

EXTRACTION OF VORTEX STRUCTURES IN 3D FLOW FIELDS

Dissertation

zur Erlangung des akademischen Grades

Doktoringenieur (Dr.-Ing.)

angenommen durch die Fakultät für Informatik
der Otto-von-Guericke-Universität Magdeburg

von Dipl.-Math. Jan Sahner
geboren am 19. März 1977 in Eckernförde

Gutachter:
Prof. Dr. Holger Theisel
Prof. Dr. Geric Scheuermann
Prof. Dr. Ronny Peikert

Magdeburg, den 30. April 2009

Für Uta, Adi, Ferdi und Ani.

Abstract

The analysis of flow datasets becomes more and more challenging due to the sheer data size produced by modern simulations and experimental measurements. Feature extraction is necessary for a significant reduction of data fluid mechanicians are faced with. Among a variety of interesting flow features, vortex features are the most prominent.

This thesis is devoted to the development of algorithms for vortex feature extraction in 3D and 3D time dependent flow fields. It contains advances in two categories: vortex features defined by swirling motion, and vortex features defined by extremal structures of scalar feature indicators. For the former, we present a novel extraction approach for swirling stream line motion cores in snapshots of the flow field and show how those structures can be tracked in time by appropriate feature flow fields. While this approach is based on stream lines, we extend this work to path lines to extract the center lines of swirling particle motion for the first time. For the latter, we define vortex core lines as ridge lines of certain scalar feature indicators. As this approach requires second derivatives of the scalar, we conclude that a reduction of the required degree of derivatives is desirable. We achieve this by employing methods from scalar field topology to define the vortex and strain skeleton of flow fields. With this notion, the treatment of large, parameter dependent data sets becomes possible. We further improve the technique by developing the first derivative free feature line extraction method for surface meshes. We show the effectiveness of our methods by applying them to a number of data sets.

All work presented in this thesis has been published in peer-reviewed conference proceedings and journals.

Zusammenfassung

Die Analyse von Strömungsdaten wird aufgrund der ständig zunehmenden Datenmengen aus Simulation und Experiment zu einer immer größeren Herausforderung. Die Extraktion von Strömungsmerkmalen wird daher immer wichtiger, um die Datenmengen für die Strömungsphysiker auf ein beherrschbares Maß zu reduzieren. Unter den vielen Strömungsmerkmalen, die es zu betrachten lohnt, bilden Wirbelmerkmale einen besonders wichtigen Bereich.

In dieser Dissertation stellen wir neue Extraktionsverfahren für Wirbelmerkmale in zwei Kategorien vor: Die eine Kategorie umfasst Methoden zur Detektion von Zentrallinien, um die entweder Stromlinien oder Pfadlinien zirkulieren. Zum Stromlinienansatz präsentieren wir neue *Feature Flow Fields*, mit denen wir die Zentrallinien sowohl extrahieren als auch zeitlich verfolgen können. Anschließend wird der weltweit erste Ansatz zur Extraktion und Zeitverfolgung von Zentrallinien präsentiert, um die Partikel, also Pfadlinien, zirkulieren. Die andere Kategorie umfasst Methoden zur Extraktion von Wirbeln als Extremalstrukturen von abgeleiteten skalaren Wirbelgrößen. Der erste Ansatz extrahiert diese als Kammlinien unter Gebrauch von zweiten Ableitungen der skalaren Wirbelgröße. Da dies zu Rauschanfällig ist, entwickeln wir Methoden, die nur die erste Ableitung benötigen und auf Skalarfeldtopologie aufbauen. Damit können wir große, parameterabhängige Datensätze analysieren. Wir entwickeln diese Technik weiter und präsentieren die erste ableitungsfreie Methode zur Extraktion von Merkmalslinien triangulierter Flächen. Die Nützlichkeit unserer Methoden wird durch eine Vielzahl von Anwendungsbeispielen unterstrichen.

Alle in dieser Dissertation vorgestellten Entwicklungen wurden in begutachteten internationalen Konferenzbänden und Zeitschriften veröffentlicht.

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 9 |
| 1.1 | Related Work | 12 |
| 1.1.1 | Exploratory Vortex Detection | 12 |
| 1.1.2 | Scalar Feature Indicators | 12 |
| 1.1.3 | Vortex Core Line Extraction Methods | 13 |
| 1.2 | Vector and Scalar Field Topology | 14 |
| 1.3 | Feature Extraction Frameworks | 15 |
| | | |
| I | Theory | 17 |
| | | |
| 2 | Flow Field Background | 19 |
| 2.1 | Notation | 19 |
| 2.2 | Vector Fields and Flow Fields | 19 |
| 2.3 | Stream Lines and Path Lines | 19 |
| 2.3.1 | Acceleration of path lines | 21 |
| 2.4 | Invariances | 21 |
| 2.4.1 | Galilean Invariance | 22 |
| 2.4.2 | Galilean Invariance of Particle Acceleration | 23 |
| 2.4.3 | Objectivity | 23 |
| 2.5 | Flow Topology | 23 |
| 2.5.1 | Continuous Extraction of Flow Topology | 24 |
| 2.5.2 | Scalar Topology / Morse Smale Theory | 25 |
| 2.5.3 | Discrete Extraction | 27 |
| 2.6 | Ridge and Valley Lines | 29 |
| | | |
| 3 | Existing Vortex Detection Methods | 31 |
| 3.1 | Vortex Region Techniques | 31 |
| 3.1.1 | Okubo Weiss | 32 |
| 3.1.2 | Lambda 2 | 33 |
| 3.1.3 | M_Z | 33 |
| 3.1.4 | Pressure | 35 |
| 3.1.5 | Rotation Strength | 35 |
| 3.2 | Vortex Core Line Techniques | 35 |
| 3.2.1 | Sujudi/Haimes | 36 |
| 3.2.2 | Banks/Singer | 36 |
| 3.2.3 | Roth/Peikert | 37 |
| 3.2.4 | Pressure Minima | 37 |

| | | |
|------------|---|-----------|
| 3.2.5 | Parallel Vectors Operator | 37 |
| II | Detection Methods based on Swirling Motion | 39 |
| 4 | Extraction of PV Surfaces | 41 |
| 4.1 | Extracting PV Lines in Static Fields | 42 |
| 4.1.1 | Obtaining the Feature Flow Field \mathbf{f} | 43 |
| 4.1.2 | Starting Points for integrating \mathbf{f} | 45 |
| 4.2 | Feature Flow Fields for Tracking | 46 |
| 4.3 | Local Bifurcations | 47 |
| 4.3.1 | Inner Bifurcations | 48 |
| 4.3.2 | Boundary Bifurcations | 50 |
| 4.3.3 | Further Bifurcations | 50 |
| 4.4 | The Algorithms | 51 |
| 4.5 | Applications | 53 |
| 5 | Cores of Swirling Particle Motion | 57 |
| 5.1 | Swirling Motion of Stream Lines | 58 |
| 5.2 | Cores of Swirling Particle Motion | 59 |
| 5.2.1 | Swirling Particle Motion in 2D | 60 |
| 5.2.2 | Swirling Particle Motion in 3D | 62 |
| 5.2.3 | A Unified Notation of Swirling Motion Cores | 64 |
| 5.3 | Extraction of Swirling Particle Cores | 65 |
| 5.3.1 | Formulation using Parallel Vectors | 65 |
| 5.3.2 | Extraction using Parallel Vectors | 66 |
| 5.4 | Galilean Invariance of Swirling Particle Cores | 66 |
| 5.5 | Applications | 67 |
| III | Detection Methods based on Extremal Structures | 71 |
| 6 | Galilean Invariant Extraction of Vortex Core Lines | 75 |
| 6.1 | Ridges and Valleys for Vortex Core Lines | 75 |
| 6.2 | Extraction of Vortex Core Lines | 77 |
| 6.2.1 | Feature Flow Field Setup | 77 |
| 6.2.2 | Interpolation issues | 78 |
| 6.3 | Iconic Representation | 79 |
| 6.4 | Applications | 80 |
| 6.5 | Limitations | 82 |
| 7 | Vortex and Strain Skeletons | 85 |
| 7.1 | Dual Vortex and Strain Quantities | 86 |
| 7.1.1 | The Okubo-Weiss Criterion | 86 |
| 7.1.2 | The \mathbf{M}_Z -Criterion | 87 |
| 7.1.3 | Implementation of \mathbf{M}_Z | 87 |
| 7.2 | Vortex and Strain Skeletons | 89 |
| 7.2.1 | Extremal Feature Definition using Scalar Topology | 90 |
| 7.2.2 | Separatrices vs. Height Ridges | 91 |
| 7.3 | Extremal Feature Extraction | 93 |

| | | |
|-----------|--|------------|
| 7.3.1 | Morse-Smale Approximation by Watershed Segmentation . . . | 93 |
| 7.3.2 | Critical points | 94 |
| 7.3.3 | 1D-Separatrices | 94 |
| 7.3.4 | 2D-Separatrices | 95 |
| 7.4 | Applications | 95 |
| 7.5 | Outlook | 97 |
| 8 | Application: Feature Based Multiparameter Analysis of a Wing Model Flow | 99 |
| 8.1 | Active Flow Control Simulation Data | 99 |
| 8.1.1 | Wing Model | 100 |
| 8.1.2 | Unexcited Flow | 101 |
| 8.1.3 | Excitation Mechanism | 101 |
| 8.1.4 | Goal of the Excitation | 101 |
| 8.2 | Vortex Skeleton Parameter Study | 102 |
| 8.2.1 | Frequency of Air Injection | 102 |
| 8.2.2 | Intensity of Air Injection | 105 |
| 9 | Extraction of Feature Lines on Surface Meshes | 107 |
| 9.0.3 | Previous Work | 108 |
| 9.1 | Discrete Feature Line Extraction Method | 108 |
| 9.1.1 | Scalar Feature Indicators | 108 |
| 9.1.2 | Morse-Smale Complex and Feature Lines | 109 |
| 9.1.3 | Persistence Cancellation | 109 |
| 9.1.4 | Feature Line Preserving Cancellation | 109 |
| 9.1.5 | Discarding Topologically Forced Lines | 112 |
| 9.1.6 | Optional Line Smoothing | 112 |
| 9.1.7 | Implementational Issues | 112 |
| 9.2 | Results | 112 |
| 9.2.1 | Insensitivity to Noise | 114 |
| 9.2.2 | User Guided Parameter Traversal | 115 |
| 9.2.3 | Limitations | 115 |
| IV | Conclusions and Future Work | 117 |
| 10 | Conclusions | 119 |
| 10.1 | Future Work | 120 |
| | Bibliography | 124 |
| | List of Figures | 135 |
| | List of Tables | 140 |

Chapter 1

Introduction

Flow fields play a vital role in many areas. Examples are burning chambers, turbomachinery and aircraft design in industry as well as blood flow in medicine. As the resolution of numerical simulations and experimental measurements like PIV have evolved significantly in the last years, the challenge of understanding the intricate flow structures within massive result data sets has increased. In particular, manual inspection of the data by means of direct visualization techniques becomes more and more time consuming and inaccurate, as soon as the whole data set cannot be considered. This development has made automatic feature extraction necessary.

In this thesis, a *feature* refers to geometric objects like points, lines and surfaces satisfying special well defined mathematical properties that characterize the flow. All methods developed in this thesis aim at automatic feature extraction. Those methods aid in analyzing data sets in the following way:

- **Data Size Reduction**

By extracting just a sparse set of geometric features, the massive size of the data set can be reduced to a small fraction. The geometric objects can be annotated by several physical quantities to increase the expressiveness of the reduced result set.

- **Batch Processing**

Keeping the number of involved parameters low makes it possible to extract the features as a batch job prior to the visual analysis of a scientist. For example, the batch processing can be performed online during the simulation on a supercomputer, potentially saving vast disk space capacity for the full result data.

- **Faster Analysis**

The scientist is only faced with a much smaller amount of data that can be analyzed faster. Due to the reduced data size, the analysis is possible at interactive rates.

Among the features of interest, vortices are the most prominent: they play a major role due to their wanted or unwanted effects on the flow. As an example, Figure 1.1a shows the vortex in the wake of an airplane. Its influence on the flow at the runway lasts for several minutes and is high enough to cause serious trouble for other airplanes that follow too closely. Also, the lift of an airplane is affected by its induced vortices. Scientists aim at a passive or active control of those vortices to gain a higher lift (e.g.

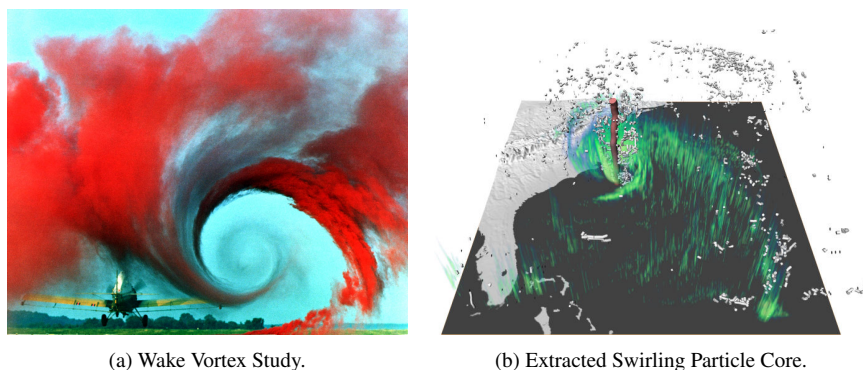


Figure 1.1: Wake vortex study from NASA Langley Research Center [NAS90]. The flow around a starting agricultural plane is made visible using smoke injection. A huge pattern of swirling particle motion is created by the aircraft's wake vortex. Chapter 5 aims at extracting the cores of such areas as can be seen in 1.1b. There, the hurricane Isabel data set at $t = 33.5$ is shown with the dominating swirling particle core line (red) and a volume rendering of the cloud moisture mixing ratio.

for reducing the runway length) or reducing the drag of the airplane (e.g. to save fuel). Understanding vortex structures is therefore very important in order to manipulate and control them successfully.

During the last decades, several definitions of what constitutes a vortex have been proposed, but none can be regarded as all-embracing. However, some desirable requirements of such a notion can be formulated from both a physical and a practical viewpoint:

1. The definition should respect the basic invariants of the underlying physics. As the Navier-Stokes equations are Galilean invariant, it is desirable to define flow features that are Galilean invariant as well.
2. The extraction should rely on as few parameters as possible to make batch processing feasible.
3. The extraction should be stable, and hence, the definition must be well-conditioned.

The methods in this thesis have been developed with those requirements in mind, and we have tried to obey them wherever possible. The thesis is divided into four parts. In the first part, the most relevant theoretical background is summarized, and a few basic concepts of flow fields and flow topology are presented. Chapter 3 summarizes the most important vortex extraction schemes that were commonly used prior to this thesis. The scientific contributions of this thesis are presented in parts II and III.

Part II is devoted to vortex extraction schemes based on the detection of swirling motion in time dependent 3D flow fields. The method developed in Chapter 4 is based on streamlines and obeys no invariances. The method of Chapter 5 extracts cores of swirling particle motion - a novel, Galilean invariant mathematical definition based on pathlines. This method is capable of automatically extracting core lines as visually seen in figure 1.1a. A result of this method can be inspected in Figure 1.1b where the core line of the hurricane Isabel dataset was extracted using our method.

Part III is devoted to vortex extraction by means of the extraction of extremal lines of scalar quantities that are derived from the flow field, such as λ_2 , the Okubo-Weiss

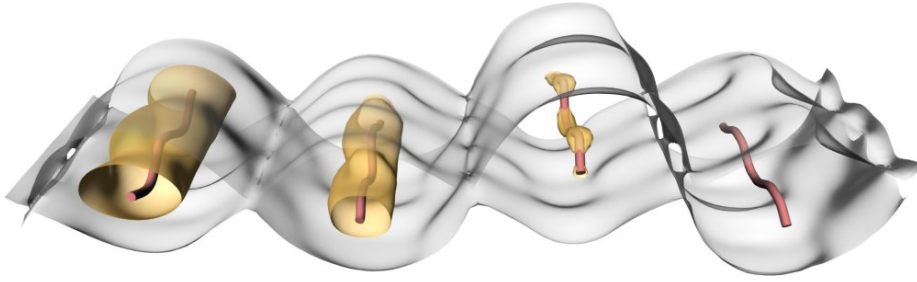
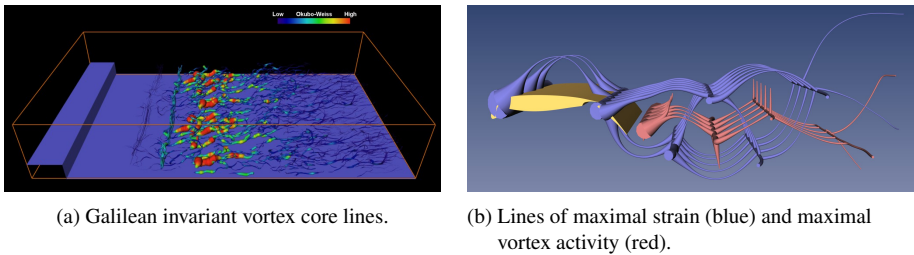


Figure 1.2: Extremal lines of scalar quantities can be regarded as center lines of iso surfaces.



(a) Galilean invariant vortex core lines.

(b) Lines of maximal strain (blue) and maximal vortex activity (red).

Figure 1.3: Extremal lines extracted with two methods developed in this thesis. Left, the ridge lines of the Okubo-Weis criterion are shown as extracted in Chapter 6. Right, the vortex and strain skeleton of the SCCH wing is shown as extracted in Chapter 7.

criterion Q or the \mathbf{M}_Z -criterion due to Haller. Such extremal lines can be regarded as the center lines of all isosurfaces (see Fig. 1.2). Those methods therefore extract line skeletons of the used quantities.

Here, the main emphasis is put on the robustness of the approaches that is mainly governed by the degree of derivatives being involved. The Galilean Invariant Vortex Core Lines extracted as height ridge lines in Chapter 6 (using third derivatives) points at the need of developing extremal extraction techniques using smaller derivative degrees (see Fig. 1.3a). A first approach is given by the vortex and strain skeletons of Chapter 7, where one- and two-dimensional vortex and strain features are extracted by using first derivatives only (Fig. 1.3b). This enhancements allowed us to process very large datasets, see Chapter 8 for a 2-parameter study of the high lift configuration of an aircraft. The extremal extraction approach is developed further in Chapter 9, where extremal structures on surfaces are extracted without any gradient computations. Note that this thesis concentrates on the mere extraction techniques and does not focus on post processing. From a practical view it is in almost all cases necessary to filter the extraction results by various quantities such as line length, directional clues or region of interest. By making the extraction techniques as robust as possible, we implicitly reduce the necessity of post processing. This is especially convincing in Chapter 8, where no derivatives are employed in the extraction and no post filtering is necessary at all. Further research directions are presented along with general conclusions of this work in part IV.

The work presented in this thesis has been published in international conference proceedings and journals as follows:

- Detection methods based on streamlines [TSW⁺05, WSTH07].
- Detection methods based on extremal structures [SWH05a, SWTH07, SWLP08] and an application thereof [WSG⁺08].

1.1 Related Work

In their state-of-the-art reports [PVH⁺02, PVH⁺03] Post et al. subdivide flow visualization into the three fields *direct visualization*, *integration-based techniques* and *feature extraction*. Following this classification, we restrict the discussion of related work to the area of feature extraction and arrange the contributions into the following five subfields:

- exploratory vortex detection,
- scalar feature indicators,
- vortex core line extraction methods,
- vector and scalar field topology,
- feature extraction frameworks.

1.1.1 Exploratory Vortex Detection

Exploratory feature detection methods help scientists in finding features during interactive flow visualization. Garth et al. propose stream surface integration starting from closed lines around the expected vortex axis [GTS⁺04a]. Tricoche et al. reported [TGK⁺04] that two dimensional transfer functions for volume rendering with dual-domain interaction [KKH01] allows highlighting of vortex regions which can not be captured well by a single feature indicator. Sadlo et al. use interactively seeded stream tubes for vortex core line validation and complement the vortex exploration by specifically seeded vorticity lines [SPP04]. Sadlo et al. annotate path lines with scalar feature indicators for vorticity transport analysis [SPS06]. Wiebel et al. introduce generalized streak lines for the detection of vortices near the flow boundary [WTS⁺07]. Doleisch et al. [Dol07] propose SimVis, an exploratory analysis framework for flow data. Features of interest can be defined by means of a *Feature Definition Language*. By marking relevant portions of histogram plots, relevant areas of the flow volume can be selected to show where the defined feature is located.

1.1.2 Scalar Feature Indicators

A wide range of literature aims at defining scalar fields that indicate features. Among them, vortex region quantities are the most prominent. Pressure p is an example, as within vortices, pressure takes on its minima. The Okubo-Weiss criterion defined by Hunt et al. [Hun87] trades off vorticity and strain. The λ_2 criterion [JH95] is negative wherever pressure minima occur within some plane, and is probably the most important scalar feature indicator for fluid mechanicians today. The \mathbf{M}_Z criterion defined by Haller [Hal05] defines a vortex as the lack of stretching and folding by determining, how long a trajectory stays elliptic (non-hyperbolic). The measure Δ [CPC90, SP03] quantifies how strong stream lines spiral in a specifically chosen reference frame by measuring

the imaginary part of the complex eigenvalues of the flow gradient. See section 3.1 for more details on the aforementioned scalar feature indicators.

Lagrangian coherent structures can be identified by finite-time Lyapunov exponents (FTLE) [Lor65], measuring how fast nearby trajectories diverge. Garth et al. show how this quantity can be computed with significantly reduced trajectory computations by an adaptive seeding strategy [GGTH07]: a four-point subdivision scheme is used to predict the FTLE, and an error metric on the current level is compared to a predefined threshold to decide whether the FTLE are estimated or computed on the next level. Sadlo et al. [SP07] exploit the fact that ridges of FTLE are Lagrangian coherent structures. By using filtered AMR ridge extraction they can reduce the costly FTLE computation to regions in which ridges are present on a lower mesh refinement level. By incorporating a look-ahead step, ridges can grow longer than indicated at the present level.

Correlation measures have been developed by several groups that transfer filter techniques from image processing to the flow field area: Heiberg et al. [HEWK03] defines a convolution on vector fields using the scalar product, and Bülow [Bul99] uses the vector product from Clifford Algebra for this purpose. Ebling et al. provide a generalization of both approaches called Clifford convolution [ES03, ES05] to apply filters to vector fields, with a special emphasis on template matching. Their approach proceeds in two phases. In the first phase, a Clifford convolution is performed with the pattern kernel to get an estimate of the angle and plane in which the tested pattern corresponds to the flow field. In the next step, a scalar convolution is performed with the appropriately rotated filter to obtain the similarity measure. As it is computationally not feasible to perform this scalar convolution for each grid node of the flow field, a mask set rotated evenly distributed over the unit sphere is precomputed, and the values corresponding to the closest mask rotation are used for the similarity measure. The approach provides a mechanism for vector field segmentation [ES06]. Wiebel et al. [WS05] use similarity measures on subsequent time steps of a time dependent flow field to determine locations for *eyelet particle tracing*. A very general feature indicator for time dependent fields is derived by Jänicke et al. [JWSK07] by extending the concept of local statistical complexity (LCS) to multivariate data. LCS was introduced by Shalizi et al. [SHR⁺06] for finite state cellular automata. LCS aims at clustering regions of similar temporal evolution in the space time domain into so called causal states. By defining a Huffman-Code for the causal states, interesting regions will have longer Huffman-Codes linked to their causal states, as they occur less often. The LCS at a given point in the space-time domain is the length of the Huffman-Code of the corresponding causal state. The clustering algorithm in [JWSK07] is based on the farthest point method. Jänicke et al. provide a faster and more accurate clustering algorithm based on a density-driven Voronoi tessellation [JBTS08].

1.1.3 Vortex Core Line Extraction Methods

Several papers are devoted to a mathematically defined notion of the core line of a vortex. As they build an important foundation for the present work, the methods of Sujudi and Haimes [SH95], Banks and Singer [BS95], Roth and Peikert [RP98] and Miura and Kida [MK97] are introduced in some detail in Section 3.2. Stegmaier et al. confined the corrector-predictor approach of Banks and Singer by still predicting in direction of vorticity, but correcting towards λ_2 -minima [JH95] instead of pressure-minima. Wiebel et al. [WGS05] determine the contribution of a subregion to the global flow by performing a localized flow analysis. In the subregion, a Neumann-Laplace

problem is solved. As the solution has vanishing curl and divergence, and no flow is crossing the boundary of the subregion due to the Neumann boundary condition, subtracting the solution from the global flow reveals its curl- and divergence-specific features. In conjunction with the method of Sujudi and Haines, this approach yields a Galilean invariant vortex extraction method [WGS07].

1.2 Vector and Scalar Field Topology

Vector field topology aims at a segmentation of the vector field domain into regions of similar flow behavior. Topological methods for flow visualization have been introduced by Helman and Hesselink for 2D [HH89] and extended to 3D [HH91], based on a classification of first order critical points by an eigenvalue analysis. Topological separatrices are computed from the critical points by a flow field integration in eigenvector directions. Globus et al. [GLL91] introduce an iconic representation of flow field topology. Several approaches add further topological elements to the extraction, like closed stream lines [WS01a, TWHS04a], saddle connections and cyclic fold bifurcations [TWHS04b, TWHS05] and topological structures on the boundary [dLvL99, SHJK00, WTHS04a].

