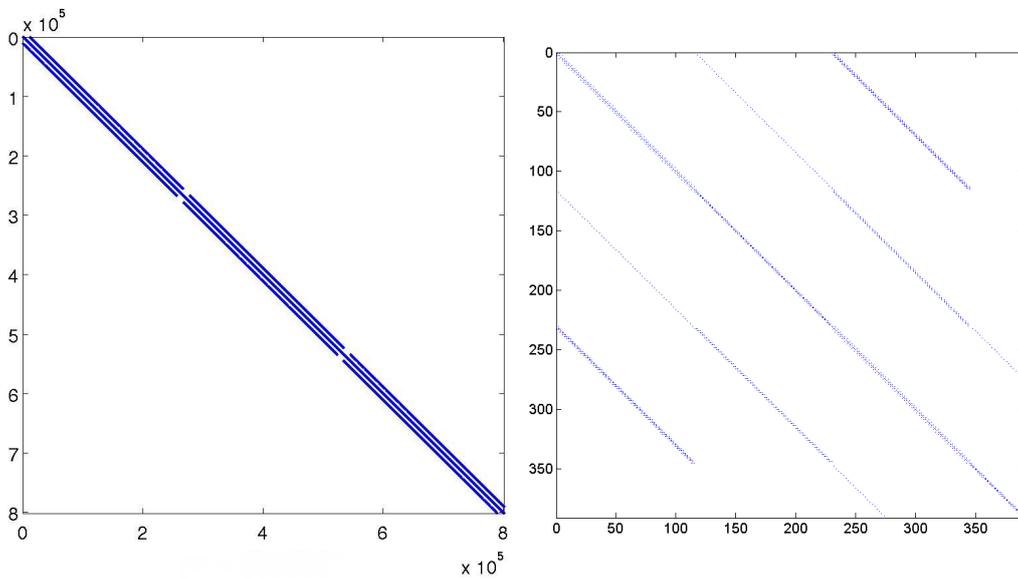


Figure 6.2: **Left panel:** The distribution of nonzero elements (blue bullets) in matrix A .
Right panel: The zoomed part of the middle diagonal in A .

6.2.3 Pre-Conjugate Gradient Method

The results obtained in previous subsection can be directly used in the conjugate method. This combination of the conjugate gradient method with preconditioner gives rise to a new method, namely to a pre-conjugate gradient method. Since it is quite similar to the conjugate gradient method, we reproduce here only its algorithm:

Preconditioned Conjugate Gradient Method

 $\mathbf{r}_0 := \mathbf{b} - A\mathbf{x}_0$
 $\mathbf{z}_0 := P^{-1}\mathbf{r}_0$
 $\mathbf{e}_0 := \mathbf{z}_0$
 $i := 0$
while $i < N$
 $\alpha_i := \frac{\mathbf{r}_i^T \mathbf{z}_i}{\mathbf{r}_i^T A \mathbf{r}_i}$
 $\mathbf{x}_{i+1} := \mathbf{x}_i + \alpha \mathbf{e}_i$
 $\mathbf{r}_{i+1} := \mathbf{r}_i - \alpha A \mathbf{e}_i$
If \mathbf{r}_{i+1} is too small **then exit the loop**
 $\mathbf{z}_{i+1} := P^{-1}\mathbf{r}_{i+1}$
 $\beta_i := \frac{\mathbf{r}_{i+1}^T \mathbf{z}_{i+1}}{\mathbf{r}_i^T \mathbf{z}_i}$
 $\mathbf{e}_{i+1} := \mathbf{z}_{i+1} + \beta_i \mathbf{e}_i$
 $k := k+1$
end while

6.3 Finite Differences

In this section we discuss the numerical implementation of the VTE-restoration algorithm using the finite-difference method (FDM) which looks relatively simpler in comparison with the FEM. Several alternative numerical schemes are considered here for the numerical discretization of the nonlinear term $\mathbf{e}(\mathbf{v}_g)$ of equation (4.12).

Introduction. The FDM is used on a regular basis for the numerical analysis of a wide variety of initial and boundary value problems. The starting point of the FDM is an introduction of a grid function u_i instead of an analytical function $u(x)$. It is defined on a grid in nodes i ($i = 1, 2, 3, \dots, N$). Once introduced, the grid function u_i allows to represent all derivatives appeared in ODEs or PDEs in terms of their finite differences. The latter, in fact, are derived from the truncated Taylor series presenting the original function u at grid points $x + h$ or $x - h$ through its value and values of its derivatives at point x . Here h is a grid step, and u_{i-1} , u_i , u_{i+1} are the values of u in nodes $i - 1$, i , $i + 1$ of the grid, respectively. Accuracy of the FDM (i.e. first-, second- or third-order approximation) depends on the fact how many terms of the truncated Taylor series are retained in the equation. Substitution of all derivatives by their finite-differences transforms ODEs and PDEs to a SLAE, which can be solved using the methods of linear algebra. As an example, we present below a list of finite differences of the first derivative most frequently used in one-dimensional problems:

- Upstream difference: $\frac{du}{dx} = \frac{u_{i+1} - u_i}{h}$, first-order accuracy, $O(h)$;
- Downstream difference: $\frac{du}{dx} = \frac{u_i - u_{i-1}}{h}$, first-order accuracy, $O(h)$;
- Central difference: $\frac{du}{dx} = \frac{u_{i+1} - u_{i-1}}{2h}$, second-order accuracy, $O(h^2)$.

The discretization of nonlinear terms. To be more specific, let's discuss an application of the FDM to the problem considered in this work. An alternating direction implicit numerical method (ADIM, see [51] for the details) was used for the discretization of the restoration algorithm. The advective terms in the VTE equation for 2D and 3D case read: $\omega(\nabla \cdot \mathbf{v})$ and $\mathbf{v}(\nabla \cdot \omega) + \omega(\nabla \cdot \mathbf{v})$, respectively. The reason for choosing of such a scheme is that according to [51] it is stable, robust and it damps high frequency noise more efficiently in comparison with high-order accurate difference schemes. To prove this fact the nonlinear term $\mathbf{e}(\mathbf{v}_g)$ was computed twice: using ADIM scheme in the first set of runs, and the fourth-order accurate central difference (FOACD) scheme [55] in the second one. The same vector field $\mathbf{v}_g = (1, \sin(x))$ with different noise containment was taken for the analysis in both cases. The results of computations were compared against the ground truth $\mathbf{e}(\mathbf{v}_g) = -\sin(x)$.

The graphs given in Fig. 6.3 represent the dependence of the noise-to-signal ratio $SNR(\mathbf{e}(\mathbf{v}_g))$ in the computed $\mathbf{e}(\mathbf{v}_g)$ output field to the original noise-to-signal ratio $SNR(\mathbf{v}_g)$

in the input vector field \mathbf{v}_g . The results of ADIM scheme is presented in black, and green colour corresponds to the FOACD scheme. Scrutiny of these graphs shows that in the close proximity to zero the $SNR(\mathbf{e}(\mathbf{v}_g))$ contains only computational errors. As a result, the green line lies below the black one, which is an expected consequence of the fact that the high-order FOACD scheme is more accurate than ADIM. However, with the increase of the noise-to-signal ratio in \mathbf{v}_g the noise level in $\mathbf{e}(\mathbf{v}_g)$ computed with the FOACD scheme increases more remarkably than in $\mathbf{e}(\mathbf{v}_g)$ computed using ADIM scheme. It is seen that the green line intersects the black line at $SNR(\mathbf{v}_g) = 0.06$. The further increase of $SNR(\mathbf{v}_g)$ leads to the situation when the error produced by ADIM scheme is substantially smaller than the error produced by FOACD scheme. Thus, the main conclusion from this experiment is that ADIM scheme is less sensitive to the noise than FOACD schemes.

The discretization of the 2D nonlinear term reads

$$(\mathbf{v}_{i,j} \cdot \nabla)\omega_{i,j} = \frac{|u_{i,j}| + u_{i,j}}{2} \frac{\omega_{i+1,j} - \omega_{i,j}}{h_x} + \frac{u_{i,j} - |u_{i,j}|}{2} \frac{\omega_{i,j} - \omega_{i-1,j}}{h_x} + \frac{|v_{i,j}| + v_{i,j}}{2} \frac{\omega_{i,j+1} - \omega_{i,j}}{h_y} + \frac{v_{i,j} - |v_{i,j}|}{2} \frac{\omega_{i,j} - \omega_{i,j-1}}{h_y}. \quad (6.31)$$

Hereafter h_x , h_y , h_z denote the spatial grid steps in x , y and z -directions, respectively. Subscripts i , j and k identify an appropriate grid node. Note that all derivatives of ω in (6.31) should satisfy zero fluxes boundary conditions at $\partial\Omega$ (for explanations see Sec. 4.2.3).

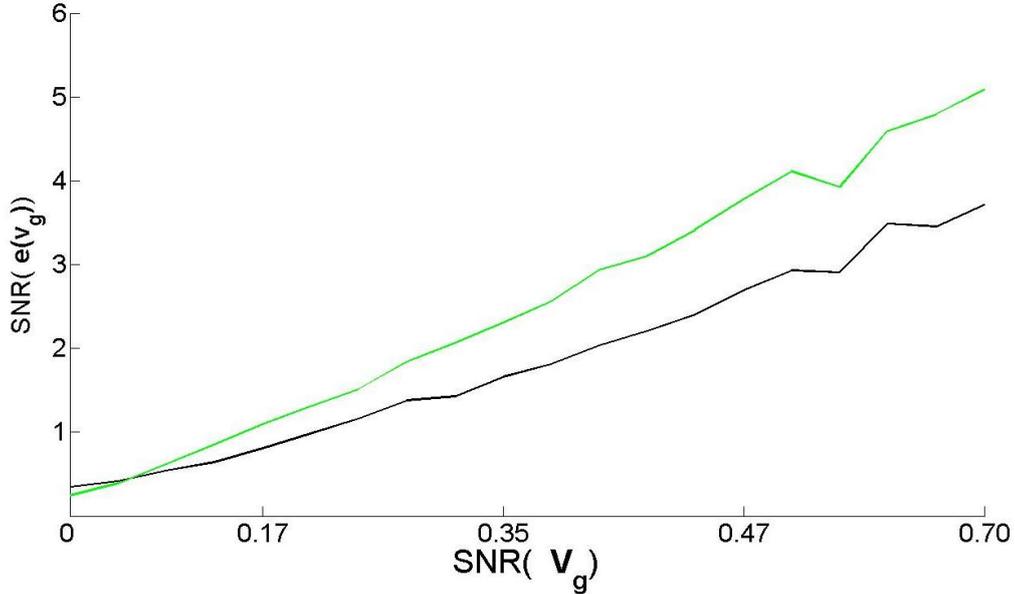


Figure 6.3: The dependence of the noise level containing in $\mathbf{e}(\mathbf{v}_g)$ computed with FOACD (green) and ADIM (black) schemes on the noise level containing in \mathbf{v}_g .

The nonlinear term in 3D case consists of two parts: $\mathbf{v}(\nabla \cdot \omega)$ and $\omega(\nabla \cdot \mathbf{v})$; their

discretization is similar to that given above. Let $\boldsymbol{\omega}_{i,j,k} = (\omega_{xi,j,k}, \omega_{yi,j,k}, \omega_{zi,j,k})^T$, then the discretization of the first part is as follows:

$$\begin{aligned} (\mathbf{v}_{i,j,k} \cdot \nabla) \boldsymbol{\omega}_{i,j,k} = & \frac{|u_{i,j,k}| + u_{i,j,k}}{2} \frac{\boldsymbol{\omega}_{i+1,j,k} - \boldsymbol{\omega}_{i,j,k}}{h_x} + \frac{u_{i,j,k} - |u_{i,j,k}|}{2} \frac{\boldsymbol{\omega}_{i,j,k} - \boldsymbol{\omega}_{i-1,j,k}}{h_x} + \\ & \frac{|v_{i,j,k}| + v_{i,j,k}}{2} \frac{\boldsymbol{\omega}_{i,j+1,k} - \boldsymbol{\omega}_{i,j,k}}{h_y} + \frac{v_{i,j,k} - |v_{i,j,k}|}{2} \frac{\boldsymbol{\omega}_{i,j,k} - \boldsymbol{\omega}_{i,j-1,k}}{h_y} + \\ & \frac{|w_{i,j,k}| + w_{i,j,k}}{2} \frac{\boldsymbol{\omega}_{i,j,k+1} - \boldsymbol{\omega}_{i,j,k}}{h_z} + \frac{w_{i,j,k} - |w_{i,j,k}|}{2} \frac{\boldsymbol{\omega}_{i,j,k} - \boldsymbol{\omega}_{i,j,k-1}}{h_z}. \end{aligned}$$

The finite-difference representation of the second part of the advection operator reads:

$$\begin{aligned} (\boldsymbol{\omega}_{i,j,k} \cdot \nabla) \mathbf{v}_{i,j,k} = & \frac{|\omega_{xi,j,k}| + \omega_{xi,j,k}}{2} \frac{\mathbf{v}_{i+1,j,k} - \mathbf{v}_{i,j,k}}{h_x} + \frac{\omega_{xi,j,k} - |\omega_{xi,j,k}|}{2} \frac{\mathbf{v}_{i,j,k} - \mathbf{v}_{i-1,j,k}}{h_x} + \\ & \frac{|\omega_{yi,j,k}| + \omega_{yi,j,k}}{2} \frac{\mathbf{v}_{i,j+1,k} - \mathbf{v}_{i,j,k}}{h_y} + \frac{\omega_{yi,j,k} - |\omega_{yi,j,k}|}{2} \frac{\mathbf{v}_{i,j,k} - \mathbf{v}_{i,j-1,k}}{h_y} + \\ & \frac{|\omega_{zi,j,k}| + \omega_{zi,j,k}}{2} \frac{\mathbf{v}_{i,j,k+1} - \mathbf{v}_{i,j,k}}{h_z} + \frac{\omega_{zi,j,k} - |\omega_{zi,j,k}|}{2} \frac{\mathbf{v}_{i,j,k} - \mathbf{v}_{i,j,k-1}}{h_z}. \end{aligned}$$

In the majority of nonstationary hydrodynamic problems all functions depend on time, as well. Taking into account this fact, one extra index, i.e. superscript, should be allocated for the temporal variable. In other words, all variables in 2D and 3D nonlinear terms should be attributed to a temporal step t . In doing so, in a 2D case we should write:

$$\mathbf{e}_{i,j}^t = \frac{\omega_{i,j}^{t+1} - \omega_{i,j}^t}{h_t} + (\mathbf{v}_{i,j}^t \cdot \nabla) \omega_{i,j}^t. \quad (6.32)$$

For the 3D case this formula reads:

$$\mathbf{e}_{i,j,k}^t = \frac{\omega_{i,j,k}^{t+1} - \omega_{i,j,k}^t}{h_t} + (\mathbf{v}_{i,j,k}^t \cdot \nabla) \omega_{i,j,k}^t + (\boldsymbol{\omega}_{i,j,k}^t \cdot \nabla) \mathbf{v}_{i,j,k}^t. \quad (6.33)$$

Here h_t denotes a temporal step, and superscript t identifies an appropriate spatial level in a space-time grid.

The discretization of the Laplacian. For the computation of eq.(4.13) and (4.18) (see Section 4.2.3 and Section 4.2.4) the Laplacian has to be presented in terms of finite differences. For the same reason of damping of high frequency noise mentioned in the previous paragraph, we will consider the discretization of the Laplacian using the second order central finite differences. As an example let's consider first the discretization of a single second derivative $\frac{\partial^2 q(x, y)}{\partial x^2}$ (i.e. 1D Laplacian). It can be presented as follows

$$\Delta_i q = \frac{\frac{dq_{i+1,j}}{dx} - \frac{dq_{i,j}}{dx}}{h_x} = \frac{\frac{q_{i+1,j} - q_{i,j}}{h_x} - \frac{q_{i,j} - q_{i-1,j}}{h_x}}{h_x} = \frac{q_{i+1,j} - 2q_{i,j} + q_{i-1,j}}{h_x^2}. \quad (6.34)$$

This central-difference differential operator has the second-order accuracy. Left panel in Fig. 6.4 is a sketch illustrated this discretization. The central node i is marked by a yellow circle, the white circles represent the neighbouring nodes. The numbers appeared in circles are the weights which are multiplied by the corresponding values q_i . Similar to 1D-Laplacian, one can discretize the 2D and 3D Laplacians, as well. Their schematic diagrams are given in the middle and right panels of Fig. 6.4, respectively.

The direct step-by-step computation of (4.13) and (4.18) is time consuming. A more preferable way regarding the numerical efficiency of the algorithm is to rewrite these equations in a form of SLAE and solve it using a linear solver (e.g. CG-method). This procedure runs as follows: at first the discrete Laplacian should be presented in a matrix form. Thereto the weighted coefficients are written in matrix rows in such an order that a multiplication of this matrix by some column gives the Laplacian of this column. If \mathcal{D} is such a matrix, then equations (4.13) and (4.18) can be presented in terms of SLAE as follows

$$(I - \nu\mathcal{D})X = F,$$

where I is the identity matrix, X is a column of unknowns, and F consists of the stacked values of $\omega_g - \mathbf{e}$ in case of (4.18), and $\mathbf{v}_g - \nabla \times \omega$ in case of (4.13). Finally, this system is solved using the CGM solver.

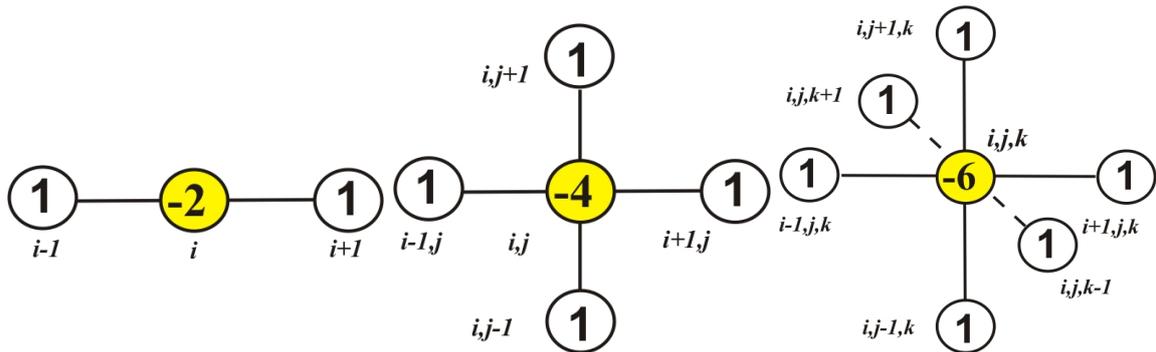


Figure 6.4: The sketches of the discretizations of the Laplacian Δ on 1D, 2D and 3D regular grids. The central elements are presented in yellow. **Left panel:** 1D case. **Middle panel:** 2D case. **Right panel:** 3D case.

6.4 The Algorithm for Total Variation

The mathematical background of the scalar-valued total variation denoising algorithm has been already discussed in Sec. 3.5.2. It is summarized briefly here again in order to complete the numerical procedure. The objectives of this method are as follows. Given some noisy input data d . The task is to compute u which is a piecewise smoothed part of

6 Discretization

d. The value d is computed as a minimum of a functional

$$J = \frac{1}{2} \|u - d\|_2^2 + \lambda \sup_{\boldsymbol{\xi}} \int_{\Omega} u(x) \operatorname{div}(\boldsymbol{\xi}(x)) dx, \quad \boldsymbol{\xi}(x) \in C_c^1(\Omega, \mathbb{R}^2), \quad \|\boldsymbol{\xi}(x)\|_2 \leq 1 \quad \forall x \in \Omega.$$

The numerical solution of this minimization problem is fulfilled by applying the algorithm presented below [17]:

The Algorithm for Total Variation Minimization

Choose $0 < \tau < 1/8$, $\mathbf{p}^0 = 0$ and set some integer number of iterations k

```
for n=1:k  
   $\mathbf{p}_{ij}^{n+1} = \frac{\mathbf{p}_{ij}^n + \tau(\nabla(\operatorname{div}(\mathbf{p}^n) - d/\lambda))_{ij}}{1 + \tau|\nabla(\operatorname{div}(\mathbf{p}^n) - d/\lambda)_{ij}|}$   
end
```

Compute $u = d - \lambda \operatorname{div}(\mathbf{p})$

7 Applications of the VTE Approach and its Comparison with other Methods

7.1 General Description

In this section we demonstrate high performance of the developed denoising procedure considering a number of practically relevant scenarios. The demonstration starts with the reconstruction of computer-simulated fluid flows. They are obtained using different numerical models and represent the initial ground truth. These fields are deliberately polluted at the first stage by adding some artificial noise. After that the polluted field is subjected to the denoising procedure, and the output field is compared against the initial ground truth. The advantage of such an analysis is that we really know the ground truth so that can objectively estimate the performance of the method by a comparison of the initial ground truth with the output denoised field.

These experiments with polluted computer-simulated flows represent typical situations which can occur in practice: the vector field is contaminated by the noise. In addition, some parts of the field can be missing. The only difference of this practical case from the methodological experiment dealing with 'synthetic' data sets is that in practice we do not know the real ground truth and cannot estimate the performance of the reconstruction procedure.

The second part of this chapter is devoted to the comparison of the developed procedure with other methods described in Sec. 3.5. And finally, the reconstruction of some real fluid flow patterns obtained in laboratory measurements is demonstrated at the end of this section.

All experiments with the computer-simulated fluid flows were conducted according to the following scheme:

- In the beginning of the procedure a numerical solution of the Navier-Stokes equations is taken as a ground truth;
- Noise is added to the numerical solution at the next stage. The algorithm is validated against different types of introduced noise, i.e. the white Gaussian noise, uniform or biased noise. Some experiments are conducted with sparse input fields and randomly located regions of missing or corrupted data;
- The obtained noisy vector field is used as an input data set for the algorithm which removes the noise and restores the vector fields;
- Finally, the result is compared with the ground truth, both qualitatively, and quantitatively.

The algorithm contains several adjustable parameters. Since the information on the input vector field is assumed to be unknown, by *default* they are set as follows: $\alpha = 1$, $\sigma = 1.34$, $\beta = 1$.

The quantitative characteristics of the denoising performance were estimated using the following measures:

$$\begin{aligned} SDR &= \frac{\|\mathbf{d} - \mathbf{g}\|_2}{\|\mathbf{u} - \mathbf{g}\|_2}, & ADR &= \frac{\|\arccos(\widehat{\mathbf{d}\mathbf{g}})\|_2}{\|\arccos(\widehat{\mathbf{u}\mathbf{g}})\|_2}, \\ NDR &= \frac{\|\mathbf{d}\|_2}{\|\mathbf{g}\|_2}, & DDR &= \frac{\|\mathbf{u}\|_2}{\|\mathbf{g}\|_2}, \end{aligned}$$

where \mathbf{g} is the ground truth flow, \mathbf{d} is the corrupted input data, \mathbf{u} is the denoised vector field.

The first measure, SDR , is the ratio of standard errors before and after denoising. It gives an overall information about the error reduction after denoising. The second measure, ADR , shows the reduction of an average angle deviation of the restored vector field relative to the ground truth. The two last measures are the ratios of an average length noisy vector and restored vector to the ground truth, respectively. Comparing these values one can get more details about the level of noise in the considered data set before and after application of the procedure.

7.2 Two Dimensional Case

7.2.1 Uniformly Distributed Noise

Figures 7.1 and 7.2 illustrate applicability of the method to the vector fields with uniformly distributed noise. In order to show that the approach works well with any type of fluid flows, two vector fields representing different flow regimes have been chosen for the analysis. A high level of noise has been chosen for this experiment in order to show the reconstructive ability of the algorithm for the restoration of vector fields with high noise-to-signal ratios. To illustrate this better, the original signals are shown in rectangular regions in the upper right corners of the noisy input fields in Fig. 7.1. These corrupted input vector fields \mathbf{d} as shown in the Fig. 7.1 can be obtained, for instance, in the laboratory or in-situ experiments when using very sensitive sensors or an inaccurate data processing method.

Comparison of the denoised output vector field \mathbf{u} (see Fig. 7.2) with the ground truth \mathbf{g} (Fig. 7.3) shows that the algorithm recovers the large-scale structures of the flow very well. So, we conclude that the *default* boundary conditions applied here give a reasonable approximation for such structures near the boundaries of the domain. The results demonstrate that quantitative and robust denoising is possible even in case of high noise-to-signal ratios.

7.2.2 Gaussian White Noise and Missing Data

A remarkable property of the presented method is that the class of physically admissible signals is modelled rather than the noise itself. As a result, the method treats the regions

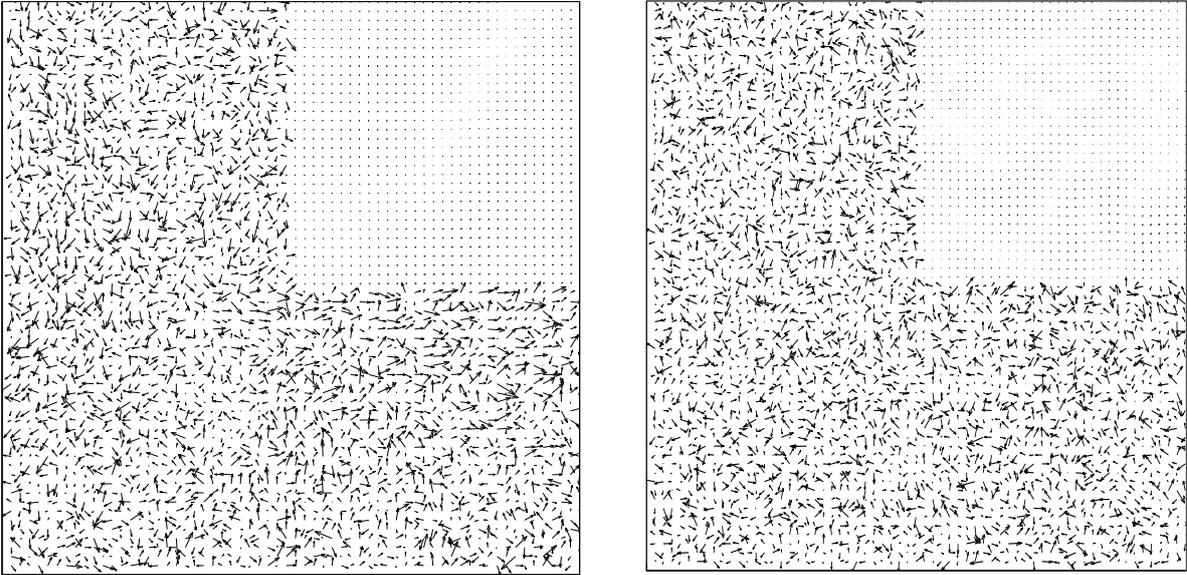


Figure 7.1: Two cases of extremely noisy data. The noise has been cut out within a rectangular region to illustrate the signal-to-noise ratio.

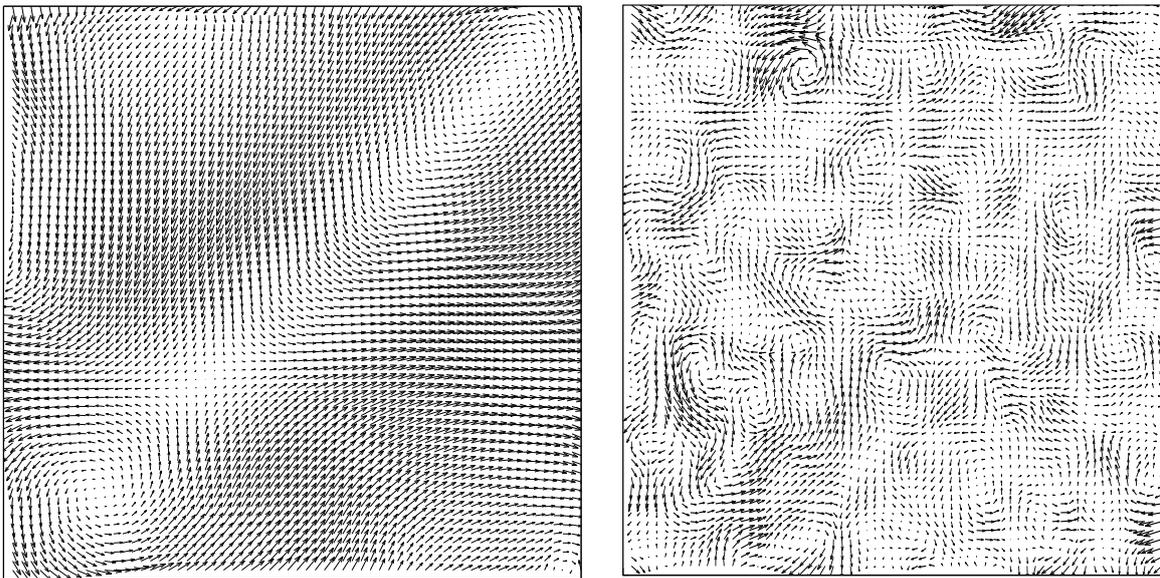


Figure 7.2: Denoised flows \mathbf{u} . **Left:** Performance measures: SDR=4.34, ADR=3.046, NDR=2.12, DDR=0.82. **Right:** SDR=15.12, ADR=2.45, NDR=9.51, DDR=1.004.