Tricoche et al. [TGK⁺04] analyze vortex breakdown by tracking topological structures in 2D slices of 3D Flow fields.

By introducing the concept of saddle connectors, Theisel et al. [TWHS03] reduce the visual clutter of 3D topology visualization significantly.

Higher order critical points in 2D flows have been studied by Firby and Gardiner [FG82], visualized by Scheuermann et al. [SHK⁺97, SKMR98] and used for flow topology compression by Tricoche et al. [TSH00]. Theisel [The02] provides a mechanism to design 2D vector fields of arbitrary topology, including higher order critical points. Higher-order topology of 3D vector fields has been introduced to the visualization community by Weinkauff et al. and applied to construct [WTHS04b] and simplify [WTS⁺05] 3D vector fields.

Tricoche et al. [TWSH02] track critical points in two-dimensional time-dependent flows and detect Hopf und Fold Bifurcations. This approach connects the computed critical points in a prism cell structure that is obtained from the input grids of two consecutive time steps. Theisel et al. use the feature flow field approach for critical point tracking over time [TS03] without using the underlying grid. A comparable method for tracking in scale space is given by Klein et al. [KE07]. A method for tetrahedral grids is given by Garth et al [GTS04b].

Scalar field topology is the vector field topology of the gradient of the scalar and therefore a subfield of vector field topology. Edelsbrunner et al. [EHZ03, BEHP04, Pas07] and Forman [For98, For02, Lew05] have presented extensions of classical Morse theory to triangulated surfaces. Those approaches have been used in various areas for as different tasks as function simplification that bundles topology and geometry [BEHP04], molecular docking applications [CCL03, NWB⁺06] and quadrangular surface remeshing using natural harmonics [DBG⁺06]. A closely linked subject is watershed segmentation, as the watersheds of a function are contained in the Morse-Smale complex [Soi99]. Mangan et al use [MW99] watershed segmentation on surface meshes for surface patchification. Recent advances introduce skeletons of a distance field function that is assembled of parts of the 1-cells of the complex [GMDP⁺07]. The approach of Edelsbrunner et al. has been extended to 3D by Gyulassy et al. [GNP⁺05, GNP07] and applied for the study of hydrodynamic instabilities [LBM⁺06].

1.3 Feature Extraction Frameworks

Peikert and Roth [PR99] introduced the parallel vectors operator, an approach for finding loci in a domain where two vector valued quantities are parallel. They reformulated various existing methods using their approach. See section 3.2.5 for more details. Weinkauff et al. [WST⁺07] has formulated the *Unified Feature Extraction Architecture* by decomposing topological feature extraction methods into the core algorithms *finding zeros*, *integration of stream objects* and *intersection of stream objects*. A key ingredient of this technique is the Feature Flow Field [TS03] approach introduced by Theisel and Seidel for tracking features in time-dependent vector fields \mathbf{v} by introducing an appropriate vector field \mathbf{f} in space-time, such that a feature tracking in \mathbf{v} boils down to a stream line integration of \mathbf{f} starting at the initial feature location.

Part I
Theory

Chapter 2

Flow Field Background

2.1 Notation

We use a bold font for all vector quantities (like flow fields \mathbf{v} or locations \mathbf{x}), a regular font for all scalar quantities (pressure p).

The total differential of a quantity (including time derivatives) will be denoted by D . The spatial derivative without time derivatives will be denoted by ∇ . Partial derivatives of a quantity u with respect to the variable x will be denoted by u_x , $\partial_x u$ or $\frac{\partial}{\partial x} u$.

2.2 Vector Fields and Flow Fields

A time dependent *vector field* is a continuous map

$$\mathbf{v} : D \times I \rightarrow \mathbf{R}^3 \quad (2.1)$$

$$(\mathbf{x}, t) \mapsto \mathbf{v}(\mathbf{x}, t), \quad (2.2)$$

where $D \subset \mathbf{R}^3$ and $I \subset \mathbf{R}$.

Flow fields are vector fields that are either measured or modeled physical flows, or simulated solutions solving the incompressible Navier-Stokes equation, the second order partial differential equation given by

$$\rho \frac{\partial \mathbf{v}}{\partial t} + \rho \nabla \mathbf{v} \cdot \mathbf{v} = -\nabla p + \eta \Delta \mathbf{v} + (\lambda + \eta) \nabla (\nabla \cdot \mathbf{v}) + \mathbf{f}, \quad (2.3)$$

where \mathbf{f} is a volume force like gravitation, λ and η are the Lamé viscosity coefficient, ρ is the density of the material, and p denotes pressure. Please refer to e.g. [Bat67] for an introduction to Navier-Stokes theory, and [SPS06] for a link to vortex analysis.

A vector field is called *steady* if $\mathbf{v}(\mathbf{x}, t) = \mathbf{v}(\mathbf{x}, t_0)$, i.e., it is constant over time.

2.3 Stream Lines and Path Lines

In a time-dependent vector field $\mathbf{v}(\mathbf{x}, t)$ there are two important types of characteristic curves: stream lines and path lines. In a space-time point (\mathbf{x}_0, t_0) we can start a *stream line* (staying in time slice $t = t_0$) by integrating

$$\frac{d}{d\tau} \mathbf{x}(\tau) = \mathbf{v}(\mathbf{x}(\tau), t_0) \quad \text{with} \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (2.4)$$

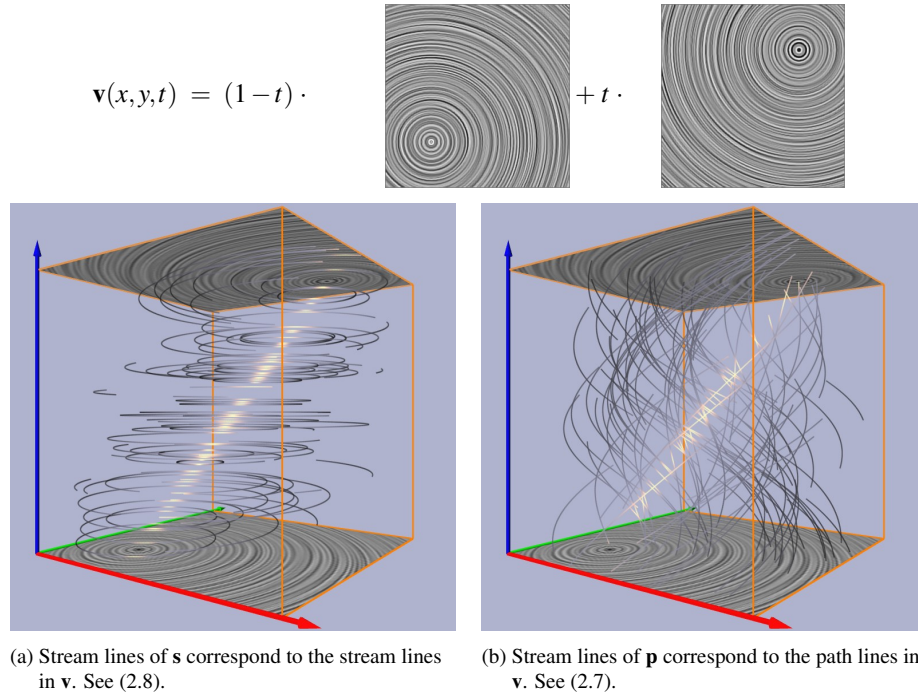


Figure 2.1: Characteristic curves of a simple 2D time-dependent vector field shown as illuminated field lines. The red/green coordinate axes denote the (x,y) -domain, the blue axis shows time.

or a *path line* by integrating

$$\frac{d}{dt} \mathbf{x}(t) = \mathbf{v}(\mathbf{x}(t), t) \quad \text{with} \quad \mathbf{x}(t_0) = \mathbf{x}_0. \quad (2.5)$$

Path lines describe the trajectories of massless particles in time-dependent vector fields. The ODE system (2.5) can be rewritten as an autonomous system at the expense of an increase in dimension by one, if time is included as an explicit state variable:

$$\frac{d}{dt} \begin{pmatrix} \mathbf{x} \\ t \end{pmatrix} = \begin{pmatrix} \mathbf{v}(\mathbf{x}(t), t) \\ 1 \end{pmatrix} \quad \text{with} \quad \begin{pmatrix} \mathbf{x} \\ t \end{pmatrix} (0) = \begin{pmatrix} \mathbf{x}_0 \\ t_0 \end{pmatrix}. \quad (2.6)$$

In this formulation space and time are dealt with on equal footing – facilitating the analysis of spatio-temporal features. Path lines of the original vector field \mathbf{v} in ordinary space now appear as stream lines of the vector field

$$\mathbf{p}(\mathbf{x}, t) = \begin{pmatrix} \mathbf{v}(\mathbf{x}, t) \\ 1 \end{pmatrix} \quad (2.7)$$

in space-time. To treat stream lines of \mathbf{v} , one may simply use

$$\mathbf{s}(\mathbf{x}, t) = \begin{pmatrix} \mathbf{v}(\mathbf{x}, t) \\ 0 \end{pmatrix}. \quad (2.8)$$

Figure 2.1 illustrates \mathbf{s} and \mathbf{p} for a simple example vector field \mathbf{v} . It is obtained by a linear interpolation over time of two linear vector fields.

In steady vector fields, stream lines and path lines coincide and are given as the solution of

$$\frac{d}{d\tau} \mathbf{x}(\tau) = \mathbf{v}(\mathbf{x}(\tau)) \quad \text{with} \quad \mathbf{x}(0) = \mathbf{x}_0. \quad (2.9)$$

A number of visualization techniques originally designed for steady vector fields can be applied to unsteady fields by considering each time step independently. In this case equations (2.9) and (2.4) coincide – hence, these approaches address the behavior of stream lines. Examining the behavior of path lines (particles) requires to consider time explicitly.

2.3.1 Acceleration of path lines

For a path line $\phi : \mathbf{R} \rightarrow \mathbf{R}^3$ of a time dependent vector field $\mathbf{v} : \mathbf{R}^4 \rightarrow \mathbf{R}^3$, we have

$$\phi'(t) = \mathbf{v}(\phi(t), t)$$

by (2.5). The *particle acceleration*, defined by

$$\mathbf{a}(t) := \phi''(t), \quad (2.10)$$

is useful for measuring swirling motion. Using the chain rule, this can be expressed as

$$\mathbf{a}(t) = \phi''(t) = D(\mathbf{v}(\phi(t), t)) \quad (2.11)$$

$$= D\mathbf{v}|_{(\phi(t), t)} \circ D(\phi(t), t) \quad (2.12)$$

$$= D\mathbf{v}|_{(\phi(t), t)} \circ \begin{pmatrix} \mathbf{v} \\ 1 \end{pmatrix} \quad (2.13)$$

$$= v_t + \nabla v \cdot v \quad (2.14)$$

by means of the flow field alone.

2.4 Invariances

It is a fundamental property of physical laws to be independent of the observer to a certain extent. The classical laws of Newtonian mechanics hold for observers in the railway station as well as for those in the train that is passing by at constant speed. Technically, *observer* hereby stands for a coordinate transformation, and *invariance* means that the described laws are independent of the transformation. The most basic group of invariance is Galilean invariance describing the above train ride (see 2.4.1). A general transformation T is given by an invertible map

$$T(\mathbf{x}, t) : \mathbf{R}^3 \times \mathbf{R} \rightarrow \mathbf{R}^3, \quad (2.15)$$

$$(\mathbf{x}, t) \mapsto \tilde{\mathbf{x}} \quad (2.16)$$

A point \mathbf{x} at time t may be mapped to an arbitrary point, but time is not being transformed. Of course, only very restricted transformations are relevant in physics, and we shortly introduce the most relevant groups of transformations, namely *Galilean* and *Objective* transformations.

When changing coordinates, all incorporated quantities have to be transformed to reflect the coordinate transformation. We exemplify this for the flow field \mathbf{v} under

transformation T changing from coordinates \mathbf{x} to $\tilde{\mathbf{x}}$. The transformed flow field $\tilde{\mathbf{v}}$ can then be obtained using a trajectory $\tilde{\mathbf{x}}$ in new coordinates by

$$\tilde{\mathbf{v}}(\tilde{\mathbf{x}}, t) = \frac{d}{dt} \tilde{\mathbf{x}}(t) \quad (2.17)$$

$$= \frac{d}{dt} T(\mathbf{x}(t), t) \quad (2.18)$$

$$= \nabla T(\mathbf{x}, t) \begin{pmatrix} \frac{d}{dt} \mathbf{x}(t) \\ 1 \end{pmatrix} \quad (2.19)$$

$$= \nabla T(\mathbf{x}, t) \begin{pmatrix} \mathbf{v}(\mathbf{x}, t) \\ 1 \end{pmatrix} \quad (2.20)$$

$$= \nabla T(T^{-1}(\tilde{\mathbf{x}}), t) \begin{pmatrix} \mathbf{v}(T^{-1}(\tilde{\mathbf{x}}), t) \\ 1 \end{pmatrix} \quad (2.21)$$

2.4.1 Galilean Invariance

Galilean transformations are coordinates traveling at constant speed:

$$\tilde{\mathbf{x}} = G(\mathbf{x}, t) = \mathbf{x} + t\mathbf{v}^0. \quad (2.22)$$

The derivative of this transformation is given by

$$\nabla G(\mathbf{x}, t) = \begin{pmatrix} 1 & 0 & 0 & v_1^0 \\ 0 & 1 & 0 & v_2^0 \\ 0 & 0 & 1 & v_3^0 \end{pmatrix} \quad (2.23)$$

and following (2.21), the flow field transformation reads

$$\tilde{\mathbf{v}}(\tilde{\mathbf{x}}, t) = \mathbf{v}(\mathbf{x}, t) + \mathbf{v}^0 = \mathbf{v}(\tilde{\mathbf{x}} - t\mathbf{v}^0, t) + \mathbf{v}^0. \quad (2.24)$$

Obviously, $\nabla \tilde{\mathbf{v}} = \nabla \mathbf{v}$ for all Galilean transformations, and hence, the spatial gradient of a flow field is Galilean invariant. Hence, all quantities that are derived from the derivatives of the flow field are Galilean invariant, too.

2.4.2 Galilean Invariance of Particle Acceleration

As an example we show that particle acceleration (2.14) is Galilean invariant. As the spatial gradient of the flow field is Galilean invariant, we note

$$\tilde{v}_t + \nabla \tilde{v} \cdot \tilde{v} = \frac{d}{dt} (\tilde{v}(\tilde{\mathbf{x}}, t)) + \nabla \tilde{v}|_{(\tilde{\mathbf{x}}, t)} \tilde{v}(\tilde{\mathbf{x}}, t) \quad (2.25)$$

$$= \frac{d}{dt} (v(\tilde{\mathbf{x}} - t\mathbf{v}^0, t) + v^0) \quad (2.26)$$

$$+ \nabla (v|_{(\tilde{\mathbf{x}} - t\mathbf{v}^0, t)} + v^0) (v(\tilde{\mathbf{x}} - t\mathbf{v}^0, t) + v^0) \quad (2.27)$$

$$= Dv|_{(\tilde{\mathbf{x}} - t\mathbf{v}^0, t)} \begin{pmatrix} -v^0 \\ \mathbf{1} \end{pmatrix} \quad (2.28)$$

$$+ \nabla (v|_{(\tilde{\mathbf{x}} - t\mathbf{v}^0, t)} + v^0) (v(\tilde{\mathbf{x}} - t\mathbf{v}^0, t) + v^0) \quad (2.29)$$

$$= -\nabla v|_{(\tilde{\mathbf{x}} - t\mathbf{v}^0, t)} v^0 + \frac{d}{dt} v|_{\tilde{\mathbf{x}} - t\mathbf{v}^0} \quad (2.30)$$

$$+ \nabla v|_{(\tilde{\mathbf{x}} - t\mathbf{v}^0, t)} v(\tilde{\mathbf{x}} - t\mathbf{v}^0, t) + \nabla v|_{(\tilde{\mathbf{x}} - t\mathbf{v}^0, t)} v^0 \quad (2.31)$$

$$= \frac{d}{dt} v|_{\mathbf{x}, t} + \nabla v|_{(\mathbf{x}, t)} v(\mathbf{x}, t) \quad (2.32)$$

$$= v_t + \nabla v \cdot v. \quad (2.33)$$

We need this property in section 5.4 for the proof of the Galilean invariance of swirling particle cores.

2.4.3 Objectivity

Objective transformations are rotating coordinates traveling at time-varying, but constant-in-space speed:

$$\tilde{\mathbf{x}} = Q(\mathbf{x}, t) = Q(t)\mathbf{x} + b(t), \quad (2.34)$$

where $Q(t) \in SO(3)$ is a proper orthogonal matrix. Its derivative is given by

$$\nabla Q(\mathbf{x}, t) = (Q(t) | Q'(t)\mathbf{x} + b'(t)), \quad (2.35)$$

and following the same lines as above, the flow field transformation reads

$$\tilde{v}(\tilde{\mathbf{x}}, t) = Q(t)v(\mathbf{x}, t) + Q'(t)\mathbf{x} + b'(t), \quad (2.36)$$

and resubstituting $\mathbf{x} = Q^{-1}(t)(\tilde{\mathbf{x}} - b(t))$, we obtain

$$\tilde{v}(\tilde{\mathbf{x}}, t) = Q(t)v(Q^{-1}(t)(\tilde{\mathbf{x}} - b(t)), t) + Q'(t)(Q^{-1}(t)(\tilde{\mathbf{x}} - b(t))) + b'(t). \quad (2.37)$$

2.5 Flow Topology

One objective of flow topology is to partition the domain into patches of similar streamline behaviour. Mathematically, it is described using α - and ω -limit sets, see e.g. Scheuermann et al. [SHJK00, SS07]. Here α -limit means *backward in time*, ω -limit means *forward in time*. The α -limit set $A(\mathbf{s})$ of a streamline \mathbf{s} is the set of all accumulation points of \mathbf{s} backward in time. In formulae,

$$A(\mathbf{s}) = \{\mathbf{p} \in \mathbf{R}^3 | \exists (t_n), t_n \rightarrow -\infty, \lim \mathbf{s}(t_n) = \mathbf{p}\}. \quad (2.38)$$

Analogously, $\Omega(\mathbf{s})$ is the set of all accumulation points of \mathbf{s} forward in time:

$$\Omega(\mathbf{s}) = \{\mathbf{p} \in \mathbf{R}^3 \mid \exists(t_n), t_n \rightarrow \infty, \lim \mathbf{s}(t_n) = \mathbf{p}\}. \quad (2.39)$$

The catchment basin $B_\alpha(A)$ of $A(\mathbf{s})$ is the union of all streamlines with α -limit set A :

$$B_\alpha(A) = \{a \in D \mid A(s_a) = A\}, \quad (2.40)$$

and analogously

$$B_\omega(\Omega) = \{a \in D \mid \Omega(s_a) = \Omega\}. \quad (2.41)$$

The flow field domain D can now be partitioned as follows:

$$D = \bigcup_{i,j} B_\alpha(A_i) \cap B_\omega(\Omega_j). \quad (2.42)$$

2.5.1 Continuous Extraction of Flow Topology

The partition (2.42) consists of points, lines, surfaces and volumes for 3D domains. Usually, the volumes are displayed only implicitly by their bounding separation surfaces. In the following, we shortly introduce the extraction of critical points, 1D-separatrices and 2D-separatrices.

Critical Points

Several numerical methods for continuous critical point extraction of the gradient vector field are at hand. In tetrahedral grids, zeros can be computed explicitly [SH95]. In regular and curvilinear grids, we use a simple subdivision approach: a cell is checked whether one of the three components of the gradient is positive/negative at *all* 8 corners of the cell. If so, no zero is found inside. Otherwise, we recursively subdivide into 8 subcells until their size is smaller than a certain threshold.

1D-Separatrices

The 1D-separatrices are stream lines of the vector field, being integrated from saddle points in direction of their eigenvector corresponding to the unique negative or positive eigenvalue, see Figures 2.3c and 2.3b. Two seeding points are placed stepping away from the saddle in that eigenvector's direction. Afterwards, a forward integration from those two points yields the separation line for a positive eigenvalue, and backward for a negative eigenvalue.

2D-Separatrices

A continuous method for 2D-separatrix extraction is a stream surface integration from saddle points. As a seeding structure, a closed curve centered at the critical point can be used that lies in the plane spanned by the two eigenvectors of matching sign. From this seeding line a forward or backward stream surface integration is performed, depending on the sign of the eigenvalues.

2.5.2 Scalar Topology / Morse Smale Theory

The topology of a scalar function $s : D \rightarrow \mathbf{R}$ can be regarded as the flow topology of the gradient vector field ∇s of the scalar. The existence of a potential gives additional structure to this topology, and it can be described with more ease, as no closed streamlines (limit cycles) can occur. It is for this reason that simple, discrete methods could be developed for the topology extraction in this setting as an application of discrete Morse Smale theory. We restrict the discussion to surfaces embedded in \mathbf{R}^3 for use in chapter 9.

A smooth function f on a smooth manifold surface M without boundary embedded in \mathbf{R}^3 is a *Morse function* if its Hessian has full rank at every critical point of f where its gradient vanishes [Mil63, Ban70, PM82, EHZ03]. *Gradient curves* are maximal lines of f that are everywhere tangent to the gradient. Gradient curves start and end at critical points c of f . The stable and unstable manifolds $S(c)$ and $U(c)$ are the collections of all points that are part of a gradient curve ending at c or originating from c , respectively. The stable and unstable manifolds of maxima and minima can be regarded as hills and valleys, respectively. Critical points that have both a lower dimensional stable and unstable manifold are saddles. Both the collection of stable manifolds and the collection of unstable manifolds partition the surface, see Fig. 2.2. For Morse functions f , the intersection of both partitions is the *Morse-Smale complex* of f . This complex subdivides the domain into quadrangles with alternating maximum, saddle, minimum, saddle, see Fig. 2.2d. The one dimensional arcs in the decomposition link maxima, minima and saddles and are called ascending (descending) 1-cells, if they connect saddles with maxima (minima). Descending (ascending) 1-cells separate hills (valleys) and constitute their boundary.

Cancellation

Pairwise cancellation of saddle/minimum or saddle/maximum can be performed to coarsen the initial complex, as long as the alternating quad property is maintained. A cancellation is called valid in this case. Fig. 2.2e illustrates how a saddle/maximum cancellation usually erases the descending 1-cell of the canceled saddle. The ascending 1-cell stays in the complex, as it still divides the unstable manifolds of the minima adjacent to the canceled saddle. In the special case of a terminal minimum or maximum, both the unstable and the stable manifolds of the saddle vanish, see Fig. 2.2f. By providing a sequence of valid cancellations, the complex can be coarsened, until no valid cancellation is left. In practice this is done by quantifying each saddle individually by a cancellation criterion. The cancellations are then performed in increasing order of the cancellation criterion.

2D-Scalar topology is closely linked to watershed lines in a 2D-terrain (cf. Figure 7.6). At certain line structures, rain water separates in the sense that nearby water assembles in different valleys. Those maximum lines partition the domain into valleys. Within valleys all water flows towards the same minimum. Similarly, the domain is partitioned into hills separated by minimal lines called watercourses. On hills all water runs down from one maximum.

The generalization to 3D is straight-forward. Here the watersheds are surfaces, and additional 1D-separatrices come into play. The partitioning of the domain in hills and valleys as well as the corresponding separatrices are the subject of scalar topology. If the scalar function is differentiable, its topology can also be obtained as the vector field topology of its gradient.

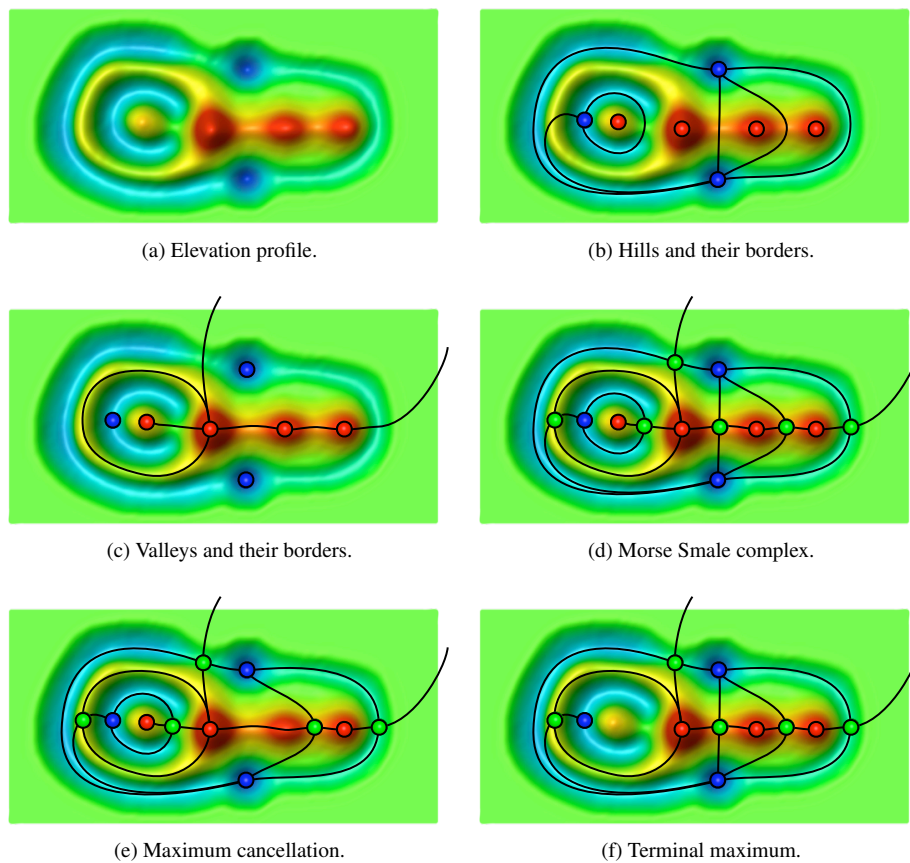


Figure 2.2: The elevation profile of a landscape as seen from above. Lines depict the partition of the landscape into hills, valleys and their intersection, the MS complex. Blue, red and green dots represent minima, maxima and saddles. Bottom row: simplifications of the MS complex by different types of cancellations and their influence on vanishing 1-cells.

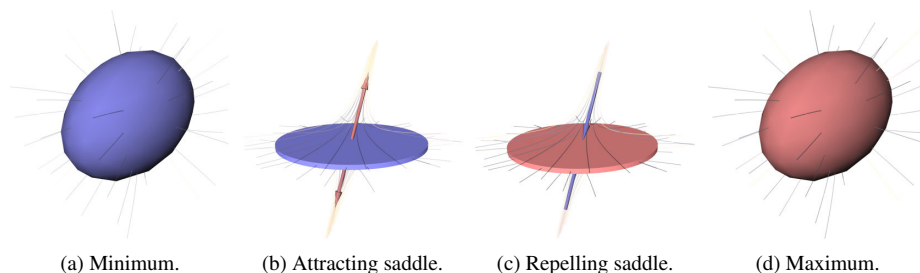


Figure 2.3: Classification of critical points in scalar fields.

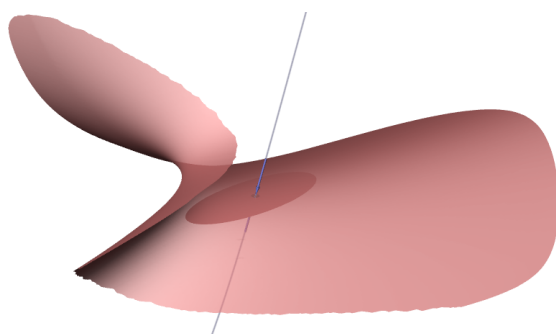


Figure 2.4: Separatrices originating from a repelling saddle. The line is a minimal line of steepest descent, the surface is a watershed.

Depending on the eigenvalues of the Hessian matrix of the scalar function, critical points can be classified into four categories (see Figure 2.3): When all three eigenvalues are positive, the critical point is a minimum, and all gradient lines are leading away from the critical point, so minima are sources (see Figure 2.3a). A point with exactly one negative eigenvalue is called a repelling saddle (Figure 2.3c). The unstable manifold emanating from this point has the plane spanned by the eigenvectors corresponding to the positive eigenvalues as its tangent plane. Within this surface, gradient lines lead away from the point. Those surfaces are maximal features, separating minima from each other. We refer to those surfaces as *2D-separatrices*, *maximum surfaces* or *watersheds*. The one-dimensional separatrix tangential to the eigenvector corresponding to the one negative eigenvalue is a minimum line in the scalar field, leading to a minimum following the steepest descent (Figure 2.4). Those lines are meaningful features, as they can be regarded as centers of isosurfaces, see Figure 7.3 for an illustrative example and Figure 7.1 for a real world application. A point with two negative eigenvalues is called attracting saddle (Figure 2.3b), and its *2D-separatrix* is a *watercourse*, as it separates two maxima from each other. We also refer to the watercourses as *minimum surfaces*¹. Finally, all eigenvalues are negative at maxima (Figure 2.3d) and all gradient lines lead into the point, so maxima are sinks.

2.5.3 Discrete Extraction

Based on Banchoff's extension of classical Morse theory to polyhedral surfaces [Ban70], Edelsbrunner et al [EHZ03] construct the MS complex for piecewise linear functions. The approach is refined by Bremer et al [BEHP04, Pas07].

¹*Minimum surfaces* should not to be mixed up with *Minimal Surfaces*, denoting surfaces of zero mean curvature in mathematics

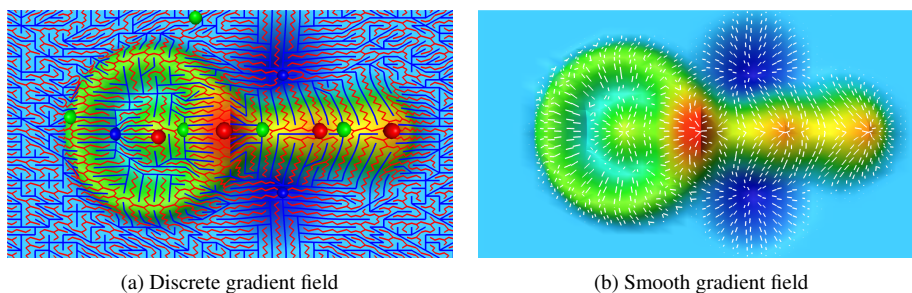


Figure 2.5: Forman’s discrete gradient vector field of the height function of an elevation profile is shown with red maxima, green saddles and blue minima in (a). Red lines are gradient curves ascending to maxima, blue lines are gradient curves descending to minima. In (b) its smooth analogue is shown, being discontinuous and constant on triangles.

Forman [For98, For02] generalizes Morse Theory to the more general category of combinatorial complexes. This generalization is a very strong theoretical foundation for simple and intuitive algorithms. Therefore we base our algorithms on this theory.

A function f defined on the vertices of a triangular surface can be considered a Morse function if all vertex values are different. Lewiner et al [LLT03b, LLT03a, CCL03] show how to construct the discrete gradient vector field (also for non-Morse functions) as a pair of interlaced primal and dual weighted spanning forests of the triangular surface that partition the surface into stable and unstable manifolds.

- The weighted *primal graph* PG is the graph of vertices and edges of the surface mesh. Edges are assigned the mean value of the incident vertices.
- The vertices and edges of the *dual graph* DG are the triangles and edges of the surface mesh, respectively. Triangles are assigned the mean value of the incident vertices.

The maxima of the dual graph serve as roots for a spanning forest representing stable manifolds. Edges are added in decreasing order. Similarly, minima of the primal graph serve as roots for a spanning forest representing unstable manifolds. Edges are added in increasing order, but only if they are not contained in the dual spanning forest. The critical elements of f are the roots of the primal (minima) and dual (maxima) graph and the edges that are not contained in either spanning trees (saddles). Lewiner shows that the extracted locations of critical points correlate with the locations Banchoff describes [Ban70]. The spanning forests can be regarded as the union of all gradient curves on the mesh. Descending gradient curves are traced as concatenated edges in the primal spanning tree as paths to the root, ascending gradient curves as triangle strips in the dual spanning tree. See Fig. 2.5 for the discrete gradient vector field of the height function of an elevation profile.

The computation of the 1-cells can now be performed as in the smooth setting. For a saddle edge $e = (v_1, v_2)$ that is incident to the faces f_1, f_2 , the stable manifolds are computed as decreasing paths in the primal spanning tree starting with v_1, v_2 respectively until the roots m_1, m_2 (that are minima) are reached. Analogously, unstable manifolds are traced as triangle strips in the dual spanning tree starting from f_1, f_2 respectively, until the maxima M_1, M_2 are reached.

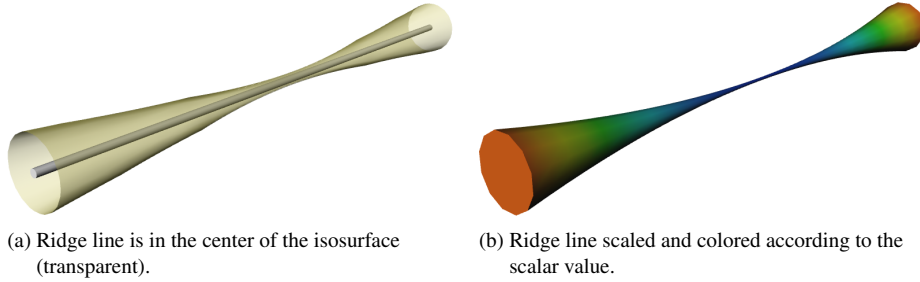


Figure 2.6: Ridge line of a simple scalar field.

2.6 Ridge and Valley Lines

Several notions of extremum lines, ridge and valley lines of a function $f : \mathbf{R}^3 \rightarrow \mathbf{R}$ have been developed in the literature. We use the height ridge definition detailed in [Ebe96], which is a one dimensional generalization of the well known zero dimensional notion of an extremum point. We choose this definition as it requires just second derivatives of the vector field rather than fourth order derivatives like ridge definitions that are based on curvature extrema, see [EGM⁺94] for a thorough introduction and comparison of several ridge line extraction schemes and [KvD93] for a historical survey of the development of extremum lines.

A sufficient condition for a local maximum point \mathbf{x} of a function $f : \mathbf{R}^3 \rightarrow \mathbf{R} \in \mathcal{C}^2(\mathbf{R}^3; \mathbf{R})$ is a vanishing gradient $\nabla f(\mathbf{x}) = 0$ coupled with a negative definite Hessian $Hf(\mathbf{x})$ implying a set of three negative eigenvalues $\gamma_1 \leq \gamma_2 \leq \gamma_3$ corresponding to orthogonal eigenvectors $\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3 \in \mathbf{R}^3$ satisfying $H\mathbf{c}_i = \gamma_i\mathbf{c}_i$. As H is symmetric, such an orthogonal eigensystem corresponding to real eigenvalues always exists.

Aiming at a one dimensional generalization of a local maximum, we note that negative eigenvalues γ_i imply that the graph of f is convex in a small neighbourhood of the maximum. In direction of \mathbf{c}_3 , the eigenvector corresponding to the largest eigenvalue γ_3 , the maximum is least stable, as this is the direction of smallest convexity. The softest relaxation hence is to relax convexity just in direction of \mathbf{c}_3 . As a ridge line (when looking at a terrain) should intuitively follow the steepest ascent, it is natural to require $\mathbf{c}_3 = \mathbf{const} \cdot \nabla f$ whenever $\nabla f \neq 0$, resulting in the requirement $H(\nabla f) = \gamma_3(\nabla f)$. This makes ridge line extraction applicable to the parallel vectors operator as stated in [PR99]. From the orthogonality of \mathbf{c}_i , it directly follows that $(\nabla f)\mathbf{c}_1 = 0, (\nabla f)\mathbf{c}_2 = 0$. Vice versa, $(\nabla f)\mathbf{c}_1 = (\nabla f)\mathbf{c}_2 = 0, \nabla f \neq 0$ implies that $\mathbf{c}_3 = \mathbf{const} \cdot \nabla f$, also from orthogonality.

This intuition leads to the following definition cited from [Ebe96].

Definition 1 Let $f \in \mathcal{C}^2(\mathbf{R}^3; \mathbf{R})$, ∇f its gradient and Hf its Hessian with eigenvectors $\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3$ and corresponding eigenvalues $\gamma_1 \leq \gamma_2 \leq \gamma_3$.

1. Then a ridge line consists of all points \mathbf{x} where

- $A := (\nabla f(\mathbf{x}))\mathbf{c}_1 = 0$ and $B := (\nabla f(\mathbf{x}))\mathbf{c}_2 = 0$ and
- $\gamma_2 < 0$.

2. This has a d -dimensional generalization. A d -dimensional ridge consists of all points \mathbf{x} where

- (a) $P_i(x) := \nabla f(\mathbf{x})\mathbf{c}_i = 0$ for all $i = 1, \dots, 3 - d$ and
 (b) $\gamma_{3-d} < 0$.

3. d -dimensional valleys of f are defined as d -dimensional ridges of $-f$.

Note that the structures defined here are d -dimensional manifolds in most cases due to the regular value theorem justifying the terminology of d -dimensional ridges.

As an example let $\gamma_1 \leq \gamma_2 < 0, \gamma_3 > \gamma_2$ and consider the function $f(x, y, z) = \gamma_1 x^2 + \gamma_2 y^2 + \gamma_3 z^2$. Then $\nabla f(x, y, z) = (\gamma_1 x, \gamma_2 y, \gamma_3 z)^t$, $Hx(x, y, z) = \text{diag}(\gamma_1, \gamma_2, \gamma_3)$ with eigenbasis $\mathbf{c}_i = \mathbf{e}_i$, \mathbf{e}_i denoting the Euclidean standard basis. At $x = y = 0$ we have $\nabla f(\mathbf{x})\mathbf{a} = \nabla f(\mathbf{x})\mathbf{b} = 0$ and $\gamma_2 < 0$. Hence, the z -axis is a ridge line. Figure 2.6 illustrates this for $\gamma_1 = -100, \gamma_2 = -99, \gamma_3 = 1$.

See section 7.2.2 for a discussion of connections and differences of ridges compared to separatrices in scalar topology.

Chapter 3

Existing Vortex Detection Methods

A large variety of vortex detection methods have been derived in the literature. In this chapter we summarize the main existing methods that were available prior to this thesis. Good overviews are also in the literature [PVH⁺02, PR99].

Vortex detection schemes can be classified in two major categories:

- *Vortex region detection* is based on scalar *vortex region quantities* that are used to define a vortex as a spatial region where the quantity exhibits a certain value range. Examples are regions of high magnitude of vorticity or negative λ_2 -criterion. Isosurfaces or volume rendering are common approaches for visualizing these quantities. We give a thorough overview in Section 3.1.
- *Vortex core line extraction* aims at finding line type features that are regarded as centers of vortices, where the understanding of *center* is defined by each method individually. Examples are the intuitive notion of swirling motion for Sujudi and Haines [SH95, PR99], and the more physical notion of integrated vorticity for banks and Singer [BS95]. We give a thorough overview in Section 3.2.

Many vortex core line extraction methods can be implemented using the parallel vectors (PV) operator, a popular line feature extraction approach for static flows fields [PR99] which we sketch in Section 3.2.5.

3.1 Vortex Region Techniques

The most widely used vortex region quantities are based on a decomposition of the flow field gradient

$$\nabla \mathbf{v} = \mathbf{S} + \mathbf{\Omega} \quad (3.1)$$

into its symmetric part, the strain tensor

$$\mathbf{S} = \frac{1}{2}(\nabla \mathbf{v} + \nabla \mathbf{v}^t) \quad (3.2)$$

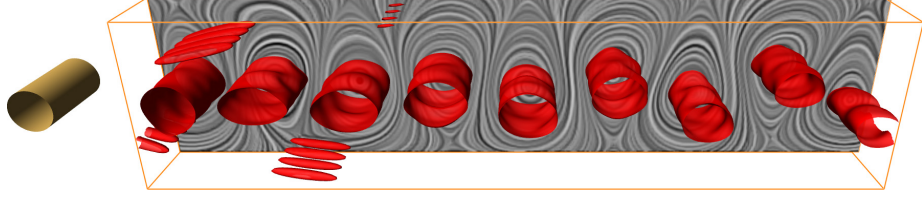


Figure 3.1: Isosurfaces of Q showing Eulerian vortex regions in a flow behind a circular cylinder. The LIC-plane is indicating corresponding vortical streamline patterns in the reference frame relative moving with the convection velocity.

and its antisymmetric part, the vorticity tensor

$$\Omega = \frac{1}{2}(\nabla \mathbf{v} - \nabla \mathbf{v}^t) = \frac{1}{2} \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix}, \quad (3.3)$$

where $\omega = (\omega_1, \omega_2, \omega_3) = \nabla \times \mathbf{v}$ denotes the vorticity. While Ω assesses vortical activity, the strain tensor \mathbf{S} measures the amount of stretching and folding which drives mixing to occur. To motivate the latter, some words are necessary on perturbation advection: in a time dependent flow field consider the path line $\mathbf{x}(t; \mathbf{x}_0)$ started at \mathbf{x}_0 defined by $\dot{\mathbf{x}}(t; \mathbf{x}_0) = \mathbf{v}(\mathbf{x}(t; \mathbf{x}_0), t)$ and initial condition $\mathbf{x}(t_0, \mathbf{x}_0) = \mathbf{x}_0$. Comparing \mathbf{x} to $\tilde{\mathbf{x}}(t, \tilde{\mathbf{x}}_0)$ with $\tilde{\mathbf{x}}_0$ being an infinitesimal perturbation of \mathbf{x}_0 , the propagated perturbation $\xi(t) = \tilde{\mathbf{x}}(t, \tilde{\mathbf{x}}_0) - \mathbf{x}(t, \mathbf{x}_0)$ is governed by the linearized dynamical system

$$\dot{\xi} = \nabla \mathbf{v}(\mathbf{x}(t; \mathbf{x}_0), t) \xi, \quad (3.4)$$

see e.g. [Hal80]. The strain tensor now gives the answer to the question, how the magnitude of the perturbation $|\xi|$ evolves in time when both initial conditions are advected by the flow. This evolution is described by the Lyapunov function

$$V(\xi, t) := \frac{1}{2} \frac{d}{dt} |\xi|^2 = \langle \xi, \dot{\xi} \rangle = \langle \xi, \nabla \mathbf{v} \xi \rangle = \langle \xi, \mathbf{S} \xi \rangle, \quad (3.5)$$

where (3.4) was used as well as the fact that the symmetric scalar product $\langle \cdot, \cdot \rangle$ only sees the symmetric part of $\nabla \mathbf{v}$. Where $V < 0$, initial perturbations decay over time, while $V > 0$ indicates their growing. Where perturbations decay or grow drastically the flow exhibits a saddle like pattern – a pattern that drives mixing of fluid particles. Now the Euclidean matrix norm $\|\mathbf{S}\|$, called the rate of strain, supplies a measure for this initial perturbation evolution due to

$$|V(\xi, t)| = |\langle \xi, \mathbf{S} \xi \rangle| \leq |\xi|^2 \|\mathbf{S}\|. \quad (3.6)$$

Note that although the initial perturbation analysis uses path lines, the quantity \mathbf{S} that measures the perturbation growth is completely Eulerian – it is built out of quantities in a specific snapshot of time.

3.1.1 Okubo Weiss

The Okubo-Weiss criterion Q is defined by

$$Q := \frac{1}{2}(\|\Omega\|^2 - \|\mathbf{S}\|^2) = \frac{1}{4}\|\omega\|^2 - \frac{1}{2}\|\mathbf{S}\|^2. \quad (3.7)$$

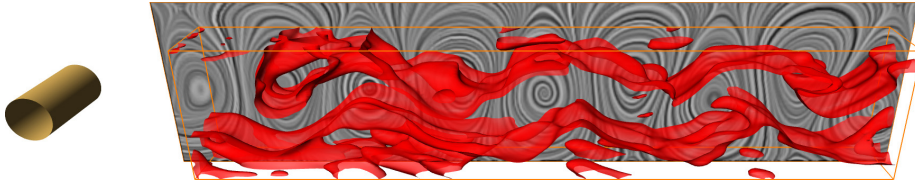


Figure 3.2: Isosurfaces of \mathbf{M}_Z showing Lagrangian vortex regions.

Where Q is positive, the vorticity magnitude dominates the rate of strain. Hence it is natural to define vortex regions as regions where $Q > 0$. Unlike λ_2 , Q has a physical meaning also where $Q < 0$. Here the rate of strain dominates the vorticity magnitude. The Q -criterion is an Eulerian quantity. Figure 3.1 shows isosurfaces of $Q > 0$ behind a circular cylinder indicating vortex features. The visualization additionally contains a LIC plane which shows rotational stream line behavior in the frame of reference corresponding to the convection velocity. The isosurfaces correspond to the circular pattern shown in the LIC plane. Note however that Q is a Galilean invariant quantity which is independent of such translational changes of the reference frame.

Q can be computed quickly using the identity $Q = -\sum_{i,j}(\nabla v)_{ij}(\nabla v)_{ji}$.

Note that in incompressible fields, the Okubo Weiss criterion coincides with the criterion of Hunt [Hun87].

3.1.2 λ_2

λ_2 , derived by [JH95], is closely related to Q . Consider the three real eigenvalues $\lambda_1 \leq \lambda_2 \leq \lambda_3$ of the symmetric matrix $\mathbf{S}^2 + \Omega^2$. In [JH95] it is deduced from the Navier Stokes equations that for a local pressure minimum two negative eigenvalues of this matrix are necessary. They define a vortex region where $\lambda_2 < 0$. In their work they show that $Q = -\frac{1}{2}(\lambda_1 + \lambda_2 + \lambda_3)$. Despite of this strong link they show that the λ_2 -criterion detects vortex regions more reliably especially under a strong external strain. Nevertheless, the λ_2 -criterion, unlike the Q -criterion, lacks a direct interpretation for regions where $\lambda_2 > 0$.

Q and λ_2 are related to the Navier Stokes equations and reflect the amount of strain and vortical motions in the vector field. Due to this fact those quantities are the most popular among fluid mechanicians.

Despite the convincing physical interpretation of Q and λ_2 , those quantities are of limited applicability in some settings. In [RP96] it is shown that for turbomachinery flow fields λ_2 is negative almost everywhere. So for highlighting regions of strong vortical activity thresholding is necessary, leaving the scientist with the question of choosing an appropriate isovalue. So vortex region detection has the drawback of being parameter dependent.

3.1.3 M_Z

Haller has recently proposed the \mathbf{M}_Z -criterion that also discriminates vortex and strain regions in incompressible flows similar to the Q -criterion, but in contrast to this based on a Lagrangian analysis [Hal05]. Figure 3.2 gives an example. As both the underlying theory and the implementation are quite involved, we give a deeper introduction here.

The \mathbf{M}_Z -criterion is based on a strain analysis along path lines. Loosely spoken, Haller proves that path lines along which the strain acceleration tensor M (the tensor

describing the first time derivative of V as defined in equation (3.5)) is positive definite are of saddle type – so called *hyperbolic* lines of maximal strain. In contrast he defines vortices as path lines along which M is indefinite. Such structures are called *elliptic*.

More precisely, Haller argues that in incompressible flows, the function V takes both positive and negative values, as \mathbf{S} has at least one negative and one positive eigenvalue. Hence, the set

$$Z = \{\xi | \langle \xi, \mathbf{S}(x(t), t) \xi \rangle = 0\} \quad (3.8)$$

is never empty and usually a two dimensional surface, as it separates the regions $V < 0$ and $V > 0$. Physically, within Z , initial bifurcations do not change their magnitude. Hence within Z , the bifurcation evolution is governed by the first time derivative

$$\frac{d}{dt}V(\xi(t), t). \quad (3.9)$$

Haller proves that path lines for which $\frac{d}{dt}V$ is positive for ξ in Z for all times are of saddle type in the sense that they form stable and unstable manifolds that drive advective mixing in the fluid.

To be able to decide this positivity, we state that

$$\begin{aligned} \frac{d}{dt}V(\xi(t), t) &= \frac{d}{dt}\langle \xi(t), \mathbf{S}(x(t), t) \xi(t) \rangle \\ &= \langle \dot{\xi}, \mathbf{S}(x(t), t) \xi \rangle + \langle \xi, \frac{d}{dt}(\mathbf{S}(x(t), t) \xi) \rangle \\ &= \langle \nabla v(x(t), t) \xi, \mathbf{S}(x(t), t) \xi \rangle \\ &\quad + \langle \xi, \frac{d}{dt}(\mathbf{S}(x(t), t)) \xi + \mathbf{S}(x(t), t) \dot{\xi} \rangle \\ &= \langle \xi, (\nabla v' \mathbf{S} + \frac{d}{dt}(\mathbf{S}(x(t), t)) + \mathbf{S} \nabla v) \xi \rangle \\ &= \langle \xi, M \xi \rangle \end{aligned} \quad (3.10)$$

where the strain acceleration tensor M is defined as

$$M = \nabla v' \mathbf{S} + \frac{d}{dt}(\mathbf{S}(x(t), t)) + \mathbf{S} \nabla v. \quad (3.11)$$

The \mathbf{M}_Z -criterion is now defined as follows:

Definition 2 \mathbf{M}_Z -criterion - total ellipticity time Let (x_0, t_0) be an arbitrary point in the space-time domain of the flow and $x(t)$ the path line with $x(t_0) = x_0$.

1. A point on the path line is called *hyperbolic*, if $\langle \xi, M \xi \rangle$ is positive for all ξ in Z at that point. Physically, such a point describes a saddle type or strain point. Otherwise, the point on the path line is called *elliptic* and indicates vortex behavior.
2. $\mathbf{M}_Z(x_0, t_0)$ is defined as the sum of the lengths of all time intervals, in which the path line started at (x_0, t_0) is elliptic.

Where $\mathbf{M}_Z = 0$, the path lines are hyperbolic for all times, and hence form stable or unstable manifolds as described above. In contrast, where \mathbf{M}_Z equals the integration time of the path lines, maximal vortex activity is present. However, in numerical implementations the path line seeding is necessarily sparse. Thus it is unlikely to find a path line that is completely hyperbolic or completely elliptic. But still, a qualitative property holds for the \mathbf{M}_Z -criterion computed on a sparse grid:

- The lower \mathbf{M}_Z , the more strain is present (the majority of points on the path line are hyperbolic).
- The higher \mathbf{M}_Z , the more vortical behavior is present (the majority of points on the path line are elliptic).