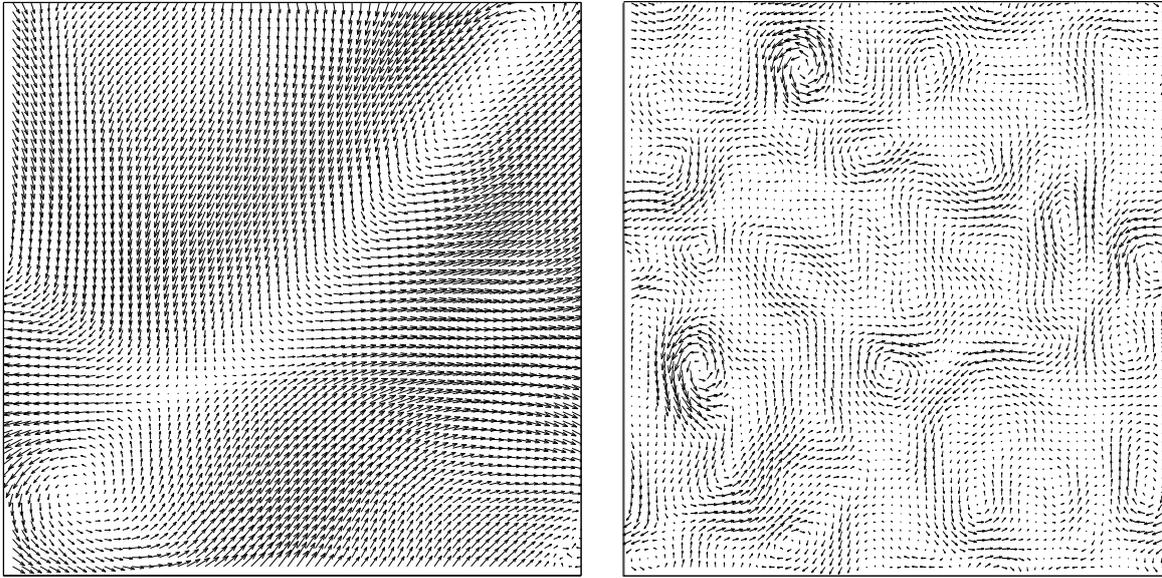


Figure 7.3: Ground truth \mathbf{g} of flows depicted in Fig. 7.1.

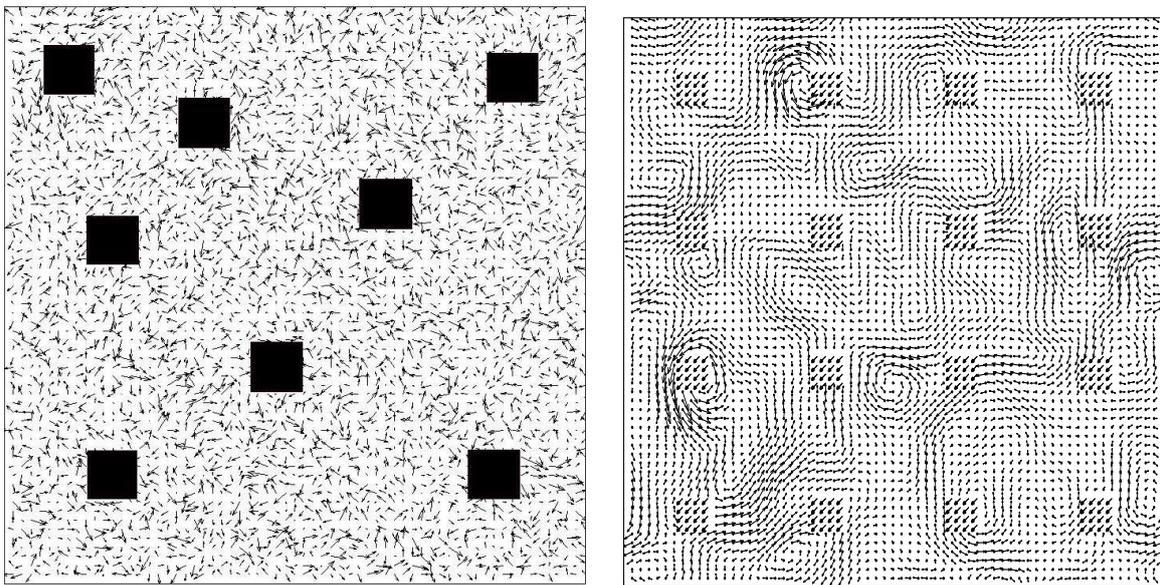


Figure 7.4: Two examples of corrupted data \mathbf{d} with missing values or outliers within rectangular regions. The location of these regions is assumed to be *unknown*.

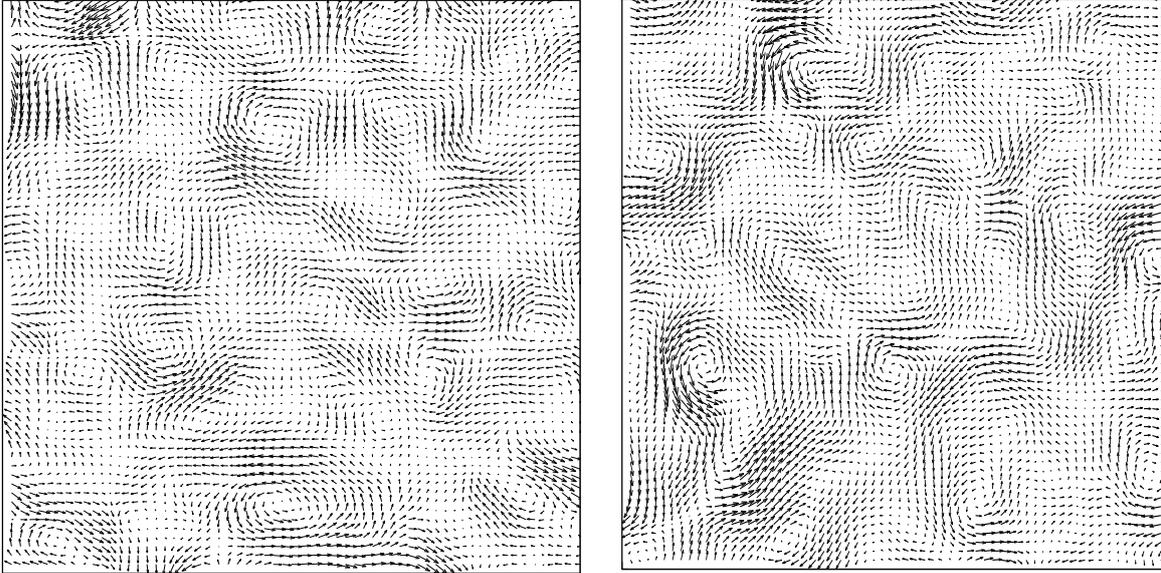


Figure 7.5: Restored vector field \mathbf{u} . **Left:** Performance measures: SDR=3.90, ADR=3.7, NDR=2.48, DDR=0.88. **Right:** Performance measures: SDR=1.645, ADR=0.532, NDR=1.0087, DDR=0.924.

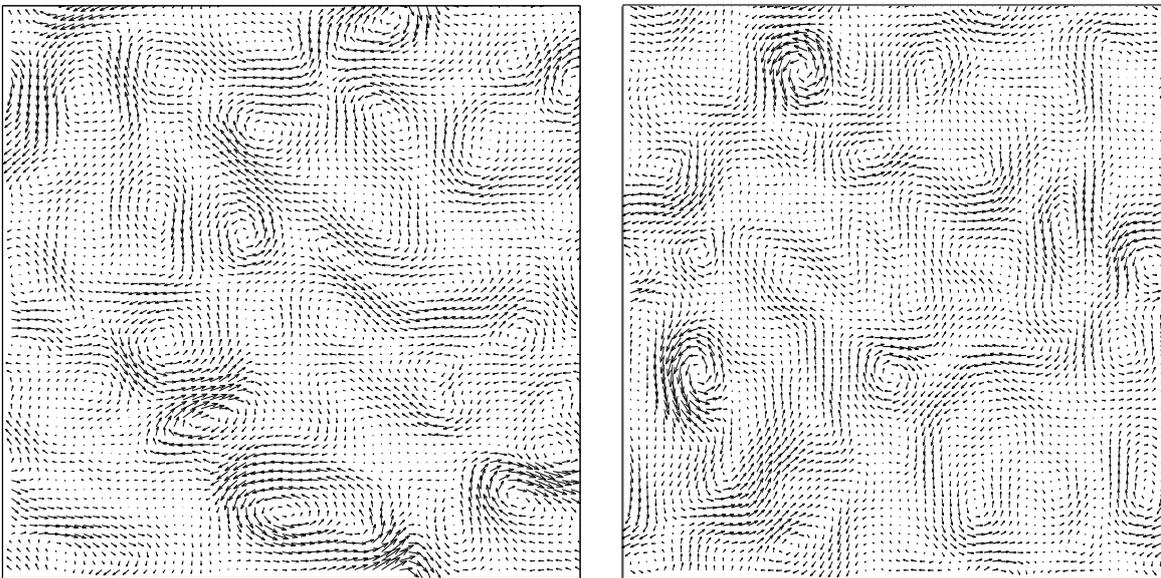


Figure 7.6: Ground truth \mathbf{g} of flows depicted in Fig. 7.4.

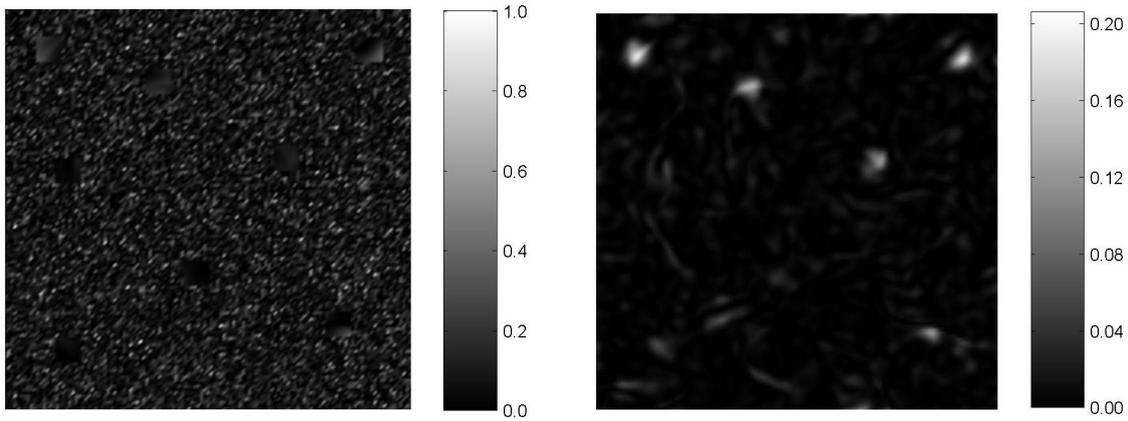


Figure 7.7: Deviation from the ground truth before denoising (**left**) and after denoising (**right**). Note the different scales to the right of each panel.

with missing or wrongly estimated data as “noise” (an example of such data is given in Fig. 7.4).

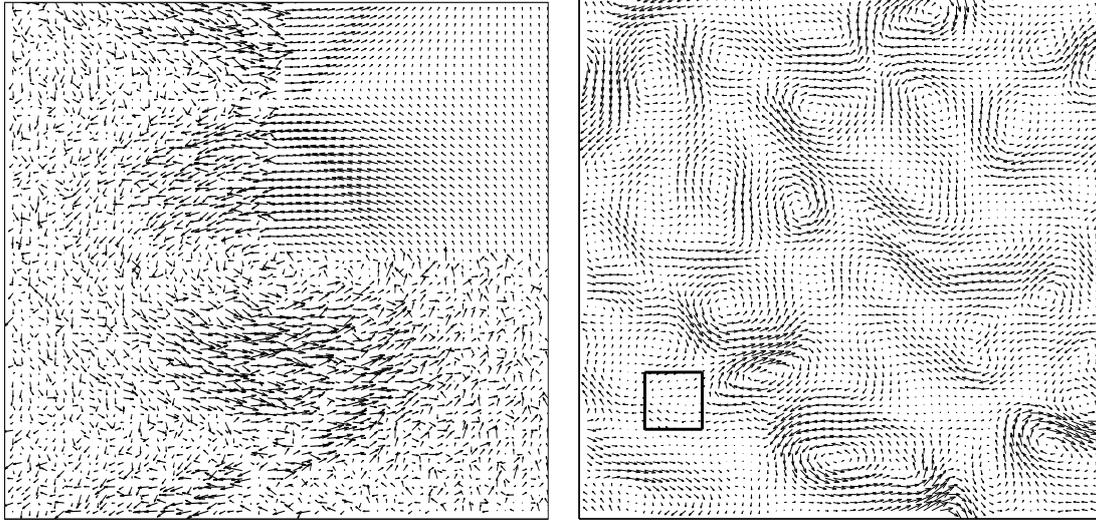
It is assumed that some regions of the field do not provide any information at all. The origin of these regions often accounts for losses of correlation that usually happen to the PIV velocimetry output in some areas. These regions are presented by the left panel in Fig. 7.4 as black rectangles. The positions of the regions were randomly set and are assumed to be *unknown*. In addition, white noise was added to these data. Comparison of the left panels of Fig. 7.5 and Fig. 7.6 shows that the algorithm successfully recovered most of the details of the vector field. This remark concerns not only the originally recorded data, but also the regions with missing data.

Figure 7.7 depicts in gray colour an absolute deviation of the vector field from the ground truth before and after denoising. The quantitative scale of the gray values is plotted to the right of the panels (two *different* scales are used for high- and low-noise cases). It is interesting to note that even in problematic regions corresponding to the black rectangles with missing data (Fig. 7.4, left panel) the deviation drops from about 0.7 (Fig. 7.7, left panel) to less than 0.2 (Fig. 7.7, right panel).

Note that spatial distribution of error does not affect an overall result of the reconstruction procedure. The method works equally well with randomly distributed noise and corruptions (previous examples) and with periodically distributed error. The right panel of Fig. 7.4 shows the vector field corrupted by the periodical distribution of error represented by the groups of parallel outliers. The result of the reconstruction and the ground truth are given in the right panels of Fig. 7.5 and Fig. 7.6, respectively.

7.2.3 Low Noise Level

Figures 7.8(a), 7.9(a) and 7.10(a) illustrate and confirm the fact that the VTE-based algorithm does not depend on any particular model of noise and errors. Figure 7.8(a)



(a) A Kelvin-Helmholtz billow corrupted by low-level noise.

(b) Ground truth turbulent flow \mathbf{g} .

Figure 7.8: **Left:** Experiment on denoising of Kelvin-Helmholtz flow. **Right:** Experiment on comparison of VTE and TV-based denoising approaches.

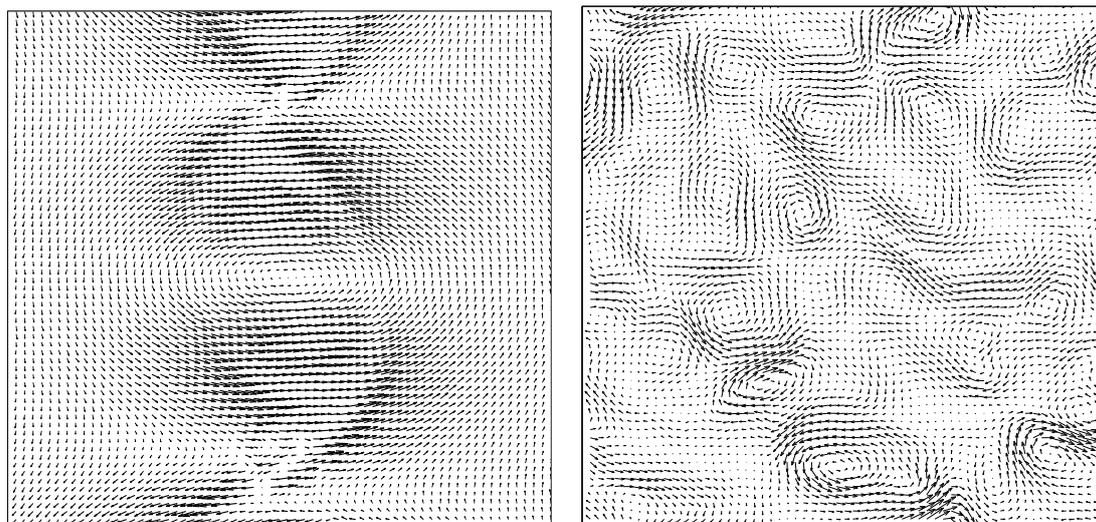
shows an example of polluted flow (a Kelvin Helmholtz billow) with a very low signal-to-noise ratio. The comparison of Fig. 7.9(a) with Fig. 7.10(a) reveals that the reconstructed vector field almost coincides with the ground truth (quantitative results are given in the figure captions). So, the limitation of the method is evident: the quality of the restoration of small-scale structures decreases with the increase of signal-to-noise ratio.

7.3 Comparison with other Methods

In this section the VTE-based denoising approach is compared with some other conventional post-processing methods discussed in Sec. 3.5. The presented here experiments are aimed to show that the filters which are based on a correct physical background can restore the corrupted vector fields much better than some other conventional filters based on any particular model of noise. Moreover, the VTE-based restoration procedure cope with some particular types of noise even better than the methods specially developed to remove these types of noise.

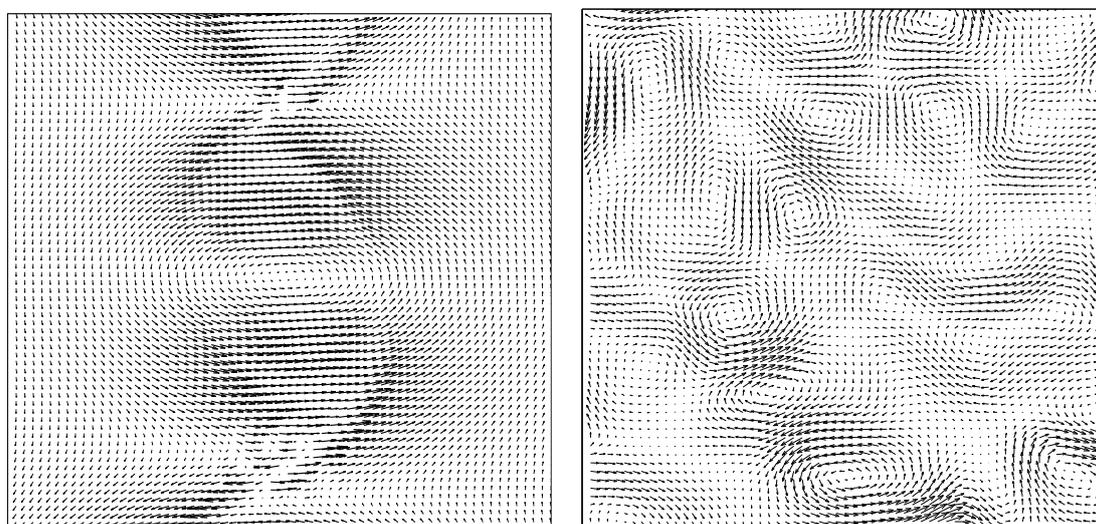
7.3.1 Comparison with Scalar Valued Total Variation Denoising

We have compared our approach with variational denoising method [70, 17] which exploits a total variation (TV) minimization procedure. The ground truth vector field \mathbf{g} depicted



(a) Restored Kelvin-Helmholtz flow. Performance measures: SDR=4.43, ADR=6.032, NDR=1.12, method **g**.
 (b) Fluid flow restored by VTE-based restoration. Performance measures: SDR=4.43, ADR=6.032, NDR=1.12, method **g**.
 DDR=0.999.

Figure 7.9: **Left:** Experiment on denoising of Kelvin-Helmholtz flow. **Right:** Experiment on comparison of VTE and TV-based denoising approaches.



(a) Ground truth of Kelvin-Helmholtz flow (b) Fluid flow **g** restored by TV-denoising method.

Figure 7.10: **Left:** Experiment on denoising of Kelvin-Helmholtz flow. **Right:** Experiment on comparison of VTE and TV-based denoising approaches.

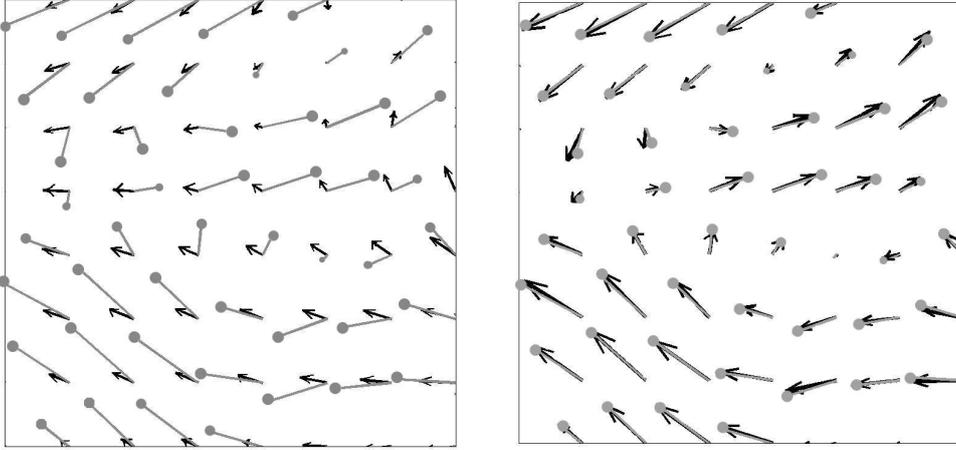


Figure 7.11: **Left:** Close-up view of the result shown in Fig. 7.10(b) for the section marked in Fig. 7.8(b) together with the ground truth vectors (gray vectors with circled heads). **Right:** Similar result obtained by the developed here VTE-based restoration approach. The TV-denoising method (left) fails to reproduce the flow structure correctly.

in Fig. 7.8(b) has been chosen as a test bed. The TV-based denoising was separately applied to both components of the input data set, i.e. $\mathbf{g} = (g_1, g_2)^\top$.

At first glance the results depicted in Fig. 7.10(b) and Fig. 7.9(b) look similar. Close inspection, however, reveals some obvious differences. The TV-based denoising creates some significant nonphysical flow structures even in the case when the input vector field does not contain any noise at all.

Figure 7.11 shows two close-up views of the field obtained for the fragments of the input field that are marked in Fig. 7.8(b). These results illustrate that approximately 30% of all TV-based denoised vectors have their directions opposite to the ground truth flow, while the result returned by our method nearly matches the ground truth.

It is clear, however, that our results not *exactly* reproduce the ground truth, as well. One of the reasons could be the unknown 'correct' boundary conditions. Another potential source of errors could be the neglect of nonstationary effects. It is clear that omitting of the time derivative in the VTE (see the discussion in two sections followed by eqn. (4.14)) introduces some noise, and this noise can not be removed. However, the right panel of Fig 7.11 shows that these errors are small.

7.3.2 Comparison with Vector Valued Total Variation Denoising

The VTE-based method was compared against the vector-valued TV-denoising approach which is the state-of-the-art in the area of vector field denoising. Both high and low SNR were chosen for the test runs (namely, 10 and 4). The noisy input, the obtained results and the ground truth are presented in Figs. 7.12-7.16. The vector fields given in Figs. 7.12 and 7.14 (left panels) were reconstructed using the TV-denoising algorithm (right panels) with $\lambda = 0.01$.

As it can be seen from these figures, the TV-denoising procedure gives reasonable results only in the case when the noise component in the input data field is negligibly small in comparison with the signal (see Fig. 7.12). If the noise level is comparable with the signal, the reconstruction becomes poor, see Fig. 7.14. With the increase of λ an impact of the smoothing term becomes stronger, and, as a result, the output vector field becomes over-smoothed. Figure 7.16 demonstrates such an example in which λ was taken five times larger than in the previous experiment.

In comparison with the TV-denoising, the VTE-based algorithm reconstructs the vector fields fairly well for both high and low SNRs, see Fig. 7.13 and Fig. 7.15. Moreover, in the case of low noise contamination the biases still can be observed in the vector field reconstructed with the TV-based denoising algorithm (see Fig. 7.12) while the same input data denoised with VTE approach are more smoothed. As a result, the achieved errors for the VTE-based approach are at least 1.25 times better than similar values for the TV reconstructive method.

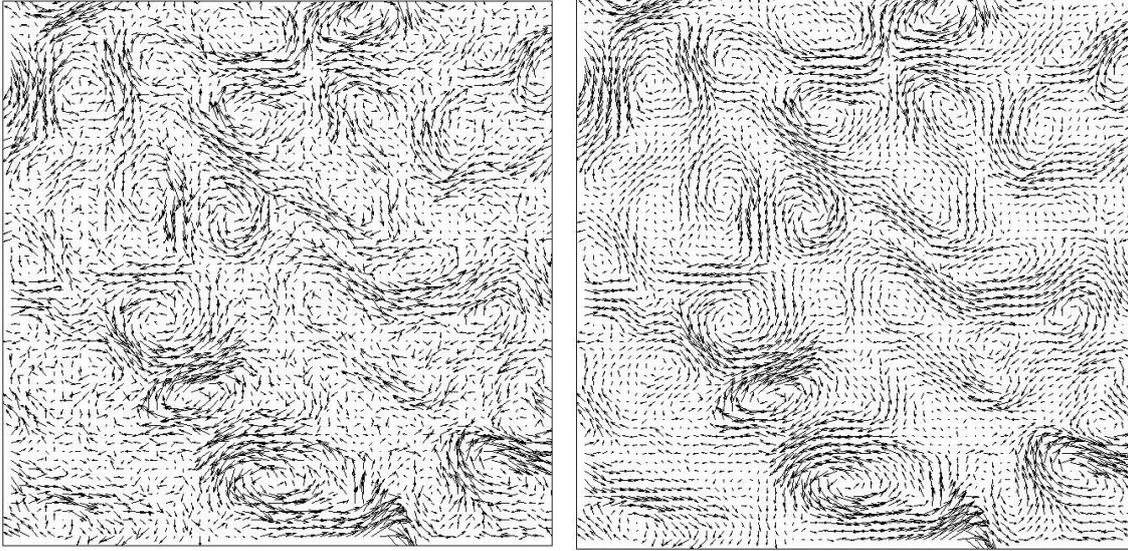


Figure 7.12: **Left:** Input vector field with SNR=10. **Right:** The vector field reconstructed with vector valued TV-based restoration approach ($\lambda = 0.001$). The corresponding error measures SDR=2.07, ADR= 2.05, LDR=0.99.

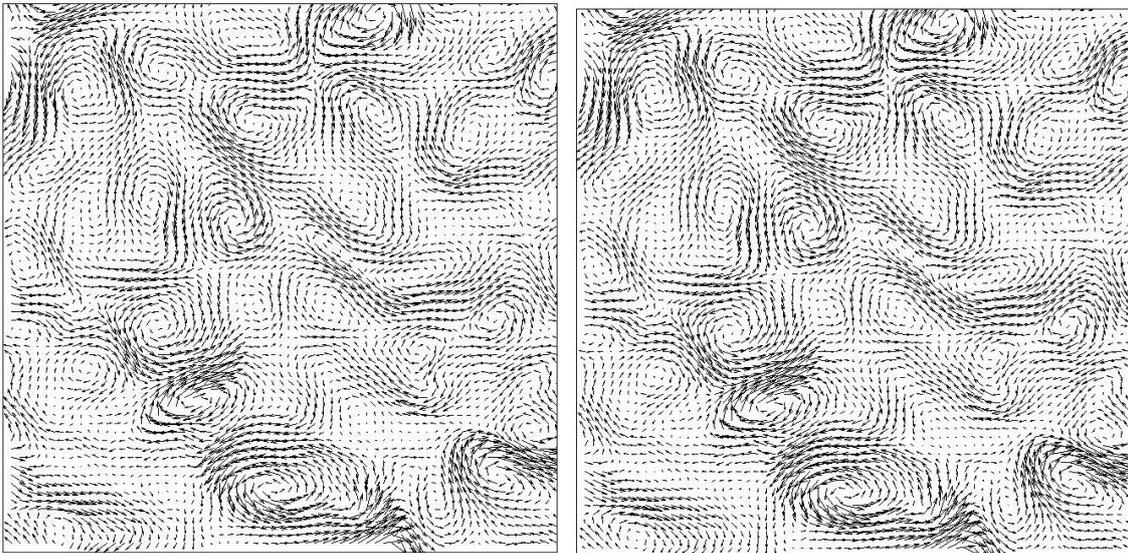


Figure 7.13: **Left:** Ground truth vector field. **Right:** The noisy vector field depicted in Fig. 7.12 reconstructed by the VTE-based restoration approach. The corresponding error measures SDR=2.68, ADR= 3.18, LDR=0.93.

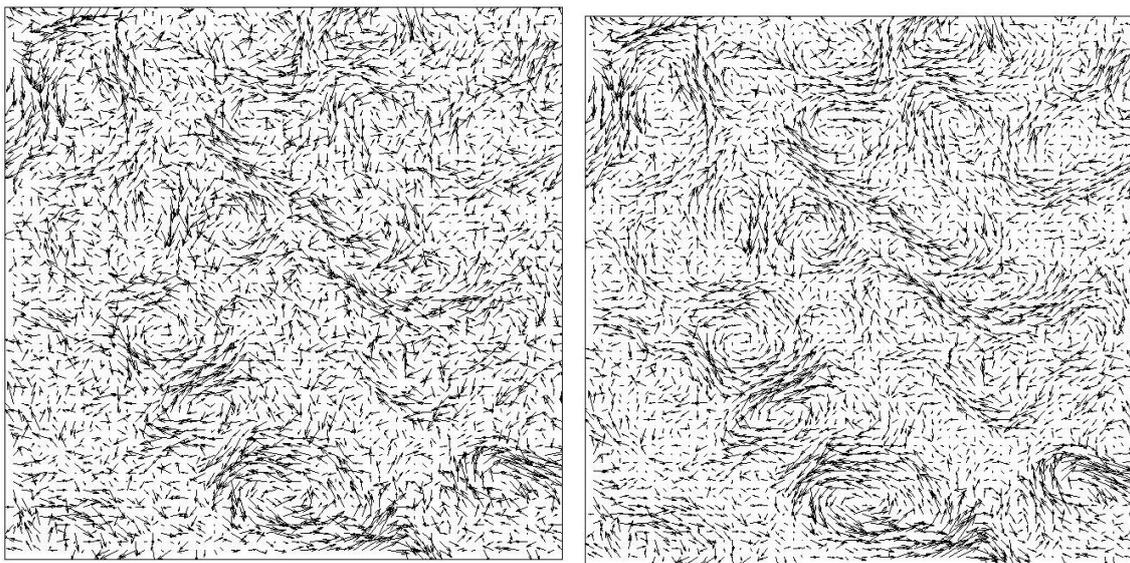


Figure 7.14: **Left:** Input vector field with SNR=4. **Right:** The vector field reconstructed using the vector valued TV-based restoration approach ($\lambda = 0.001$). The corresponding error measures: SDR=1.52, ADR= 1.44, LDR=1.11.