While Q and λ_2 are Galilean invariant, Haller states that \mathbf{M}_Z is even objective, see section 2.4.3.

3.1.4 Pressure

Within vortices, the pressure is lower than in the ambient flow field. Hence, pressure p is a vortex region quantity that indicates vortices at its minima. p does not relate to the flow gradient decomposition directly, but of course, p is linked to Ω and \mathbf{S} by the Navier-Stokes equation. Indeed, in the deduction of the λ_2 method, Jeong and Hussain used this link to prove that $\lambda_2 < 0$ is a necessary condition for the occurrence of a pressure minimum within some plain [JH95].

3.1.5 Rotation Strength

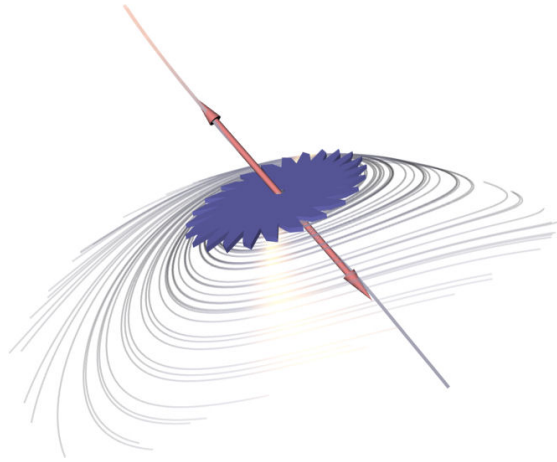
Rotation strength Δ as used in [SP03], see also [CPC90] is linked to the intuitive understanding that a vortex exhibits spiraling stream lines with respect to some specific reference frame. Within this reference frame, the stream line pattern of a flow field is dominated by its Jacobian Jv . If J has a conjugate pair of complex eigenvalues, the flow locally spirals in a plane corresponding to those eigenvectors. Δ is then defined as the magnitude of the imaginary part of those complex conjugate eigenvalues. So large values of Δ indicate strong spiraling patterns within the right reference frame. Where $\Delta = 0$, no such reference frame can be found. By considering the orientation of the corresponding eigenbasis, a rotation angle $\varphi \in (-\pi, \pi)$ can also be extracted. When $\varphi > 0$, the flow spirals counter clockwise around the eigenvector corresponding to the real eigenvalue, clockwise, if $\varphi < 0$.

3.2 Vortex Core Line Techniques

Several algorithms aim at extracting a line feature called the vortex core line. The motivation arises from the intuitive observation that a vortex might be regarded as circular particle movement around a common line.

The advantage of the schemes presented here over the vortex region approach is that they can be applied without user interaction, for instance as a batch job prior to visualization or during the simulation.

Figure 3.3: Flow pattern around a focus saddle. The blue plane is spanned by the eigenvectors corresponding to the complex eigenvalues. Swirling motion takes place in this plane around the real eigenvector denoted by the red arrows.



3.2.1 Sujudi/Haimes

Centers of swirling motion have first been treated by Sujudi and Haimes [SH95]. Their inspiration was the flow pattern around a certain type of critical point: a focus saddle (see Figure 3.3 for an illustration). Here, the Jacobian of the flow field has one real and two complex eigenvalues. The eigenvectors corresponding to the complex eigenvalues span a plane in which the flow spirals around the critical point. The eigenvector corresponding to the real eigenvalue denotes the axis of rotation.

Sujudi and Haimes generalized this flow pattern to non-critical points by considering the so called reduced velocity. At a point \mathbf{x} , the reduced velocity $\mathbf{w}(\mathbf{x})$ is given as the projection of the steady flow field $\mathbf{v}(\mathbf{x})$ to the plane normal to the real eigenvector \mathbf{e} by

$$\mathbf{w}(\mathbf{x}) = \mathbf{v}(\mathbf{x}) - (\mathbf{v}(\mathbf{x}) \cdot \mathbf{e}(\mathbf{x})) \mathbf{e}(\mathbf{x}). \quad (3.12)$$

They show that centers of swirling flow are line-type structures where $\mathbf{w}(\mathbf{x}) = 0$. Peikert et al. [PR99] formulated this using the Parallel Vectors operator and showed that $\mathbf{w}(\mathbf{x}) = 0$ is equivalent to $\mathbf{v}(\mathbf{x}) \parallel \nabla v$, i.e., \mathbf{v} and ∇v are parallel, avoiding eigenvalue computation this way.

3.2.2 Banks/Singer

Banks and Singer suggested specially adjusted vorticity lines [BS95]. Browsing through the grid vertices of the flow volume, seed points are detected by means of both a low pressure and a high vorticity threshold. Starting from those seed points, the algorithm proceeds as follows:

- integrate vorticity for some time
- span a plane perpendicular to vorticity and update the current location to the next local pressure minimum within the plane

They also give a method for computing the vortex hull by means of further pressure and vorticity threshold and give a space saving Fourier representation of the result.

3.2.3 Roth/Peikert

Roth and Peikert suggested a higher order method for vortex core line extraction of lines with zero torsion [RP98], which also involves the vector field v directly by finding line structures where v points into the direction of $(\nabla a)v$ where $a = (\nabla v)v$ is the steady acceleration of the vector field. This method is also capable of finding vortex cores that are bent strongly.

3.2.4 Pressure Minima

It was suggested by Miura and Kida [MK97] to extract valley lines of pressure. This approach was applied locally only and resulted in disconnected line segments. See Section 2.6 for an introduction to valley line extraction.

3.2.5 Parallel Vectors Operator

The idea of the PV approach is to derive two vector fields w_1, w_2 out of a given 3D vector field v , such that the desired vortex core lines are the locations where w_1 and w_2 are parallel. Several ways of extracting these lines exist [PR99], either based on extracting and intersecting isosurfaces [MR96], Newton iterations on grid faces, analytic solutions for triangular faces, or curve following schemes [BS94].

Part II

Detection Methods based on Swirling Motion

Chapter 4

Extraction of PV Surfaces

Flow dynamicists are interested in tracking vortex core lines over time for several reasons:

- When the impact of a feature on the flow is measured by certain criteria (rotation strength [SP03], pressure [MK97], Okubo-Weiss-criterion [Hun87]), feature tracking becomes necessary to answer the question, whether the impact of a feature on the flow increases or decreases, when time evolves, as the correspondence problem of features in different time steps is not a trivial task. With the full feature surface of our method at hand, the problem can be solved by simply checking, if two features at different times lie in the same connected component of the surface.
- Also, the spatial evolution over time is interesting e.g. in burning chambers, where the location and extent of vortices are the key ingredient for a complete burning process.

A first approach to tracking PV-based vortex core lines over time was given in [BP02] which focused on scale-space as the additional dimension. There, a marching-cubes-like algorithm is performed to extract 4D triangular structures in regular 4D hypercubes building the cells of the space-time domain.

In this chapter we introduce a new method to extract and track vortex core lines which are based on a PV formulation. This method is based on the concept of feature flow fields (FFF) [TS03]: we derive appropriate vector fields such that the searched vortex core lines are stream lines of them. This way, the extraction/tracking of vortex core lines is reduced to a simple stream line/surface integration of vector fields. We choose this approach because of the following reasons:

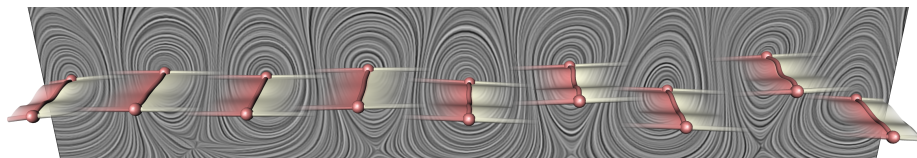


Figure 4.1: Flow behind a circular cylinder. Shown are vortex core lines in a certain frame of reference. Their evolution over time is tracked by our algorithm and depicted using transparent surfaces. Red color encodes the past while gray shows the future.

- Numerical stream line/surface integration is well-understood in the Visualization community. A variety of fast and stable algorithms exist for this purpose. [Hul92, vW93, Gel01, SBM⁺01]
- The stream surface integration approach is independent of an underlying grid, giving a subcell accuracy and relieving us of finding appropriate local connection strategies.
- Bifurcations (i.e. events of sudden changes of the behavior of vortex core lines over time) play an important role in the understanding of the dynamical behavior of vortex core lines. Contrary to pre-existing methods, the FFF approach permits to localize, characterize and classify these bifurcations. To the best of our knowledge, this has not been done in the Visualization community before.

The rest of the chapter is organized as follows: sections 4.1–4.4 describe our FFF based approach of extracting and tracking vortex core lines. Since the approach is exclusively based on the PV formulation, we describe the approach independently of the vortex core background. We call the solutions of the PV operator in the static case *PV lines*, while their sweeping over time in time-dependent fields are called *PV surfaces*. Section 4.1 explains the FFF approach to extract PV lines in static fields. Based on this, section 4.2 introduces the feature flow fields to tracking PV lines over time. Section 4.3 gives a complete classification of local bifurcations of PV lines over time. Section 4.4 describes the final algorithms to tracking PV lines. Section 4.5 shows the application of our technique to a number of test data sets, among them the particular PV realization for vortex core lines defined by Sujudi and Haimes [SH95] in a Galerkin model of a flow behind a circular cylinder.

Notation: In this chapter we consider points, vectors and vector fields both in 3D and 4D. To make a clear distinction between them, we write a 4D structure as $\tilde{\mathbf{p}}$, $\tilde{\mathbf{v}}$, ..., while for 3D structures we simply write \mathbf{p} , \mathbf{v} .

4.1 Extracting PV Lines in Static Fields

In this section we shortly describe the parallel vectors (PV) operator [PR99] and explain how to use the concept of feature flow fields (FFF) in order to extract PV lines in a 3D static field. This will later be the foundation of an FFF-based algorithm for tracking PV lines in unsteady fields.

Given two continuous 3D vector fields \mathbf{w}_1 and \mathbf{w}_2 , the PV operator extracts all points in the domain where the vectors of \mathbf{w}_1 and \mathbf{w}_2 point in the same direction, i.e. $\mathbf{w}_2 = \lambda \mathbf{w}_1$ for some real λ ¹, or $\mathbf{w}_1 \parallel \mathbf{w}_2$ in shorthand notation.

Peikert and Roth [PR99] list examples of reasonable choices of \mathbf{w}_1 and \mathbf{w}_2 to extract line type features like vortex core lines in flow fields or extremum lines in scalar fields. As the theory presented here can be formulated completely in terms of the PV operator, we keep the derivation as general as possible and just choose particular vector fields \mathbf{w}_1 and \mathbf{w}_2 in section 4.5.

Aiming at extracting PV lines of $(\mathbf{w}_1, \mathbf{w}_2)$ in the domain $D = [x_{min}, x_{max}] \times [y_{min}, y_{max}] \times$

¹ $\lambda = \pm\infty$ is also allowed, i.e., $\mathbf{w}_1 \parallel \mathbf{w}_2$ holds if \mathbf{w}_1 or \mathbf{w}_2 vanishes.

$[z_{min}, z_{max}]$, we define the vector field \mathbf{s} as

$$\mathbf{s}(x, y, z) = \begin{pmatrix} k(x, y, z) \\ m(x, y, z) \\ n(x, y, z) \end{pmatrix} = \mathbf{w}_1 \times \mathbf{w}_2. \quad (4.1)$$

Then the PV lines consist of all locations (x, y, z) with $\mathbf{s}(x, y, z) = (0, 0, 0)^T$. If \mathbf{w}_1 and \mathbf{w}_2 are continuous, then the PV lines are indeed continuous line structures, i.e. point sets of dimensionality 1 [PR99]². PV structures of dimensionality 0 or dimensionality 2 are structurally unstable in 3D, i.e. they disappear by adding noise to the data. For this reason we do not consider them here.

The FFF approach [TS03] (originally introduced to track critical points of time-dependent vector fields) was already used for extracting particular Galilean invariant vortex core lines without using the PV approach [SWH05a], see chapter 6. To apply the FFF concept to PV lines, the following steps are necessary:

1. A vector field \mathbf{f} is defined which fulfills the FFF property. This means that given a point $\mathbf{x}_0 = (x_0, y_0, z_0)$ with $\mathbf{s}(\mathbf{x}_0) = (0, 0, 0)^T$, each point \mathbf{x} on the stream line of \mathbf{f} starting from \mathbf{x}_0 fulfills $\mathbf{s}(\mathbf{x}) = (0, 0, 0)^T$ as well. In other words: PV lines of $(\mathbf{w}_1, \mathbf{w}_2)$ are stream lines of \mathbf{f} .
2. A set of starting points is defined which guarantees that the stream line integration of \mathbf{f} starting from them covers all PV lines.

Then all PV lines of $(\mathbf{w}_1, \mathbf{w}_2)$ can simply be extracted by applying a stream line integration of \mathbf{f} . We treat the two parts of the approach in the following subsections.

4.1.1 Obtaining the Feature Flow Field \mathbf{f}

In this section we show that \mathbf{f} essentially consists of an appropriate combination of the first order partials s_x, s_y, s_z of \mathbf{s} . We denote the gradients of the components of \mathbf{s} in (4.1) by

$$\nabla k = \begin{pmatrix} k_x \\ k_y \\ k_z \end{pmatrix}, \quad \nabla m = \begin{pmatrix} m_x \\ m_y \\ m_z \end{pmatrix}, \quad \nabla n = \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix}.$$

Then \mathbf{f} should point into a direction where the components of \mathbf{s} remain constant. This means that \mathbf{f} has to be perpendicular to the gradients of the components of \mathbf{s} . We define

$$\mathbf{f}_1 = \nabla m \times \nabla n = \begin{pmatrix} \det(\mathbf{s}_y, \mathbf{s}_z, (1, 0, 0)^T) \\ \det(\mathbf{s}_z, \mathbf{s}_x, (1, 0, 0)^T) \\ \det(\mathbf{s}_x, \mathbf{s}_y, (1, 0, 0)^T) \end{pmatrix}. \quad (4.2)$$

²Note that this statement gives that these line structures cannot be obtained by replacing $\mathbf{s}(x, y, z) = (0, 0, 0)^T$ by $\|\mathbf{s}(x, y, z)\| = 0$ and applying a simple scalar field analysis of $\|\mathbf{s}\|$, since the zeros of general scalar fields are structures of dimensionality 2.

which is perpendicular to ∇m and ∇n . Hence, all points on a stream line of \mathbf{f}_1 have constant components m and n . In a similar way we define \mathbf{f}_2 and \mathbf{f}_3 as

$$\mathbf{f}_2 = \nabla n \times \nabla k = \begin{pmatrix} \det(\mathbf{s}_y, \mathbf{s}_z, (0, 1, 0)^T) \\ \det(\mathbf{s}_z, \mathbf{s}_x, (0, 1, 0)^T) \\ \det(\mathbf{s}_x, \mathbf{s}_y, (0, 1, 0)^T) \end{pmatrix} \quad (4.3)$$

$$\mathbf{f}_3 = \nabla k \times \nabla m = \begin{pmatrix} \det(\mathbf{s}_y, \mathbf{s}_z, (0, 0, 1)^T) \\ \det(\mathbf{s}_z, \mathbf{s}_x, (0, 0, 1)^T) \\ \det(\mathbf{s}_x, \mathbf{s}_y, (0, 0, 1)^T) \end{pmatrix}. \quad (4.4)$$

In general, $\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3$ describe different directions. However, if we are on a PV line, we can write $\mathbf{w}_2 = \lambda \mathbf{w}_1$ for a certain λ . Inserting this into the computation of the derivatives of (4.1), we get

$$\begin{aligned} \mathbf{s}_x &= (\mathbf{w}_{1x} \times \mathbf{w}_2) + (\mathbf{w}_1 \times \mathbf{w}_{2x}) = \mathbf{w}_1 \times (\mathbf{w}_{2x} - \lambda \mathbf{w}_{1x}) \\ \mathbf{s}_y &= (\mathbf{w}_{1y} \times \mathbf{w}_2) + (\mathbf{w}_1 \times \mathbf{w}_{2y}) = \mathbf{w}_1 \times (\mathbf{w}_{2y} - \lambda \mathbf{w}_{1y}) \\ \mathbf{s}_z &= (\mathbf{w}_{1z} \times \mathbf{w}_2) + (\mathbf{w}_1 \times \mathbf{w}_{2z}) = \mathbf{w}_1 \times (\mathbf{w}_{2z} - \lambda \mathbf{w}_{1z}) \end{aligned} \quad (4.5)$$

which shows that $\mathbf{s}_x, \mathbf{s}_y, \mathbf{s}_z$ are coplanar perpendicular to \mathbf{w}_1 and \mathbf{w}_2 . (4.5) and (4.2)–(4.4) give that $\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3$ are parallel on a PV line. Thus, almost every linear combination of $\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3$ can act as feature flow field. Setting $\mathbf{f} = \alpha \mathbf{f}_1 + \beta \mathbf{f}_2 + \gamma \mathbf{f}_3$ and $\mathbf{a} = (\alpha, \beta, \gamma)^T$, we obtain

$$\mathbf{f} = \begin{pmatrix} e \\ f \\ g \end{pmatrix} = \begin{pmatrix} \det(\mathbf{s}_y, \mathbf{s}_z, \mathbf{a}) \\ \det(\mathbf{s}_z, \mathbf{s}_x, \mathbf{a}) \\ \det(\mathbf{s}_x, \mathbf{s}_y, \mathbf{a}) \end{pmatrix}. \quad (4.6)$$

Choosing the vector field \mathbf{a} :

In order to choose a suitable vector field \mathbf{a} , we rewrite \mathbf{f} as

$$\mathbf{f} = \begin{pmatrix} \mathbf{b}_1 \cdot \mathbf{a} \\ \mathbf{b}_2 \cdot \mathbf{a} \\ \mathbf{b}_3 \cdot \mathbf{a} \end{pmatrix} \quad (4.7)$$

with

$$\mathbf{b}_1 = \mathbf{s}_y \times \mathbf{s}_z, \quad \mathbf{b}_2 = \mathbf{s}_z \times \mathbf{s}_x, \quad \mathbf{b}_3 = \mathbf{s}_x \times \mathbf{s}_y. \quad (4.8)$$

On a PV line, the coplanarity of $\mathbf{s}_x, \mathbf{s}_y, \mathbf{s}_z$ together with (4.8) gives $\mathbf{w}_1 \parallel \mathbf{w}_2 \parallel \mathbf{b}_1 \parallel \mathbf{b}_2 \parallel \mathbf{b}_3$. Consequently, the only condition we have to put on \mathbf{a} is that it must not be perpendicular to \mathbf{w}_1 and \mathbf{w}_2 respectively on a PV line. If we know that \mathbf{w}_1 never vanishes on a PV line, the simple choice $\mathbf{a} = \mathbf{w}_1$ does the job. A similar statement holds for \mathbf{w}_2 . In case that both \mathbf{w}_1 and \mathbf{w}_2 may vanish on a PV line, we choose

$$\mathbf{a} = \begin{cases} \mathbf{w}_1 & \text{if } \|\mathbf{w}_1\| \geq \|\mathbf{w}_2\| \\ \mathbf{w}_2 & \text{otherwise,} \end{cases}$$

which guarantees \mathbf{a} to be continuous in direction but not in orientation. Thus, \mathbf{f} has to be integrated as an orientation-free vector field (similar e.g. to an eigenvector field of a tensor field), i.e., the local orientation has to be obtained from the information where the integration of the line has come from.

A number of vortex core line extraction concepts based on parallel vectors use $\mathbf{w}_1 = \mathbf{v}$ and $\mathbf{w}_2 = \mathbf{M} \mathbf{v}$ (see for example [SH95] with $\mathbf{M} = \nabla \mathbf{v}$ or [TM98] with $\mathbf{M} =$

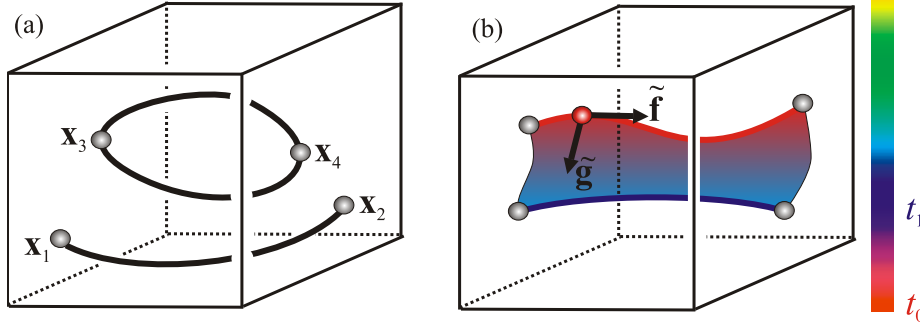


Figure 4.2: (a) two kinds of PV lines: ending in the boundary points $\mathbf{x}_1, \mathbf{x}_2$, or closed: in this case we extract two starting points $\mathbf{x}_3, \mathbf{x}_4$; (b) PV structures are surfaces in 4D.

$(\nabla \mathbf{v})^T$). For these approaches, both \mathbf{w}_1 and \mathbf{w}_2 vanish at critical points of \mathbf{v} causing both $\mathbf{s}(\mathbf{x}) = (0, 0, 0)^T$ and $\mathbf{f}(\mathbf{x}) = (0, 0, 0)^T$ there. To deal with this problem, we equivalently reformulate \mathbf{w}_2 as an appropriate eigenvector of \mathbf{M} . This eigenvector does not vanish along the vortex core line, but since eigenvectors have no orientation, an orientation-free integration of \mathbf{f} is necessary here as well.

4.1.2 Starting Points for integrating \mathbf{f}

Given the definitions above, we first analyze whether an integration of \mathbf{f} along a PV line may get stuck in a critical point of \mathbf{f} . This happens at a location $\mathbf{x} \in D$ with $\mathbf{s}(\mathbf{x}) = (0, 0, 0)^T$ and $\mathbf{f}(\mathbf{x}) = (0, 0, 0)^T$. Both conditions independently build structures of dimensionality 1 in D , i.e., they are line structures³. If these line structures intersect, the intersection points are structurally unstable, i.e., they disappear by adding noise to the data. Because of this we assume that such intersections do not exist in D . However, we mention that $[\mathbf{s}(\mathbf{x}) = (0, 0, 0)^T, \mathbf{f}(\mathbf{x}) = (0, 0, 0)^T]$ gives stable solutions in a time-depending setting. We treat this in section 4.3.

If $[\mathbf{s}(\mathbf{x}) = (0, 0, 0)^T, \mathbf{f}(\mathbf{x}) = (0, 0, 0)^T]$ does not have solutions in D , every PV line ends either on the boundary faces of D (for both forward and backward integration), or builds a closed stream line in \mathbf{f} [PR99]⁴. To get the starting points for the first case, we search for the intersections of the PV lines with the boundary of D : for the boundary face $x = x_{min}$, we search for all points (y, z) with $\mathbf{s}(x_{min}, y, z) = (0, 0, 0)^T$. To do so, different numerical solvers can be applied. We use a simple subdivision approach in the (y, z) -domain: a rectangular cell C is checked whether one of the components k, m, n is positive/negative at *all* 4 corners of C . If so, no PV line intersection is found inside C . Otherwise, we recursively subdivide C into 4 subcells until their size is smaller than a certain threshold. In a similar way we compute the intersections of the PV lines with the remaining 5 faces.

To find a starting point on a closed PV line, it is sufficient to identify an arbitrary

³This has been shown for \mathbf{s} , as it defines PV lines [PR99]. To show that $\mathbf{f}(\mathbf{x}) = (0, 0, 0)^T$ builds line structures as well, we have to show that \mathbf{f} can also be formulated using the PV operator. (4.2)–(4.4) give that we can rewrite \mathbf{f} as $\mathbf{f} = (\alpha \nabla m - \beta \nabla k) \times (\nabla n - \frac{\gamma}{\beta} \nabla m)$. Hence, $\mathbf{f} = (0, 0, 0)^T$ corresponds to $(\alpha \nabla m - \beta \nabla k) \parallel (\nabla n - \frac{\gamma}{\beta} \nabla m)$.

⁴This statement implies that our approach does not have to incorporate algorithms to detecting closed stream lines in flow fields [WS01b], since we know in advance that our stream lines of interest in \mathbf{f} are closed.

point on the line. We have chosen to extract points \mathbf{x} with

$$[\mathbf{s}(\mathbf{x}) = (0,0,0)^T, \quad e(\mathbf{x}) = 0] \quad (4.9)$$

with e from (4.6). To do so, we apply a similar 3D subdivision approach as described above for the 2D case. Since a closed PV line must consist of points with both positive and negative e -components, each closed PV line must consist of at least two points fulfilling (4.9). Figure 4.2a illustrates an example. Clearly, (4.9) may also deliver solutions on open PV lines, but it guarantees to find at least two solutions for each closed PV line⁵. Finally, we do an integration of \mathbf{f} starting from all detected points and remove multiply obtained curves.