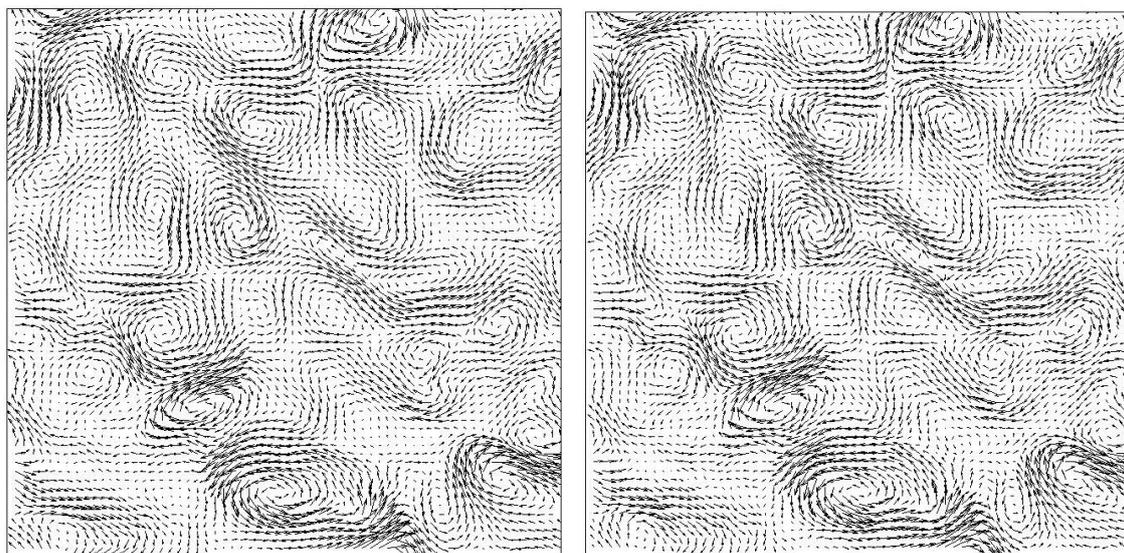


Figure 7.15: **Left:** Ground truth vector field. **Right:** The noisy vector field depicted in Fig. 7.12 reconstructed by the VTE-based restoration approach. The corresponding error measures: SDR=3.87, ADR= 3.63, LDR=0.94.

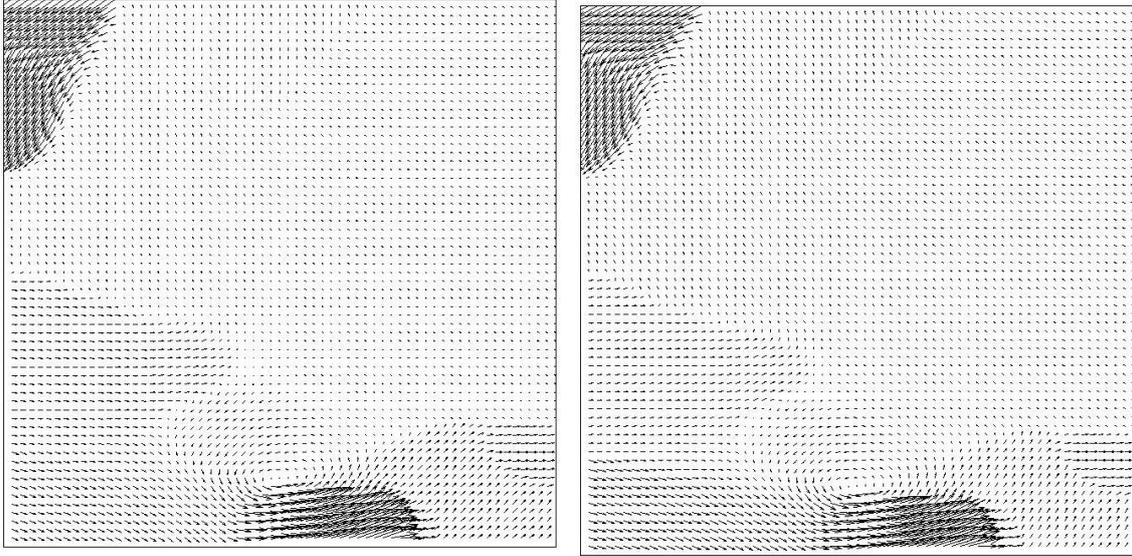


Figure 7.16: The vector fields reconstructed by the vector valued TV-based restoration approach. The adjustable parameter λ was set to 0.05. The SNRs for both input vector fields were **Left: 10; Right: 4**.

7.3.3 Comparison with the Median and Mean Value Operator Denoising

Median operator denoising. 'Spurious vectors' is a typical noise commonly produced by the PIV measurements (Sec. 3.5). It usually appears in a single or small groups of outliers randomly distributed in the output vector field. The routine procedure for their detection and elimination is based on the application of a median filter [100]. A vector field corrupted by such a noise is given to the left in Fig. 7.17. For the comparison reason the spurious vectors of this vector field were removed separately by the median filter (right panel of Fig. 7.17) and the VTE-reconstructive method (left panel of Fig. 7.18). For the simplicity of the comparison the 'cleaned' vector field represented by black vectors is overlaid with the ground truth presented by red vectors. The ground truth is given on the right panel of Fig. 7.18, as well.

As it can be seen from Fig. 7.17, the median filter removes single outliers quite efficiently. However, the result of the denoising is getting worse when the neighbouring grid nodes contain two or more outliers (find the areas in Fig. 7.17 surrounded by green rectangles). At the same time all spurious vectors removed by the VTE-based approach nearly perfectly coincide with the ground truth (Fig. 7.18).

Mean operator denoising. Here we demonstrate the comparison of the VTE-based method with the mean value operator denoising. The input data field was deliberately polluted and corrupted by the Gaussian noise (see Fig. 7.19) so that the mean value operator should give the best performance. These data were restored independently by the mean

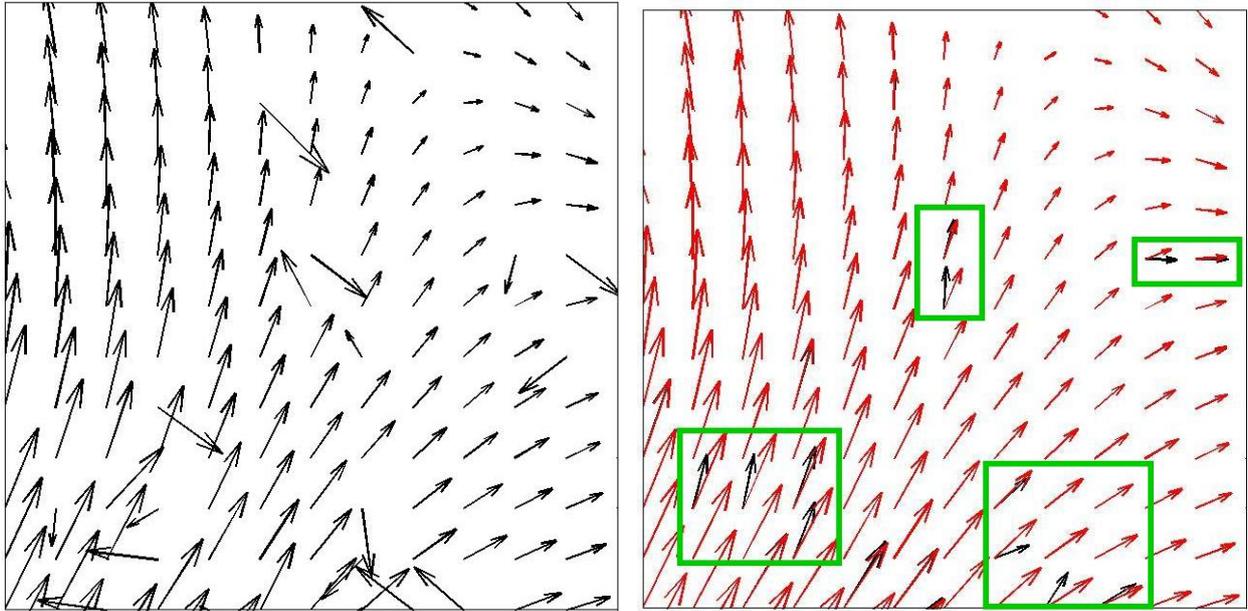


Figure 7.17: **Left:** Input vector field containing outliers. **Right:** The same vector field but after reconstruction by the median filter (black) overlaid with the ground truth (red). The green squares show the areas where the input field (left plane) nodes contained two or more outliers before denoising.

value filter and the VTE-based procedure. In the case of mean value filter the corrupted data were substituted by the corresponding mean values. Figures 7.20 and 7.21 show the result of the denoising at 1.2 noise-to-signal ratio. The left panels in Figs. 7.20-7.21 represent the result after the filtering, and the right hand side panels show the ground truth.

The mean value filter was iteratively applied to the noisy input field until the best values of SDR and ADR were achieved (the further filtering was not reasonable because of over-smoothing of the output vector field). Looking at the flow structures after the denoising it is clear that the data still contain some biased errors. While the result of the VTE-based approach given in Fig. 7.21 is smoother, the biases are removed from the output vector field and, as a result, the corresponding quantitative error reduction measures are approximately 10% better than the error values in two previous experiments. Thus, considered examples illustrate that the VTE-based filter removes the homogeneously distributed noise even better than the filters based on this particular model of noise.

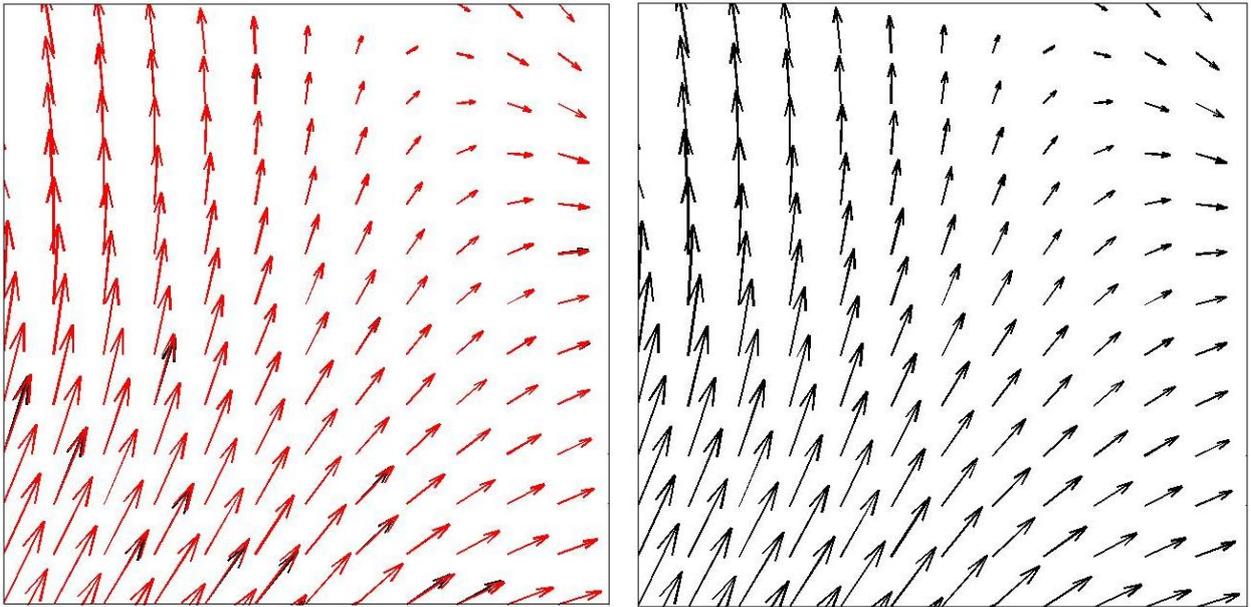


Figure 7.18: **Left:** The same input vector field as in Fig. 7.17 but after reconstruction by the VTE filter (black) overlaid with the ground truth (red). **Right:** Ground truth.

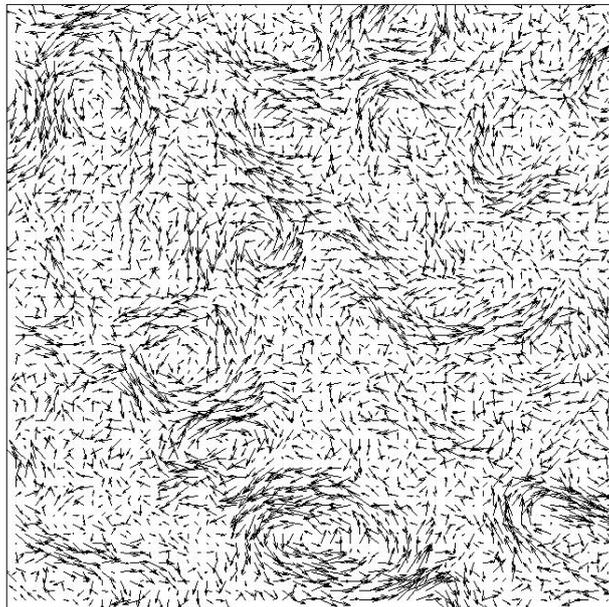


Figure 7.19: A vector field containing the Gaussian noise.

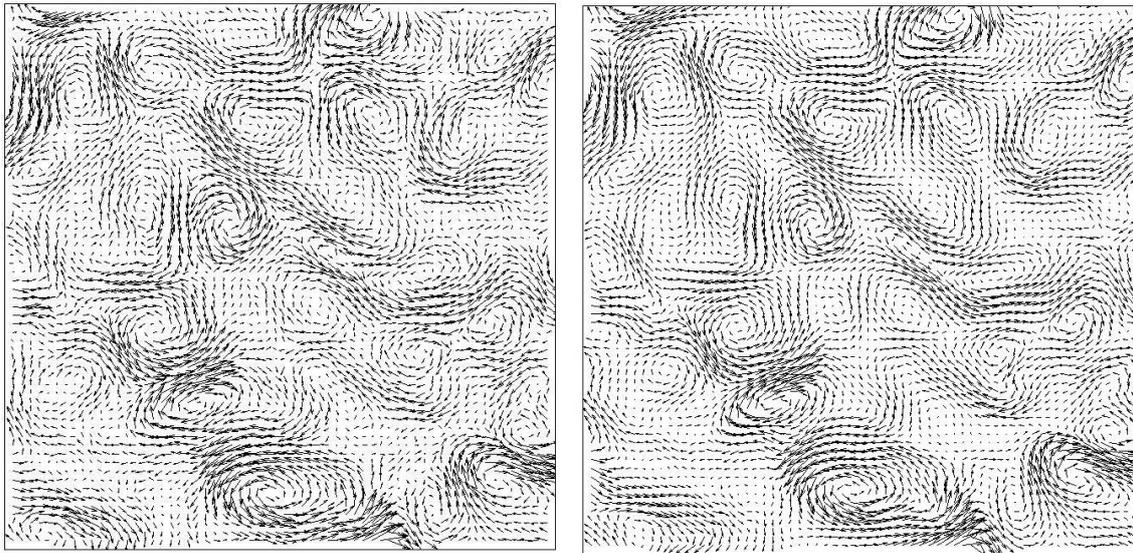


Figure 7.20: **Left:** A vector field reconstructed by the mean value filter. The corresponding error values are as follows: SDR=2.32, ADR=2.26, NDR=1.18, DDR=0.94. **Right:** Ground truth.

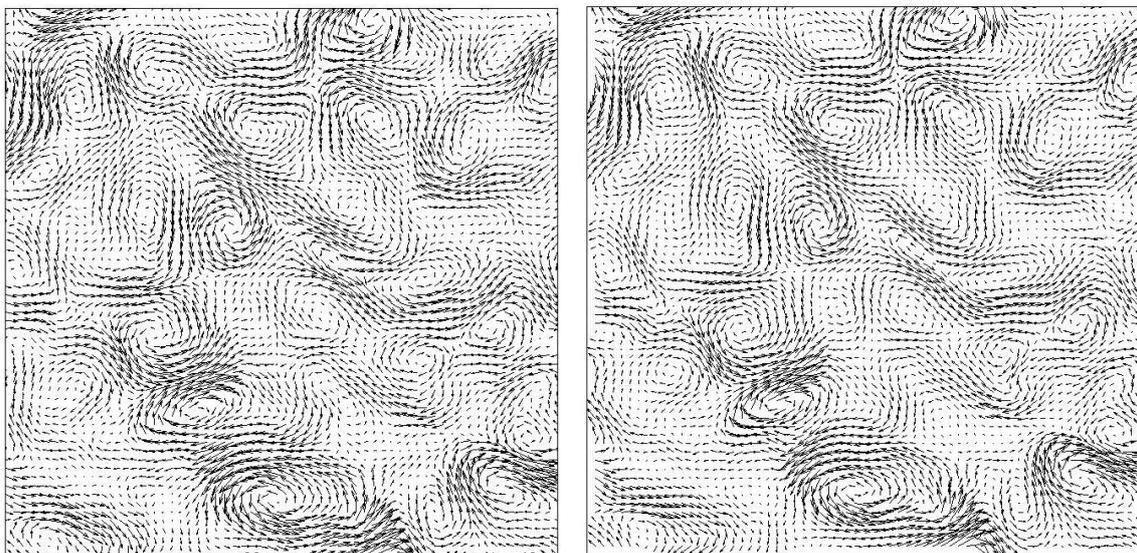


Figure 7.21: **Left:** A vector field reconstructed by the VTE-based filter. The corresponding error values are: SDR=2.57, ADR=2.51, NDR=1.18, DDR=0.92. **Right:** Ground truth flow \mathbf{g} .

7.4 Three Dimensional Case

We started the demonstration of the performance of the VTE-based restoration approach considering its 2D version. All flows tested so far were assumed to be two-dimensional. Now we turn to the discussion of a more general three-dimensional case. The efficiency of the method (Sec. 4.3) in three dimensions in recovering both velocity and vorticity of any fluid flow can be illustrated by the following examples. We consider here two vector fields. The first one is obtained in numerical experiments with a four-cell gravity convection calculated in the framework of a fully-nonlinear non-hydrostatic numerical model MITgcm [59]. The second one reproduces a velocity vector field around a cylinder streamlining by a turbulent flow [26]. Both these fields were corrupted by different types of noise containing various signal-to-noise ratios. The size of each vector field was $128 \times 128 \times 128$ grid steps. Figures 7.22-7.32 represent the noisy, denoised, and ground truth flows, respectively. Each test flow is accompanied by two series of diagrams: the diagrams in Figs. 7.22-7.25 depict a 3D vorticity iso-surfaces and Figs. 7.26-7.32 give the velocity slices taken in surface-to-bottom plane or in parallel to bottom plane.

7.4.1 Uniformly Distributed Noise

Let us consider first the case of uniformly distributed noise. Figures 7.22, 7.27 and 7.30 represent two fluid flows corrupted by the Gaussian noise with different signal-to-noise ratios. The biggest noisy signal has been introduced to the convective flow. The signal-to-noise ratio in this case was equal to 1. In the cylinder-flow experiment the appropriate value was only 0.125. The comparison of the recovered flows (Figs. 7.23, 7.28 and 7.31) to ground truth (Figs. 7.24, 7.29 and 7.32) reveals that the denoising procedure restored nearly all details in the recovered vector fields in both cases.

It's important to note that the level of noise normally is much higher in the vorticity field rather than in velocity. This takes place because vorticity is the first derivative from the velocity. However, the denoising procedure reconstructs both with pretty good accuracy (compare zoomed cuts from right panels of Figs. 7.23, 7.24 in Figs. 7.25, 7.26). After recovering the square root error in the four-cell convection and cone flow vector field is reduced by 8.76 and 1.34 times, respectively.

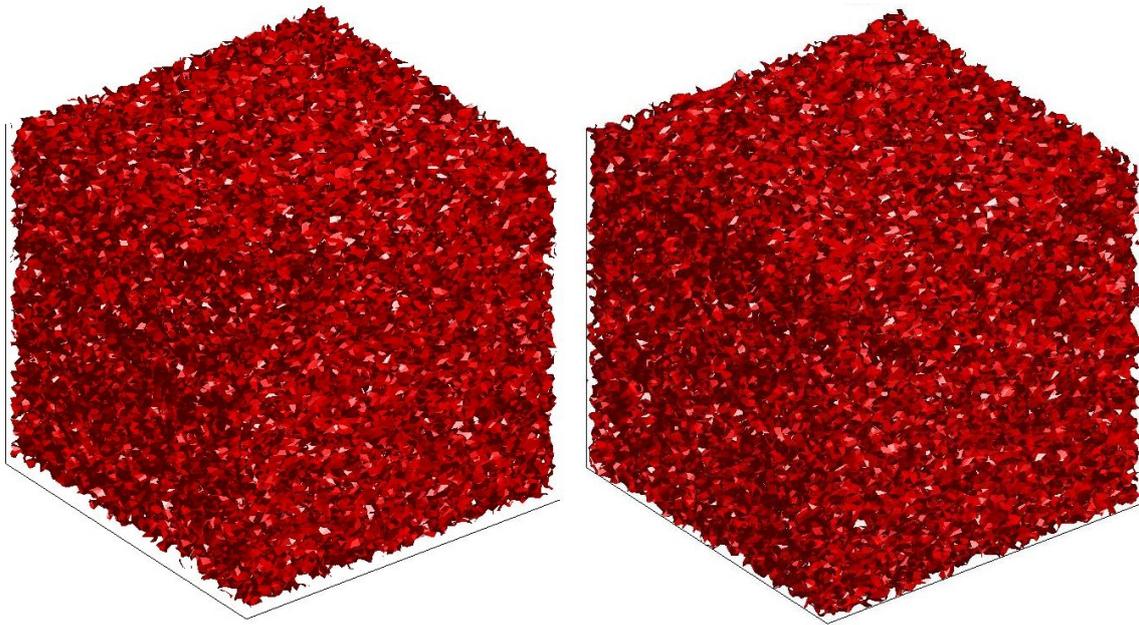


Figure 7.22: Three-dimensional vorticity iso-surfaces of noisy fluid flows: the four-cell vertical convection (left) and the flow around the cylinder (right).

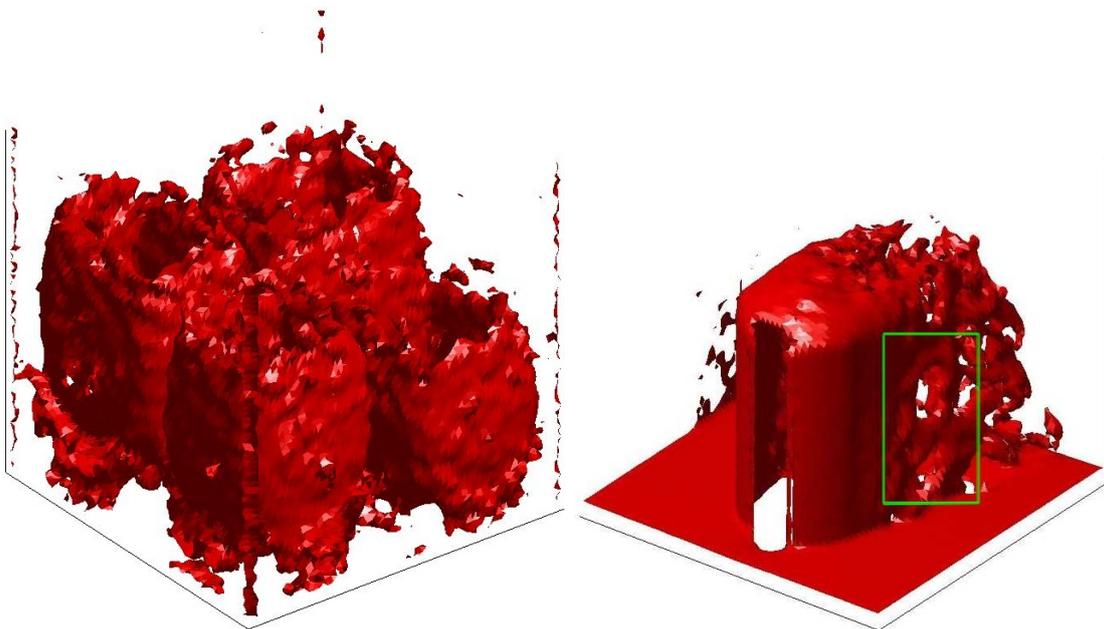


Figure 7.23: The same as above but after application of the VTE-based denoising procedure.

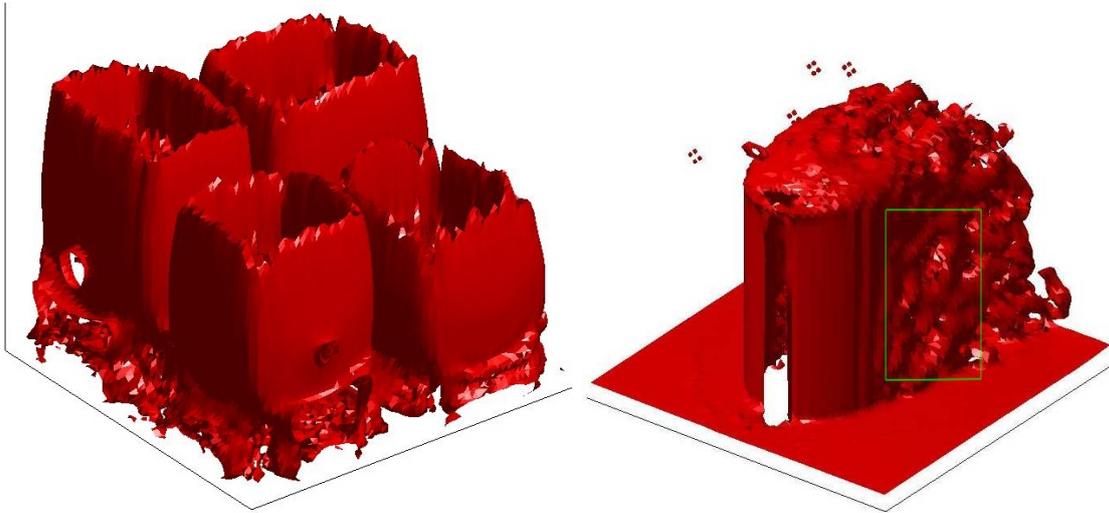


Figure 7.24: The ground truth of the vorticity for: four-cell vertical convection (left) and flow around the cylinder (right).

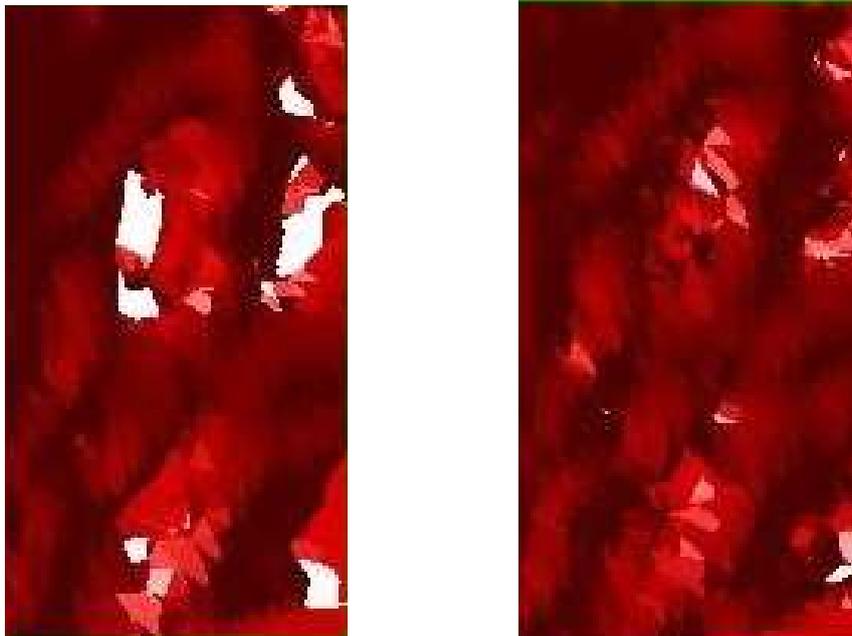


Figure 7.25: Zooms of the rectangular fragments depicted in Figures 7.23 (left) and 7.24 (right). They represent the denoised field and the ground truth, respectively.

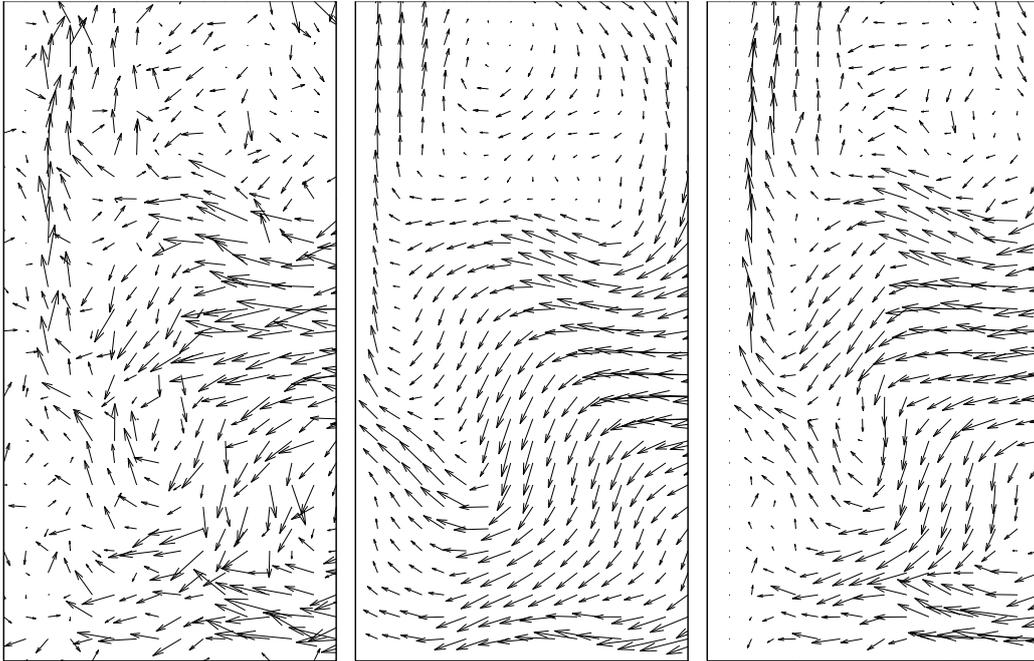


Figure 7.26: Vertical cross-sections of the velocity field through the center of the fluid flow over cylinder: noisy (left), denoised (middle), ground truth (right) signals.

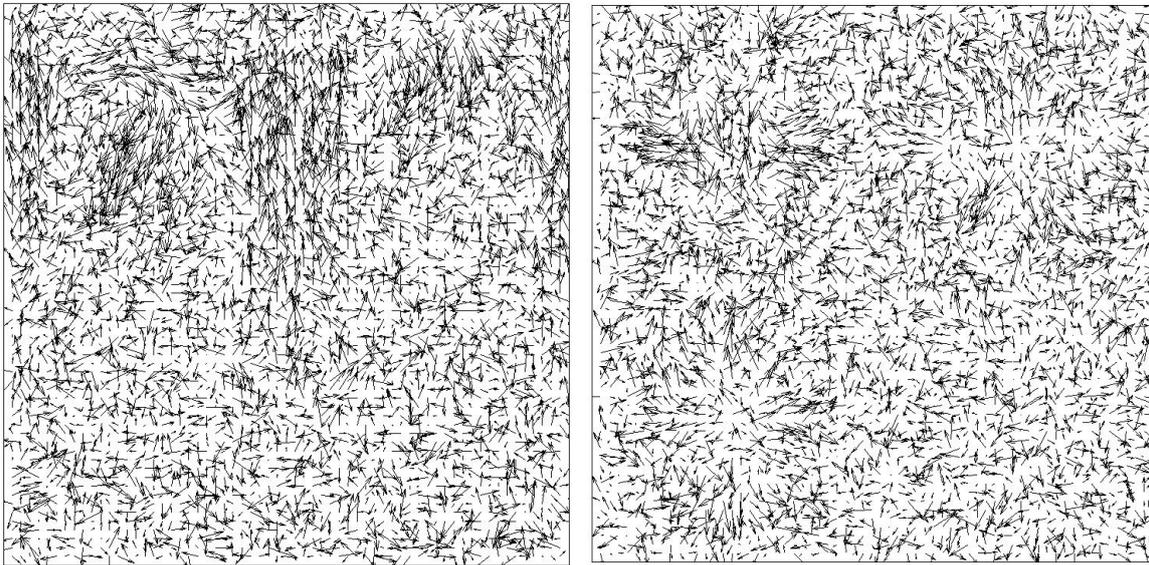


Figure 7.27: Vertical (left) and horizontal (right) cross-sections of the velocity field in the area of vertical convective cells.

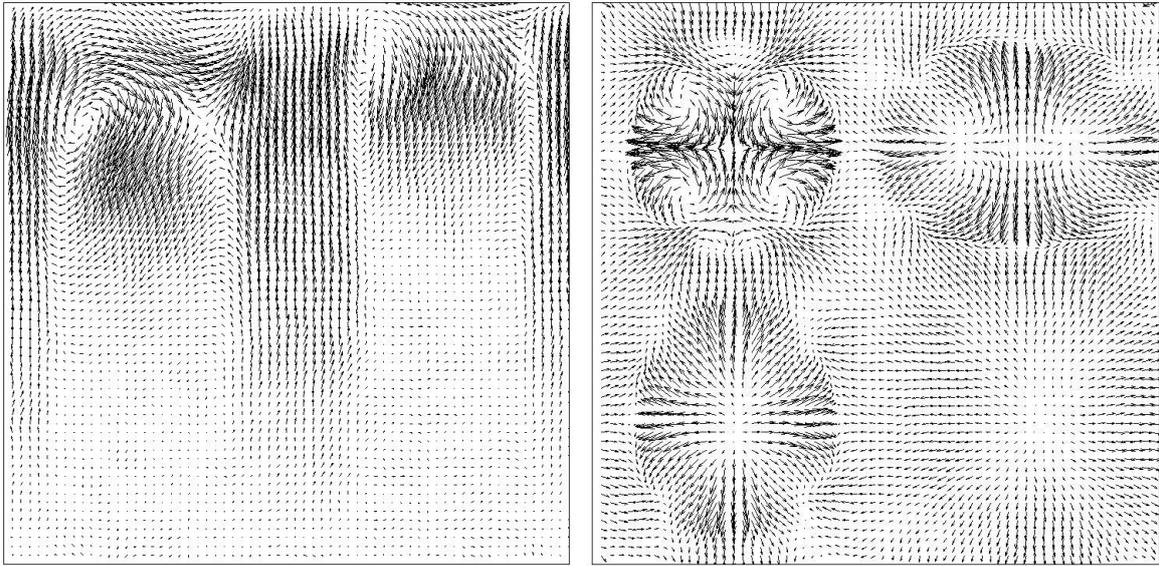


Figure 7.28: The same as above but after application of the VTE-based denoising procedure.

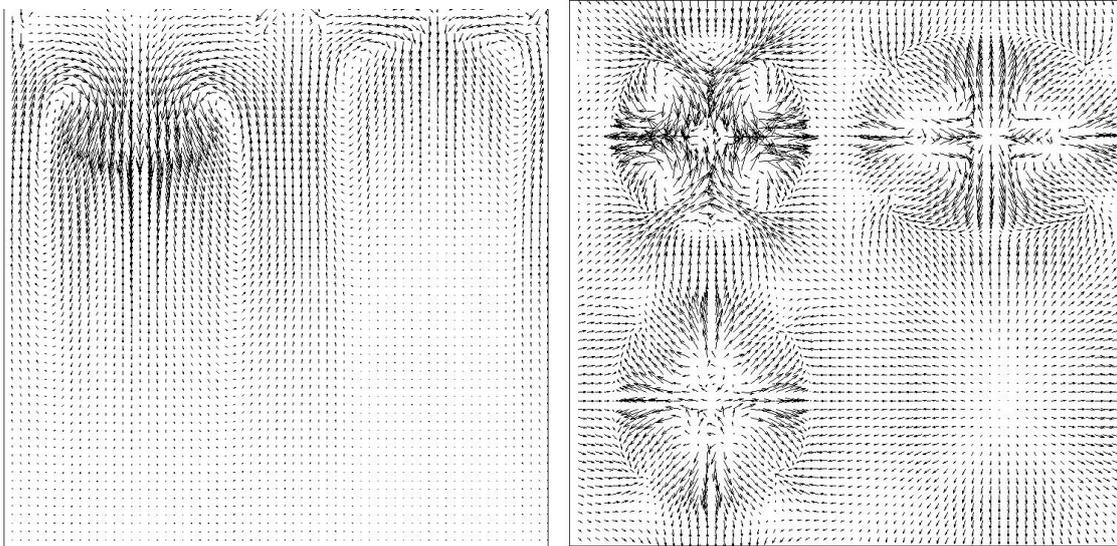


Figure 7.29: The actual ground truth of the velocity in convective cells used as an initial data set.

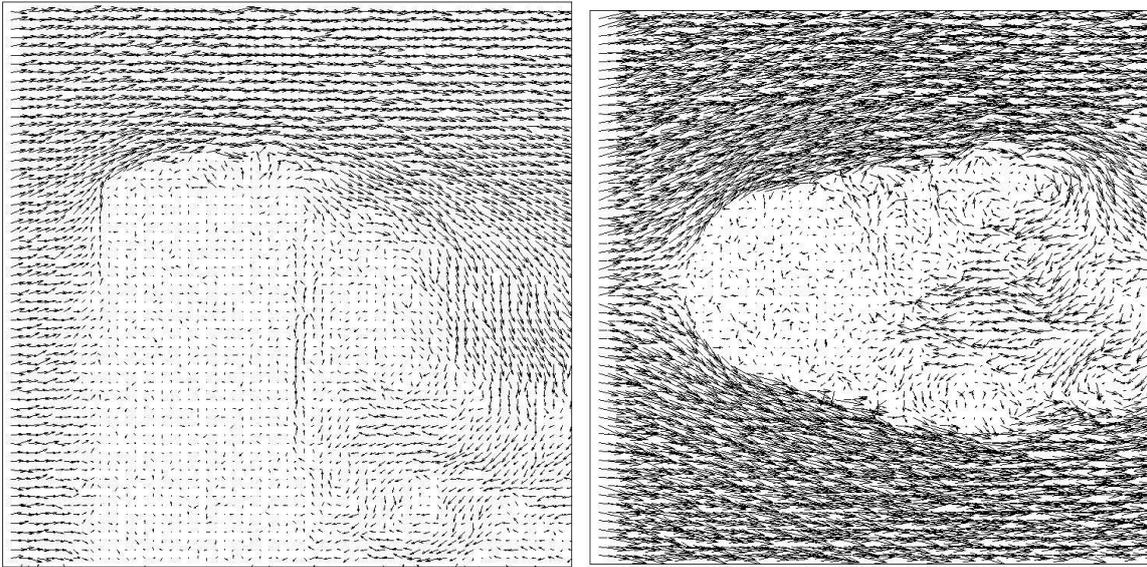


Figure 7.30: Vertical (left) and horizontal (right) cross-sections of the noisy velocity through the centres of the flow over cylinder.

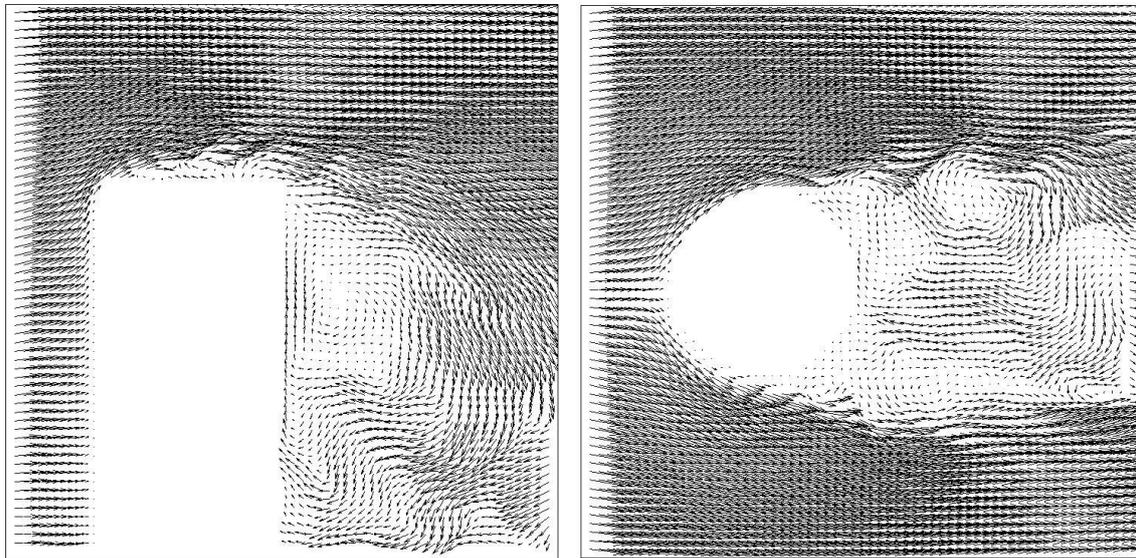


Figure 7.31: The same as above but after the application of the VTE-based denoising procedure.

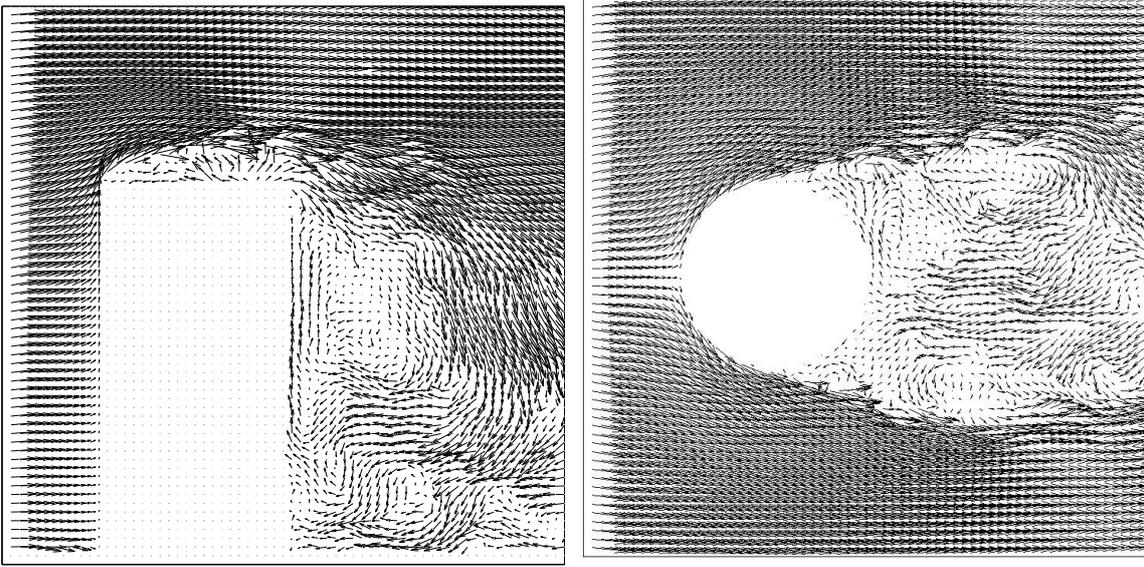


Figure 7.32: The actual ground truth of the velocity over cylinder used as an initial data set.

7.5 Super-resolution Approach: Results

7.5.1 Data Sets and Set-Up

Data sets. The first data set used in testing of the super-resolution method was taken from the numerical experiments with the four-cell deep-water convection discussed above. As long as this method is targeted at restoration of the lost fragments of fluid flows, the initial ground truth has been deliberately corrupted. Only 10% of randomly distributed velocity vectors were taken as an input data set (see Fig. 7.33).

In second series of experiments the result of a CFD-simulation of the turbulent flow behind a cylinder was taken as an initial ground truth. In this case only 20% of randomly distributed velocity vectors were taken as input data set (see Fig. 7.34). The background (ground truth) vector fields for both examples were taken the same as used in the previous section.

Quantization errors. One of the most typical errors in PIV or PTV experiments is related to the incorrect estimation of particle displacements. Some others appear due to the sub-pixel uncertainty in particle's displacement and wrong estimation of the displacement gradients. They lead to bias errors in the output vector fields (for more details see Sec. 3.5). We simulate such real measurements by corrupting both data sets with this type of noise.

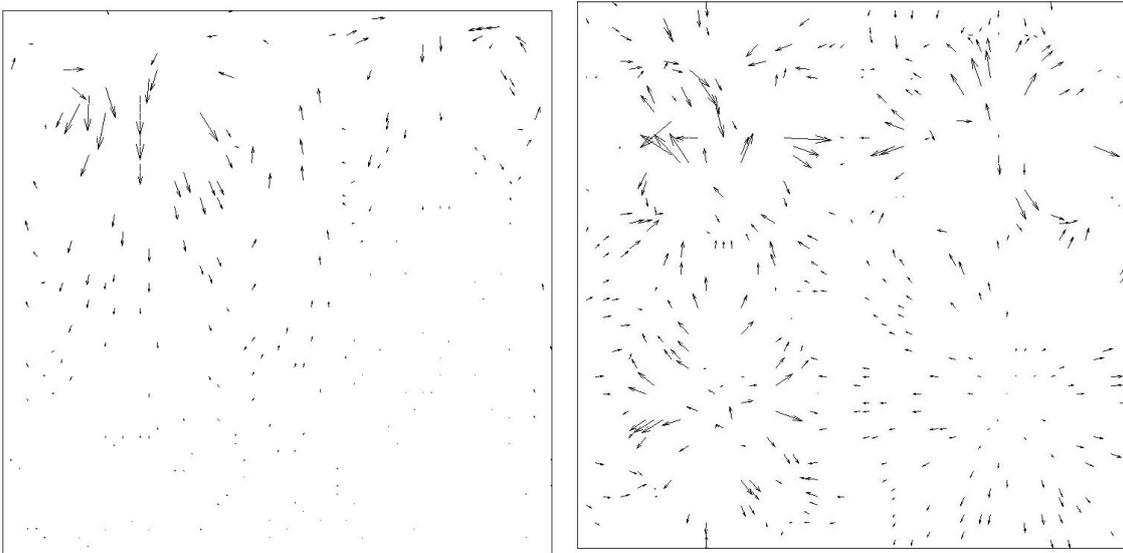


Figure 7.33: Vertical (left) and horizontal (right) cross-sections of the velocity developing in a four-cell convection. The sparsity in these input velocity fields equals 10%.

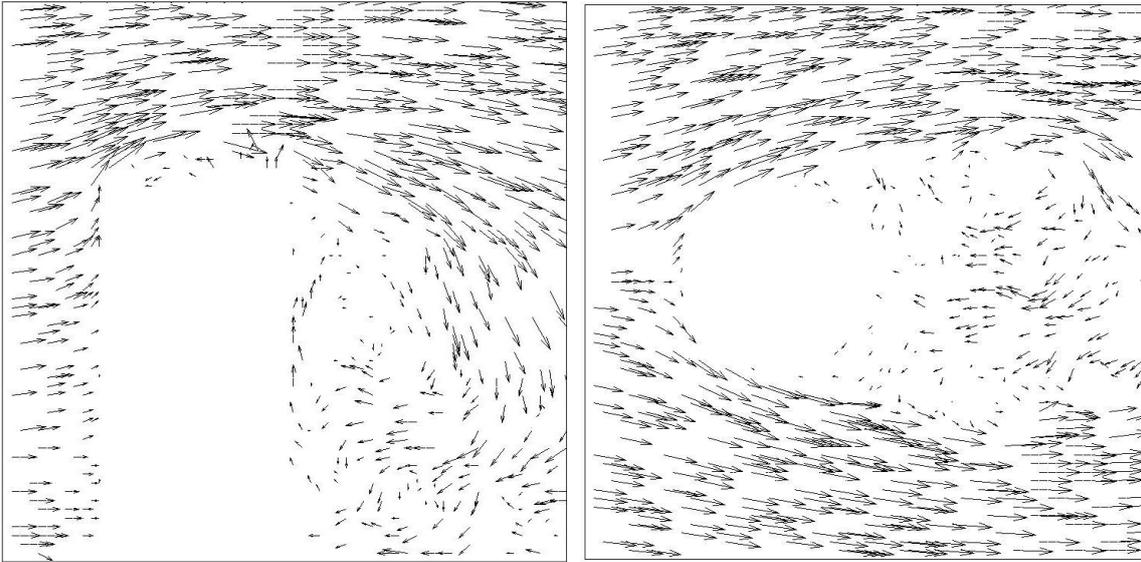
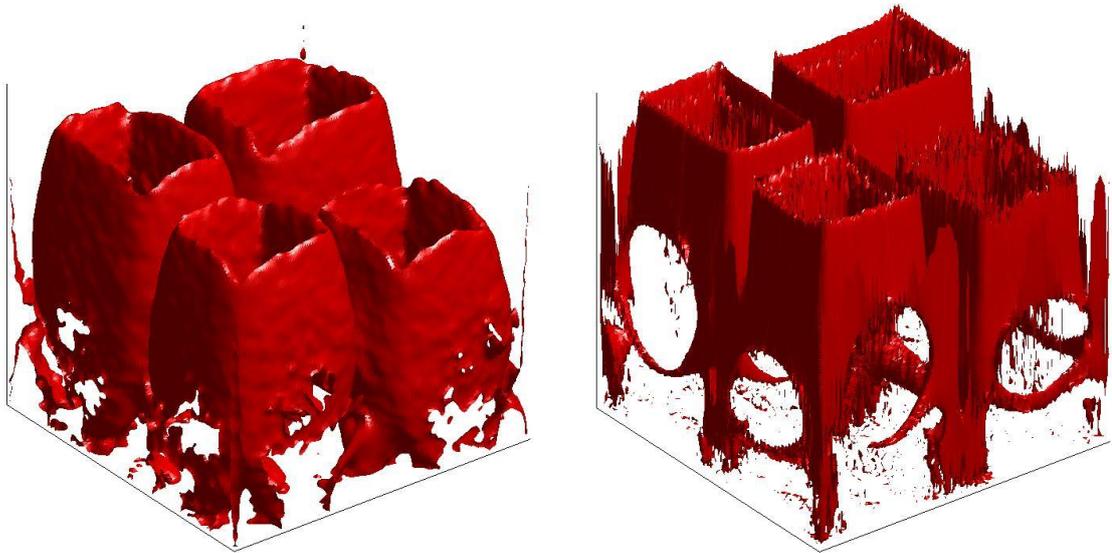


Figure 7.34: Vertical (left) and horizontal (right) cross-sections of the velocity field developing in a turbulent flow around a cylinder. The sparsity of the input velocity fields is equal to 20%.

7.5.2 Results and Comparison

Weak Noise

Figure 7.35 shows the vorticity fields corresponding to both ground-truth and restoration of the four-cell vertical convection. The restored vorticity field was computed using our super-resolution approach. The restored and ground truth velocity vector fields, namely the cross-sections taken in vertical and horizontal planes, are given in Figs. 7.36 and 7.37. Taking into account that only 10% of the data were available for the analysis as an input data set, the quality of the restoration is fairly good enabling inspection of all physical properties of the flow. The error measures in this case were: $SDR = 3.03$, $LDR = 0.80$.



b

Figure 7.35: Instantaneous snapshot of the vorticity iso-surface of a four-cell vertical convection in three dimensions: reconstructed result using super-resolution method (left) and ground truth (right).

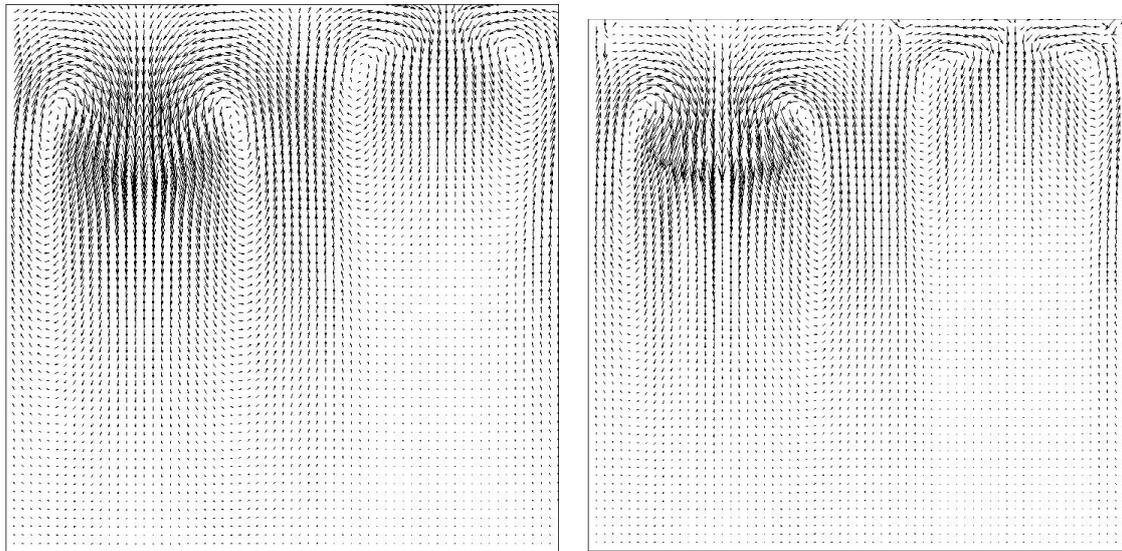


Figure 7.36: Vertical cross-sections of the reconstructed (left) and ground truth (right) velocity fields through the centres of convective cells. These vector fields correspond to the vorticities shown in Fig 7.35. Error measures are as follows: $SDR = 3.03$, $LDR = 0.80$.

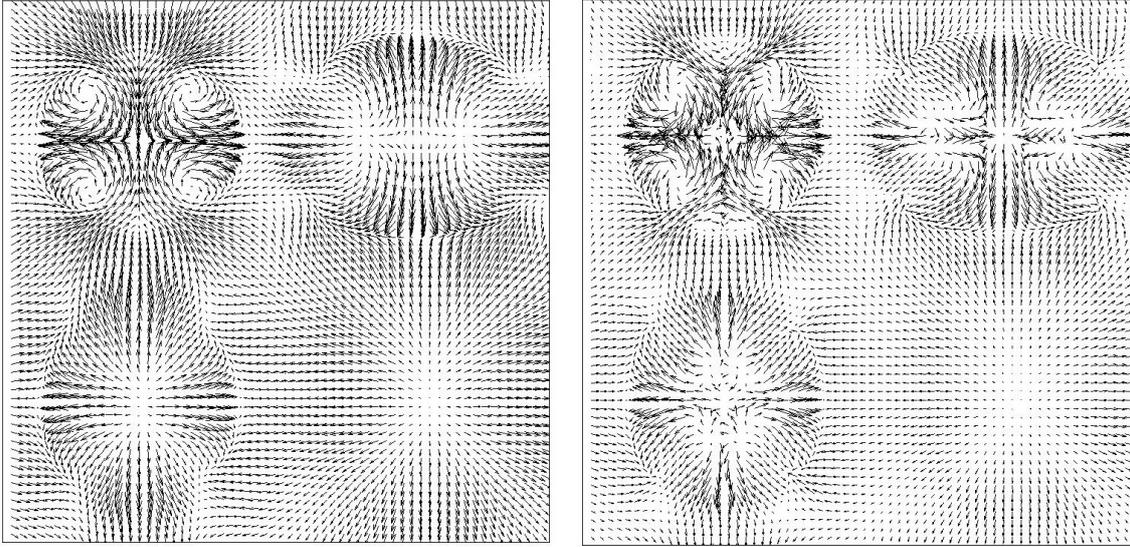


Figure 7.37: Horizontal cross-sections of the reconstructed (left) and ground truth (right) velocity fields through the centres of convective cells. These vector fields correspond to the vorticities shown in Figure 7.35. Error measures are: $SDR = 3.03$, $LDR = 0.80$.

Comparison with cubic spline interpolation. In order to compare the VTE-based approach with the state-of-the-art methods of image processing, i.e. cubic spline interpolation and resampling ([90, 91]), we reconstructed the second data set using both methods. The interpolation result looks quite similar to ground truth (see Figs. 7.38 and 7.39, left and right panel). However, closer inspection and comparison with ground truth flow (Figs. 7.38 and 7.39, center) reveal that spline interpolation creates some *nonphysical* flow structure: the resulting flow contains artificial “waves” that appear in all parts of the vector field. They have appeared even in the regions where the flow is supposed to be almost constant (Fig. 7.40, upper section). Their propagation along the current is independent on distance, which contradicts the turbulence theory [50]. Moreover, some tiny parts of the flow were reconstructed in a wrong way. As a result, the corresponding vorticity field looks very poor (Fig. 7.38). The corresponding error measures, $LDR = 0.65$ and $SDR = 1.91$, confirm this fact.

As distinct from that, the super-resolution approach conforms hydrodynamic principles and does not produce any nonphysical structures (Fig. 7.40, bottom). The resulting vector field looks like a smoothed version of the ground truth. At the same time the main flow structures are clearly recovered. Small-scale parts of the flow have been also restored with satisfactory accuracy, so that the vorticities of the reconstructed and ground truth flow are similar (Fig. 7.38). The corresponding error measures are: $LDR = 0.94$, $SDR = 8.0$.