4.2 Feature Flow Fields for Tracking

Now we consider PV structures in time-dependent 3D vector fields $\mathbf{w}_1(x, y, z, t), \mathbf{w}_2(x, y, z, t)$. To do so, we first note that all the 3D static vector fields $\mathbf{w}_1, \mathbf{w}_2, \mathbf{s}, \nabla k, \nabla m, \nabla n, \mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3, \mathbf{f}, \mathbf{a}$ which were introduced in section 4.1 can be defined in a similar way for the time-dependent case as well. In the following we consider these vector fields to be time-dependent, i.e. they are maps from the 4D domain $\tilde{D} = D \times [t_{min}, t_{max}]$ to \mathbf{R}^3 .

PV structures in $(\mathbf{w}_1(x, y, z, t), \mathbf{w}_2(x, y, z, t))$ can be considered as lines in D sweeping over time while smoothly changing their shape and location. In addition, certain bifurcations may occur. Hence, the PV structures in \tilde{D} have the dimensionality 2, i.e. they are surfaces in \tilde{D} . Figure 4.2b gives an illustration. Here a PV line at time t_0 (red) moves to the blue line at time t_1 . Each point on the swept surface between the two lines is actually a 4D point: in addition to the spatial values it is provided with a t -value. In figure 4.2b (as well as in the following figures) we color code the t -values of points, lines and surfaces.

In order to extract the PV surfaces in \tilde{D} , we need to define two 4D feature flow fields $\tilde{\mathbf{f}}$ and $\tilde{\mathbf{g}}$. The first one can easily be defined as

$$\tilde{\mathbf{f}}(x, y, z, t) = \begin{pmatrix} \mathbf{f}(x, y, z, t) \\ 0 \end{pmatrix} \quad (4.10)$$

where \mathbf{f} is defined in (4.6). It gives a PV line at a certain time level, i.e. all points on a stream line of $\tilde{\mathbf{f}}$ have the same t -value. The evolution in time of a PV line should be covered by the 4D feature flow field $\tilde{\mathbf{g}}$. Keeping in mind that PV structures in \tilde{D} are surfaces, a family of different $\tilde{\mathbf{g}}$ could be chosen such that each linear combination of $\tilde{\mathbf{f}}$ and $\tilde{\mathbf{g}}$ is a FFF. Among them, we choose the $\tilde{\mathbf{g}}$ with $\tilde{\mathbf{f}} \perp \tilde{\mathbf{g}}$. This gives a unique $\tilde{\mathbf{g}}$ (except for scaling). We obtain

$$\tilde{\mathbf{g}}(x, y, z, t) = \begin{pmatrix} \mathbf{h} \times \mathbf{f} \\ \|\mathbf{f}\|^2 \end{pmatrix} = \begin{pmatrix} \mathbf{h} \times \mathbf{f} \\ e^2 + f^2 + g^2 \end{pmatrix} \quad (4.11)$$

with

$$\mathbf{h}(x, y, z, t) = \begin{pmatrix} \det(\mathbf{s}_x, \mathbf{s}_t, \mathbf{a}) \\ \det(\mathbf{s}_y, \mathbf{s}_t, \mathbf{a}) \\ \det(\mathbf{s}_z, \mathbf{s}_t, \mathbf{a}) \end{pmatrix} \quad (4.12)$$

⁵If a closed PV line completely lies in the $y-z$ plane by chance, (4.9) gives many solutions. In this case, $e(\mathbf{x}) = 0$ can simply be replaced by $f(\mathbf{x}) = 0$ in (4.9) to reduce the number of solutions.

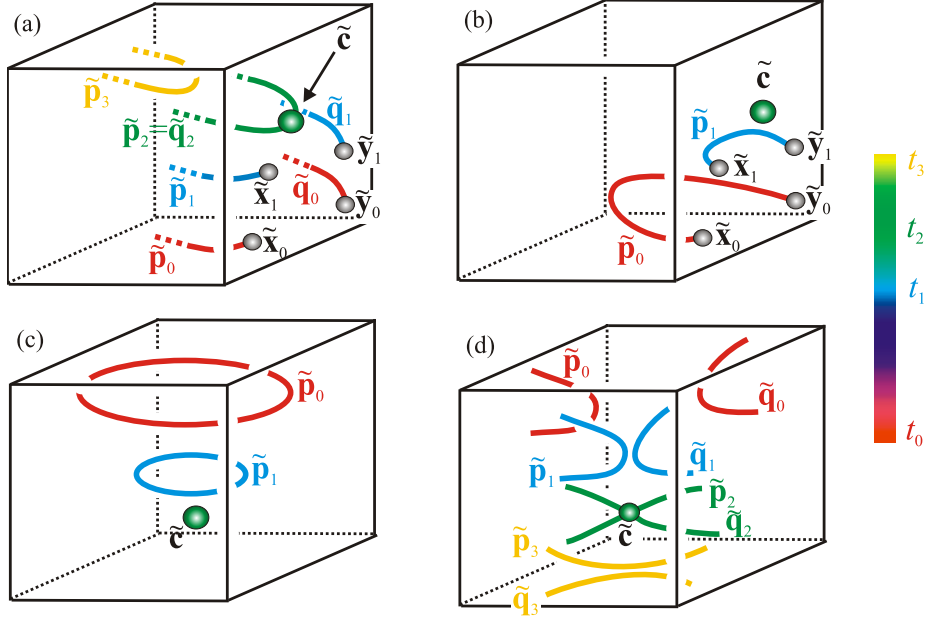


Figure 4.3: local bifurcations of PV lines: (a) inflow boundary bifurcation; (b) outflow boundary bifurcation; (c) closed collapse bifurcation; (d) saddle bifurcation.

and \mathbf{f} defined in (4.6). Figure 4.2b illustrates $\tilde{\mathbf{f}}$ and $\tilde{\mathbf{g}}$ at a certain point (red) on the PV surface.

To prove that $\tilde{\mathbf{g}}$ is indeed the desired feature flow field, we consider the gradients in \tilde{D} of the components of \mathbf{s} :

$$\tilde{\nabla}k = \begin{pmatrix} k_x \\ k_y \\ k_z \\ k_t \end{pmatrix}, \quad \tilde{\nabla}m = \begin{pmatrix} m_x \\ m_y \\ m_z \\ m_t \end{pmatrix}, \quad \tilde{\nabla}n = \begin{pmatrix} n_x \\ n_y \\ n_z \\ n_t \end{pmatrix}.$$

Then we have to show that from $\mathbf{w}_1 \parallel \mathbf{w}_2$ (i.e. for $\mathbf{s} = (0,0,0)^T$) the following four properties can be deduced:

$$\tilde{\nabla}k \cdot \tilde{\mathbf{g}} = \tilde{\nabla}m \cdot \tilde{\mathbf{g}} = \tilde{\nabla}n \cdot \tilde{\mathbf{g}} = \tilde{\mathbf{f}} \cdot \tilde{\mathbf{g}} = 0$$

where \cdot denotes the 4D dot product. This can be shown by a straightforward exercise in algebra.

We note that Theisel et al. already proposed a FFF for PV tracking [TS03] which appears not to work: the FFF proposed there is constantly vanishing on a PV line and therefore unable to track it.

4.3 Local Bifurcations

Although in general PV lines change smoothly over time, there are certain points \tilde{D} in which the behavior of the PV lines changes abruptly. These bifurcations are a vital

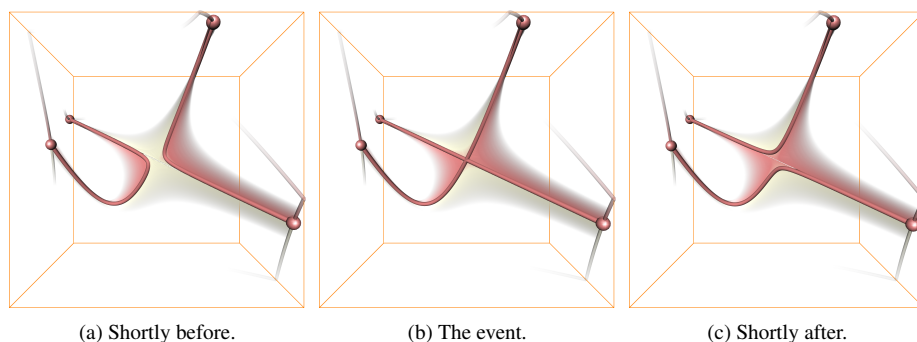


Figure 4.4: Saddle bifurcation.

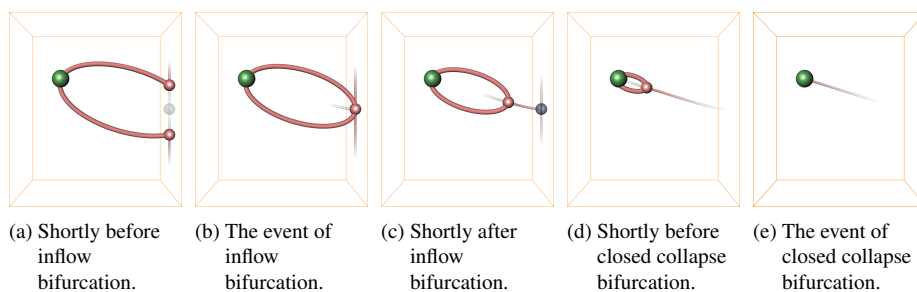


Figure 4.5: Inflow bifurcation and closed collapse bifurcation.

ingredient for assuring the complete extraction of PV surfaces. Furthermore, knowing what kind of bifurcations may occur contributes to understanding the parallel vectors operator. In this section we characterize local bifurcations and show how to extract them. In general, we can distinguish between two kinds of bifurcations: inner bifurcations and boundary bifurcations. We treat them separately in the following sections.

4.3.1 Inner Bifurcations

An inner bifurcation is characterized by the fact that the integration of $\tilde{\mathbf{f}}$ on a PV surface in \tilde{D} ends in a critical point of $\tilde{\mathbf{f}}$. This means that an inner bifurcation occurs at a location $\tilde{\mathbf{c}} \in \tilde{D}$ with

$$[\mathbf{s}(\tilde{\mathbf{c}}) = (0,0,0)^T , \mathbf{f}(\tilde{\mathbf{c}}) = (0,0,0)^T]. \quad (4.13)$$

Since both $\mathbf{s}(\tilde{\mathbf{c}}) = (0,0,0)^T$ and $\mathbf{f}(\tilde{\mathbf{c}}) = (0,0,0)^T$ gives surfaces in \tilde{D} as solutions⁶, their intersections are stable isolated points in \tilde{D} . To get them, we use a subdivision approach in 4D similar to the one already explained in 2D and 3D.

In order to analyze the behavior of the PV lines around an inner bifurcation, we analyze the Jacobian matrix of $\tilde{\mathbf{f}}$ in $\tilde{\mathbf{c}}$ which is a common approach in the field of vector field topology to classify critical points of 2D [HH89] and 3D [WTHS04a] vector fields. On $\tilde{\mathbf{c}}$ fulfilling (4.13) we know from (4.7) that $\mathbf{b}_1 = \mathbf{b}_2 = \mathbf{b}_3 = (0,0,0)^T$. This

⁶This is due to the fact that both $\mathbf{s}(\tilde{\mathbf{c}}) = (0,0,0)^T$ and $\mathbf{f}(\tilde{\mathbf{c}}) = (0,0,0)^T$ can be interpreted as sweeping lines over time.

and (4.8) gives that $\mathbf{s}_x \parallel \mathbf{s}_y \parallel \mathbf{s}_z$, i.e. we can set $\mathbf{s}_y = p \mathbf{s}_x$ and $\mathbf{s}_z = q \mathbf{s}_x$ for certain p, q . Inserting this into the derivatives of (4.8), we get

$$\begin{aligned}
\mathbf{b}_{1x} &= (\mathbf{s}_{yx} \times \mathbf{s}_z) + (\mathbf{s}_y \times \mathbf{s}_{zx}) = \mathbf{s}_x \times (p \mathbf{s}_{zx} - q \mathbf{s}_{yx}) \\
\mathbf{b}_{1y} &= (\mathbf{s}_{yy} \times \mathbf{s}_z) + (\mathbf{s}_y \times \mathbf{s}_{zy}) = \mathbf{s}_x \times (p \mathbf{s}_{zy} - q \mathbf{s}_{yy}) \\
\mathbf{b}_{1z} &= (\mathbf{s}_{yz} \times \mathbf{s}_z) + (\mathbf{s}_y \times \mathbf{s}_{zz}) = \mathbf{s}_x \times (p \mathbf{s}_{zz} - q \mathbf{s}_{yz}) \\
\mathbf{b}_{2x} &= (\mathbf{s}_{zx} \times \mathbf{s}_x) + (\mathbf{s}_z \times \mathbf{s}_{xx}) = \mathbf{s}_x \times (q \mathbf{s}_{xx} - \mathbf{s}_{zx}) \\
\mathbf{b}_{2y} &= (\mathbf{s}_{zy} \times \mathbf{s}_x) + (\mathbf{s}_z \times \mathbf{s}_{xy}) = \mathbf{s}_x \times (q \mathbf{s}_{xy} - \mathbf{s}_{zy}) \\
\mathbf{b}_{2z} &= (\mathbf{s}_{zz} \times \mathbf{s}_x) + (\mathbf{s}_z \times \mathbf{s}_{xz}) = \mathbf{s}_x \times (q \mathbf{s}_{xz} - \mathbf{s}_{zz}) \\
\mathbf{b}_{3x} &= (\mathbf{s}_{xx} \times \mathbf{s}_y) + (\mathbf{s}_x \times \mathbf{s}_{yx}) = \mathbf{s}_x \times (\mathbf{s}_{yx} - p \mathbf{s}_{xx}) \\
\mathbf{b}_{3y} &= (\mathbf{s}_{xy} \times \mathbf{s}_y) + (\mathbf{s}_x \times \mathbf{s}_{yy}) = \mathbf{s}_x \times (\mathbf{s}_{yy} - p \mathbf{s}_{xy}) \\
\mathbf{b}_{3z} &= (\mathbf{s}_{xz} \times \mathbf{s}_y) + (\mathbf{s}_x \times \mathbf{s}_{yz}) = \mathbf{s}_x \times (\mathbf{s}_{yz} - p \mathbf{s}_{xz}).
\end{aligned} \tag{4.14}$$

This shows that the 9 vectors in (4.14) are coplanar perpendicular to $\mathbf{s}_x, \mathbf{s}_y, \mathbf{s}_z$. In addition, the following statements follow directly from (4.14):

$$\mathbf{b}_{1x} + p \mathbf{b}_{2x} + q \mathbf{b}_{3x} = (0, 0, 0)^T \tag{4.15}$$

$$\mathbf{b}_{1y} + p \mathbf{b}_{2y} + q \mathbf{b}_{3y} = (0, 0, 0)^T \tag{4.16}$$

$$\mathbf{b}_{1z} + p \mathbf{b}_{2z} + q \mathbf{b}_{3z} = (0, 0, 0)^T \tag{4.17}$$

$$\mathbf{b}_{1x} + \mathbf{b}_{2y} + \mathbf{b}_{3z} = (0, 0, 0)^T. \tag{4.18}$$

Keeping $\mathbf{b}_1 = \mathbf{b}_2 = \mathbf{b}_3 = (0, 0, 0)^T$ in mind, we can write the Jacobian matrix of $\mathbf{f}(\tilde{\mathbf{c}})$ as

$$\mathbf{J}_{\mathbf{f}(\tilde{\mathbf{c}})} = \begin{pmatrix} \mathbf{b}_{1x} \cdot \mathbf{a} & \mathbf{b}_{1y} \cdot \mathbf{a} & \mathbf{b}_{1z} \cdot \mathbf{a} \\ \mathbf{b}_{2x} \cdot \mathbf{a} & \mathbf{b}_{2y} \cdot \mathbf{a} & \mathbf{b}_{2z} \cdot \mathbf{a} \\ \mathbf{b}_{3x} \cdot \mathbf{a} & \mathbf{b}_{3y} \cdot \mathbf{a} & \mathbf{b}_{3z} \cdot \mathbf{a} \end{pmatrix}. \tag{4.19}$$

(4.15)–(4.17) show that the lines of $\mathbf{J}_{\mathbf{f}(\tilde{\mathbf{c}})}$ are not independent, which gives $\det(\mathbf{J}_{\mathbf{f}(\tilde{\mathbf{c}})}) = 0$. Hence, one eigenvalue of $\mathbf{J}_{\mathbf{f}(\tilde{\mathbf{c}})}$ is zero. From (4.18) we infer that the trace of $\mathbf{J}_{\mathbf{f}(\tilde{\mathbf{c}})}$ is zero. As the trace of a matrix equals the sum of its eigenvalues, we see that also the remaining eigenvalues of $\mathbf{J}_{\mathbf{f}(\tilde{\mathbf{c}})}$ add to zero. So they can be written as

$$0, -\sqrt{r}, \sqrt{r}$$

for some real, possibly negative r . Hence we can classify three kinds of inner bifurcations. The first, $r = 0$, is a generally unstable higher order inner bifurcation and not considered here. For $r \neq 0$, exactly two stable kinds of inner bifurcations are possible depending on the sign of r :

A *closed collapse bifurcation* appears if (4.13) and $r < 0$ hold. In this case, the two non-zero eigenvalues of $\mathbf{J}_{\mathbf{f}(\tilde{\mathbf{c}})}$ are purely imaginary indicating a rotational behavior of \mathbf{f} around $\tilde{\mathbf{c}}$. While figure 4.3c illustrates this, figure 4.5 depicts this bifurcation using a test data set, see section 4.5 for details on the used visualization scheme. Now imagine a closed PV line $\tilde{\mathbf{p}}_0$ at the time t_0 . While moving forward in time ($t = t_1$), the closed PV line $\tilde{\mathbf{p}}_1$ becomes smaller until at a certain time t_2 it collapses to a point $\tilde{\mathbf{c}}$ and disappears. Note that the inverse case of a closed collapse bifurcation exists as well indicating the birth of a (small) closed PV line.

A *saddle bifurcation* appears if (4.13) and $r > 0$ hold. See figure 4.3d for an illustration and figure 4.4 for an example from a test data set. Two PV lines $\tilde{\mathbf{p}}_0, \tilde{\mathbf{q}}_0$ at the time $t = t_0$ move towards each other ($t = t_1$), share a common point $\tilde{\mathbf{c}}$ at the time $t = t_2$, and move away from each other ($t = t_3$). The directions of the PV lines out of $\tilde{\mathbf{c}}$ are the directions of the two eigenvectors of $\mathbf{J}_{\mathbf{f}(\tilde{\mathbf{c}})}$ corresponding to the non-zero eigenvalues.

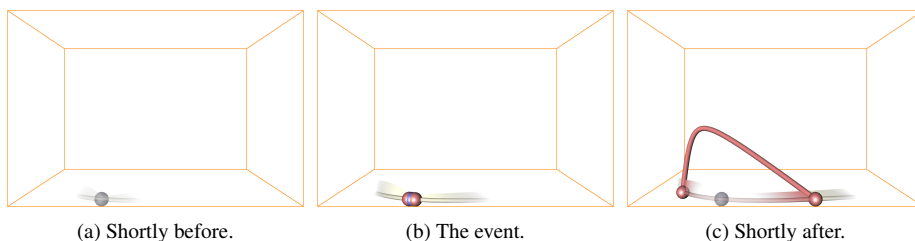


Figure 4.6: Reversed outflow boundary bifurcation.

4.3.2 Boundary Bifurcations

A boundary bifurcation is characterized by the fact that $\tilde{\mathbf{f}}$ on a PV surface is tangential to the boundary surface of \tilde{D} . A boundary bifurcation $\tilde{\mathbf{c}} = (x_{\tilde{\mathbf{c}}}, y_{\tilde{\mathbf{c}}}, z_{\tilde{\mathbf{c}}}, t_{\tilde{\mathbf{c}}})$ on the boundary face $x = x_{max}$ is the solution of

$$[\mathbf{s}(\tilde{\mathbf{c}}) = (0, 0, 0)^T, \quad x_{\tilde{\mathbf{c}}} = x_{max}, \quad e(\tilde{\mathbf{c}}) = 0] \quad (4.20)$$

which gives isolated points in the stable case. To get them, we may apply a 3D subdivision approach (in (x, y, t) -space) similar as described in section 4.1.2. However, there is a faster approach which will be explained later in section 4.4. In a similar way we compute the boundary bifurcations for the remaining boundary faces of D .

At a boundary bifurcation $\tilde{\mathbf{c}}$, the integration of \mathbf{f} starting from $\tilde{\mathbf{c}}$ (both in forward and backward direction) may enter \tilde{D} , or it may leave \tilde{D} immediately after starting the integration. To distinguish these two kinds of behavior, we check whether directional derivative $\nabla \mathbf{f} \cdot \mathbf{f}$ of \mathbf{f} points inside or outside D . In the first case, we have an *inflow boundary bifurcation*. See figure 4.3a for an illustrating example. Imagine two PV lines $\tilde{\mathbf{p}}_0$ and $\tilde{\mathbf{q}}_0$ at the time $t = t_0$ which leave \tilde{D} at the points $\tilde{\mathbf{x}}_0$ and $\tilde{\mathbf{y}}_0$ respectively. While moving forward in time ($t = t_1$), the exit points $\tilde{\mathbf{x}}_1, \tilde{\mathbf{y}}_1$ of the current PV lines $\tilde{\mathbf{p}}_1, \tilde{\mathbf{q}}_1$ move towards each other until at a certain time t_2 they collapse to a point $\tilde{\mathbf{c}}$. At $\tilde{\mathbf{c}}$ the current PV lines $\tilde{\mathbf{p}}_2, \tilde{\mathbf{q}}_2$ get smoothly connected and build a single PV line $\tilde{\mathbf{p}}_3$ from this moment on. The point $\tilde{\mathbf{c}}$ denotes the inflow boundary collapse bifurcations.

An *outflow boundary bifurcation* illustrated in figure 4.3b and shown in figure 4.6 within a test data set. Here, the PV line $\tilde{\mathbf{p}}_0$ at the time t_0 enters and leaves \tilde{D} at the points $\tilde{\mathbf{x}}_0$ and $\tilde{\mathbf{y}}_0$. While moving forward in time ($t = t_1$), the exit points $\tilde{\mathbf{x}}_1, \tilde{\mathbf{y}}_1$ move towards each other, until at a certain time t_2 they collapse in the point $\tilde{\mathbf{c}}$ making the PV line disappear.

Also for boundary bifurcations the reverse cases exist. At an inflow boundary bifurcation a PV line may split up into two lines, and at an inflow boundary bifurcation a PV line may appear.

4.3.3 Further Bifurcations

After introducing the local bifurcations above, one may ask whether there are more bifurcations possible. In particular we check whether fold bifurcations of (open or closed) PV lines exist. A fold bifurcation occurs when two PV lines move toward each other, merge at a certain time and immediately disappear after that⁷. It turns out that

⁷In the field of time-dependent vector field topology, we have similar fold bifurcations for critical points [TSH01] and isolated closed stream lines (cyclic fold bifurcation [TWHS04b]).

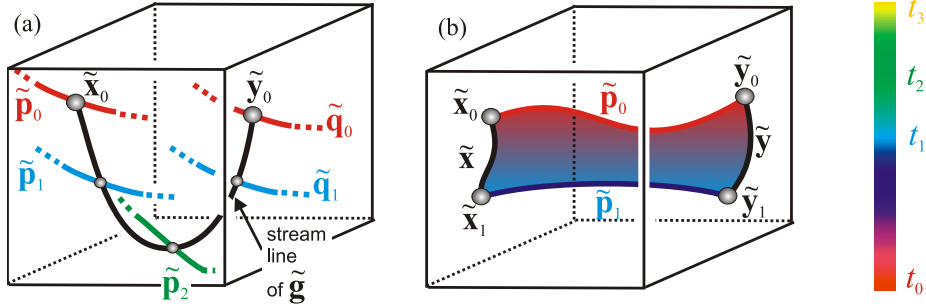


Figure 4.7: (a) PV line fold bifurcations do not exist! (b) tracking an open PV line.

such a bifurcation cannot exist for PV lines. To show this, imagine two PV lines $\tilde{\mathbf{p}}_0, \tilde{\mathbf{q}}_0$ at the time t_0 as illustrated in figure 4.7a. While moving forward in time ($t = t_1$), the current PV lines $\tilde{\mathbf{p}}_1, \tilde{\mathbf{q}}_1$ move towards each other until they merge in the line $\tilde{\mathbf{p}}_2$ and disappear. If we pick a point $\tilde{\mathbf{x}}_0$ on $\tilde{\mathbf{p}}_0$ and start a stream line integration of $\tilde{\mathbf{g}}$ from $\tilde{\mathbf{x}}_0$, we end in a point $\tilde{\mathbf{y}}_0$ on $\tilde{\mathbf{q}}_0$. Since $\tilde{\mathbf{x}}_0$ and $\tilde{\mathbf{y}}_0$ have the same t -value $t = t_0$, the integration of $\tilde{\mathbf{g}}$ must go both forward and backward in time. This is a contradiction to (4.11) which shows that the last component of $\tilde{\mathbf{g}}$ (specifying the evolution in time) is always non-negative. Therefore, PV fold bifurcations do not exist.

4.4 The Algorithms

Before we formulate the algorithms for PV surface extraction in \tilde{D} , we explain the main ideas on a number of simple examples.