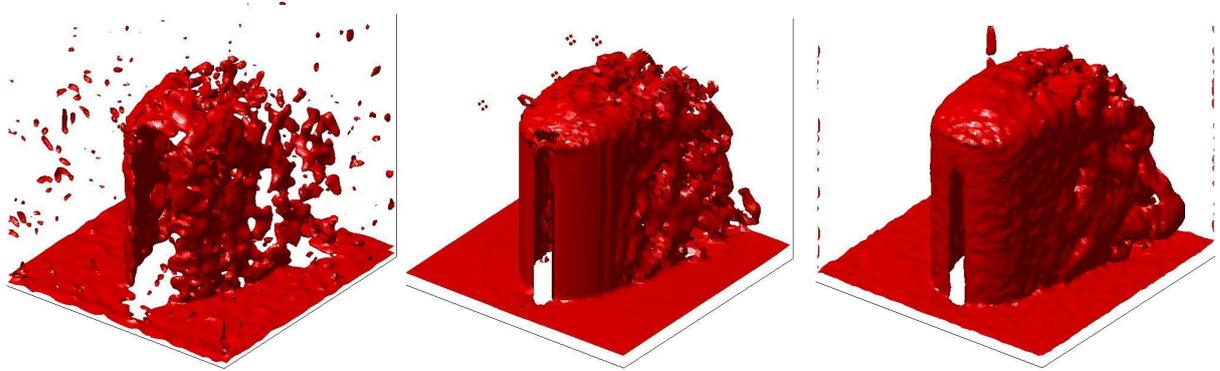


Figure 7.38: Instantaneous vorticity snapshots: restored by cubic spline interpolation (left), ground truth (center), and using the super-resolution approach (right).

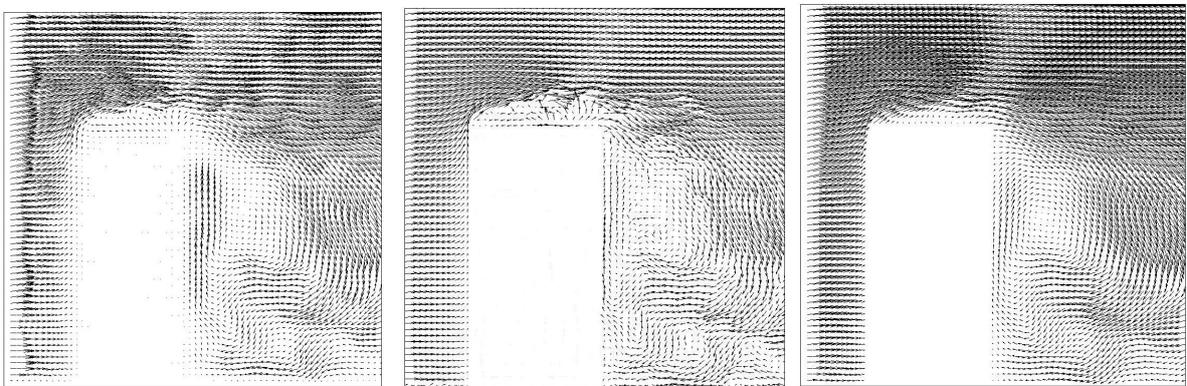


Figure 7.39: Three vertical cross-sections of the velocity fields through the center of the flow around a cylinder: cubic spline interpolation (left), ground truth (center), reconstruction by the super-resolution approach (right). Nonphysical structure is created on the left picture which is not observed to the right (see also Fig. 7.40).

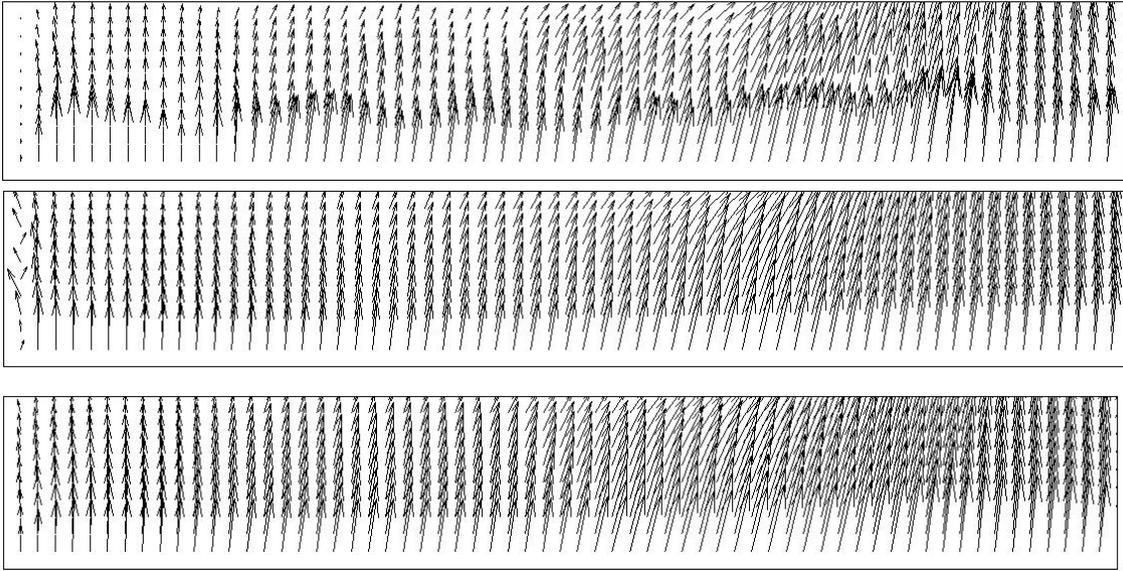


Figure 7.40: Close-up views of the steady velocity vector fields in front of the cylinder (Fig. 7.39), rotated by 90 degrees for convenience. **Top:** Cubic spline reconstruction; **middle:** ground truth; **bottom:** super-resolution approach reconstruction. Nonphysical “waves” dominate in the restored field shown in the top panel and in Fig. 7.39, left panel.

Strong Noise

The case of a high-level noise is studied in this section. The vector fields considered above (see Fig. 7.33 and Fig. 7.34) are taken here as a test bed. The sparsity in data is the same as in the previous examples: 90% for the deep-water convection, and 80% for the flow over cylinder. The SNR ratio for both vector fields was chosen equal to 0.4.

Note, that it is quite unlikely to meet such a high level of noise in laboratory experiments. However, in the present case the experiments with such intensity of noise can give a clear answer to the question on the ability of the methods to recover heavily polluted data sets.

Figures 7.41 - 7.42 represent the noisy and sparse input fields of the four-cell convection (Fig. 7.41) and the flow over the cylinder (Fig. 7.42), and Figs. 7.43 and 7.45 give the cross sections of the reconstructed and ground truth flows. As it can be seen from Fig. 7.43, the main structures of the flow have mostly been reconstructed by the method. However, due to the high sparsity and low SNR value, a lot of artifacts have been introduced into the field by the noise. As a result, the vorticity was poorly reconstructed, see Fig. 7.44.

Note that in the case of a flow over cylinder the SNR value was the same. However, both the velocity, and the vorticity vector fields were reconstructed at least with satisfactory quality. Comparing the results given in Figs. 7.43, 7.44 and 7.45, 7.46 with similar results from previous chapter given in Figs. 7.35, 7.36 and 7.38, 7.39 one can conclude that the SNR value is less important for the vector field reconstruction than the level of sparsity.

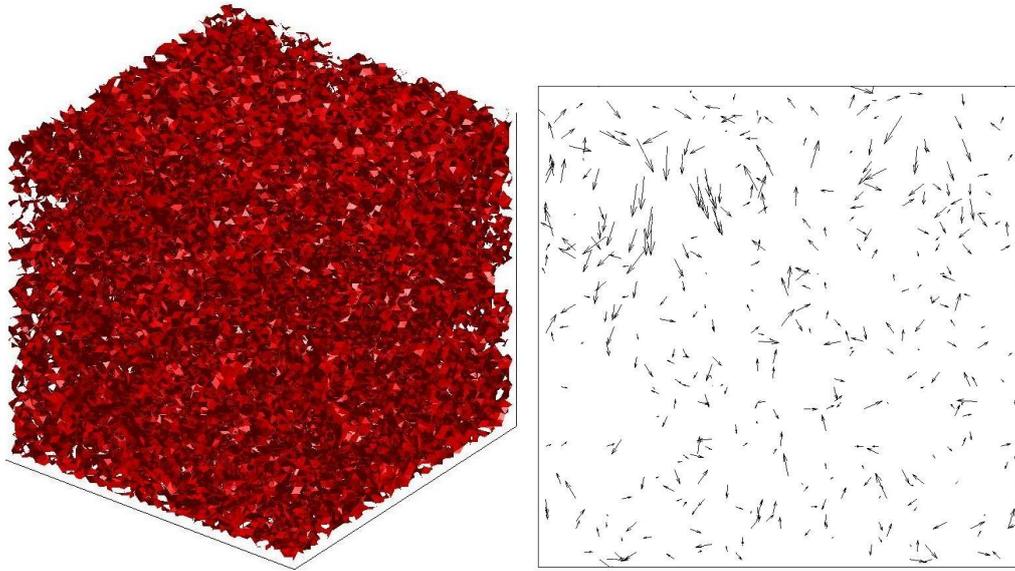


Figure 7.41: The noisy input. **Left:** Instantaneous snapshot of the vorticity iso-surface of a four-cell vertical convection in three dimensions. **Right:** The vertical cross-sections of the velocity vector field.

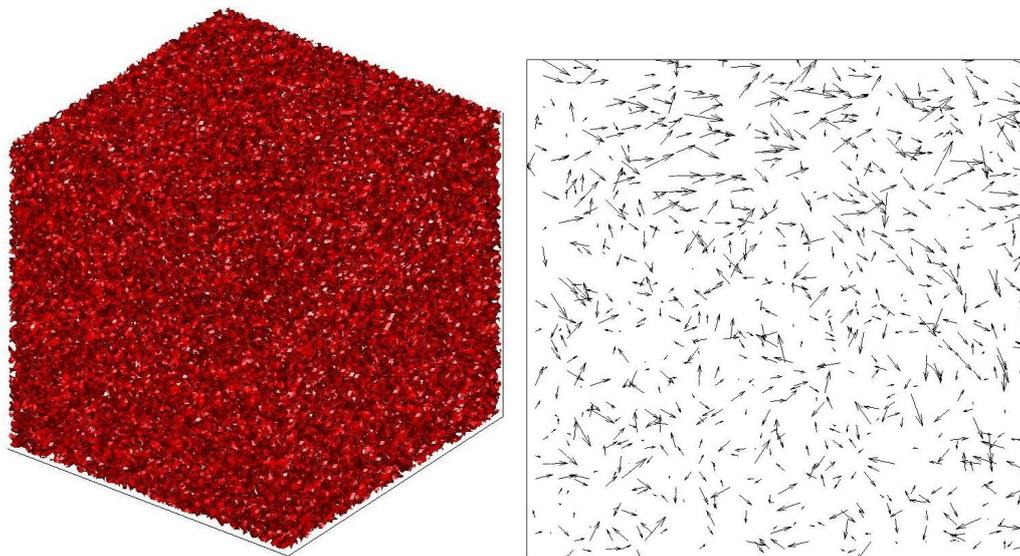


Figure 7.42: The noisy input. **Left:** Instantaneous snapshot of the vorticity iso-surface of a flow over the cylinder in three dimensions. **Right:** The vertical cross-sections of the velocity vector field.

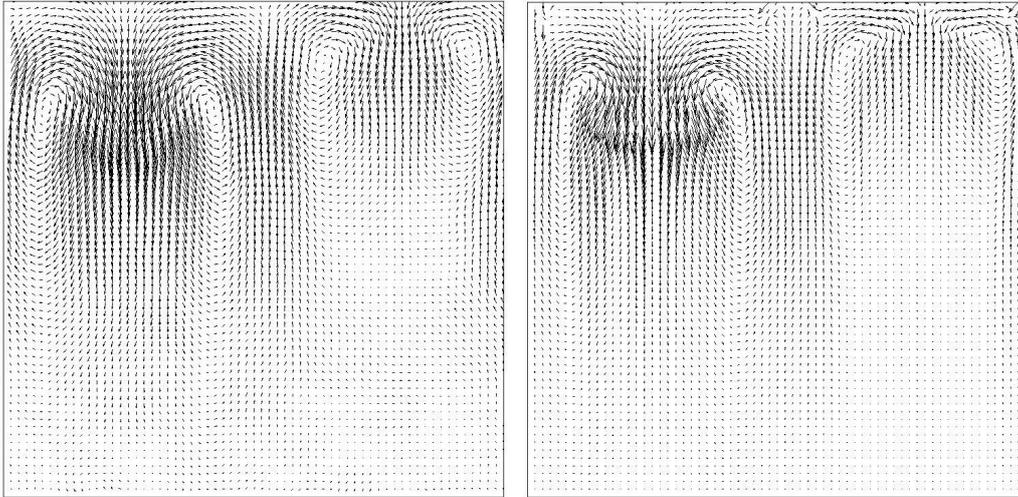


Figure 7.43: Vertical cross-sections of the reconstructed (left) and ground truth (right) velocity fields through the centers of convective cells. These vector fields correspond to the vorticities shown in Figure 7.44. Error measures: $SDR = 3.11$, $LDR = 0.79$.

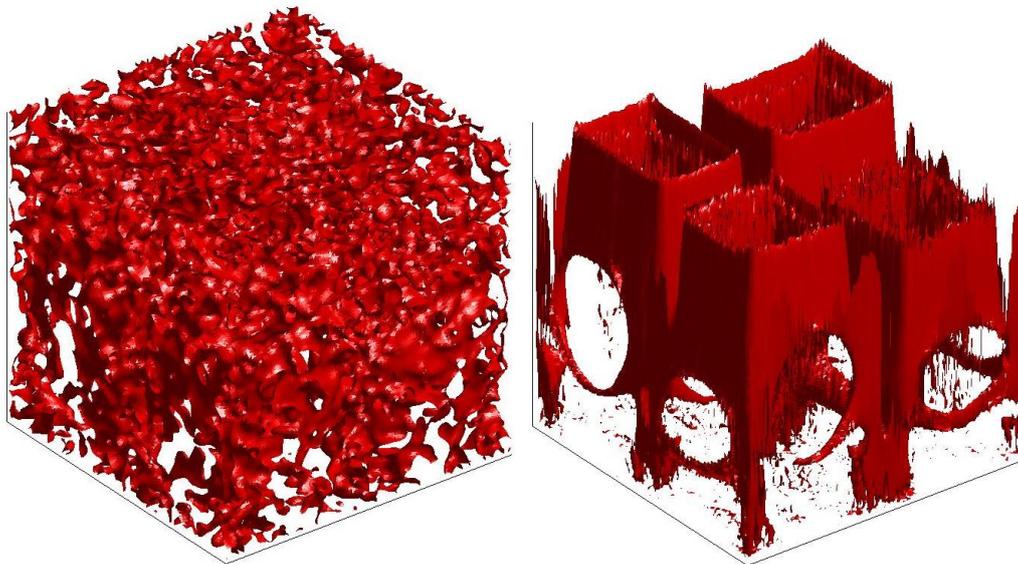


Figure 7.44: Instantaneous snapshot of the vorticity iso-surface of the convective cells in three dimensions: reconstructed result using super-resolution method (left) and ground truth (right).

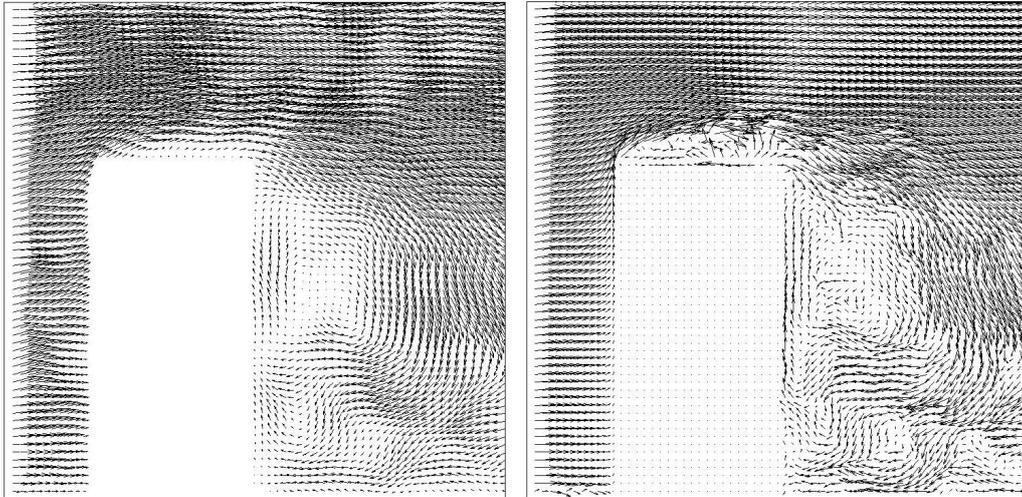


Figure 7.45: Vertical cross-sections of the reconstructed (left) and ground truth (right) velocity fields through the centre of a flow over cylinder. These vector fields correspond to the vorticities shown in Figure 7.46. Error measures: $SDR = 5.63$, $LDR = 0.90$.

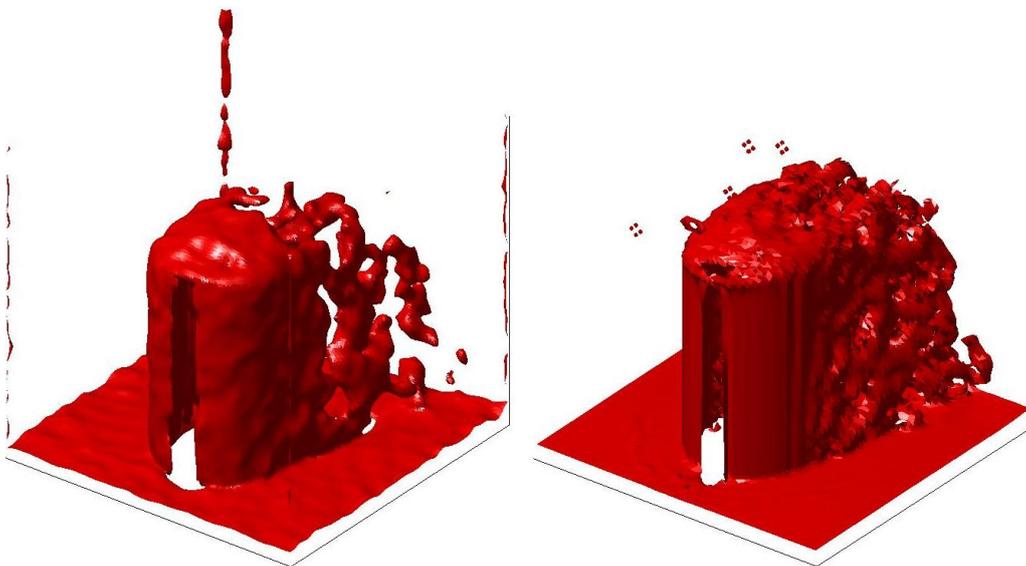


Figure 7.46: Instantaneous snapshot of the vorticity iso-surface of a flow over cylinder in three dimensions: reconstructed result using super-resolution method (left) and ground truth (right).

7.6 Experiments with Real Data

It was interesting to test the developed method by its application to some observational data obtained in real PIV measurements. One of the options was to use observations of a turbulent boundary layer discussed in [81]. The top panel in Fig. 7.47 shows a measured velocity field. Though the flow general structures can be visually recognized, the details are perturbed by the noise.

The restored velocity vector field is depicted in the bottom panel of Fig. 7.47. Comparison of the top and bottom panels reveals that vortices which are hardly recognized in the input data (see regions $[x=12, y=15]-[x=21, y=20]$, $[x=0, y=30]-[x=5, y=40]$) become clearly visible after restoration. For example, the group of outliers in region $[x=19, y=24]-[x=21, y=26]$ is effectively denoised.

Note that the input vector field exhibits abrupt changes and discontinuities which violate the incompressibility constraint (2.3). Our procedure smoothes over the flows in such a way to yield a physically plausible approximation of the input data.

Some remarkable modifications can be seen in the vorticity field after denoising (compare two vorticity iso-surfaces given in two panels of Fig. 7.48). The top panel shows the vorticity iso-surface before denoising. Obviously, no concrete flow structures can be distinguished there. Some of the large scaled vortex tubes can be distinguished in the vorticity iso-surface of the denoised data, however their resolution is so poor that the details are completely lost.

Figure 7.49 shows the same iso-surface after the implementation of the superresolution approach in which the resolution went up by a factor of four. The smoothed vorticity iso-surface tightly stretches around each vortex so that the boundaries of numerous swirls are clearly seen.

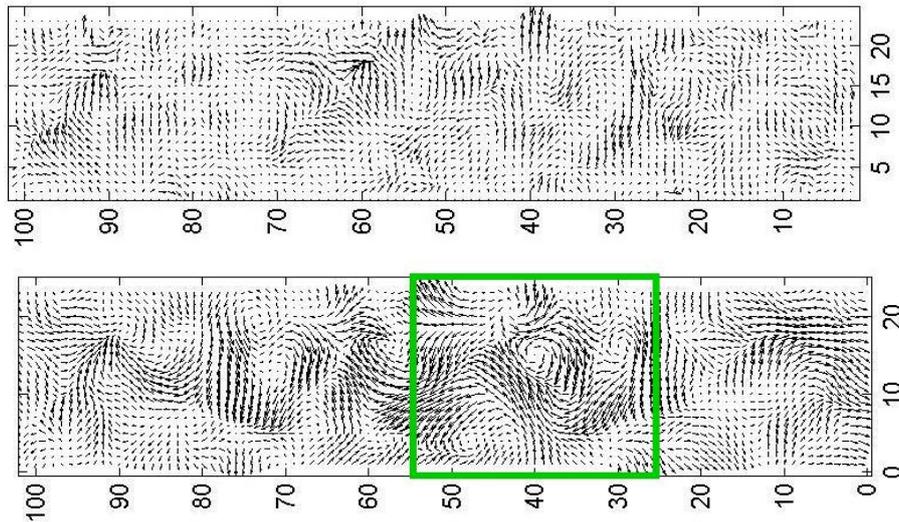


Figure 7.47: A 2D-slice of measured velocity vector field. **Top:** Before denoising. **Bottom:** After denoising.

7 Applications of the VTE Approach and its Comparison with other Methods

Two velocity vector fields given in Fig. 7.50 show the fragment of denoised vector field marked by green rectangular in Fig. 7.47. They represent a standard output (left) and four-time increased resolution field (right) obtained after the implementation of the super-resolution approach. As it can be seen from the figure, the vortex which is not seen in the denoised vector field in the area marked by red rectangular (Fig. 7.50 left panel) appears in the appropriate area after the increase of resolution.

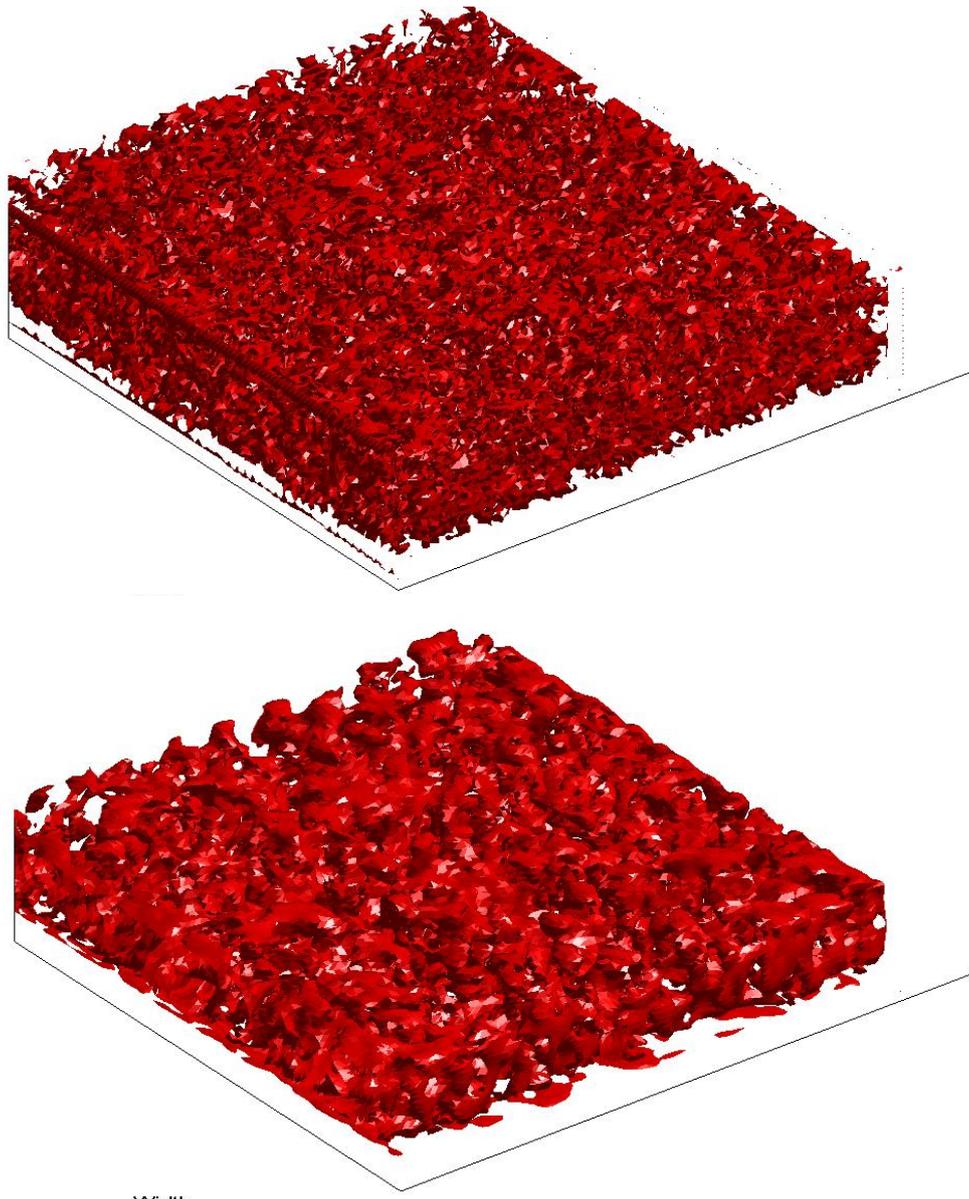


Figure 7.48: Measured vorticity iso-surface. **Top:** Before denoising. **Bottom:** After denoising.

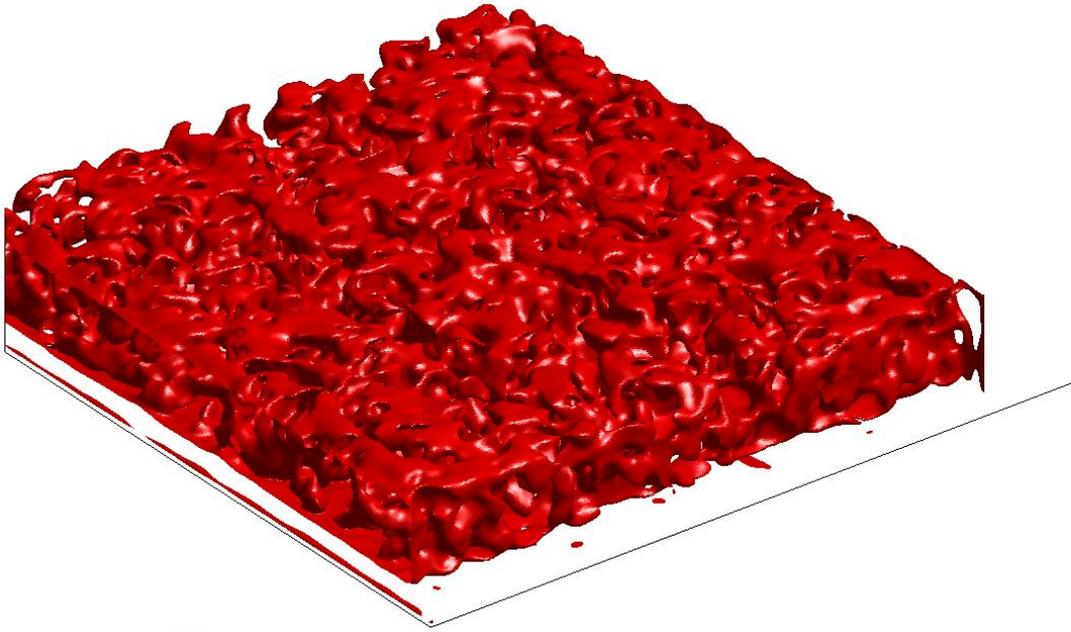


Figure 7.49: The denoised vorticity iso-surface with 4 times increased resolution.

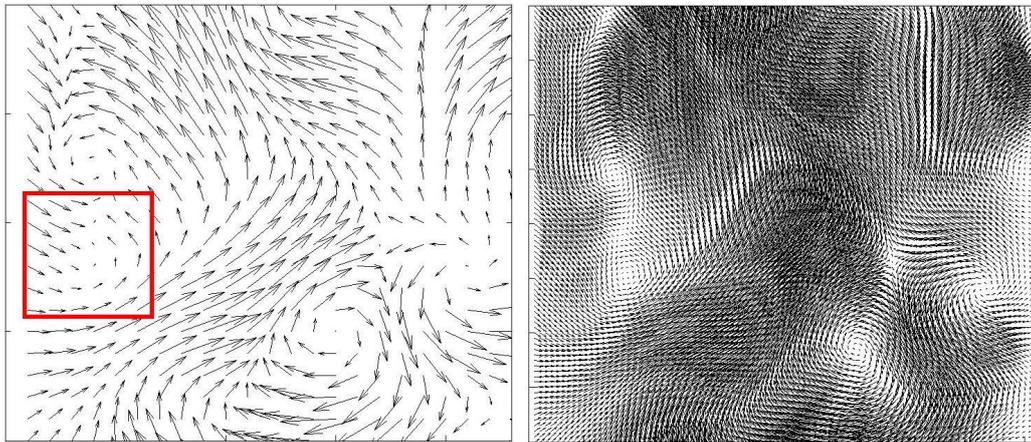


Figure 7.50: **Left:** The denoised velocity vector field. **Right:** The denoised velocity vector field with 4 times increased resolution.

7.7 Summary

Formulating a short summary of this section one can conclude that the developed here VTE-based method of denoising can be successfully applied to a wide variety of fluid flows with various types of noise, corruptions, and data loses. Surprisingly, its comparison with some other noise-based filters which were developed for effective removal of some particular types of noise reveals that the VTE-based procedure copes with these types of noise even better than specially designed for doing that.

The reconstructive ability of the VTE-based approach of working effectively with turbulent fluid flows is not restricted by only the reconstruction of the velocity vector fields. It can also give good estimations of the derivatives of the velocity, as well (vorticity, for instance). Due to its physical nature based on the continuum mechanics equations, the VTE-based method does not introduce any 'non-physical' features that usually appear after application of many other filters.

In comparison with some other hydrodynamically-based reconstructive approaches mentioned in the introduction (see Chap. 1), our method includes a complete system of hydrodynamical equations which make its application suitable for all types of fluid motions. Moreover, the developed method can work well with a single vector field when the concrete boundary conditions are unknown, whereas the advanced reconstructive methods mentioned in Chap. 1 require at least a couple of vector fields as an input data set.

8 Properties of Restoration Approach

8.1 Choice of Parameter Values

Efficient performance of the denoising algorithm depends on the correct choice of all adjustable parameters. In worst case wrong choice of a single parameter can lead to a strong undesirable modification of the whole fluid flow in the worst case. In this chapter we discuss how the choice of the model parameters can affect the resulting fluid flow.

Parameter ν in (4.12). This parameter has a clear physical meaning. It relates to the laminar viscosity of the fluid flow. Depending on the application its value can be increased or decreased with immediate effect on the level of smoothing of the vorticity field ω .

Parameters α in (4.14) and β in (4.16). Parameter α determines the contribution of the second term in (4.14) which represents the VTE at the correction step. Its impact can be demonstrated by a simple example. Let us apply the algorithm two times to the same vector field which does not contain noise. In these experiments, let us chose $\alpha = 1$ and $\alpha = 1000$ for the first and second runs, respectively. If the viscosity parameter has been chosen correctly, then the VTE should be satisfied, and the second term in (4.14) must be equal to zero. As a result, there should be no dependence on the magnitude of α , see Fig. 8.1 (left panel).

As one can see from this figure, the vector field computed with $\alpha = 1$ (black arrows) coincides almost perfectly with the vector field computed with $\alpha = 100$ (red arrows). Next, let us take the value of viscosity 100 times larger than in the previous experiment and conduct two additional runs. The increase of viscosity reduces the Reynolds number 100 times and, as a result, the solution of the VTE should be less turbulent. The results of these two runs are given in the middle panel of Fig. 8.1 where the discrepancy between two vector fields is already seen. The further increase of viscosity results in the reconstruction of output field in such a way that the flow looks almost laminar (right panel of Fig. 8.1).

Parameter β provides a balance between the vector fields obtained at second and third steps. If $\beta < 1$, then the output field is closer to the vector field obtained at second step; it becomes closer to the result of the third step if $\beta > 1$. It was found empirically that the most reasonable choice of α and β in (4.16) is $\alpha = \beta = 1$. These values were used in *all* experiments.

Parameter σ in (4.9a) and (4.9b). Parameter σ describes the cutoff frequency of the lowpass filtering step, as described in Sec. 4.2.2. Although “the best” value depends on the particular data at hand, a broad range of values exists that can lead to a good

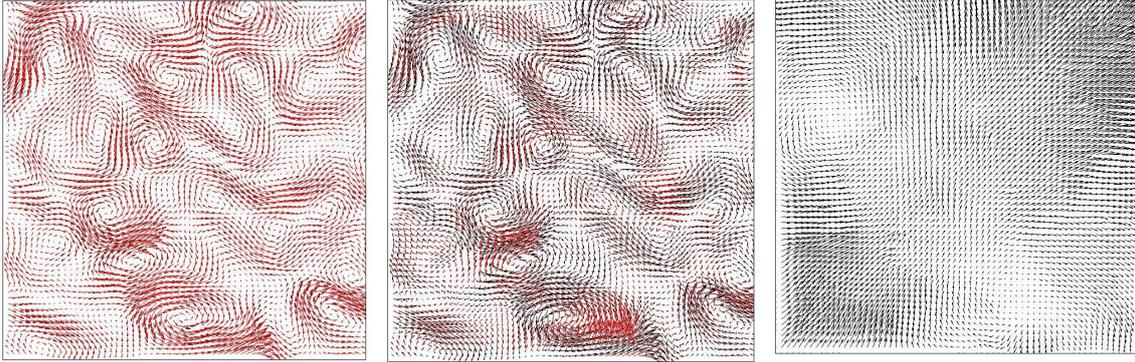


Figure 8.1: Two overlapping noise-free vector fields after application of the VTE-based approach obtained with $\alpha = 1$ (black) and $\alpha = 1000$ (red) for $\nu = 1$ (**left**) and $\nu = 1000$ (**middle**). **Right:** The vector field obtained for $\alpha = 1000$, $\nu = 10000$.

performance of the overall algorithm. To illustrate this point we conducted an experiment using two extreme values of the parameter σ . Figure 8.2, left panel, shows the result for the very large value $\sigma = 20$. Obviously, essential parts of the initial fluid flow pattern (shown in the middle panel) have been blurred by the algorithm. The result for an opposite extreme case with $\sigma \ll 1$ is shown in the right panel of Fig. 8.2. Noise significantly affects the solution, yet all flow structures are visible.

As a result of a large number of experiments with different values of the cutoff frequency parameter, $\sigma = 1.34$ has been taken by default (see explanation in Sec. 4.2.2) which filters out the noise quite effectively (Fig. 7.5, bottom panel) and appears to be a reasonable and robust choice for a broad range of applications.

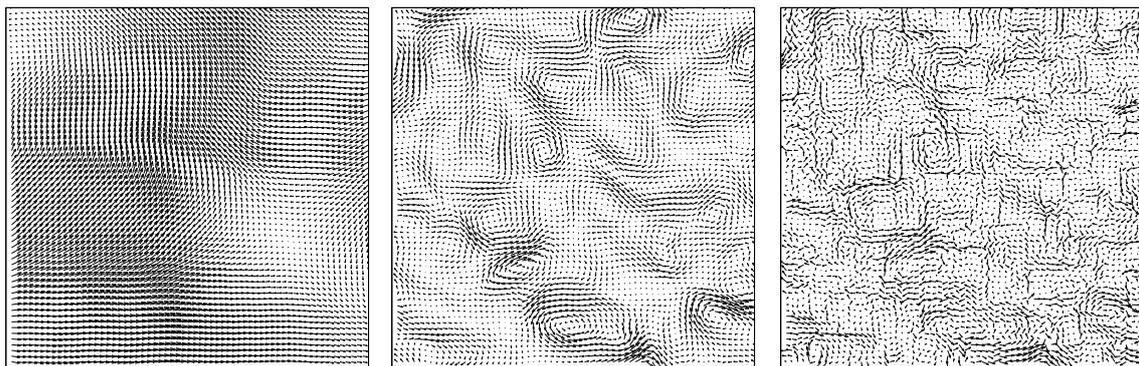


Figure 8.2: **Left:** Output field obtained with $\sigma = 20$. **Middle:** Ground truth. **Right:** Vector field denoised with $\sigma = 0.1$.

8.2 Vorticity Melting

The choice of wrong viscosity value and incorrectness of 'default' boundary conditions (Sec. 2.8) have a weak influence on the output vector field after a single application. However, with the increase of a number of iterations the error can accumulate, and its impact usually becomes remarkable. For instance, the choice of high level of viscosity together with the 'default' boundary conditions (i.e. zero fluxes at the boundaries) will result in smoothing of the input vector field in such a way that after each iteration it becomes closer and closer to the low Reynolds number fluid flow.

To illustrate this effect we chose the ground truth velocity vector field of a flow over cylinder (Fig. 8.3) and apply the reconstructive procedure. The viscosity value ν was set to 1 which is relatively high for such kind of motions. The maximum number of iterations was set to 120. The step with the Gaussian filtering was omitted in order to avoid a smoothing effect from this step and demonstrate the role of the boundary conditions and viscosity values on the output result.

Figure 8.4 shows the result of implementation of the reconstructive approach after 35 iterations. The output vector field looks quite smoothed in comparison with the ground truth, so that tiny details seen in Fig. 8.3 are partially or completely lost in Fig. 8.4. The increase of the number of iterations amplifies the smoothing effect (see Figs. 8.5-8.7), and the modification of the flow vector field is continued until all small details completely vanish. The resultant fluid flow consists of only large flow structures which characterize the flows with low Reynolds numbers (Fig. 8.7). More evidently the smoothing effect is seen in the vorticity vector field (left panel in Figs. 8.4-8.7) which is 'melting' with the increase of iterations.

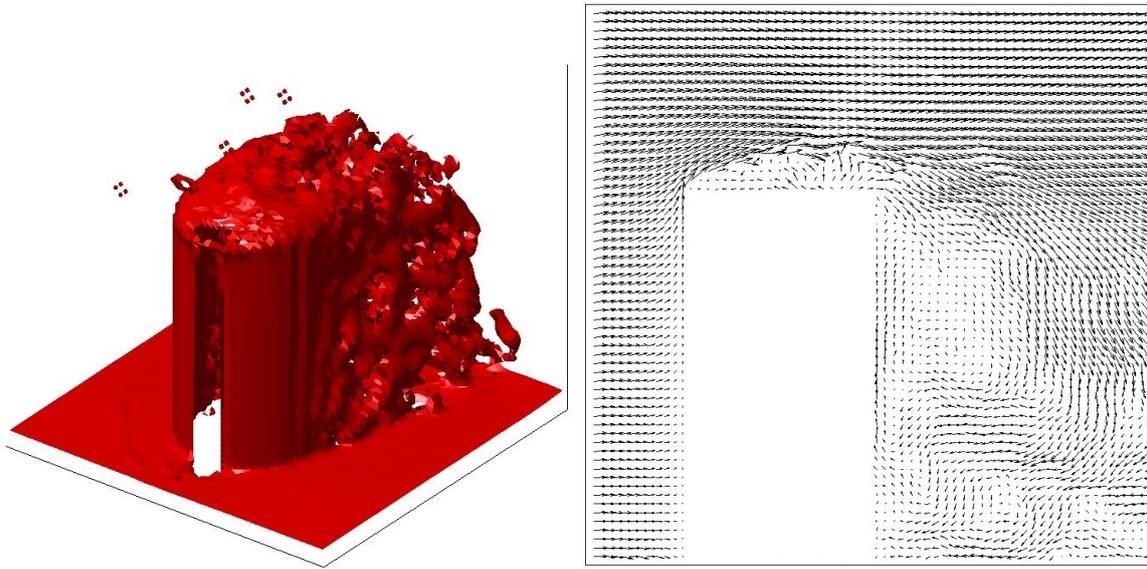


Figure 8.3: The ground truth of 3D vorticity iso-surface (left) and vertical cross-section of velocity (right).

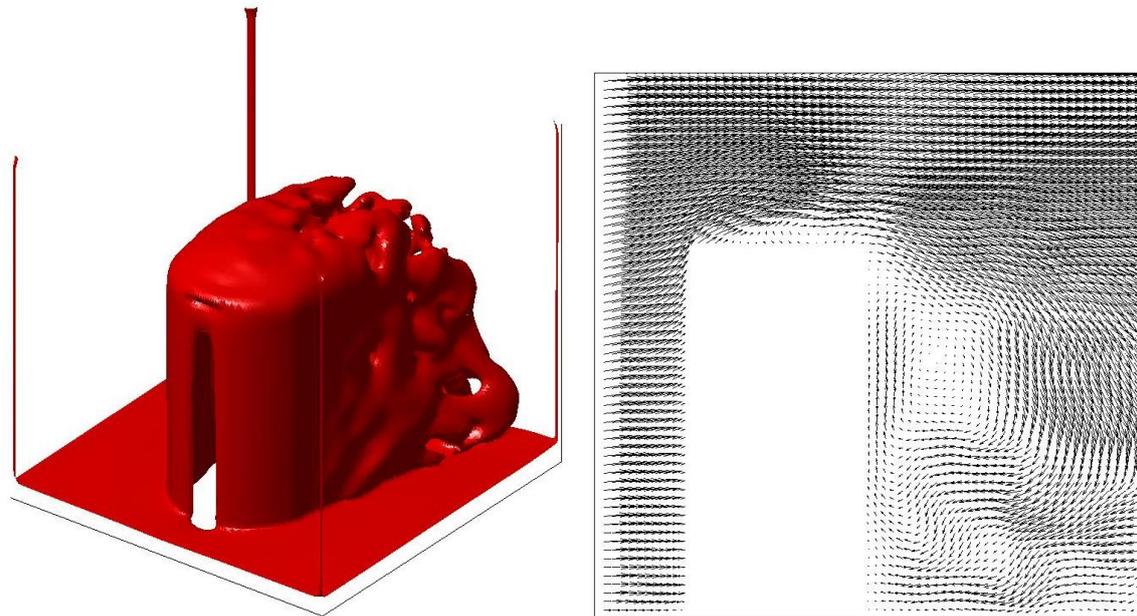


Figure 8.4: The 3D vorticity iso-surface (left) and vertical cross section of velocity (right) after 35 iterations.

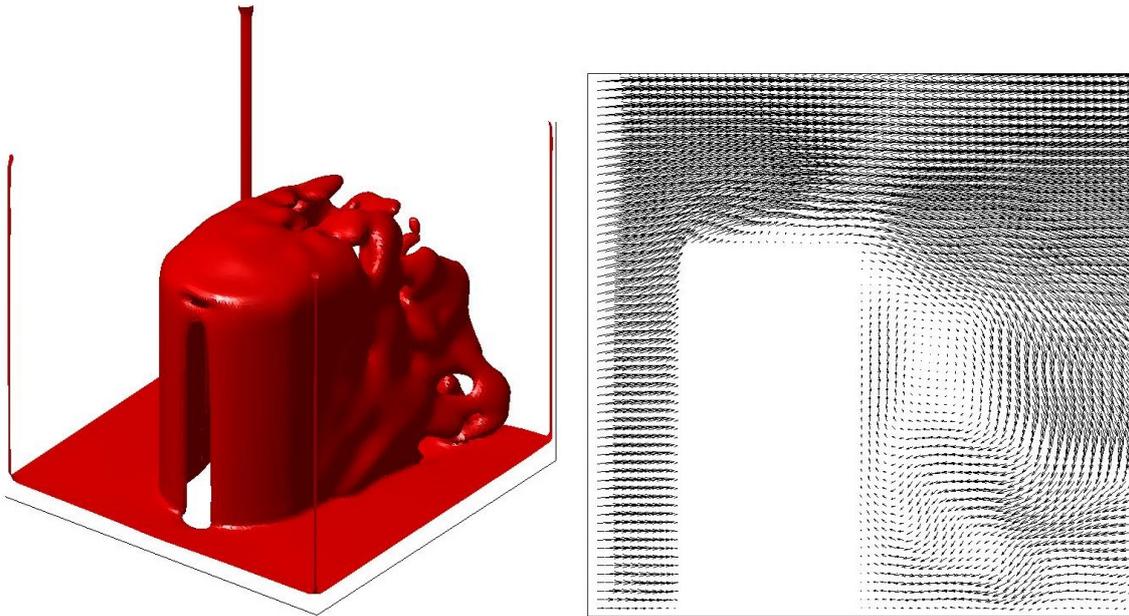


Figure 8.5: The 3D vorticity iso-surface (left) and vertical cross section of velocity (right) after 65 iterations.

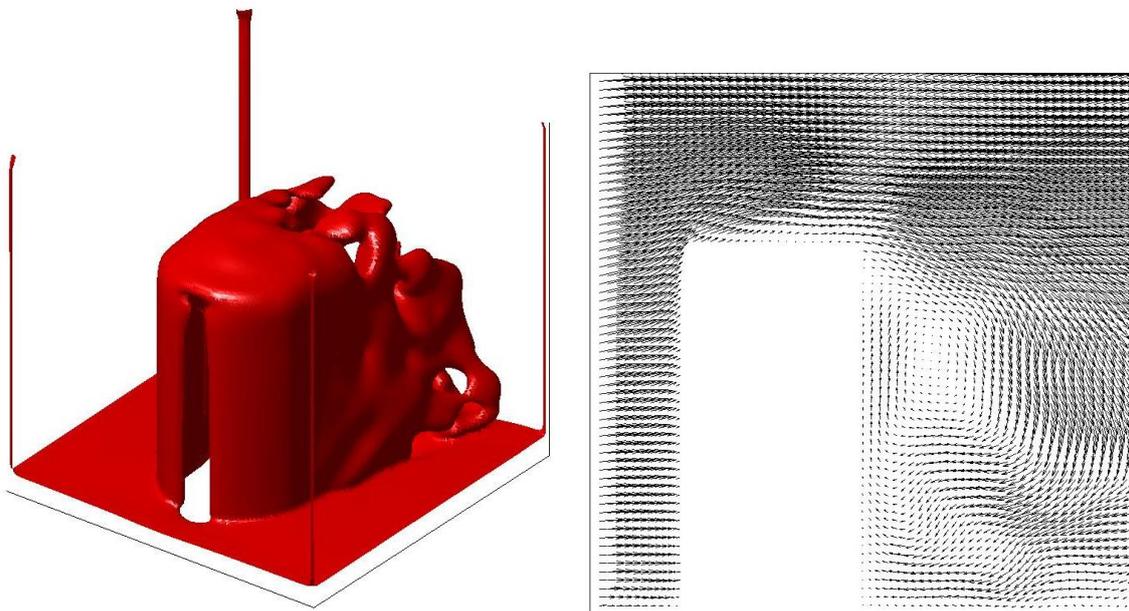


Figure 8.6: The 3D vorticity iso-surface (left) and vertical cross section of velocity (right) after 100 iterations.

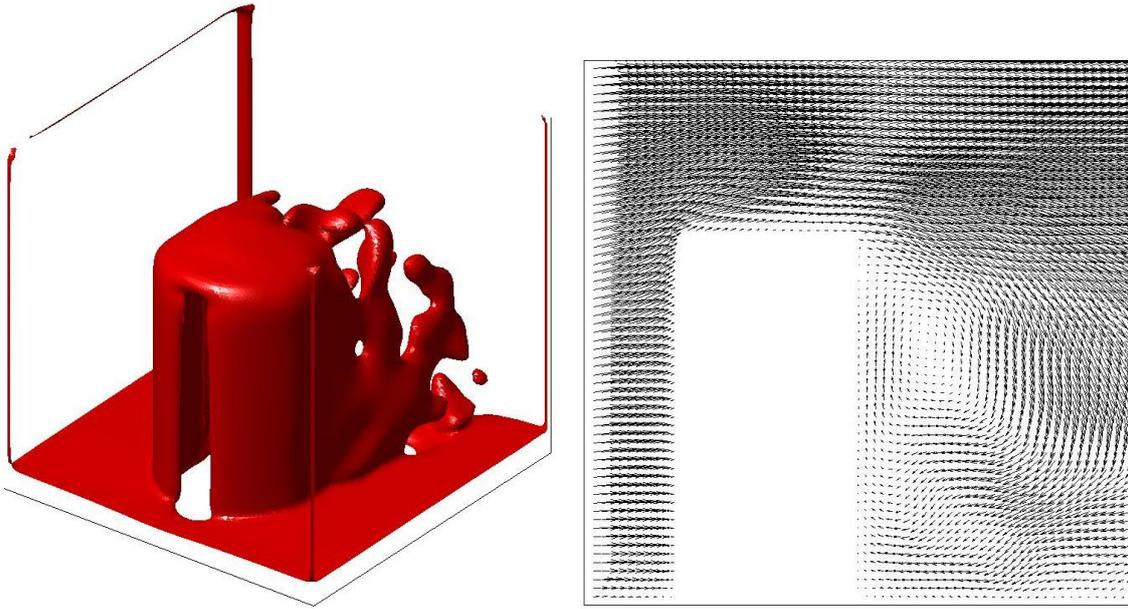


Figure 8.7: The 3D vorticity iso-surface (left) and vertical cross section of velocity (right) after 120 iterations.

Conclusion: the high viscosity values result in the reduction of the Reynolds number of the output vector fields and make the flow structures more laminar while low viscosity increases the Reynolds number and turbulizes the output fields. Thus, the incorrect choice of viscosity changes the flow regime, which results in undesirable artifacts if the viscosity value is too low, or in damping the flow features if the viscosity value is too high.

8.3 Multi Scale

The existed interpolation procedures do not guarantee that the interpolated fields satisfy the hydrodynamical properties of fluid flows. Therefore, the choice of re-scaling methods discussed in Sec. 4.4 was governed only by the intention to find an optimum between simplicity of the method and computational efficiency. Since the chosen interpolation methods are linear, they give only a rough approximation to the ground truth with some losses of details. The super-resolution restoration approach, however, rectify the errors and partially restore the missed flow details. The restoration quality depends on the grid resolution and sparsity of the input data. It becomes better in prediction when the grid resolution and/or the amount of non-sparse data increase (see Figs. 8.8-8.15). This dependence accounts for two factors. The input nonzero fragments of the input fields is a basis for the restoration of sparse regions. The amount of input nonzero vectors increases with the increase of the grid resolution and, as a consequence, makes the reconstruction performance better. On the other hand, the detalization is better on a finer grid, so moving from coarse towards fine grid we increase the detalization (texture) of a flow and make the reconstructed vector field better matching to the ground truth.

Figure 8.14 depicts the relative RMS level (SDR) obtained for 10 reconstructed vector fields at four different grids. As a test example the cylinder flow was taken with 80% of sparsity. The blue line corresponds to the coarsest grid with size $16 \times 16 \times 16$. The reconstruction on this grid is quite poor, so the SDR value is close to 1. The corresponding reconstructed velocity vector field and vorticity iso-surface are shown in Fig. 8.9. The turbulent behavior of a flow is not recognizable there. However, the situation is much better on a twice finer grid (Fig. 8.11): the large vortex behind cylinder is clearly seen and looks similar to that presented in the ground truth (Fig. 8.10) while the corresponding RMS graph is close to 1.7. Doubling of the grids resolution reveals additional details in the reconstructed fluid flow (see Fig. 8.13) and increases its similarity to ground truth (see Fig. 8.12). As a consequence, the SDR level rises to 3.2 and 4 for $64 \times 64 \times 64$ and $128 \times 128 \times 128$ grids, respectively.

Energy rise. Figure 8.15 shows an increase of energy in the velocity vector field after each iteration. Black line corresponds to 95% of sparsity, green line corresponds to 85%, cyan line corresponds to 75%, blue line corresponds to 50%, and the red line is the energy of ground truth. A remarkable fact is that this graph is in consistency with the assumption made in paragraph 'termination criterion' in Sec. 4.4: with the increase of the number of iteration the energy of the reconstructed field asymptotically tends to a some constant value.

8 Properties of Restoration Approach

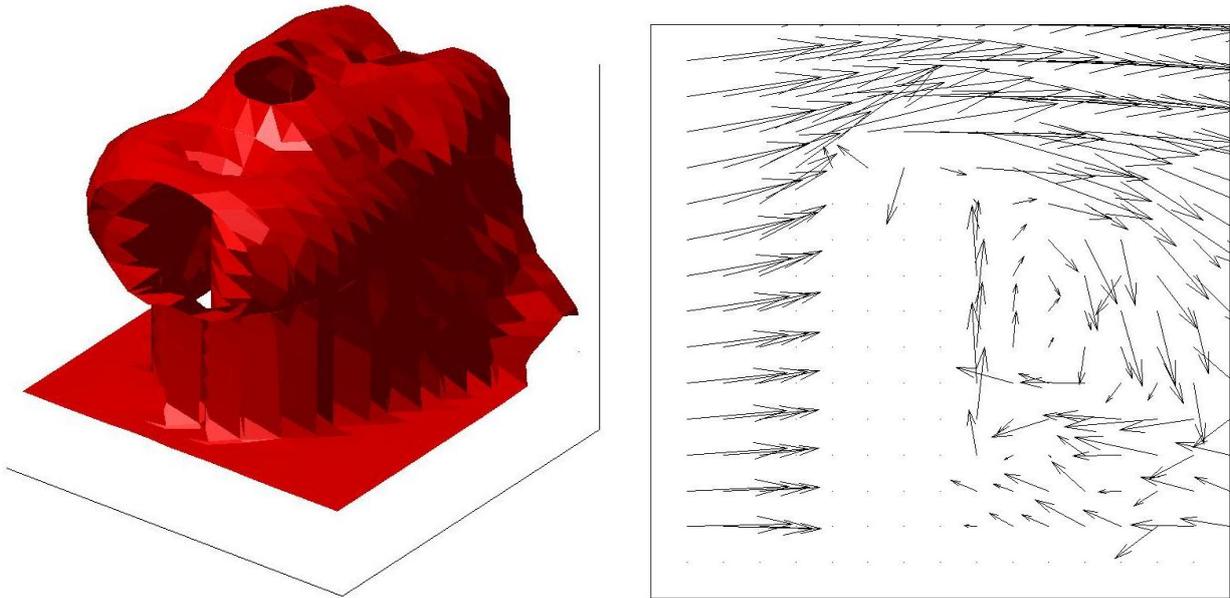


Figure 8.8: The ground truth 3D vorticity iso-surfaces and velocity vertical cross section on 16x16x16 grid.

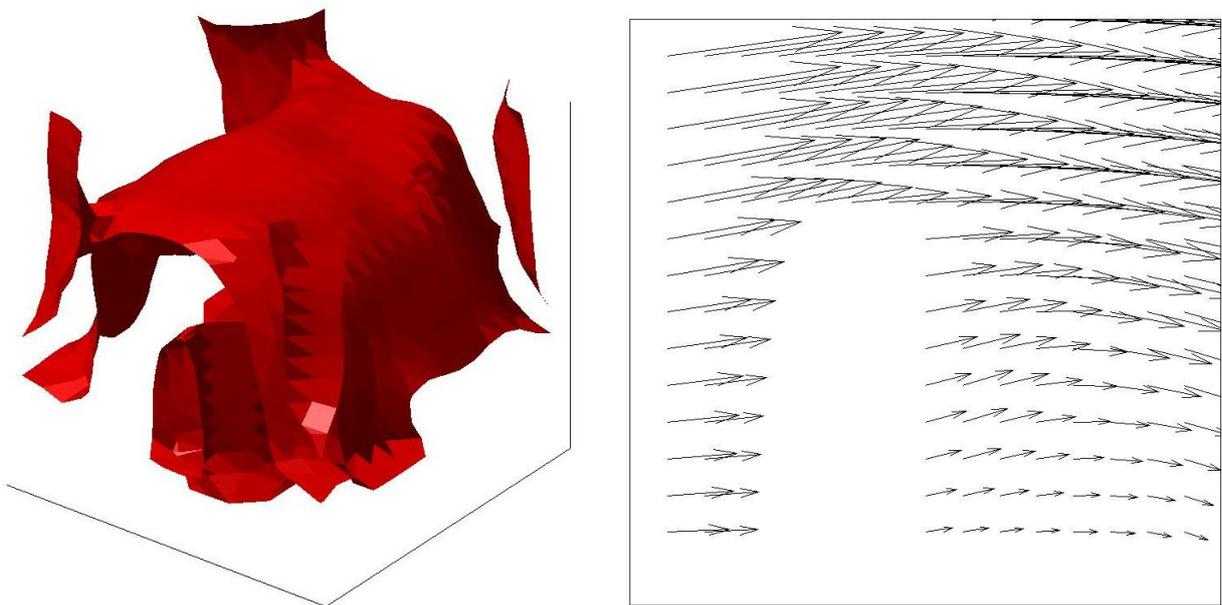


Figure 8.9: The reconstructed 3D vorticity iso-surfaces and velocity vertical cross section on 16x16x16 grid.

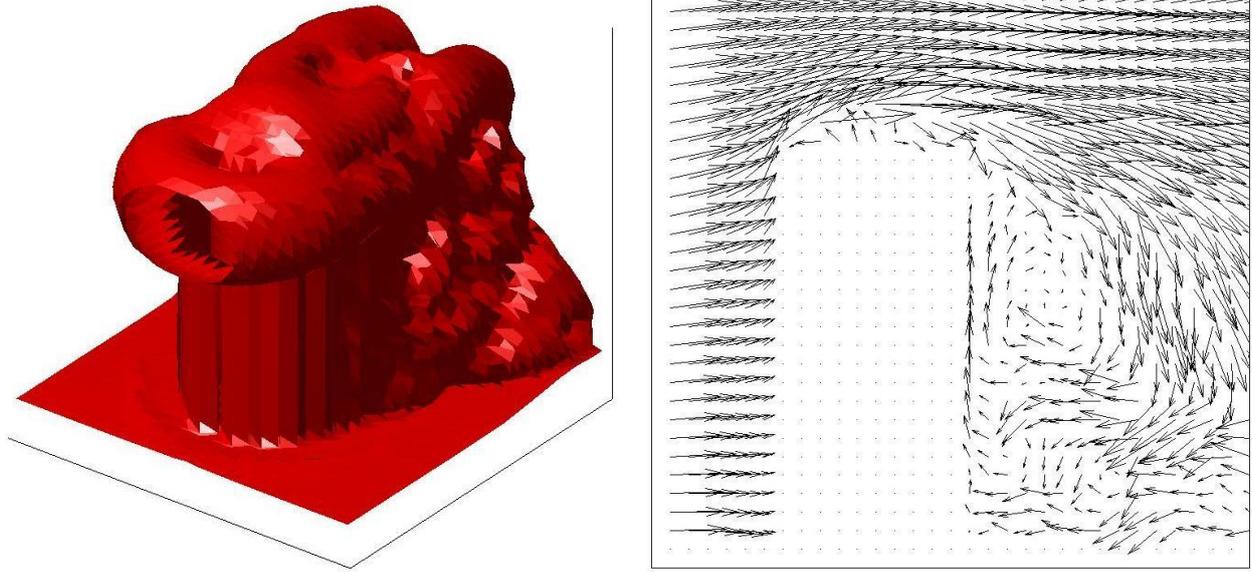


Figure 8.10: The 3D vorticity iso-surfaces and velocity vertical cross section on $32 \times 32 \times 32$ grid.