Consider figure 4.7b: suppose there is a PV line $\tilde{\mathbf{p}}_0$ at the time $t = t_0$ which leaves \tilde{D} in the points $\tilde{\mathbf{x}}_0, \tilde{\mathbf{y}}_0$. While moving forward in time until $t = t_1$, $\tilde{\mathbf{p}}_0$ sweeps to the line $\tilde{\mathbf{p}}_1$ which leaves \tilde{D} in $\tilde{\mathbf{x}}_1, \tilde{\mathbf{y}}_1$. Doing this sweeping, the points where the PV lines leave \tilde{D} form two lines $\tilde{\mathbf{x}}, \tilde{\mathbf{y}}$ on the boundary of \tilde{D} : $\tilde{\mathbf{x}}$ connects $\tilde{\mathbf{x}}_0$ and $\tilde{\mathbf{x}}_1$, while $\tilde{\mathbf{y}}$ connects $\tilde{\mathbf{y}}_0$ and $\tilde{\mathbf{y}}_1$. In order to extract the PV surfaces (i.e. the surface bounded by the curves $\tilde{\mathbf{p}}_0, \tilde{\mathbf{y}}, \tilde{\mathbf{p}}_1, \tilde{\mathbf{x}}$), we have the choice between two approaches: One approach is to start with an extraction of $\tilde{\mathbf{p}}_0$ and using it as seeding curve for a stream surface integration of $\tilde{\mathbf{g}}$ until we reach $\tilde{\mathbf{p}}_1$ (or reversely, integrate $\tilde{\mathbf{g}}$ backward from $\tilde{\mathbf{p}}_1$ until we reach $\tilde{\mathbf{p}}_0$). The second approach is to extract $\tilde{\mathbf{x}}$ and use it as seeding curve of a stream surface integration of $\tilde{\mathbf{f}}$ until it reaches $\tilde{\mathbf{y}}$ (or reversely, integrating $\tilde{\mathbf{f}}$ from $\tilde{\mathbf{y}}$ to $\tilde{\mathbf{x}}$). The first approach has two disadvantages over the second one: first, a stream surface integration of $\tilde{\mathbf{f}}$ is cheaper than a stream surface integration of $\tilde{\mathbf{g}}$ because $\tilde{\mathbf{f}}$ has a simpler structure (see section 4.2). Second, a stream surface integration of $\tilde{\mathbf{g}}$ starting from $\tilde{\mathbf{p}}_0$ may partially leave \tilde{D} before reaching $\tilde{\mathbf{p}}_1$. Hence we prefer the second approach. The extraction of $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ turns out to be simple: in the example, $\tilde{\mathbf{x}}$ consists of all points (x_{min}, y, z, t) with $\mathbf{s}(x_{min}, y, z, t) = (0, 0, 0)^T$. Keeping x_{min} constant, this can be considered as finding PV lines in a steady 3D flow field in the (y, z, t) -domain. Thus, $\tilde{\mathbf{x}}$ can be found by applying the algorithm of section 4.1 for the (y, z, t) -domain. Similarly we find $\tilde{\mathbf{y}}$.

Another simple example is shown in figure 4.8a. Here we have a closed PV line $\tilde{\mathbf{p}}_0$ at the time t_0 which moves over time to the line $\tilde{\mathbf{p}}_1$ at $t = t_1$. To extract the PV surface, we might integrate $\tilde{\mathbf{g}}$ starting from $\tilde{\mathbf{p}}_0$. In order to be consistent with the example before,

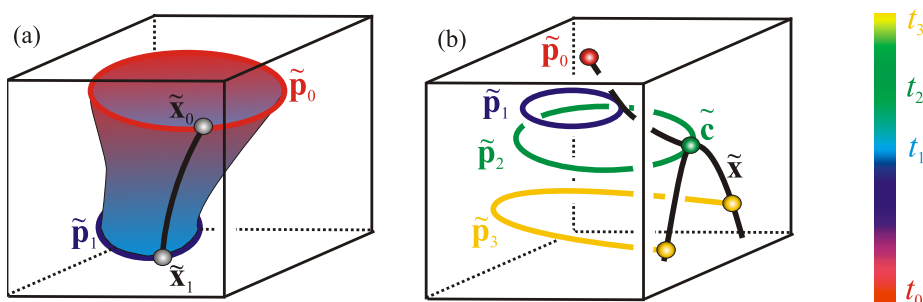


Figure 4.8: seeding lines (black) for simple examples.

we prefer to pick a point \tilde{x}_0 on \tilde{p}_0 ⁸ and apply a stream line integration of \tilde{g} starting from \tilde{x}_0 until it leaves \tilde{D} in a point \tilde{x}_1 . This stream line is used as seeding structure for a stream surface integration of \tilde{f} .

The last simple example is shown in figure 4.8b. Here a closed PV line appears at the time t_0 in the closed collapse bifurcation point \tilde{p}_0 , grows over time ($t = t_1$) to \tilde{p}_1 until at $t = t_2$ it touches the boundary of \tilde{D} in the inflow boundary bifurcation point \tilde{c} . From this moment on it is an open PV line \tilde{p}_3 which creates an intersection curve \tilde{x} with the boundary of \tilde{D} . In order to get a seeding structure for this example, we first extract \tilde{x} similar to the example in figure 4.7b. Then we apply a stream line integration of \tilde{g} starting from \tilde{c} until it ends in \tilde{p}_0 .

In order to extract PV surfaces in \tilde{D} , we provide two algorithms. Algorithm 1 describes how to get a seeding structure, i.e., a set of lines in \tilde{D} such that a stream surface integration of \tilde{f} starting from them gives the complete PV surface. Based on this, algorithm 2 describes how to extract and visualize the PV surfaces for a particular time interval.

Algorithm 1 (getting the seeding lines):

1. Compute the intersection curves of the PV surface with the spatial boundaries of \tilde{D} (see figure 4.7b for an example).
2. Extract all local bifurcations introduced in section 4.3.
3. Extract closed PV lines at the times $t = t_{min}$ and $t = t_{max}$ respectively, pick a point on each extracted closed line, and apply a stream line integration of \tilde{g} starting from them until they leave \tilde{D} or end in a closed collapse bifurcation. Figure 4.8a shows an example.
4. Start a stream line integration of \tilde{g} from all inflow boundary bifurcations until it ends in a closed collapse bifurcation or leaves \tilde{D} (see figure 4.8b for an example).

Then the set of all lines obtained in steps 1–4 is the searched seeding structure. This algorithm needs some comments:

To 1: We have to find the PV lines at the faces of \tilde{D} . This means to apply six times the algorithm for extracting PV lines of static vector fields as described in section 4.1.

To 2: All inner bifurcations can be found by applying a recursive subdivision approach as described earlier. Boundary bifurcations can be found by checking the lines from

⁸We use one of the points which were necessary to extract \tilde{p}_0 – see section 4.1

step 1 of algorithm 1 for local extremal values of the t -component.

To 3: We get the closed stream line by applying the static algorithm of section 4.1 to find closed PV lines in the first and last time step.

Algorithm 1 can be considered as a preprocess of the actual PV extraction algorithm described in the following

Algorithm 2 (extract and visualize the PV surface for a time interval $[t_0, t_1]$ with $t_{min} \leq t_0 \leq t_1 \leq t_{max}$):

1. Load the seeding lines obtained from algorithm 1.
2. Identify all parts of the seeding lines with t -values between t_0 and t_1 .
3. Starting from these seeding lines, apply a stream surface integration of $\tilde{\mathbf{f}}$ until it leaves \tilde{D} or returns to its starting point.
4. Visualize the stream surfaces obtained in 3.

Note that although algorithm 2 guarantees that all PV surfaces are found, it does not guarantee that each surface is found only once. In fact, an open PV surface is extracted twice, by integrating from both exit curves of \tilde{D} . In the current implementation we did not consider this problem and visualized parts of the PV surfaces twice.

Out-of-core considerations:

3D time-dependent fields tend to be larger than the main memory of high-end workstations. Thus, an out-of-core data handling is preferable. We show for our algorithm that only a certain part of the data has to be in memory at once, and that (in worst case) the whole data set has to be loaded only twice. We assume that certain time intervals of the data can be loaded into memory separately, e.g. the data may come as a sequence of static 3D vector fields, one for each time step: by loading the vector fields of two consecutive time steps t_i and t_{i+1} and applying a linear interpolation, we obtain the time-dependent vector field in that interval.

Algorithm 1 can be executed by treating time slices consecutively, but not in one sweep through the data since $\tilde{\mathbf{g}}$ needs to be integrated in both directions. Thus, we make one forward sweep through the data collecting the local bifurcations and integrating $\tilde{\mathbf{g}}$ in forward direction. While doing this, we build up 6 static 3D vector fields representing the spatial boundaries of the domain over time. They serve as input for step 1 of this algorithm. In a following backward sweep we integrate $\tilde{\mathbf{g}}$ backwards starting from the already collected seeding points.

Since a stream line of $\tilde{\mathbf{f}}$ always stays in the same time level, the stream surface integration of algorithm 2 can be applied to smaller subintervals independently if the data of the original time interval $[t_0, t_1]$ does not fit into main memory.

4.5 Applications

Figures 4.4–4.6 show examples of local bifurcations in constructed quadrilinear vector fields $\mathbf{w}_1, \mathbf{w}_2$. We show them both to illustrate the bifurcations again and to explain our visualization technique. As projecting the complete PV surface to space leads to selfintersections already in quite simple settings, we use the following approach to visualize the evolution of PV structures: at a given time we draw the PV lines as red tubes inside the PV surface that is displayed only for a certain time range for future

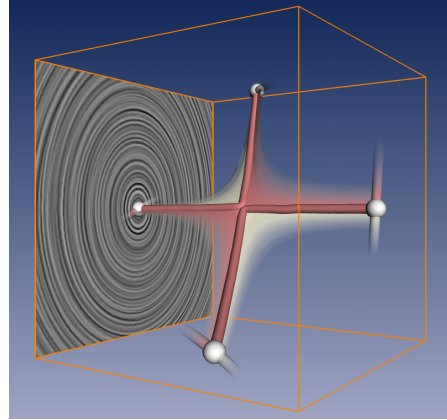
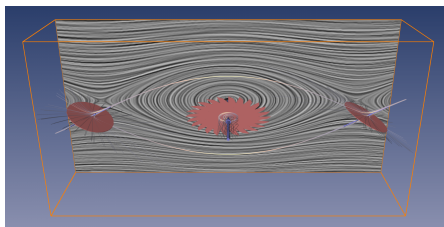


Figure 4.9: Saddle bifurcation of vortex core lines as defined by $\mathbf{v} \parallel \nabla \mathbf{v} \cdot \mathbf{v}$ in a quadrilinear vector field.

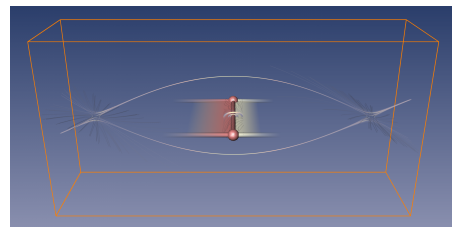
and past. At the boundary the corresponding seeding lines from algorithm 1 are given for a larger time interval. Both the surfaces and the seeding lines fade out away from the current time. We use color coding to indicate past (red) and future (gray). Figure 4.4 shows the evolution of a saddle bifurcation. Note that the width of the surface in figures 4.4a and 4.4c confirms the intuition that the most drastic movements of the PV line over time takes place near the bifurcation points.

Figure 4.6 shows a reversed outflow boundary bifurcation leading to the birth of a PV line. We omitted to display the PV surface for this and the following example. Figure 4.5 shows an inflow bifurcation and a subsequent closed collapse bifurcation in the green point. Note that in figure 4.5a, the location of the future inflow bifurcation is already shown by the grey semi-transparent point.

Now we proceed to applying our parallel vector based theory to vortex core line extraction. To do so, we consider the vortex core line concept defined by Sujudi and Haines [SH95] searching for all locations with $\mathbf{v} \parallel \nabla \mathbf{v} \cdot \mathbf{v}$ in regions where $\nabla \mathbf{v}$ has a pair of complex eigenvalues. As already mentioned in section 4.1.1, we have equivalently chosen $\mathbf{w}_1 = \mathbf{v}$ and \mathbf{w}_2 as the eigenvector corresponding to the only real eigenvalue of $\nabla \mathbf{v}$ in the regions of interest. Before we apply the technique to a real data set, we analyze whether the bifurcations introduced in section 4.3 may appear for vortex core lines defined by [SH95] for piecewise low-degree vector fields. It turns out that the inner bifurcations do not exist inside a cell for a piecewise linear vector field in



(a) Critical points and surrounding stream lines.



(b) Vortex core line together with its PV surface showing the evolution of the vortex over time.

Figure 4.10: Stuart Vortex moving over time from left to right.

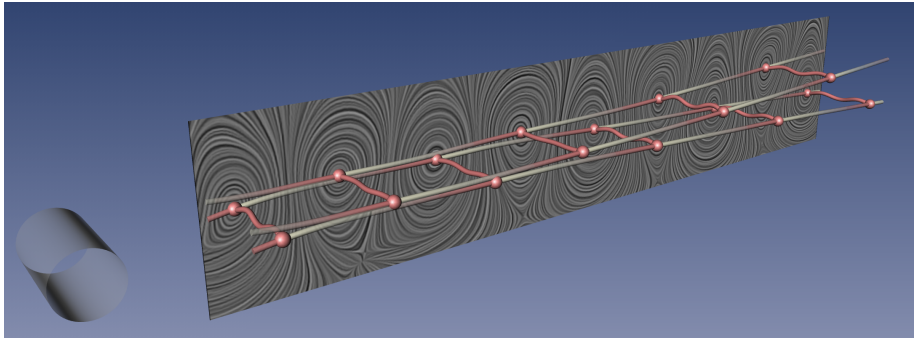


Figure 4.11: Flow behind a circular cylinder. The extracted seeding lines elucidate the alternating evolution of the vortical structures in transverse direction.

space-time⁹. For piecewise quadrilinear vector fields, all bifurcations can occur inside a cell. Figure 4.9 shows an example of a quadrilinear vector field containing a saddle bifurcation.

Consider Figure 4.10 that demonstrates our vortex core line tracking approach for visualizing a moving Stuart vortex. A Stuart vortex is a well-known vortical structure in fluid dynamics which can be described by a closed formula. Figure 4.10a shows that there is a critical point on the moving vortex core line. Figure 4.10b shows the vortex core line together with the PV surfaces indicating its past and future behavior. It shows that our FFF integration did not get stuck in the critical point.

Figures 4.1 and 4.11 demonstrate the results of our method applied to vortex core line tracking in a flow behind a circular cylinder. As above, we chose the vortex core line definition $\mathbf{v} \parallel \nabla \mathbf{v} \cdot \mathbf{v}$. The data set was derived by Bernd R. Noack (TU Berlin) from a direct numerical Navier Stokes simulation by Gerd Mutschke (FZ Rossendorf). It resolves the so called ‘mode B’ of the 3D cylinder wake at a Reynolds number of 300 and a spanwise wavelength of 1 diameter. The data is provided on a $265 \times 337 \times 65$ curvilinear grid as a low-dimensional Galerkin model [NE94, ZFN⁺95]. The examined time range is $[0, 2\pi]$. The flow exhibits periodic vortex shedding leading to the well known von Kármán vortex street. This phenomenon plays an important role in many industrial applications, like mixing in heat exchangers or mass flow measurements with vortex counters. However, this vortex shedding can lead to undesirable periodic forces on obstacles, like chimneys, buildings, bridges and submarine towers.

⁹This is similar to the fact that e.g. fold bifurcations do not exist inside a cell for piecewise linear time-dependent vector fields.

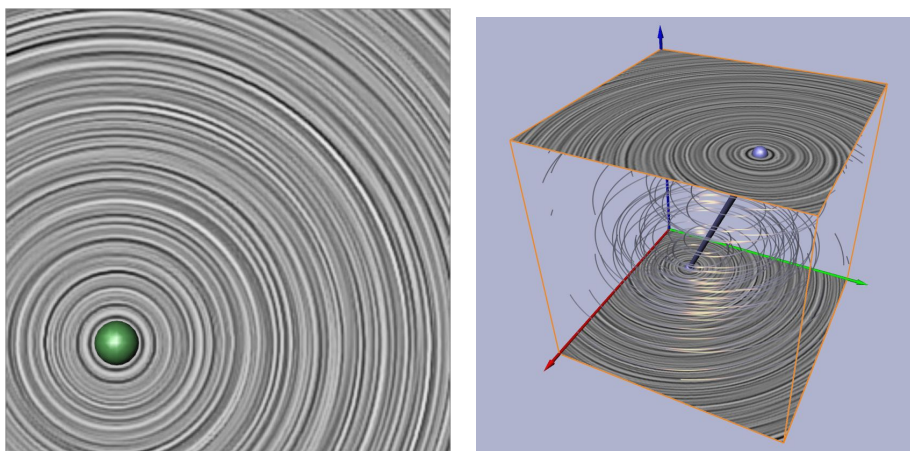
Chapter 5

Cores of Swirling Particle Motion

One way to assess vortices in experiments is to emit particles into the flow and to examine their behavior: patterns of swirling flow indicate vortices. This has been done in the experiment shown in Figure 1.1a. By injecting smoke, i.e., a huge amount of particles, swirling flow caused by the wake vortex becomes visible. For numerical and measured data sets, Sujudi and Haimes [SH95] proposed a scheme to extract centers of swirling flow. Peikert et al. formulated the idea of Sujudi/Haimes using the Parallel Vectors operator and presented a fast and robust extraction technique [PR99]. Bauer et al. [BP02] and Theisel et al. [TSW⁺05] (chapter 4) proposed different algorithms to track these centers over time in unsteady flows. All these approaches have in common that they assess the behavior of *stream lines* only.

However, most flow phenomena are unsteady in nature. In unsteady flows (as shown in Figure 1.1a), particle motion is described by *path lines* instead of stream lines (cf. section 2.3). This generally gives different swirling patterns. In this chapter we aim at extracting the cores of swirling particle motion in unsteady flows based on the behavior of path lines. To do so, we develop a novel mathematical characterization of such cores as a generalization of the original idea of Sujudi/Haimes. We do this for 2D and 3D flows. In the latter case, the resulting core structures are Galilean invariant lines sweeping over time, i.e., surfaces in the space-time domain. At a single time step, particles group around these core lines forming patterns of swirling motion similar to Figure 1.1a. That is why we refer to those features as *swirling particle cores*. Mathematically, they are characterized by the coplanarity of three 4D vectors. In order to extract them, we show how to re-formulate the problem using the Parallel Vectors operator [PR99] and apply it accordingly.

The chapter is organized as follows: In section 5.1 we summarize the basic facts concerning swirling motion of stream lines. In section 5.2 we develop our description of cores of swirling particle motion. Furthermore, we introduce a unified notation of swirling motion in 2D and 3D flows. In section 5.3 we re-formulate this description using the Parallel Vectors operator and show how to extract the cores. In section 5.5 we apply our technique to a number of 2D and 3D flows.



(a) Cores of swirling motion in 2D steady flow fields are critical points where the Jacobian has a pair of conjugate complex eigenvalues.

(b) Cores of spiraling stream lines in 2D unsteady flows are certain critical points (foci and center) tracked over time. Example vector field from Figure 2.1a.

Figure 5.1: Swirling motion of stream lines in 2D steady and unsteady flows.

5.1 Swirling Motion of Stream Lines

Patterns of spiraling stream lines in 2D and 3D flows have already been treated in the literature. These patterns are assessed by examining eigenvalues and eigenvectors of the first derivative \mathbf{J} (the Jacobian) of the respective flow field \mathbf{v} . A necessary condition for spiraling stream lines in \mathbf{v} is that \mathbf{J} has a pair of conjugate complex eigenvalues. In the following we give a short overview of the literature.

2D Flows

A steady 2D flow field is given as

$$\mathbf{v}(x,y) = \begin{pmatrix} u(x,y) \\ v(x,y) \end{pmatrix}. \quad (5.1)$$

The Jacobian of this field has either two real or one pair of conjugate complex eigenvalues. Swirling motion occurs in the latter case only – stream lines spiraling around a common point (see Figure 5.1a). The velocity at this point must be zero, i.e., $\mathbf{v}(x,y) = \mathbf{0}$. This means that cores of swirling motion in 2D steady flow fields are certain types of critical points, namely foci and centers. Thus, they can be treated using steady flow field topology as described in [HH89].

An unsteady 2D flow field is given as

$$\mathbf{v}(x,y,t) = \begin{pmatrix} u(x,y,t) \\ v(x,y,t) \end{pmatrix}. \quad (5.2)$$

Following (2.8), the stream lines of this field always stay in the same given time slice t_0 (Figure 2.1a). Thus, swirling motion in a single time slice can be captured by applying the scheme known from the steady case. By changing the given time slice, the critical points will move over time and form line-type structures in space-time. In order to

extract all locations of all foci and centers one has to apply algorithms for tracking critical points as known from [TWSH02, TS03, TWHS05]. Figure 5.1b shows this for the simple example vector field known from Figure 2.1a.

Throughout this chapter, swirling stream line cores will be colored blue, whereas swirling particle cores will be colored red.

3D Flows

In steady 3D flow fields

$$\mathbf{v}(x, y, z) = \begin{pmatrix} u(x, y, z) \\ v(x, y, z) \\ w(x, y, z) \end{pmatrix} \quad (5.3)$$

centers of swirling motion can be extracted by the method of Sujudi and Haimes [SH95], see Section 3.2.1 for an introduction.

Figure 5.2a shows the core line of swirling flow for the Stuart vortex, a simple vortex model given by

$$\mathbf{v}(x, y, z) = \begin{pmatrix} \sinh(y)/(\cosh(y) - 0.25 \cos(x)) \\ -0.25 \sin(x)/(\cosh(y) - 0.25 \cos(x)) \\ z \end{pmatrix}. \quad (5.4)$$

As expected, the swirling patterns in the LIC image correlate with the extracted center of swirling flow.

An unsteady 3D flow field is given as

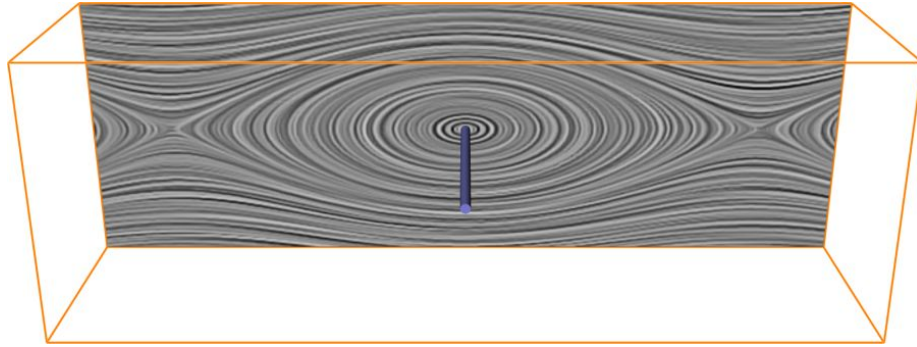
$$\mathbf{v}(x, y, z, t) = \begin{pmatrix} u(x, y, z, t) \\ v(x, y, z, t) \\ w(x, y, z, t) \end{pmatrix}. \quad (5.5)$$

At each time step t_0 , locations of swirling stream line behavior can be extracted using the method known from the steady case. This yields lines sweeping over time, resulting in 4D surfaces in the space-time domain. Two approaches for extracting those surfaces exist [BP02, TSW⁺05]. Both address swirling motion of stream lines, not particles.

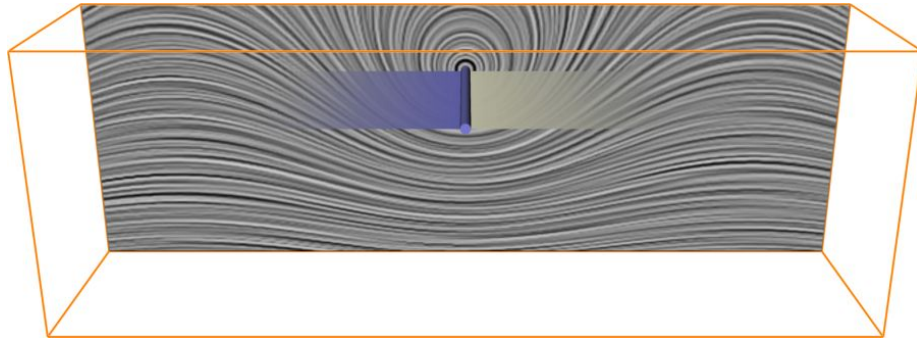
Figure 5.2b gives an example for sweeping stream line cores, extracted by the method in [TSW⁺05]. The flow field is derived from the steady model (5.4) by superimposing a constant flow of velocity $(1, 0, 0)^T$. This leads to structures moving constantly from left to right over time. The swirling stream line core at $t = 0$ is depicted as a blue line. Past and future is encoded in the surface, where blue means the past and grey encodes future. Note that by adding the convection velocity, the swirling stream line loci moved up relative to the steady case.

5.2 Cores of Swirling Particle Motion

In the last section we reviewed the approaches for extracting swirling motion of *stream lines* in steady and unsteady flows. To the best of our knowledge, there exists no approach for capturing swirling motion of *path lines*. We believe that studying the behavior of path lines is important since particle motion in unsteady flows is described by path lines instead of stream lines. Furthermore, many flow phenomena are unsteady



(a) Cores of swirling motion in 3D steady flows are lines where $\mathbf{v} \parallel \mathbf{e}$. Shown is a simple vortex model: the Stuart vortex (5.4).



(b) Cores of spiraling stream lines in 3D unsteady flows are lines sweeping over time. Blue denotes the past of the core, gray shows the future. Shown is a certain time step of the Stuart vortex moving over time from left to right with a constant velocity. This added convection velocity results in an upward shift of the stream line core compared to the steady case. The LIC plane shows the stream line pattern at the chosen time step.

Figure 5.2: Swirling motion of stream lines in 3D steady and unsteady flows.

and examining them solely based on the behavior of stream lines in certain time steps may not give the complete picture.

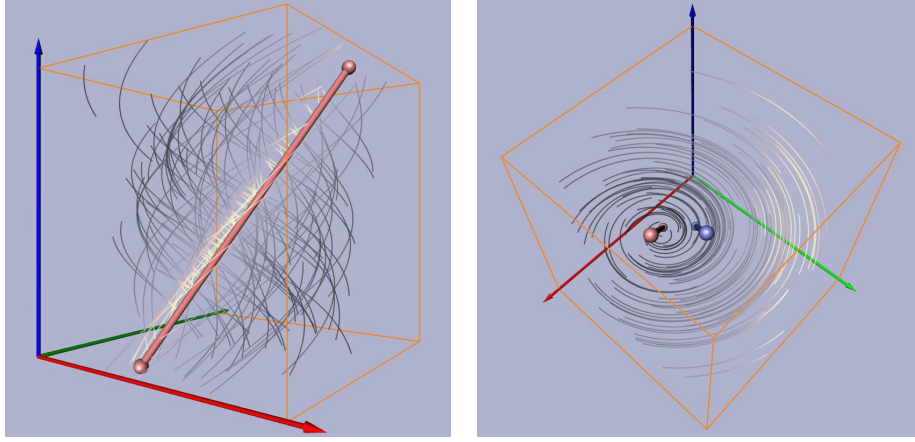
In the course of this section we develop a novel mathematical characterization of swirling particle motion in 2D and 3D unsteady flows – sections 5.2.1 and 5.2.2 respectively. Afterwards, we give a comprehensive summary of all discussed types of swirling flow leading to a more generalized notation (section 5.2.3).

5.2.1 Swirling Particle Motion in 2D

Following (2.7), the path lines of an unsteady 2D flow field $\mathbf{v}(x, y, t)$ are given as the stream lines of the steady 3D vector field

$$\mathbf{p}(x, y, t) = \begin{pmatrix} \mathbf{v}(x, y, t) \\ 1 \end{pmatrix} = \begin{pmatrix} u(x, y, t) \\ v(x, y, t) \\ 1 \end{pmatrix}. \quad (5.6)$$

This formulation of path lines as stream lines in a higher-dimensional vector field reduces the identification of swirling particle motion to a known case: it can be treated



(a) View similar to Figure 2.1b.

(b) View from top. Additionally, the core line of spiraling stream lines is shown in blue (cf. Figure 5.1b).

Figure 5.3: Core line (shown in red) of swirling particle motion of a simple 2D unsteady vector field \mathbf{v} (see Figure 2.1). The path lines of \mathbf{v} spiral around this core line.

similar to swirling stream line motion of a steady 3D vector field by applying the original approach of Sujudi/Haimes [SH95] to \mathbf{p} . This yields line structures where the Jacobian of \mathbf{p} has a pair of conjugate complex eigenvalues and the only real eigenvector is parallel to \mathbf{p} . The Jacobian of \mathbf{p} is

$$\mathbf{J}(\mathbf{p}) = \begin{bmatrix} u_x & u_y & u_t \\ v_x & v_y & v_t \\ 0 & 0 & 0 \end{bmatrix} \quad (5.7)$$

and has the eigenvalues $e_1, e_2, 0$ with the respective eigenvectors

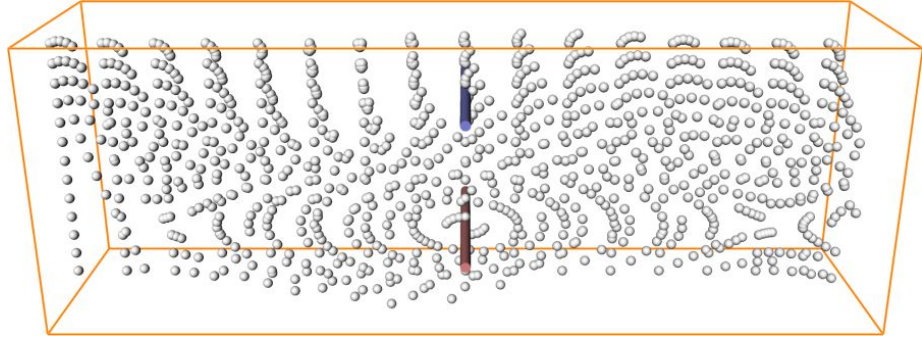
$$\begin{pmatrix} \mathbf{e}_1 \\ 0 \end{pmatrix}, \begin{pmatrix} \mathbf{e}_2 \\ 0 \end{pmatrix}, \mathbf{f}, \quad (5.8)$$

where $e_1, e_2, \mathbf{e}_1, \mathbf{e}_2$ constitute the eigensystem of the spatial Jacobian $\begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix}$ and

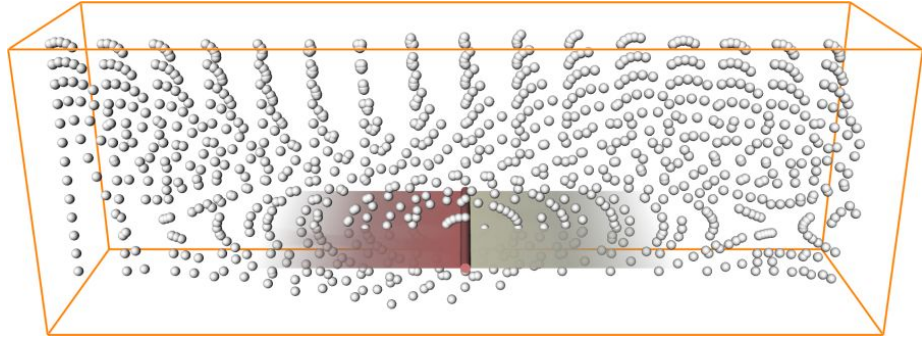
$$\mathbf{f} = \begin{pmatrix} \det(\mathbf{v}_y, \mathbf{v}_t) \\ \det(\mathbf{v}_t, \mathbf{v}_x) \\ \det(\mathbf{v}_x, \mathbf{v}_y) \end{pmatrix}. \quad (5.9)$$

Note, that \mathbf{f} is the Feature Flow Field for tracking critical points in time-dependent 2D vector fields known from [TS03]. In order to track critical points of \mathbf{v} , this field \mathbf{f} was designed such that the values of \mathbf{v} do not change along the stream lines of \mathbf{f} . In other words, the directional derivative of \mathbf{v} in direction of \mathbf{f} is zero. This means that $\mathbf{J}(\mathbf{v}) \cdot \mathbf{f} = \mathbf{0}$ and consequently $\mathbf{J}(\mathbf{p}) \cdot \mathbf{f} = \mathbf{0} \cdot \mathbf{f}$. Hence, \mathbf{f} necessarily is an eigenvector of $\mathbf{J}(\mathbf{p})$ corresponding to the eigenvalue 0.

Figure 5.3 shows the core of swirling particle motion for the simple 2D unsteady vector field introduced in Figure 2.1. The core line is depicted in red. Additionally, Figure 5.3b shows the core line of spiraling stream lines in blue. It can clearly be seen that both structures are different and that the particle core line lies in the center of the spiraling path lines.



(a) Swirling particle core as a line at $t = 0$ shown in red. Particles are grouping around that line in a spiraling pattern. Additionally, the core of swirling stream line motion is shown in blue. This clearly shows that stream lines and path lines spiral around different centers (compare with Figure 5.2b).



(b) Swirling particle cores in 3D unsteady flows are lines sweeping over time, i.e., surfaces in the 4D space-time domain. Red denotes the past of the core line, gray shows its future.

Figure 5.4: As an illustration of our new technique, swirling particle motion is shown in the example of the Stuart vortex moving over time from left to right with a constant velocity.

5.2.2 Swirling Particle Motion in 3D

We aim at extracting swirling particle cores of an unsteady 3D flow field $\mathbf{v}(x, y, z, t)$, i.e., locations around which spiraling patterns of path lines occur. Following (2.7), path lines of \mathbf{v} are stream lines of the steady 4D vector field

$$\mathbf{p}(x, y, z, t) = \begin{pmatrix} \mathbf{v}(x, y, z, t) \\ 1 \end{pmatrix} = \begin{pmatrix} u(x, y, z, t) \\ v(x, y, z, t) \\ w(x, y, z, t) \\ 1 \end{pmatrix}. \quad (5.10)$$

There is no existing tool to identify swirling motion in a steady 4D vector field. In the following we develop a new approach. The key is, again, the eigensystem of the Jacobian of \mathbf{p} . In the 3D unsteady setting, the Jacobian of \mathbf{p} is

$$\mathbf{J}(\mathbf{p}) = \begin{bmatrix} u_x & u_y & u_z & u_t \\ v_x & v_y & v_z & v_t \\ w_x & w_y & w_z & w_t \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (5.11)$$

and has the eigenvalues $e_1, e_2, e_3, 0$ with the respective four eigenvectors

$$\begin{pmatrix} \mathbf{e}_1 \\ 0 \end{pmatrix}, \begin{pmatrix} \mathbf{e}_2 \\ 0 \end{pmatrix}, \begin{pmatrix} \mathbf{e}_3 \\ 0 \end{pmatrix} =: \mathbf{e}, \mathbf{f}, \quad (5.12)$$

where $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ are the eigenvectors of the spatial Jacobian

$$\mathbf{J}_s(\mathbf{v}) = \begin{bmatrix} u_x & u_y & u_z \\ v_x & v_y & v_z \\ w_x & w_y & w_z \end{bmatrix} \quad (5.13)$$

and the fourth eigenvector \mathbf{f} can be written as

$$\mathbf{f}(x, y, z, t) = \begin{pmatrix} +\det(\mathbf{v}_y, \mathbf{v}_z, \mathbf{v}_t) \\ -\det(\mathbf{v}_z, \mathbf{v}_t, \mathbf{v}_x) \\ +\det(\mathbf{v}_t, \mathbf{v}_x, \mathbf{v}_y) \\ -\det(\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z) \end{pmatrix}. \quad (5.14)$$

Note, that \mathbf{f} is the Feature Flow Field for tracking critical points in 3D unsteady vector fields [WTHS07] – here, the same explanation as in the 2D case applies (section 5.2.1). The eigenvalue corresponding to \mathbf{f} is always zero. Therefore, $\mathbf{J}(\mathbf{p})$ has always one real eigenvalue and only the following cases can occur:

- All eigenvalues of $\mathbf{J}(\mathbf{p})$ are real.
- $\mathbf{J}(\mathbf{p})$ has a pair of conjugate complex eigenvalues and two real eigenvalues – let them be sorted such that e_1, e_2 are complex and e_3 is real.

Since complex eigenvalues are a necessary condition, swirling motion is only possible in the latter case. At any given point \mathbf{x} in the 4D domain, the eigenvectors corresponding to e_1, e_2 span a plane P_c in which locally the swirling motion occurs. The two real eigenvectors \mathbf{e} and \mathbf{f} denote the part of the flow which is independent of swirling – they span a plane P_r in which no swirling occurs at all. In order to see what the core of swirling motion in 4D is, consider the following rephrasing of the definitions of swirling motion cores in other dimensions:

Although a point \mathbf{x} on the core structure is surrounded by spiraling integral curves, the flow vector at \mathbf{x} itself is solely governed by the non-swirling part of the flow.

This is a direct generalization of the “reduced velocity”-idea of Sujudi/Haimes. For our case this means that \mathbf{x} is a point on the swirling particle core if the flow vector $\mathbf{p}(\mathbf{x})$ lies in the plane of non-swirling flow P_r , i.e., the plane spanned by \mathbf{e} and \mathbf{f} . In other words, the swirling particle cores are at locations where

$$\lambda_1 \mathbf{p} + \lambda_2 \mathbf{e} + \lambda_3 \mathbf{f} = \mathbf{0} \quad \text{with} \quad \lambda_1^2 + \lambda_2^2 + \lambda_3^2 > 0. \quad (5.15)$$

This is a coplanarity problem: swirling particle cores are at locations where the 4D vectors \mathbf{p} , \mathbf{e} and \mathbf{f} are coplanar. We call the operator solving this equation the *Coplanar Vectors* operator, which reads in the general setting

$$\lambda_1 \mathbf{a} + \lambda_2 \mathbf{b} + \lambda_3 \mathbf{c} = \mathbf{0} \quad \text{with} \quad \lambda_1^2 + \lambda_2^2 + \lambda_3^2 > 0. \quad (5.16)$$

In order to show that our approach is reasonable, we study what happens if we require the flow field \mathbf{v} to be steady, i.e., $\mathbf{v}(x, y, z, t) = \mathbf{v}(x, y, z, t_0)$. In this setting, path

| | steady stream lines | stream lines | unsteady path lines |
|----|---|---|---|
| 2D | CRITICAL POINTS $\lambda \mathbf{v} = \mathbf{0}$ <i>CP finder</i> can be treated using [HH89] | TRACKED CRITICAL POINTS $\lambda_1 \mathbf{s} + \lambda_2 \mathbf{e} = \mathbf{0}$ <i>PV operator</i> can be treated using [TS03] and, as proven, using [SH95, PR99] | SWIRLING PARTICLE CORES $\lambda_1 \mathbf{p} + \lambda_2 \mathbf{e} = \mathbf{0}$ <i>PV operator</i> treated in this thesis can be extracted using [SH95, PR99] |
| 3D | SWIRLING SL CORES $\lambda_1 \mathbf{v} + \lambda_2 \mathbf{e} = \mathbf{0}$ <i>PV operator</i> original idea of Sujudi/Haimes treated in [SH95, PR99] | TRACKED SL CORES $\lambda_1 \mathbf{s} + \lambda_2 \mathbf{e}^s + \lambda_3 \mathbf{f} = \mathbf{0}$ <i>CV operator</i> treated in [BP02, TSW ⁺ 05] | SWIRLING PARTICLE CORES $\lambda_1 \mathbf{p} + \lambda_2 \mathbf{e}^s + \lambda_3 \mathbf{f} = \mathbf{0}$ <i>CV operator</i> treated in this thesis |

Table 5.1: Summary of swirling motion in 2D and 3D flows. Depending on the dimension of the autonomous system the conditions can be written using the notations of Critical Points (CP), Parallel Vectors (PV), and Coplanar Vectors (CV).

lines coincide with stream lines and our approach needs to reduce to the steady case, i.e., the method of Sujudi/Haimes. As the temporal derivative $\mathbf{v}_t = 0$, the fourth eigenvector becomes $\mathbf{f} = (0, 0, 0, -\det(\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z))^T$ following (5.14), and the coplanarity condition (5.15) reads

$$\lambda_1 \begin{pmatrix} \mathbf{v} \\ 1 \end{pmatrix} + \lambda_2 \begin{pmatrix} \mathbf{e}_3 \\ 0 \end{pmatrix} + \lambda_3 \begin{pmatrix} 0 \\ 0 \\ 0 \\ -\det(\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z) \end{pmatrix} = \mathbf{0}. \quad (5.17)$$

The last component of this equation requires $\lambda_1 = \lambda_3 \det(\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z)$. Hence, in the steady setting our approach reduces to $\mathbf{v} \parallel \mathbf{e}_3$, i.e., the method of Sujudi/Haimes.

In Figure 5.4 the core of swirling particle motion has been extracted and visualized for the moving Stuart vortex. In Figure 5.4a the core is shown for $t = 0$ as a red line. A number of particles have been seeded uniformly and advected over time. They form a spiraling pattern around the core line. That is why we call the structures fulfilling (5.15) *swirling particle cores*. Additionally, Figure 5.4a shows the core of swirling stream line motion in blue. It can clearly be seen that the structures are different. Note, that swirling particle cores of 3D unsteady fields are surfaces, i.e., the particle core lines of a single time step sweep over time. This is shown in Figure 5.4b where the past of the core line is encoded in red and its future in gray.

In section 5.3 we will show how to extract swirling particle cores. In the next section we will give a summary on swirling motion in general.

5.2.3 A Unified Notation of Swirling Motion Cores

As we have seen in the previous sections, one may find swirling motion of

- stream lines in steady fields,
- stream lines in unsteady fields,
- path lines in unsteady fields.

All three cases can be found in 2D as well as 3D fields, summing up to a total of six cases. In the following we show how all those cases can be written using a unified notation.

Let \mathbf{V} be the autonomous system of the characteristic curves in question, i.e., \mathbf{v} for the steady case (2.9), \mathbf{s} for the stream lines of an unsteady flow (2.8), and \mathbf{p} for path lines (2.7). Let \mathbf{e}_i be the eigenvectors corresponding to the real eigenvalues of $\mathbf{J}(\mathbf{V})$. The point \mathbf{x} is part of the respective core of swirling motion, if $\mathbf{V}(\mathbf{x})$ lies in the span of $\mathbf{e}_i(\mathbf{x})$. In other words, this reads

$$\lambda_1 \mathbf{V}(\mathbf{x}) + \sum \lambda_i \mathbf{e}_i(\mathbf{x}) = \mathbf{0} \quad \text{with} \quad \sum \lambda_i^2 > 0. \quad (5.18)$$

We want to illustrate this: depending on the dimension of \mathbf{V} , this is equivalent to

- extraction of critical points in 2D,
- solving the Parallel Vectors operator in 3D,
- solving the Coplanar Vectors operator in 4D.

Table 5.1 illustrates this. The first column of this table covers the steady case, i.e., critical point extraction in 2D and the method of Sujudi/Haimes in 3D. They have been discussed in section 5.1. The third column refers to swirling particle cores as developed above. It remains to explain the second column where swirling motion of stream lines in unsteady flows is treated. In the 2D unsteady case, the Parallel Vectors operator is applied to \mathbf{s} and its only real eigenvector \mathbf{e} . We need to show that this describes critical points (foci and centers) tracked over time as discussed in section 5.1. Indeed,

$$\lambda_1 \mathbf{s} + \lambda_2 \mathbf{e} = \lambda_1 \begin{pmatrix} u \\ v \\ 0 \end{pmatrix} + \lambda_2 \begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix} = 0 \quad (5.19)$$

requires $\lambda_2 = 0$ and hence $\mathbf{v} = \begin{pmatrix} u \\ v \end{pmatrix} = 0$, as needed. This means that the Parallel Vectors operator can be used to track critical points in time-dependent 2D vector fields. In fact, the critical point in Figure 5.1b has been tracked this way. See also Figures 5.6a-b.

A similar statement holds for the 3D unsteady case, where the Coplanar Vectors operator is used to describe swirling stream line cores swept over time.

5.3 Extraction of Swirling Particle Cores

In the following we show how to extract swirling particle cores of unsteady 3D flows. In the next section we re-formulate the coplanarity problem (5.15) using the Parallel Vectors operator [PR99] – a common tool for feature extraction in the visualization community, which we will briefly explain in section 5.3.2.

5.3.1 Formulation using Parallel Vectors

We identified cores of swirling particle motion in unsteady 3D flows as locations where the three 4D vector fields $\mathbf{p}, \mathbf{e}^s, \mathbf{f}$ are coplanar. This is given by (5.15), which reads component-wise

$$\lambda_1 \begin{pmatrix} u \\ v \\ w \\ 1 \end{pmatrix} + \lambda_2 \begin{pmatrix} e_1^s \\ e_2^s \\ e_3^s \\ 0 \end{pmatrix} + \lambda_3 \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} = \mathbf{0}, \quad (5.20)$$

By setting $\lambda_1 = -\lambda_3 f_4$ we can eliminate the fourth component, and the reformulation reads

$$\lambda_2 \underbrace{\begin{pmatrix} e_1^s \\ e_2^s \\ e_3^s \end{pmatrix}}_{\mathbf{a}} + \lambda_3 \underbrace{\left(\begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} - f_4 \begin{pmatrix} u \\ v \\ w \end{pmatrix} \right)}_{\mathbf{b}} = \mathbf{0}. \quad (5.21)$$

This is a 3D Parallel Vectors problem. The reformulation $\mathbf{a} \parallel \mathbf{b}$ is equivalent to the coplanarity of the vector fields $\mathbf{p}, \mathbf{e}^s, \mathbf{f}$, and hence $\mathbf{a} \parallel \mathbf{b}$ is satisfied exactly at the cores of swirling particle motion in unsteady flow fields. With this reformulation at hand we can use the powerful extraction techniques available for the Parallel Vectors operator.

Note that although the eigenvectors corresponding to the eigenvalue zero can be calculated explicitly using formulae (5.9) and (5.14), it is more stable to calculate all involved eigenvectors using an eigenvector solver, especially in degenerate cases where $\det(\mathbf{v}_x, \mathbf{v}_y) = 0$ in 2D or $\det(\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z) = 0$ in 3D.

5.3.2 Extraction using Parallel Vectors

Applying the Parallel Vectors operator to our problem yields lines sweeping over time, i.e., surfaces in the 4D space-time domain. Different methods of extracting these surfaces exist [BP02, TSW⁺05] – both being extensions of the original approach presented in [PR99], which can be summarized as follows: to extract the solution lines for a specific time step, one has to iterate over all faces of the grid and search for points where $\mathbf{a} \parallel \mathbf{b}$. For example, in a triangle (being the face of a tetrahedral mesh) one can find such points by solving a certain three-by-three eigenvalue problem. The extracted points have to be connected to lines in a post-processing step by considering the solution points at the faces belonging to the same volume element (tetrahedron, voxel, etc.). On volume elements with more than two adjacent solutions, a decision is necessary which points should be connected. We use a simple angle criterion: the line segments with the smallest angle to each other are connected.

As Peikert et al. [PR99] already pointed out, one expects swirling stream line cores to point in direction of the flow field, but the parallelity condition $\mathbf{a} \parallel \mathbf{b}$ does not ensure this. Indeed, the solution lines can be orthogonal to the input vectors. Whenever this is not desired, one can filter the output such that only lines are displayed that do not exceed a defined threshold angle towards \mathbf{a} , or \mathbf{b} . For both the swirling stream line and particle cores we use the angle between the solution lines and \mathbf{e} as a criterion.

Note that the extraction of core lines of swirling motion is nonlinear in general, since eigenvectors do not depend linearly on the input fields. While this makes the extraction using linear techniques more difficult in general, we found that filtering the resulting lines by length was sufficient to rule out the nonlinearity.

5.4 Galilean Invariance of Swirling Particle Cores

Peikert has shown that swirling particle cores are Galilean invariant [Pei08]. We give his proof here in full detail. To this end we show that (5.21) is equivalent to $\mathbf{e} \parallel \mathbf{v}_t + \nabla \mathbf{v} \cdot \mathbf{v}$, the particle acceleration (2.14).

To show the equivalence, we show both implications. Let $\mathbf{e} \parallel (f_1, f_2, f_3)^T - f_4 \mathbf{v}$ hold. This means, $(f_1, f_2, f_3)^T - f_4 \mathbf{v}$ is the only eigenvector of $\nabla \mathbf{v}$ corresponding to a

real eigenvalue.

$$\lambda \left(\begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} - f_4 \mathbf{v} \right) = \nabla \mathbf{v} \left(\begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} - f_4 \mathbf{v} \right) \quad (5.22)$$

$$= -f_4 v_t - f_4 \nabla \mathbf{v} \cdot \mathbf{v} \quad (5.23)$$

$$= -f_4 (v_t + \nabla \mathbf{v} \cdot \mathbf{v}) \quad (5.24)$$

because $\nabla \mathbf{p} \mathbf{f} = 0$ and therefore $\nabla \mathbf{v} (f_1, f_2, f_3)^T - f_4 \mathbf{v} = 0$. Hence, $v_t \nabla \mathbf{v} \cdot \mathbf{v}$ and $(f_1, f_2, f_3)^T - f_4 \mathbf{v}$ point in the same direction.

To show that the converse is true, let $v_t + \nabla \mathbf{v} \cdot \mathbf{v}$ be the only eigenvector of $\nabla \mathbf{v}$ corresponding to a real eigenvalue. By multiplying with f_4 we obtain

$$-f_4 \nabla \mathbf{v} (v_t + \nabla \mathbf{v} \cdot \mathbf{v}) = -f_4 \lambda (v_t + \nabla \mathbf{v} \cdot \mathbf{v}) \quad (5.25)$$

$$= \lambda (-f_4 v_t - f_4 \nabla \mathbf{v} \cdot \mathbf{v}) \quad (5.26)$$

$$= \lambda \left(\nabla \mathbf{v} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} - f_4 \nabla \mathbf{v} \cdot \mathbf{v} \right) \quad (5.27)$$

$$= \lambda \nabla \mathbf{v} \left(\begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} - f_4 \mathbf{v} \right). \quad (5.28)$$

I.e., $(f_1, f_2, f_3)^T - f_4 \mathbf{v}$ and $v_t + \nabla \mathbf{v} \cdot \mathbf{v}$ is mapped to the same one-dimensional subspace. As the eigenspace of $\nabla \mathbf{v}$ corresponding to the only real eigenvalue λ is necessarily one-dimensional, $(f_1, f_2, f_3)^T - f_4 \mathbf{v}$ and $v_t + \nabla \mathbf{v} \cdot \mathbf{v}$ are parallel, as desired.