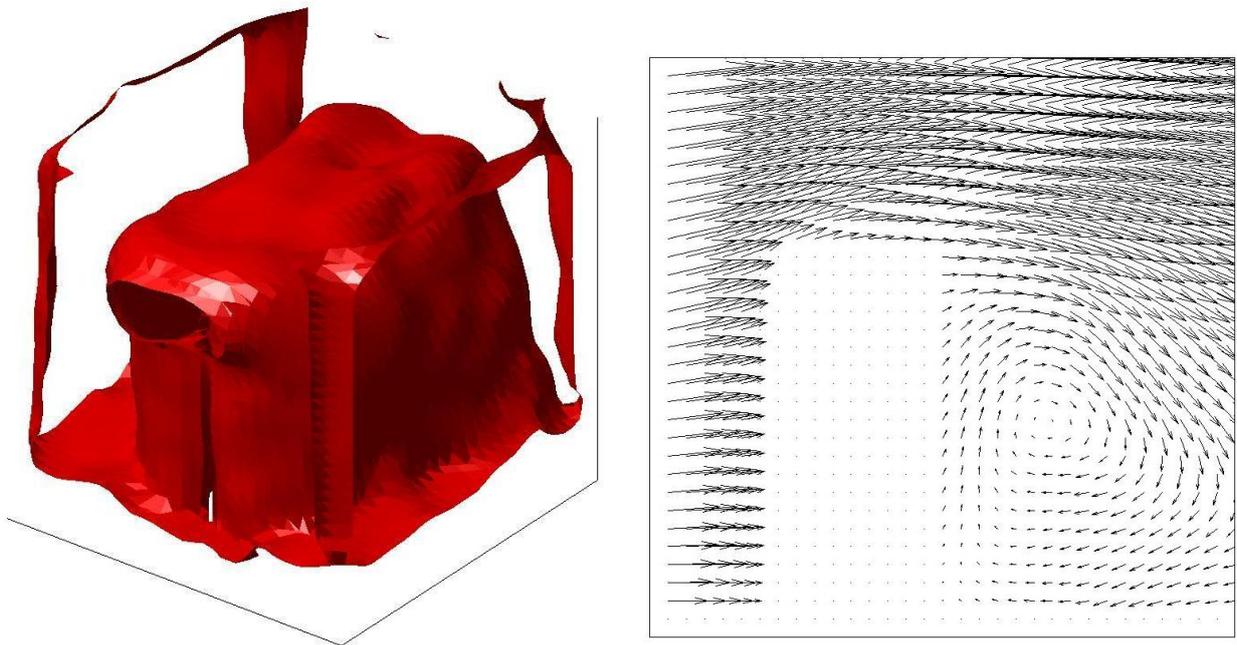


Figure 8.11: The reconstructed 3D vorticity iso-surfaces and velocity vertical cross section on $32 \times 32 \times 32$ grid.

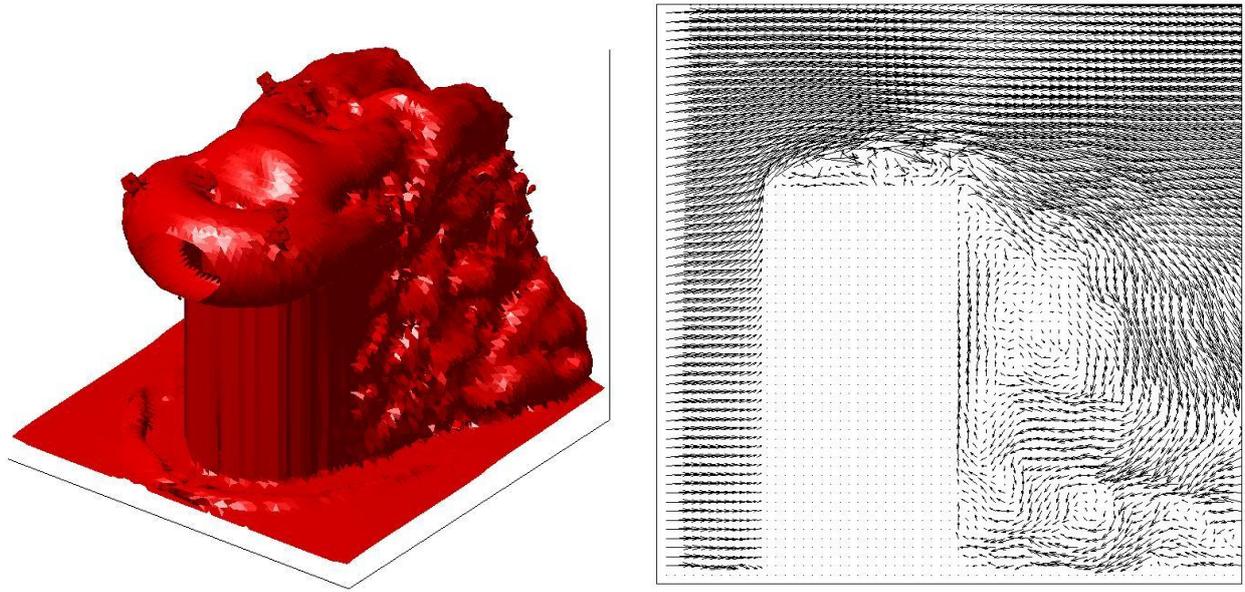


Figure 8.12: The 3D vorticity iso-surfaces and velocity vertical cross section on $64 \times 64 \times 64$ grid.

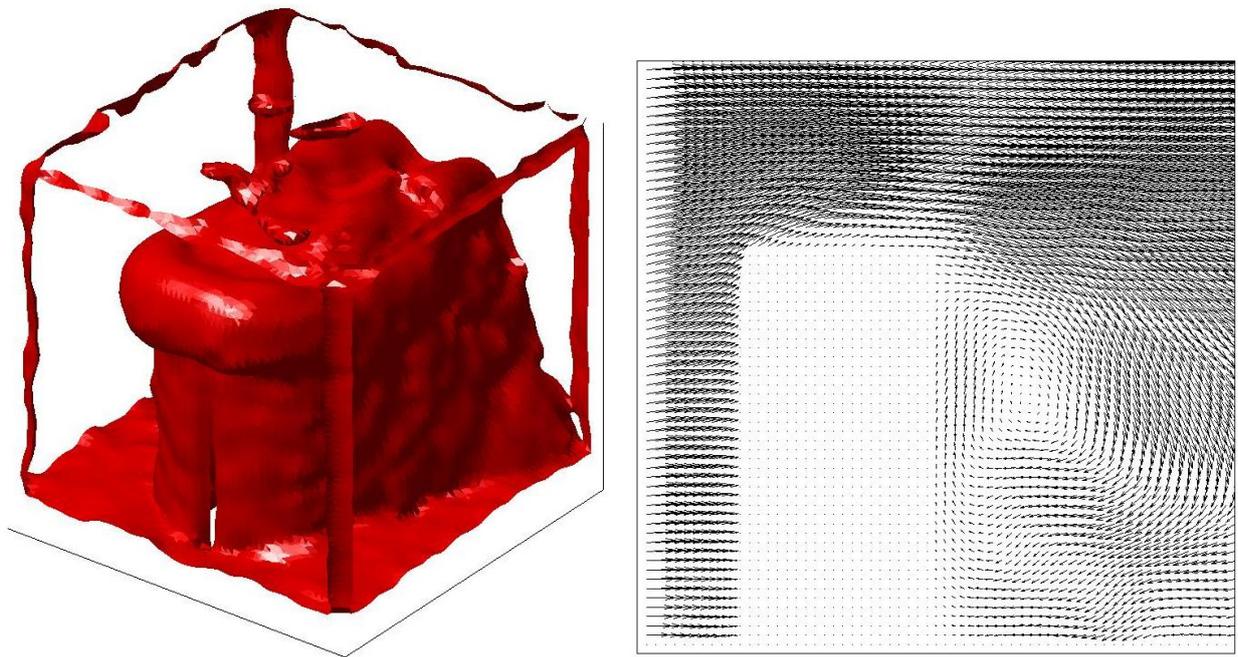


Figure 8.13: The reconstructed 3D vorticity iso-surfaces and velocity vertical cross section on $64 \times 64 \times 64$ grid.

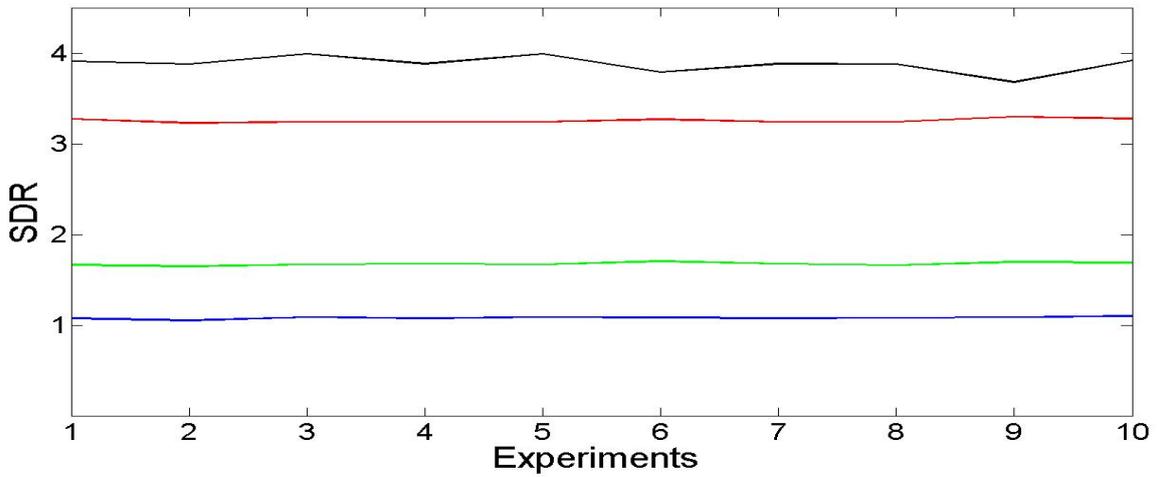


Figure 8.14: The error level (SDR) for the grids with different resolution: blue line corresponds to 16x16x16 grid, green line corresponds to 32x32x32 grid, red line corresponds to 64x64x64 grid, black line corresponds to 128x128x128 grid.

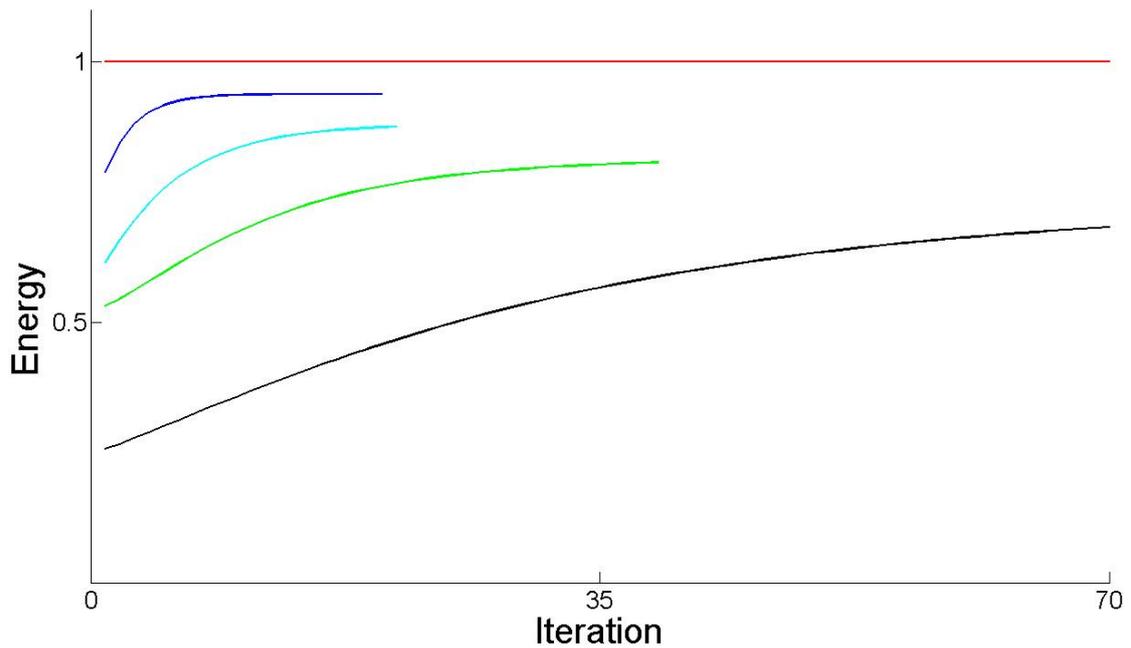


Figure 8.15: The normalised energy graph for different levels of data sparsity: black line corresponds to 95% of sparsity, green line corresponds to 85%, cyan line corresponds to 75%, blue line corresponds to 50%.

9 Conclusions and Future Work

9.1 Conclusions

The present work is devoted to development of a variational approach aimed at solution of the problem of denoising and restoration of fluid flow data sets obtained by image velocimetry traditionally used in experimental fluid dynamics. This approach is presented here in two versions: as an algorithm for reconstruction of noisy or corrupted velocity vector fields, and as a super-resolution method which fills the gaps and increases the resolution of poorly presented initial data sets. Using the hydrodynamical principles, the method restores fluid flow structures in a physically plausible way consistent with the laws of continuum mechanics and increases the resolution of the output fields. Some prominent features of the developed method are as follows:

- As distinct from many standard methods that are based on the direct explicit modelling of the noise, the suggested restoration procedure utilises the background a-priori knowledge of the physical structure of the input vector fields. As a consequence, the method copes with various types of corruptions and noise fairly well and the result of reconstruction is almost independent on their origin. The performance of the method is independent on the type of fluid flow in a wide range of signal-to-noise ratios. This fact was confirmed in a number of numerical experiments with different ground truth data sets and various types of error;
- The developed approach involves four relatively simple computational steps. Even though the method itself exploits the vorticity transport equation (VTE) that is nonlinear and has no clear and well elaborated methods for its solution, the resultant mathematical procedure presented here is reduced to three standard PDE boundary value problems and Gaussian filtering which makes it quite fast and efficient in computational sense;
- The method is computationally efficient regardless of two- or three-dimensional fluid flows are under scrutiny. It has a broad range of applicability to a wide variety of real situations and does not depend on the type of introduced corruptions. The only important thing is that the considered fluid flows should obey the hydrodynamics equations;
- In contrast to many standard similar methods discussed in the introduction, the suggested procedure can be successfully applied to a single instant vector field, i.e. it does not require the information about the fluid flow at several moments of time.

The convergence of the algorithm was checked and proven in Chap. 5 where some qualitative estimations on the error reduction in cases of low noise-to-signal ratios was obtained, as well. It has been mathematically justified that in the case of low noise contamination the result of the VTE-based restoration approach always contains less error than the input field, while the error reduction rate depends on a set of chosen disposable parameters.

The comparison with other denoising methods discussed in Chap. 7 has shown that the presented procedure reconstructs the corrupted vector field better than algorithms commonly used for this purpose. The present version can be used in two ways, i.e. either as a denoising procedure aimed at reduction of the level of a non-physical component of the input signal, or as a resolution restoration technique targeted the fragments of fluid flow with low density or even lost information in order to restore the characteristics of the fluid flow.

The reconstructing procedure is easy in operation, and in the most cases it does not require any special preliminary knowledge on hydrodynamics from the potential users. Moreover, in many practical cases it can successfully reconstruct the velocity vector fields even when the information about the type of fluid flow or boundary conditions is missing. Thus, the developed method can be used as a useful practical tool for the post-processing of the observational data sets obtained in experimental fluid dynamics.

An extension of the VTE-based method to its super-resolution version makes possible the reconstruction of the sparse vector fields obtained from PTV-based measurements. Unlike some other super-resolution procedures (e.g. [85]) the suggested method does not necessary require that the sparse vector field should be noise free or smoothed, as it is demanded in [84]. The algorithm can successfully increase the resolution even if the input vector field contains small level of noise. Moreover, the reconstructed vector field has no limitations on a number of derivations and does not introduce nonphysical structures which is common for the methods heavily based on splines [92].

9.2 Future Work

Combination with TomoPIV. As a potential area for some possible future applications, the present work might be useful for various modifications of the tomographical methods used in 3D-PIV velocimetry [23]. An obvious benefit of such an improvement would be a substantial acceleration of the computational procedure. Really, the conventional TomoPIV implies reconstruction of the particles' positions (particle volumes) using the initial image as an input data set. Since the images are normally quite large (dozens of millions of pixels), the procedure assumes to be quite time consuming. However, the computation time can be substantially reduced if one rescales down the problem and solve it on a coarser grid instead of direct computation of the particle volumes for these given data sets. After the completion of this operation the obtained from these volumes vector fields could be rescaled up using the super-resolution approach. Knowing the approximate velocities of particles in the volume from these vector fields, one can predict their possible appearance at the next time step. Thus, the construction of particle volume at this time step is just simplified to adjustment of the particles positions. Such kind of predictions

are also useful for the reduction of the noise appeared due to account of ghost particles.

Discretization improvement. The developed in Chap. 6 numerical schemes possess quite important properties such as simplicity of practical implementation and computational stability during their application. These two characteristics are of vital importance for modern numerical procedures. Note, however, that there is still enough room for further improvements. One of the options could be implementation of more advanced higher-order numerical schemes: the used so far schemes were of the first-order accuracy. Such a choice was justified at the initial stage of method development until all principal difficulties were fixed, but more accurate numerical procedures can boost the reconstructive properties and give even more realistic results.

Automatic choice of parameters. The algorithm contains several disposable parameters, i.e. α, β, σ , which are supposed to be set by the user on the basis of his understanding of physical characteristics of the studied fluid flow. Their choice is one of the key elements of the method. It is not clear in advance what set of the parameters leads to a better denoising, although multiple application of the method in a wide range of α, β, σ and analysis of the sensitivity of the generated output can give the right answer. In this respect the development of an automatic procedure for the justified selection of the adjustable parameters would be a further modification and improvement of the algorithm which makes the user independent on their choice, and the method itself more robust for data post-processing.

Applications in experimental thermodynamics. Another potential area of the method application could be the modification and correction of the data sets obtained in experimental thermodynamics. Temperature is one of the most important characteristics of many fluid flows. It is well known that the buoyancy force is a key driving factor for such hydrodynamic processes as convection, thermohaline currents, baroclinic wave motions, etc. The information about the volume temperature distribution is quite important for a large number of industrial and scientific applications dealing with the heat transfer processes.

The modern techniques for the temperature measurements can be conventionally classified as direct contact measurements and remote infrared (IR) surface sensing. In the case of direct contact measurements, a special thermo-sensor (commonly based on thermo-pair alloys) is usually placed in the investigated area where it measures an actual in-situ temperature in the point where it is located. In the case of remote sensing, an infrared camera scans the investigated body and records the outgoing IR-radiation. The obtained IR-image is analyzed afterwards in order to reproduce the surface temperature using the Stefan-Boltzmann's law. Note that the direct in-situ measurements give the temperature records only in a restricted number of points where the sensing elements are located, and the temperature distribution in the entire volume is not detected. As an alternative, the remote sensing gives the continuous distribution of the temperature but only at the surface. So, as we can see, both methods are not ideal for the measurements of the entire

volume temperature distributions, and, as a consequence, the task of the definition of continuous volume thermal characteristics becomes quite challenging.

One of the possible solutions could be found if one tries to take advantages of the developed here VTE-based method. Having a distribution of velocity vector field obtained from PIV or PTV experiments, one can solve the inverse problem and compute a possible distribution of temperature in a fluid. A special extension of the VTE-based approach can be a proper method for such kind of problems. The inclusion of the buoyancy terms in the VTE step 4.2.3 and adding to the procedure an extra step with heat transfer equation would make the VTE-based approach quite suitable for the computation of the temperature distributions. For instance, the following scheme can be suggested: let some flow is driven by heating processes. The obtained in PIV or PTV experiments velocity vector field is cleaned from the noise; then the cleaned vector field is substituted into the VTE with buoyancy term. This equation can give a first approximation for the temperature distribution. At the next step this rough approximation can be corrected and specified in such a way to satisfies the heat transfer equation. Finally, the obtained temperature distribution is corrected again to satisfies the VTE. The procedure iterates several times until the obtained temperature and velocity distributions satisfy both equations.

Outline. Here we have summarized some most prominent characteristics of the VTE-based method for fluid flow data post-processing with focus on its obvious advantages in comparison with the traditionally used method. Special attention has been paid to further modifications, improvements and possible extensions of its application to some other areas of experimental fluid dynamics, and we believe that this list is not complete and can be remarkably enlarged.

10 Appendix

10.1 Vector Derivatives

In this section we discuss some formulae used in this work. All expressions are given in the Cartesian system of coordinate system. In two- and three dimensions the corresponding unit vectors are presented as \mathbf{i}, \mathbf{j} and $\mathbf{i}, \mathbf{j}, \mathbf{k}$, respectively.

Gradient: Suppose $f_3 = f(x, y, z)$ and $f_2 = f(x, y)$ are one-time differentiable scalar functions. We define a **3D gradient** operator as follows:

$$\mathbf{grad}(f_3) = \nabla f_3 = \frac{\partial f}{\partial x} \mathbf{i} + \frac{\partial f}{\partial y} \mathbf{j} + \frac{\partial f}{\partial z} \mathbf{k}. \quad (10.1)$$

An appropriate **2D gradient** version is:

$$\mathbf{grad}(f_2) = \nabla f_2 = \frac{\partial f}{\partial x} \mathbf{i} + \frac{\partial f}{\partial y} \mathbf{j}. \quad (10.2)$$

Divergence: Let functions $\mathbf{F}_3 = \mathbf{F}_3(F_x, F_y, F_z)$ and $\mathbf{F}_2 = \mathbf{F}_2(F_x, F_y)$ represent 3D and 2D vector fields, respectively. Suppose their components are one-time differentiable functions. We define a **3D divergence** of the vector field \mathbf{F}_3 as follows:

$$\mathit{div}(\mathbf{F}_3) = (\nabla, \mathbf{F}_3) = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} + \frac{\partial F_z}{\partial z}. \quad (10.3)$$

In two dimensions the **2D divergence** reads:

$$\mathit{div}(\mathbf{F}_2) = (\nabla, \mathbf{F}_2) = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y}. \quad (10.4)$$

Curl: Let $\mathbf{F}_3 = \mathbf{F}_3(F_x, F_y, F_z)$ and $\mathbf{F}_2 = \mathbf{F}_2(F_x, F_y)$ be 3D and 2D vector fields whose components are one-time differentiable functions, and let $f(x, y)$ be a one-time differentiable scalar function. We define the **3D curl** of vector field \mathbf{F}_3 as follows:

$$\nabla \times \mathbf{F}_3 = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ F_x & F_y & F_z \end{vmatrix} = \quad (10.5)$$

$$\left(\frac{\partial F_z}{\partial y} - \frac{\partial F_y}{\partial z} \right) \mathbf{i} + \left(\frac{\partial F_x}{\partial z} - \frac{\partial F_z}{\partial x} \right) \mathbf{j} + \left(\frac{\partial F_y}{\partial x} - \frac{\partial F_x}{\partial y} \right) \mathbf{k}.$$

A **2D curl** works with 2D vector fields as a scalar valid operator,

$$\nabla \times \mathbf{F}_2 = \left(\frac{\partial F_y}{\partial x} - \frac{\partial F_x}{\partial y} \right), \quad (10.6)$$

and with 2D scalar fields as a vector operator, i.e.

$$\nabla \times f = \frac{\partial f}{\partial y} \mathbf{i} - \frac{\partial f}{\partial x} \mathbf{j}. \quad (10.7)$$

Properties: Let \mathbf{F} , \mathbf{Q} and f be three-times differentiable functions defined in a 3D space. The equations presented below express a set of useful relationships between operators *div*, *curl* and *grad* used in this work:

$$\text{div}(\mathbf{curl}(\mathbf{F})) = (\nabla, \nabla \times \mathbf{F}) = 0; \quad (10.8)$$

$$\text{div}(\mathbf{grad}(f)) = \Delta f; \quad (10.9)$$

$$\text{div}(\mathbf{F} + \mathbf{Q}) = \text{div}(\mathbf{F}) + \text{div}(\mathbf{Q}); \quad (10.10)$$

$$\mathbf{curl}(\mathbf{grad}(f)) = (\nabla \times \nabla)f = \mathbf{0}; \quad (10.11)$$

$$\mathbf{curl}(\mathbf{curl}(\mathbf{F})) = \mathbf{grad}(\text{div}(\mathbf{F})) - \Delta \mathbf{F}. \quad (10.12)$$

Equations (10.9)-(10.12) are valid both in 2D, and 3D spaces. Note, that if \mathbf{F} represents a 2D fluid flow velocity vector field, then equation (10.8) is valid, as well. This circumstance reflects the fact that a 2D flow is only a particular case of 3D flows with zero third coordinate.

Let $u = u(x_i, x_j)$ be a two-times differentiable function. It is said that it satisfies the Maxwell's relation if

$$\frac{\partial}{\partial x_j} \left(\frac{\partial u}{\partial x_i} \right) = \frac{\partial}{\partial x_i} \left(\frac{\partial u}{\partial x_j} \right). \quad (10.13)$$

In particular, in a 3D-space similar to (10.13) Maxwell's relations for a three-times differentiable function $\mathbf{F} = \mathbf{F}(F_x, F_y, F_z)$ read:

$$\begin{aligned} \text{div}(\Delta(\mathbf{F})) &= \frac{\partial \Delta(\mathbf{F})}{\partial x} + \frac{\partial \Delta(\mathbf{F})}{\partial y} + \frac{\partial \Delta(\mathbf{F})}{\partial z} = \Delta \frac{\partial F_x}{\partial x} + \Delta \frac{\partial F_y}{\partial y} + \Delta \frac{\partial F_z}{\partial z} = \\ &= \Delta \left(\frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} + \frac{\partial F_z}{\partial z} \right) = \Delta \text{div}(\mathbf{F}). \end{aligned} \quad (10.14)$$

Cross-product for the vorticity in 2D case. Although vorticity is a scalar function in 2D case, the cross-product with other vectors can be defined. Consider a 2D velocity vector field as a particular case of a more general 3D motion with zero third component $\mathbf{v} = (u, v, 0)^T$. Then, according to (10.5) $\boldsymbol{\omega} = \mathbf{curl}(\mathbf{v}) = (0, 0, \partial v/\partial x - \partial u/\partial x)^T = (0, 0, \omega)^T$. Expanding 2D vector \mathbf{F}_2 on three dimensions we can write: $\mathbf{F}_2 = (f_1, f_2, 0)^T$. Multiplying the vorticity by \mathbf{F}_2 we obtain $\mathbf{F}_2 \times \boldsymbol{\omega} = (f_2\omega, -f_1\omega, 0)$. Finally, truncating the third zero component from this product we arrive at the rule for definition of the cross-product for scalar vorticity:

$$\mathbf{F}_2 = (f_1, f_2)^T \times \omega = (f_1, f_2)^T \times \omega = (f_2\omega, -f_1\omega)^T . \quad (10.15)$$

10.2 Quadratic Functionals

In this section we collect the results concerning the minimization of quadratic functionals over a Hilbert space, possibly subjected to linear equality constraints. The content of this chapter was combined from following sources [10, 12, 13, 18, 22]. All theorems will be given here without proof. We formulate first some formal definitions valid in a Hilbert space H .

10.2.1 Unconstrained Quadratic Functionals

Consider a functional

$$J(u) = a(u, u) - \langle f, v \rangle_{\Omega} , \quad v \in H , \quad (10.16)$$

where $a(u, u) : H \times H \rightarrow \mathbb{R}$ is a bilinear form and $f : H \rightarrow \mathbb{R}$ is a linear function. We formulate the conditions when the minimization problem (10.16) has only one solution. Let $\|\cdot\|$ be the norm of H . The H -elliptic continuous bilinear form is defined [12] as follows:

Definition 8. A bilinear form $a : H \times H \rightarrow \mathbb{R}$ is continuous if exists a constant $C \geq 0$ with

$$a(u, v) \leq C\|u\| \cdot \|v\| .$$

Definition 9. A continuous bilinear form $a(\cdot, \cdot)$ is H -elliptic, if

$$a(v, v) \geq C_e\|v\|^2, \quad \forall v \in H .$$

with a constant $C_e > 0$.

Theorem 10.2.1. [18, Lax-Milgram] Let $a : H \times H \rightarrow \mathbb{R}$ be a continuous H -elliptic bilinear form, and let $f : H \rightarrow \mathbb{R}$ be a continuous linear form. Then the following abstract variational problem has one and only one solution:

Find an element u , such that

$$\text{for } u \in H \text{ and } \forall v \in H, \quad a(u, v) = f(v) .$$

10.2.2 Sobolev Spaces

This section contains a brief introduction into Sobolev spaces. Full details can be found in the original source [24, Sec. 5.2]. Let $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)$ be a multi-index such that $|\alpha| \geq 0$, and let D^α be a differential operator. Then the weak derivative of functions can be introduced as follows:

Definition 10 (Weak Derivative). *Let \mathbf{u} and \mathbf{v} be the functions in $L^1(\Omega)$, and Ω be an open subset of \mathbb{R}^d . We say that \mathbf{v} is the α -th weak partial derivative of \mathbf{u} provided the following condition, i.e.*

$$\int_{\Omega} \mathbf{u} D^\alpha \phi dx = (-1)^{|\alpha|} \int_{\Omega} \mathbf{v} \phi dx \quad (10.17)$$

is valid for all test functions $\phi \in C_0^\infty(\Omega)$.

Definition 11 (Adjoint Differential Operator). *Define the adjoint differential operator $D^{*\alpha}$ of a linear differential operator D^α as $D^{*\alpha} = (-1)^{|\alpha|} D^\alpha$.*

Remark 3. *Equation (10.17) is equivalent to the following*

$$\langle \mathbf{u}, D^\alpha \phi \rangle_{\Omega} = \langle D^{*\alpha} \mathbf{u}, \phi \rangle_{\Omega} ,$$

and $\mathbf{v} = (-1)^{|\alpha|} D^{*\alpha} \mathbf{u} = D^\alpha \mathbf{u}$.

Having introduced the concept of weak derivatives, let us define the Sobolev space for vector functions.

Definition 12 (Sobolev space). *The d -dimensional Sobolev vector space*

$$[W^{k,p}(\Omega)]^d$$

consists of all locally summable vector functions $\mathbf{u} : \Omega \rightarrow \mathbb{R}^d$ such that for each multi-index α with $|\alpha| \leq k$, the α -s weak derivative of \mathbf{u} exists and belongs to $[L^p(\Omega)]^d$. Let $\partial\Omega$ denote the boundary of Ω . The Sobolev space consisting of all \mathbf{u} with $\mathbf{u}|_{\partial\Omega} = 0$ will be denoted as $[W_0^{k,p}(\Omega)]^d$.

Remark 4. *If $p = 2$ then $[H^k(\Omega)]^d = [W^{k,2}(\Omega)]^d$ and $[H_0^k(\Omega)]^d = [W_0^{k,2}(\Omega)]^d$ where $k = 0, 1, \dots$.*

The concept of weak forms of PDEs is introduced similarly to the definition of weak derivatives.

Definition 13 (Weak Form). *Given a problem*

$$D^\alpha \mathbf{u} = \mathbf{f} \quad (10.18)$$

where $\mathbf{f} \in [L^1(\Omega)]^d$. *Its weak formulation is defined as follows*

$$\min_{\mathbf{u}} \{ \langle \mathbf{u}, D^{*\alpha} \phi \rangle_{\Omega} - \langle \mathbf{f}, \phi \rangle_{\Omega} \} , \quad (10.19)$$

where $D^{*\alpha}$ is an adjoint differential operator of D^α , and $\phi \in [C_0^\alpha(\Omega)]^d$ is an arbitrary function.

Remark 5. Note that if a problem is formulated **only** in a weak form, then it is sufficient that $\mathbf{u} \in [H^\alpha(\Omega)]^d$.

If (10.18) has some solutions, then they should coincide with the solutions of (10.19). This fact follows from the fundamental lemma of calculus of variations:

Lemma 10.2.2. (The fundamental lemma of calculus of variations.) If some function $M(\mathbf{u})$ is continuous and

$$\int_{\Omega} M(\mathbf{u})\phi dx = 0 \tag{10.20}$$

for $\forall \phi \in C_0^\infty(\Omega)$, then $M(\mathbf{u}) \equiv 0$ within Ω .

Suppose $\mathbf{u} \in C^\alpha(\Omega)$, and let also the following conditions are satisfied: $M(\mathbf{u}) = D^\alpha \mathbf{u} - \mathbf{f}$; $M^*(\phi) = D^{*\alpha} \phi - \mathbf{f}$. Then according to Def. 13 the expression $\min_{\mathbf{u}} \langle M^*(\phi), \mathbf{u} \rangle_\Omega$ is a weak form of $M(\mathbf{u}) = 0$. The rule of integration by parts leads to: $\min_{\mathbf{u}} \langle M^*(\phi), \mathbf{u} \rangle_\Omega = \min_{\mathbf{u}} \langle M(\mathbf{u}), \phi \rangle_\Omega$. If \mathbf{u} is the minimizer, then $\langle M(\mathbf{u}), \phi \rangle_\Omega = 0$. According to the last lemma $M(\mathbf{u}) = 0$. Hence, the solutions of the weak form (10.19) and the strong form (10.18) are the same.

10.2.3 Quadratic Functionals and Constraints

Let H be a Hilbert space of functions defined on Ω , and let X and Y are two closed convex subsets of H . Consider the constrained minimization problem on the set Ω

$$\min_{u \in X} \{J(u)\} = a(u, u) - \langle f, u \rangle_\Omega \quad \text{s.t.} \quad b(u, \mu) = \langle g, \mu \rangle_\Omega, \quad \forall \mu \in Y. \tag{10.21}$$

Definition 14 (Lagrangian). The Lagrangian functional $L : X \times Y$ associated with (10.21) is defined as follows:

$$L(u, p) = J(u) + \left(b(u, p) - \langle g, p \rangle_\Omega \right),$$

where p denotes the Lagrangian variable.

Theorem 10.2.3. [44, Theorem 1.6] If the solution of (10.21) is regular, then there exists an associated Lagrange variable $p^* \in Y$.

The constrained minimization problem (10.21) is equivalent to a dual problem

$$\max_p \min_u \{L(u, p)\} .$$

According to [22], *min* and *max* are interchangeable, i.e.

$$\max_p \min_u \{J(u, p)\} = \min_u \max_p \{J(u, p)\}$$

holds. The latter is known as a saddle point problem [10]. The solution of this problem is a pair (u^*, p^*) which satisfies the *saddle point property*

$$J(u^*, p) \leq J(u^*, p^*) \leq J(u, p^*) \tag{10.22}$$

for all $u \in X$ and $p \in Y$. The unicity of the pair of (u^*, p^*) is proven by the following theorem.

Theorem 10.2.4. [13, Saddle Point Theorem] Let $V := \{v \in H \mid b(v, \mu) = 0, \forall \mu \in Y\}$. The saddle-point problem (10.21) has a unique solution $(u^*, p^*) \in H \times V$ if

1. the bilinear form a is V -elliptic:

$$a(v, v) \geq C \|v\|_V^2, \forall v \in V,$$

where C is some positive constant, and

2. the bilinear form b satisfies the so-called inf-sup (or Babuška-Brezzi) condition

$$\inf_{\mu \in Y} \sup_{v \in H} \frac{b(\mu, v)}{\|\mu\|_Y \|v\|_V} > 0 .$$

Here $\|\cdot\|_V$ and $\|\cdot\|_Y$ denote the norms in spaces V and Y , respectively.

10.3 Theorems

Theorem 10.3.1 (Cauchy - Schwarz inequality). Let f and g are two integrable functions. Then the Cauchy-Schwarz inequality is given by the formula:

$$\langle f, g \rangle_\Omega^2 \leq \langle f, f \rangle_\Omega \langle g, g \rangle_\Omega .$$

Theorem 10.3.2 (Parseval's relation). Let \hat{f} and \hat{g} be the Fourier transforms of f and g , respectively. Then

$$\int_{-\infty}^{\infty} \hat{f}(\phi) \hat{g}(\phi) d\phi = \int_{-\infty}^{\infty} f(x) g(x) dx. \quad (10.23)$$

Theorem 10.3.3 (Shannon-Whittaker sampling theorem). , [79] Let f be bandlimited. Then

$$f(x) = \sum_{k=-\infty}^{\infty} f(kT) \operatorname{sinc} \left(\frac{\pi}{T} (x - kT) \right) .$$

Given ω_c the sampling interval T can be chosen so as to exactly represent $f(x)$ by $\{f(kT)\}$.

Theorem 10.3.4 (Young's inequality [24]). Suppose f is in $L^p(\mathbb{R}^d)$ and g is in $L^q(\mathbb{R}^d)$ and $1 \leq p, q$ then the following inequality holds

$$\|\delta_1 * g\|_r \leq C_{p,q} \|\delta_1\|_q \|g\|_p , \quad \text{where } \frac{1}{l} + \frac{1}{p} = \frac{1}{r} + 1 \quad \text{and} \quad C_{p,q} < 1 .$$

Theorem 10.3.5. (Green's Formula.) Let $\Omega \subset \mathbb{R}^d$ be a simply connected domain with outer normal $\mathbf{n} \in \mathbb{R}^d$, $d = 2, 3$. If $\mathbf{v}, \mathbf{u} \in C^1(\Omega)$, are two vector functions. Then the following expression holds [88]:

$$\int_{\Omega} (\nabla \times \mathbf{u}, \mathbf{v}) dx = \int_{\Omega} (\mathbf{u}, \nabla \times \mathbf{v}) dx - \oint_{\partial\Omega} (\mathbf{u}, \mathbf{n} \times \mathbf{v}) dx . \quad (10.24)$$

In particular, if \mathbf{u} or \mathbf{v} is in $C_0^1(\Omega)$

$$\int_{\Omega} (\nabla \times \mathbf{u}, \mathbf{v}) dx = \int_{\Omega} (\mathbf{u}, \nabla \times \mathbf{v}) dx . \quad (10.25)$$

The rule of differentiation in the Fourier space. Let \hat{f} and f be two functions defined in n -dimensional Fourier and co-ordinate spaces. Then for function \hat{f} from the Fourier space the differentiation reduces to a simple multiplication:

$$\mathcal{F}\left(\frac{\partial}{\partial x_i} f(\mathbf{x})\right)(\phi) = \phi_i \hat{f}(\phi). \quad (10.26)$$

If \mathcal{F} is a Fourier transform, then

$$\mathcal{F}(f(x) * g(x)) = \hat{f}(\phi) \hat{g}(\phi). \quad (10.27)$$

Bibliography

- [1] D.J. Acheson. Elementary Fluid Dynamics. Oxford Applied Mathematics and Computing Science Series. Oxford University Press, 1990.
- [2] R.J. Adrian. Twenty years of particle velocimetry. *Experiments in Fluids*, 39(2):159–169, 2005.
- [3] R.J. Adrian. Particle-imaging techniques for experimental fluid mechanics *Annu. Rev. Fluid Mech.*, 23: 261-304, 1991.
- [4] S.N. Atluri, S. Shen. *The Meshless Local Petrov-Galerkin Method* Tech. Science Press, 2002.
- [5] Y. D. Afanasyev, E. K. Demirov. A variational filtration and interpolation technique for PIV employing fluid dynamical constraints *Experiments in Fluids*, 39: 828-835, 2005.
- [6] R. Barrett, M. Berry, T. F. Chan, et al. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods* SIAM, Philadelphia, 1994.
- [7] F. Becker, B. Wieneke, J. Yuan, C. Schnörr. A Variational Approach to Adaptive Correlation for Motion Estimation in Particle Image Velocimetry In *Pattern Recognition – 30th DAGM Symposium* volume 5096 of *LNCS*, 335–344, Springer, 2008.
- [8] F. Becker, B. Wieneke, J. Yuan, C. Schnörr. Variational Correlation Approach to Flow Measurement with Window Adaption In *14th International Symposium on Applications of Laser Techniques to Fluid Mechanics* 1.1.3, 2008.
- [9] C. Benedicks, D. Tarlet, B. Michaelis, D. Thevenin, B. Wunderlich. Use of Colored Tracers in Gas Flow Experiments for a Lagrangian Flow Analysis with Increased Tracer Density In *Pattern Recognition – 31th DAGM Symposium*, volume 5748 of *LNCS*, pages 392–401. Springer, 2009.
- [10] M. Benzi, G.H. Golub, J. Liesen. Numerical solution of saddle point problems *Acta Numerica* 14: 1–137, Cambridge University Press, 2005.
- [11] L.C. Berselli, T. Iliescu, W.J. Layton. *Mathematics of Large Eddy Simulations of Turbulent Flow* Springer Verlag, Berlin, 2006.
- [12] D. Braess. *Finite elements. Theory, fast solver, and applications in solid mechanics*. Springer, 1997.

Bibliography

- [13] F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods*. Springer, 1991.
- [14] C. Brücker. Digital-particle-image-velocimetry (DPIV) in a scanning light-sheet: 3d starting flow around a short cylinder. *Exp. in Fluids*, 19(4), 1995.
- [15] S. Burgmann, C. Brücker, and W. Schröder. Scanning PIV measurements of a laminar separation bubble. *Exp. in Fluids*, 41(2):319–326, 2006.
- [16] A. Cenedese, G. Querzoli. Lagrangian statistics and transilient matrix measurements by PTV in a convective boundary layer *Meas. Sci. Technol* 8: 1553–1561, 1997.
- [17] A. Chambolle. An algorithm for total variation minimization and applications *Journal of Mathematical Imaging and Vision* 20: 89-97, 2004.
- [18] P.G. Ciarlet. *The Finite Element Method for Elliptic Problems*. North-Holland Publishing Company, 1978.
- [19] Chi-Ho Kwon, C.D. Bertram, S.D. Hall and J.A. Reizes. Particle Tracking in Complex Multi-particle Overlapped Bead Images In *Fourth Australian Conference on Laser Diagnostics in Fluid Mechanics and Combustion* 81–84, 2005.
- [20] T. Coppetti, D. Heitz, G. Arroyo, E. Memin and A. Santa-Cruz. Fluid experimental flow estimation based on an optical-flow scheme. *Exp. in Fluids*. 40: 80-97, 2006.
- [21] R. Dautray and J.-L. Lions. *Mathematical Analysis and Numerical Methods for Science and Technology*, volume 3. Springer, 2000.
- [22] I. Ekeland and R. Temam, R. *Convex Analysis and variational Problems* Studies in Advanced Math., CRC Press, 1992.
- [23] G. Elsinga, F. Scarano, B. Wieneke, and B. van Oudheusden. Tomographic particle image velocimetry. *Experiments in Fluids*, 41(6):933–947, 2006.
- [24] L.C. Evans. *Partial Differential Equations*. American Mathematical Society, 19, 2002.
- [25] G. E. Elsinga. *Tomographic particle image velocimetry and its application to turbulent boundary layers*. PrintPartners Ipskamp B.V., 2008.
- [26] O. Frederich, E. Wassen, F. Thiele. Prediction of the flow around a short wall-mounted cylinder using LES and DES. *J. Numer. Analysis, Industrial and Appl. Mathematics* 3(3-4): 231-247, (2008).
- [27] D.G. Cacuci. On the use of adjoint operators for global optimization and data assimilation, In *Lectures Notes Workshop on Predictability, Observations, and Uncertainties in Geosciences* Florida State University, Tallahassee, 2006.

- [28] H.W. George. *Multidimensional Analysis: Algebras and Systems for Science and Engineering*. Springer-Verlag, 1995.
- [29] A.V. Getling, O. Brausch. Cellular flow patterns and their evolutionary scenarios in three-dimensional Rayleigh-Bénard convection *Physical Review* 67: 313–317, 2003.
- [30] V. Girault and P.-A. Raviart. *Finite Element Methods for Navier-Stokes Equations*. Springer, 1986.
- [31] J.M. Foucaut, J. Carlier, M. Stanislas. Post-processing of PIV records to allow derivative computation. *Proc. 10th Int. Symp. Appl. Laser Tech. Fluid Mech. (Lisbon, Portugal)* 16.4, 2000.
- [32] J.E. Fowler, Li Hua. Wavelet transforms for vector fields using omnidirectionally balanced multiwavelets *IEEE Transactions on Signal Processing* 50(12): 3018 - 3027, 2002.
- [33] D. Hart. Super Resolution by Recursive Local-Correlation. *Journal of Visualisation* 10, 1999.
- [34] K.D. Hinsch. Holographic particle image velocimetry. *Measurement Science and Technology*, 13:61–72, 2002.
- [35] H.W. Haussecker, D.J. Fleet. Computing Optical Flow with Physical Models of Brightness Variation *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(6): 661-673, 2001.
- [36] P. Heas et al. Layered estimation of atmospheric mesoscale dynamics from satellite imagery. *IEEE IEEE Transactions on Geoscience and Remote Sensing*. 45(12), 40874104, 2007.
- [37] P. Heas and E. Memin. 3D motion estimation of atmospheric layers from image sequences. *IEEE IEEE Transactions on Geoscience and Remote Sensing*. In press.
- [38] P. Heas, D. Heitz, and E. Memin. Multiscale regularization based on turbulent kinetic energy decay for PIV estimations with high spatial resolution. *in 8TH International Symposium On Particle Image Velocimetry - PIV09* Melbourne, Victoria, Australia, August 25-28, 2009.
- [39] M. R. Hestenes, E. Stiefel. Methods of Conjugate Gradient for Solving Linear Systems. *J. Res. National Bureau of Standards*, 49(6): 409–436, 1952.
- [40] B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence* 17: 185-203, 1981.
- [41] K. Hoyer, M. Holzner, B. Lüthi, M. Guala, A. Liberzon, and W. Kinzelbach. 3D scanning particle tracking velocimetry. *Experiments in Fluids*, 39(5):923–934, 2005.

Bibliography

- [42] C.R. Hunt and D.J. Carruthers. Rapid Distortion Theory and the Problems of Turbulence. *J. Fluid Mech.* 212: 497–532, 1990.
- [43] K. Hutter, K. Jöhnk. *Continuum methods of Physical Modelling*. Springer, 2003.
- [44] K. Ito, K. Kunisch. *Lagrange Multiplier Approach to Variational Problems and Applications*. SIAM, 2008.
- [45] C. Johnson. *Numerical solution of partial differential equations by the finite element method*. Cambridge University Press. 1987.
- [46] Kazuo Ohmi, Hang-Yu Li. Particle-tracking velocimetry with new algorithms *Meas. Sci. Technol.* 11: 603616, 2000.
- [47] R.D. Keane, R.J. Adrian and Y. Zhang. Super-resolution particle imaging velocimetry *Meas. Sci. Technol.* 6: 754–768, 1995.
- [48] Yu. L. Klimovich. *Turbulent Motion and the Structure of Chaos - A new Approach to the Statistical Theory of Open Systems*. Kluwer Academic Publishers, 1991.
- [49] T. Kobayashi, T. Saga, and S. Segawa. Multipoint velocity measurement for unsteady flow field by digital image processing. *Journal of Visualization* 5:197-202, 1989.
- [50] C. A.N.Kolmogorov. C.R. Acad. Sci. USSR (30):301, 1941.
- [51] Z. Kowalik, T.S Murty. Numerical Modeling of Ocean Dynamics *Advanced Series on Ocean Engineering*, Vol. 5, page 48-49. World Scientific (1993)
- [52] Y.A. Kuznetsov. Efficient iterative solvers for elliptic finite element problems on nonmatching grids *Russian J. Numer. Anal. Math. Modelling* 10: 187-211, 1995.
- [53] L.D. Landau, E.M. Lifschitz. *Fluid Mechanics(Course of theoretical physics.)* Butterworth-Heinemann, 2000.
- [54] A. Lecuona, J. Nogueira and P. A. Rodriguez. Data validation, interpolation and signal to noise increase in iterative PIV methods. In *11th International Symposium Application of Laser Techniques to Fluid Mechanics, Goulbekian Foundation*. 1–13, Lisboa, 2002.
- [55] S.K. Lele. Compact Finite Difference Schemes with Spectral-like Resolution. *Journal Of Computational Physics* 106: 16–42, 1992.
- [56] Lellmann, J. Kappes, J. Yuan, J. Becker, F. Schnörr, C. Convex Multi-Class Image Labeling by Simplex-Constrained Total Variation In *Scale Space and Variational Methods in Computer Vision (SSVM 2009)*. Volume of LNCS(5567), pp 150-162, Springer, 2009.

- [57] H.G. Maas, A. Gruen, and D. Papantoniou. Particle tracking velocimetry in three-dimensional flows. *Experiments in Fluids*, 15(2):133–146, 1993.
- [58] T. Mani, H. Misra, C. Kambhamettu, J.T. Kirby. A robust motion estimation algorithm for PIV. *Meas. Sci. and Tech.* 16: 865-877, 2005.
- [59] J. Marshall, A. Adcroft, C., Hill, L. Perelman, and C. Heisey. A finite-volume, incompressible Navier-Stokes model for studies of the ocean on parallel computers. *J. Geophys. Res.*, 102: 5733–5752, 1997.
- [60] J. Mathieu and J. Scott. An Introduction to Turbulent Flow. Cambridge University Press, 2000.
- [61] A. Melling. Tracer particles and seeding for particle image velocimetry. *Meas. Sci. Technol.* 8: 1406-1416, 1997.
- [62] M.F. Murphy, G.H. Golub, A.J. Wathen. A note on preconditioning for indefinite linear systems. *SIAM J. Sci. Comput.* 21: 1969-1972, 2000.
- [63] K. Ohmi and H.-Y. Li. Particle-tracking velocimetry with new algorithms. *Meas. Sci. Technol.* 11:603-616, 2000.
- [64] N. Papadakis, P. Heas, E. Memin. Image assimilation for motion estimation of atmospheric layers with shallow-water model. In *8th Asian Conference on Computer Vision (ACCV) 2007*.
- [65] S. Petra, A. Schröder, C. Schnörr. 3D Tomography from Few Projections in Experimental Fluid Dynamics. In: *Imaging Measurement Methods for Flow Analysis* 106: 63–72, Springer, 2009.
- [66] S.P. Pudasaini, K. Hutter. *Avalanche Dynamics: Dynamics of Rapid Flows of Dense Granular Avalanches*. Springer, 2007.
- [67] M. Raffel, M. Gharib, O. Ronnenberger, J. Kompenhans. Feasibility study of three-dimensional PIV by correlating images of particles withing parallel light sheet Planes. *Exp. Fluids* 19:69–77, 1995.
- [68] M. Raffel, C.E. Willert, S.T. Wereley, and J. Kompenhans. *Particle Image Velocimetry – A Practical Guide*. Springer, 2007.
- [69] J.L. Rogers and M.F. Schatz. Superlattice Patterns in Vertically Oscillated Rayleigh-Bénard Convection. *Physical Review Letters* (85)20: 4281-4284, 2000.
- [70] L.Rudin, S. Osher, E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D* 60, 259–268, 1992.
- [71] P. Ruhnau, C. Güttler, T. Putze, and C. Schnörr. A variational approach for particle tracking velocimetry. *Measurement Science and Technology*, 16:1449–1458, 2005.

Bibliography

- [72] P. Ruhnau and C. Schnörr. Optical stokes flow estimation: An imaging-based control approach. *Exp. Fluids*, 42:61–78, 2007.
- [73] P. Ruhnau, A. Stahl, and C. Schnörr. On-line variational estimation of dynamical fluid flows with physics-based spatio-temporal regularization. In *Pattern Recognition – 28th DAGM Symposium*, volume 4174 of *LNCS*, pages 444–454. Springer, 2006.
- [74] P. Ruhnau, A. Stahl, and C. Schnörr. Variational estimation of experimental fluid flows with physics-based spatio-temporal regularization. *Measurement Science and Technology*, 18:755–763, 2007.
- [75] Y. Saad, M.H. Schultz. A Generalized Minimal Residual Algorithm For Solving Nonsymmetric Linear Systems. *Siam J. Sci. Stat. Comput.* 7(3): 856–869. 1986.
- [76] P.K. Sahoo, T. Riedel. *Mean Value Theorems and Functional Equations*. World Scientific Publishing Co. 1998.
- [77] F. Scarano M.L. Reithmuller. Advances in Iterative Multigrid methods. *Exp. in Fluids*. 51–60, 2000.
- [78] D-G. Seol, T. Bhaumik, C. Bergmann, S.A. Socolofsky. Particle Image Velocimetry Measurements of the Mean Flow Characteristics in a Bubble Plume. *Journal Of Engineering Mechanics* 665–676, 2007.
- [79] C. E. Shannon. Communication in the presence of noise *Proc.IEEE* 86(2):447–457, 1998.
- [80] A. Stitou, M.L. Riethmuller. Extension of PIV to super resolution using PTV Meas. *Sci. Technol.* 12 : 1398–1403, 2001.
- [81] A. Schroeder, R. Geisler, K. Staack, B. Wieneke, G.E. Elsinga, F. Scarano, A. Henning. Lagrangian and Eulerian views into a turbulent boundary layer flow using time-resolved tomographic PIV. In *14th Int Symp on Applications of Laser Techniques to Fluid Mechanics*, Lisbon, Portugal, 7-10 July, 2008.
- [82] M. Stanislas, J. Westerweel. *Particle Image Velocimetry: Recent Improvements*. Springer, 2003.
- [83] N. Stashchuk, V. Vlasenko, and K. Hutter. Kelvin-Helmholz instability: Numerical modelling of disintegration of basin-scale internal waves in a tank filled with stratified water. *Nonlinear Processes in Geophysics*, 12:955–964, 2005.
- [84] H. Stüer and S. Blaser. Interpolation of Scattered 3D PTV Data to a Regular Grid. *Flow, Turbulence and Combustion*, 64: 215-232, 2000.
- [85] Takehiro Ido, Yuichi Muraib. A recursive interpolation algorithm for particle tracking velocimetry. *Flow Meas. Instr.* 17: 267-275, 2006.

- [86] Taketoshi Okuno, Yasuhiko Sugii, Shigeru Nishio. Image measurement of flow field using physics-based dynamic model. *Meas. Sci. Technol.* 11: 667-676, 2000.
- [87] D. Tarlet, C. Bendicks, R. Bordas, B. Wunderlich, D. Thevenin, B. Michaelis. Coloured Tracer Particles Employed for 3-D Particle Tracking Velocimetry (PTV) in Gas Flows In: *Imaging Measurement Methods for Flow Analysis* 106: 93–102, Springer, 2009.
- [88] A. Toselli, O. Widlung. *Domain Decomposition. Methods - Algorithms and Theory*. Springer, 2005.
- [89] C. Tropea, A.L. Yarin, and J.F. Foss, editors. *Springer Handbook of Experimental Fluid Mechanics*. Springer, 2007.
- [90] M. Unser, A. Aldroubi, M. Eden. B-Spline Signal Processing: Part I–Theory. *IEEE Trans. Signal Proc.*, 41(2): 821–832, 1993.
- [91] M. Unser, A. Aldroubi, M. Eden. B-Spline Signal Processing: Part II–Efficient Design and Applications. *IEEE Trans. Signal Proc.*, 41(2): 834–848, 1993.
- [92] M. Unser. Splines. *IEEE Signal processing magazine* 16(6): 22–38, 1999.
- [93] M. Van Dyke. *An Albumn of Fluid Motion*. The Parabolic Press, Stanford, CA, 1982.
- [94] A. Vlasenko and C. Schnörr. Physically Consistent and Efficient Variational Denoising of Image Fluid Flow Estimates. In *Trans. on Im. Proc. IEEE*, in press.
- [95] A. Vlasenko and C. Schnörr. Physically Consistent Variational Denoising of Image Fluid Flow Estimates. In *Pattern Recognition – 30th DAGM Symposium*, volume 5096 of *LNCS*, pages 406–415. Springer, 2008.
- [96] A. Vlasenko and C. Schnörr. Variational Approaches to Image Fluid Flow Estimates with Physical Priors. In *Nitsche and C. Dobriloff, editors, Imaging Measurement Methods for Flow Analysis*, volume 106 of *NNFM*, pages 247–256. Springer, 2009.
- [97] A. Vlasenko and C. Schnörr. Superresolution and Denoising of 3D Fluid Flow Estimates. In *Pattern Recognition – 31th DAGM Symposium*, volume 5748 of *LNCS*, pages 482–491. Springer, 2009.
- [98] J. Westerweel. *Digital Particle Image Velocimetry. Theory and Application*. Delft University press, 1993.
- [99] J. Westerweel. Efficient detection of spurious vectors in particle image velocimetry data. *Exp. in Fluids* 12, 236–247, 1994.
- [100] J. Westerweel, D. Dabiri, M. Gharib. The effect of a discrete window offset on the accuracy of cross-correlation analysis of digital PIV recordings. *Exp. Fluids* 23, 20–28, 1997.

Bibliography

- [101] J. Westerweel. On velocity gradients in PIV interrogation. *Exp. Fluids*. 44: 831–842, 2007.
- [102] R.P. Wildes, M. Amabile, A.M. Lanzillotto, T.S. Leu. Physically based fluid flow recovery from image sequences. *in CVPR* 969–975, IEEE 1997.
- [103] C. E. Willert and M. Gharib. Digital particle image velocimetry. *Exp. in Fluids* 10, 181–193, 1991.
- [104] L. William, B. van E. Henson, F.S. McCormick. A Multigrid Tutorial. Second edition. SIAM 2000.
- [105] J. Willneff, A. Gruen. A New Spatio-Temporal Matching Algorithm for 3D-Particle Tracking Velocimetry. The 9th of International Symposium on Transport Phenomena and Dynamics of Rotating Machinery. Honolulu, Hawaii, February 10-14, 2002.
- [106] J. Wu et al. Longitudinal vortex structures in a cylinder wake. *Phys. Fluids*. 6(9): 2833–2835, 1994.
- [107] Yuan, J. and Schnörr, C. and Mémin, E. Discrete Orthogonal Decomposition and Variational Fluid Flow. *J. Math. Imag. Vision* 28: 67–80, 2007.
- [108] Yuan, J. and Schnörr, C. and Steidl, G. Simultaneous Optical Flow Estimation and Decomposition. *SIAM J. Scientific Computing* 29(6): 2283–2304, 2007.