As the particle acceleration $v_t + \nabla \mathbf{v} \cdot \mathbf{v}$ is Galilean invariant (see Section 2.4.2), the formulation $\mathbf{e} \parallel v_t + \nabla \mathbf{v} \cdot \mathbf{v}$ is Galilean invariant. Hence, the notion of swirling particle cores is Galilean invariant.

5.5 Applications

As we use the Parallel Vectors operator for the extraction of swirling particle cores, we refer the reader to the literature [PR99] for the discussion of timings and memory consumption. In our implementation, a single time step of dimension 128^3 is processed in a single thread in about 15 seconds on an AMD64 X2 4400+.

In 3D unsteady flows, swirling particle cores are lines sweeping over time, i.e., 4D surfaces. In the previous sections we used semi-transparent surfaces to encode past and future (e.g. Figure 5.4b). However, for more complex data sets these surfaces might contain self intersections. We found that displaying the core lines only – at a certain time step or in an animation – results in clearer visualizations in most cases.

In Figures 1.1b and 5.5 we applied our method to the Hurricane Isabel data set from the IEEE Visualization 2004 contest. This is a complex 3D time-dependent data set produced by the Weather Research and Forecast (WRF) model. Figure 1.1b shows the unfiltered extraction result at $t = 33.5$ consisting of 1533 core lines. We have chosen to filter all lines shorter than 10% of the diagonal of the bounding box. The result is the single swirling particle core line in the eye of the hurricane – verified by the volume rendering of the cloud moisture (Figure 5.5).

Figure 5.6 shows an unsteady flow over a 2D cavity. This data set was kindly provided by Mo Samimy and Edgar Caraballo (both Ohio State University) [CSJ03] as

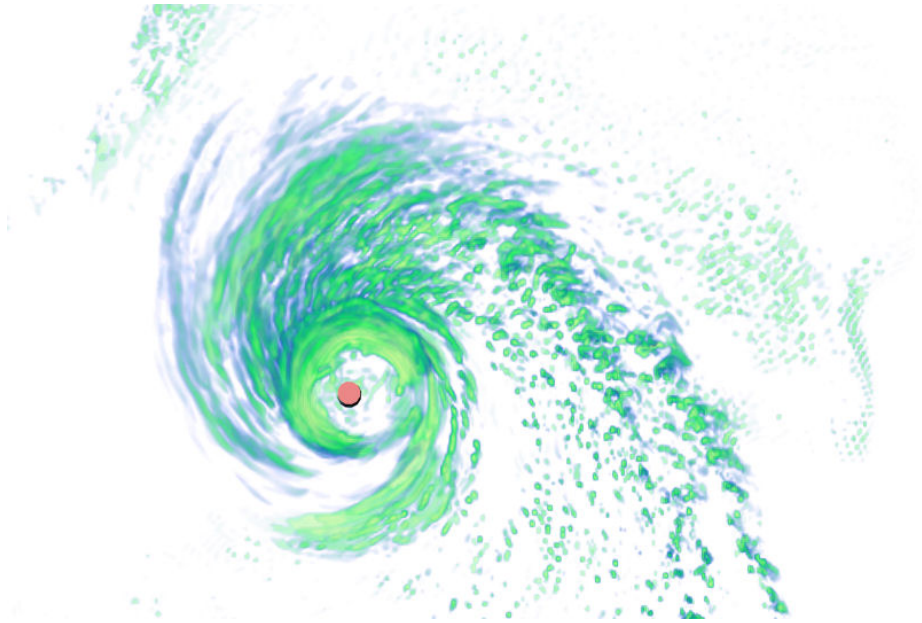


Figure 5.5: Hurricane Isabel data set at $t = 33.5$. Shown are the dominating swirling particle core line (red) and a volume rendering of the cloud moisture mixing ratio. Orthographic view from top showing the dominant particle core (red) in the center of the clouds.

well as Bernd R. Noack and Ivanka Pelivan (both TU Berlin). 1000 time steps have been simulated using the *compressible* Navier-Stokes equations. The data is almost periodic, with a period of about 100 time steps in length, and only the first 100 time steps are shown.

As shown in section 5.2.3, the Parallel Vectors operator can be used to track certain critical points (foci and centers) over time. Figures 5.6a-b exemplify this: the blue lines denote swirling motion of stream lines – once extracted by tracking critical points using Feature Flow Fields and once by applying the Parallel Vectors operator. Both results coincide very well. Note that additionally Figure 5.6a shows tracked saddle points as yellow curves. Figure 5.6c stresses again the difference between swirling particle and stream line cores: the blue swirling stream line core goes through the center of spiraling stream lines at a specific time step (shown as LIC plane), but it does not lie in the center of spiraling path lines (shown as illuminated lines). Since unsteady motion is described by path lines, existing approaches fail to capture swirling motion in unsteady flows correctly: they are based on stream lines. Our approach captures this behavior correctly as shown by the red swirling particle core. Figure 5.6d shows the 181 extracted particle core lines, where the majority (154) is shorter than 3.5% of the diagonal of the bounding box and has been filtered accordingly. Figure 5.6e shows the filtered result.

Figure 5.7 demonstrates the results of our method applied to a flow behind a circular cylinder. The data set was derived by Bernd R. Noack (TU Berlin) from a direct numerical Navier Stokes simulation by Gerd Mutschke (FZ Rossendorf). It resolves the so called ‘mode B’ of the 3D cylinder wake at a Reynolds number of 300 and a span-wise wavelength of 1 diameter. The data is provided on a $265 \times 337 \times 65$ curvilinear

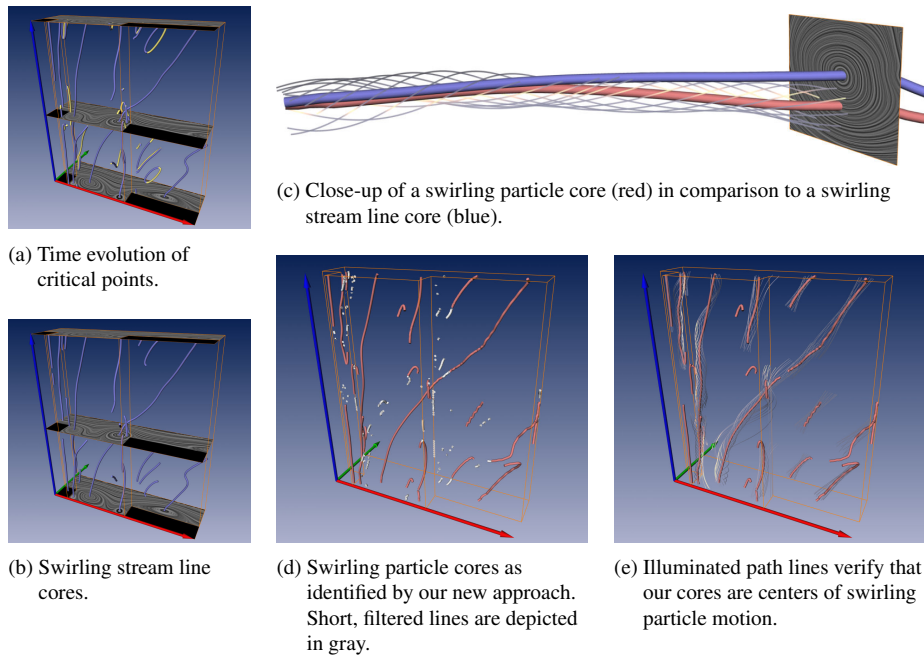
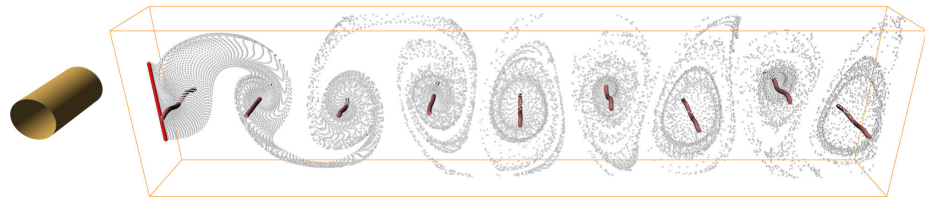


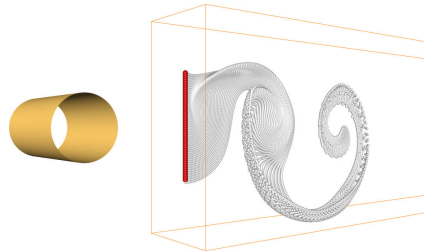
Figure 5.6: Unsteady flow over a 2D cavity. Red and green axes span the spatial domain, the blue axis denotes time.

grid as a low-dimensional Galerkin model. The flow exhibits periodic vortex shedding leading to the well known von Kármán vortex street [ZFN⁺95]. This phenomenon plays an important role in many industrial applications, like mixing in heat exchangers or mass flow measurements with vortex counters. However, this vortex shedding can lead to undesirable periodic forces on obstacles, like chimneys, buildings, bridges and submarine towers.

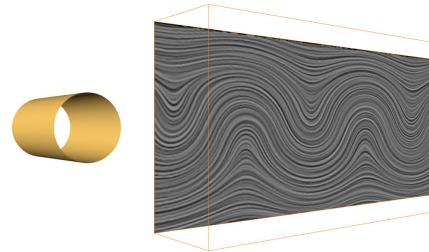
Figures 5.7a-b show particles seeded at a vertical line on the left-hand side of the bounding box. Due to the periodic vortex shedding these particles form patterns of swirling motion after some integration steps – a clear indication of the von Kármán vortex street. These patterns perfectly match up with the cores of swirling particle motion (red) extracted using our method (filtered by angle criterion with $|\cos(\ell, \mathbf{e})| < 0.3$ and by length with 0.1%). Figure 5.7c shows stream lines at a certain time step as depicted by the LIC plane: existing approaches based on stream lines have to fail to detect the von Kármán vortex street here, since the original frame of reference does not exhibit any spiraling stream lines. However, our method is based on the behavior of path lines and captures the vortex street correctly. Stream line based approaches are able to capture the features only if one chooses a reference frame matching their convection velocity. In this frame of reference, swirling motion of stream lines is present and its cores can be extracted by the method of Sujudi/Haimes. This has been done in Figure 5.7d by applying a priori knowledge [ZFN⁺95]. As a reference, the swirling particle cores (red) are also displayed. The extracted structures are very close to each other (Figure 5.7e). However, our method is able to extract these features in the original frame of reference without a priori knowledge.



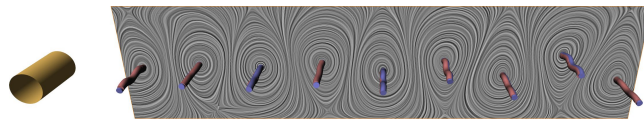
(a) Swirling particle cores (red lines) at a certain time step. The additionally shown particles verify that our core lines are at the centers of swirling particle motion.



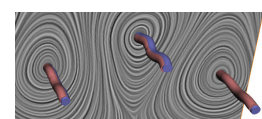
(b) Particles are injected constantly at the vertical line on the left-hand side of the bounding box and advected over time. They form patterns of swirling motion indicating the von Kármán vortex street.



(c) Stream lines at a certain time step visualized using a LIC plane. Swirling motion of stream lines can not be observed in the original frame of reference.



(d) In an *appropriately chosen* frame of reference, swirling motion of stream lines is present (indicated by the LIC plane). The extracted swirling stream line cores (blue) are displayed together with the swirling particle cores (red) of the same time step.



(e) Close-up of (d). Although extracted with different methods in different reference frames, the extracted lines are very similar.

Figure 5.7: 3D unsteady flow behind a cylinder. Existing stream line based approaches fail to capture swirling motion cores in the original frame of reference. Our new path line based method is able to extract such features without a priori knowledge.

Part III

Detection Methods based on Extremal Structures

In Chapter 3 we gave an overview of several vortex region and vortex core line techniques. While the vortex core line techniques are very intuitive, as vortices can be easily imagined as lines around which particles rotate, the vortex region techniques are physically more rigorous, both as they are derived from Navier-Stokes (e.g. λ_2), as well as they obey several invariances (Galilean invariance for λ_2 and Q , objectivity for \mathbf{M}_Z).

This part of the thesis is devoted to marrying the two approaches by extracting extremal lines of scalar vortex region quantities serving as vortex core lines. The motivation of this technique is the usual approach of fluid mechanicians to examine unknown data. Most often, isosurfaces of λ_2 are used at various thresholds ranging from low to high magnitude. In this way, the extent and location of isosurfaces is most easily grasped by direct visualization means.

Following this motivation, we extract extremal lines of the scalar quantities automatically, as extremal lines are lines around which isosurfaces collapse. They are centers of isosurfaces. Several mathematical definitions for extremal lines exist. The most common is the height ridge definition, a definition from differential geometry. The extraction method described in chapter 6 uses third derivatives of the scalar function and is therefore rather susceptible to noise requiring very small stepsizes even in moderately noisy datasets. This leads us to the necessity of reducing the involved degree of derivatives. In chapter 7 we define extrema given by the watershed definition by using just first derivatives. This boosts both robustness, speed and accuracy of the extraction process considerably, allowing us to process large datasets as the two parameter study of a high lift configuration with active flow control in chapter 8. In chapter 9 we extract extremal structures without any use of derivatives. For this challenging task we step back from 3D to 2.5D and extract extremal lines of scalar fields defined on surfaces. Here, a special focus is put on salient surface features related to surface crests. It is left for future research to generalize this one-parameter-dependent, derivative-free extremum line extraction scheme to 3D.

Chapter 6

Galilean Invariant Extraction of Vortex Core Lines

In this chapter we present an approach to extracting vortex core lines that is invariant under Galilean changes of the reference frame (cf. section 2.4.1). I.e., the extracted features remain unchanged when a constant vector is added to the flow field. Instead of using swirling stream line behavior as indication of a vortex core line, we consider ridge or valley lines of Galilean invariant vortex region quantities (Figure 6.1c). Previous prominent approaches, as those of Sujudi/Haimes [SH95], see Section 3.2.1, and Roth/Peikert [RP98], see Section 3.2.3 are dependent on the reference frame: Some specially chosen reference frame may result in a vortex core line, but when the spectator changes the reference frame significantly (for instance by moving faster than the mean velocity of the field), the feature vanishes. Figure 6.1 shows this dependence on the reference frame for the method of Sujudi and Haimes.

Furthermore, we show that those line type features have a higher dimensional generalization, e.g., surfaces.

The chapter is organized as follows: While section 6.1 treats the theory of ridge and valley lines, section 6.2 deals with implementation issue for their extraction. In Section 6.3 we present an iconic representation for vortex core lines that encodes the most relevant information like strength of the coherent structure as well as rotation direction. We apply our technique to several data sets in section 6.4.

6.1 Ridges and Valleys for Vortex Core Lines

We suggest a combination of the vortex region approaches (cf. section 3.1) and vortex core line approaches (cf. section 3.2) by extracting vortex core lines of vortex region quantities like Q and λ_2 and identify those lines by certain maximal lines of Q called ridge lines where $Q > 0$ and certain minimal lines of λ_2 where $\lambda_2 < 0$ called valley lines. In [PR99] it is pointed out how to extract such extremum lines using the parallel vectors operator. We use the Feature Flow Field approach due to [TS03] detailed in section 6.2.

By extracting vortex core lines in this way, we combine the Galilean invariance of the vortex region detection with the parameter independence of the vortex core line extraction.

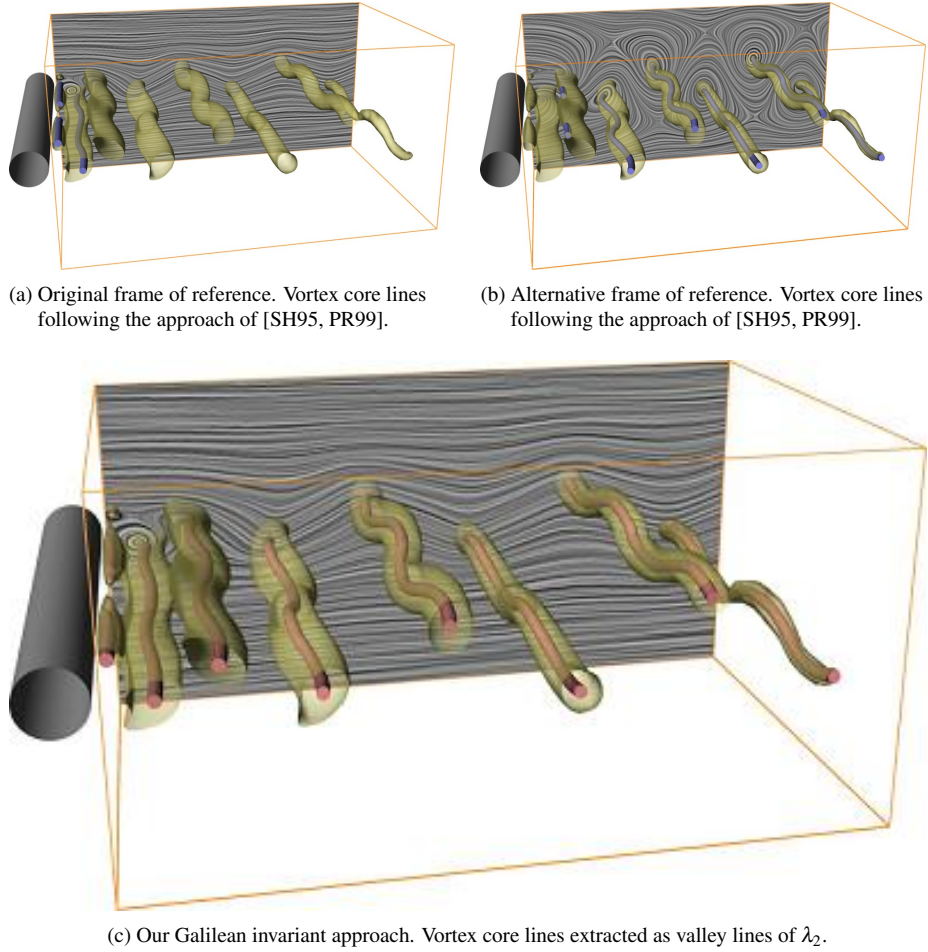


Figure 6.1: Flow behind a circular cylinder. Vortex regions visualized as transparent isosurfaces of λ_2 . Vortex core lines displayed as cylindrical lines.

With the Definition 1 (see page 29) of d -dimensional ridges at hand we can define d -dimensional **Galilean invariant vortex cores**.

Definition 3 Let s be a Galilean invariant vortex region quantity. In regions where s identifies a vortex, a d -dimensional **Galilean invariant vortex core** with respect to s is defined as d -dimensional

$$\left. \begin{array}{l} \text{ridge} \\ \text{valley} \end{array} \right\} \text{ of } s \text{ if } \left\{ \begin{array}{l} \text{large} \\ \text{small} \end{array} \right.$$

values of s indicate a vortex.

We concentrate on extracting 1-dimensional vortex cores that we suggest as an alternative definition of vortex core lines. Nevertheless, 2-dimensional vortex cores are interesting features for future research.

| vortex region quantity | vortex range | vortex core type |
|------------------------|----------------|------------------|
| p | $[0, \infty)$ | valley |
| $\ \omega\ $ | $(0, \infty)$ | ridge |
| Δ | $(0, \infty)$ | ridge |
| Q | $(0, \infty)$ | ridge |
| λ_2 | $(-\infty, 0)$ | valley |

Table 6.1: Vortex region quantities pressure p (might also be negative, as often relative pressure is considered), vorticity ω , rotation strength Δ from [SP03], Q -criterion and λ_2 criterion with the value range in which they indicate vortices. Vortex cores according to definition 3 are either ridges or valleys as shown in column 3.

Several vortex region quantities s and their vortex indicating value ranges are displayed in Table 6.1. This table also shows, whether vortex cores with respect to s are ridges or valleys of s .

6.2 Extraction of Vortex Core Lines

Let s be a vortex region quantity as used in Definition 3. We use the Feature Flow Field extraction scheme from [TS03] to extract the vortex core lines with respect to s as defined in the previous subsection.

The Feature Flow Field scheme involves two steps: In the first step certain points are extracted that lie on the extremum lines of interest. Those points are used as seed points in the second step by extracting the extremum lines as field lines of a derived flow field, the so called feature flow field. The following subsection is devoted to these two issues.

Afterwards we address interpolation issues in Subsection 6.2.2. The applications we show in Section 6.4 are based on flow fields that are interpolated from uniform grid data. We found that the widely used trilinear interpolation is not well suited for ridge extraction.

6.2.1 Feature Flow Field Setup

Again, we concentrate on the extraction of ridge lines. From Definition 1 in the previous subsection, we know that we have to extract regions where $A = B = 0, \gamma_2 < 0$.

Assuming a point $x \in \mathbf{R}^3$ fulfills this requirement, the tangent direction of the ridge x lies on can be computed as follows. As ∇A is orthogonal to the isolines of A and ∇B is orthogonal to the isolines of B , the ridge tangent is $T := \nabla A \times \nabla B$. The ridge line passing through x is then exactly the field line of T passing through x . So T is the feature flow field we are looking for. We extracted the ridge lines by Runge-Kutta-integration of T . Although T involves derivatives of high degree, we still found that integrating the features was stable.

Now we are left with computing T and finding seed points x .

Finding Seed Points

We are searching for zeros of the mapping $x \mapsto (A, B)$ from $\mathbf{R}^3 \rightarrow \mathbf{R}^2$. As we expect the zeros to be one dimensional, we can restrict the search to two dimensional subsets of the domain, e.g., the faces of some underlying structured or unstructured grid. This reduces the problem to finding roots of a function $\mathbf{R}^2 \rightarrow \mathbf{R}^2$. For this setting, several

Newton solvers can be applied, involving further differentiation. As $A = (\nabla f)c_1$ and $B = (\nabla f)c_2$ already involve second derivatives of f , we favoured a gradient free minimization of the positive function $x \mapsto A^2 + B^2$ which turned out to be more stable. We used the method described in [Ebe96] based on Powell's search [Pow64] and inverse parabolic interpolation [PFTV91].

Computing the feature flow field

The computation of the feature flow field $T := (\nabla A) \times (\nabla B)$ is quite involved. First of all, it is not trivial to state ∇A and ∇B explicitly in terms of the derivatives of f . Furthermore, A and B might be discontinuous at places where $\gamma_1 = \gamma_2$, so called partial umbilics. At such places the eigensystem γ_i is not unique, because the 2-dimensional eigenspace corresponding to $\gamma_1 = \gamma_2$ allows a range of orthonormal bases. [Ebe96] provides a remedy for this issue. As those findings are central to our algorithm, we state the ridge direction computed therein. The ridge tangent T is given by

$$T = \tilde{A} \times \tilde{B}. \quad (6.1)$$

Here, $\tilde{A}, \tilde{B} \in \mathbf{R}^3$ are given by

$$\left. \begin{array}{l} \tilde{A}_i \\ \tilde{B}_i \end{array} \right\} = \left\{ \begin{array}{l} \gamma_1 c_{1i} + \frac{(\nabla f)c_3}{\gamma_1 - \gamma_3} \sum_{j,k} c_{1j} \gamma_{3k} \partial_{x_i} \partial_{x_j} \partial_{x_k} f \\ \gamma_2 c_{2i} + \frac{(\nabla f)c_3}{\gamma_2 - \gamma_3} \sum_{j,k} c_{2j} c_{3k} \partial_{x_i} \partial_{x_j} \partial_{x_k} f \end{array} \right. \quad (6.2)$$

In [Ebe96] it is shown that T is only defined up to sign, whenever the ridge passes a partial umbilic $\gamma_1 = \gamma_2$. So in practice, when following a ridge by integrating T , the current ridge direction t_1 is compared to the previous ridge direction t_0 and replaced by $-t_1$ if the Euclidean scalar product $t_1 \cdot t_0 < 0$, i.e., if two subsequent ridge directions differ by an angle greater than $\frac{\pi}{2}$.

6.2.2 Interpolation issues

For ridge and valley line extraction, gradient and Hessian of vortex region quantities s have to be computed at arbitrary locations. As s usually involve derivatives of the flow to be Galilean invariant, the Hessian Hs involves third derivatives of the flow field. [PR99] states that extracting extremum lines requires careful filtering of the input field. We suggest here to use an appropriate interpolation scheme to remedy this problem.

Interpolating s trilinearly appeared to be both unstable and ineffective. Although some features were roughly recognized, most of them were missed completely. This is not surprising due to the high degree of smoothness required by the setting, and the fact that extremum lines are typically quadratic features that can not be resolved well by trilinear interpolation. Due to this, quadratic schemes seem a natural choice. Among those, approximation by quadratic super splines (see [RZNS04]) provides a good trade-off between smoothness and speed. As the polynomials involved are of total degree 2, (6.2) simplifies significantly, as here $\partial_{x_i} \partial_{x_j} \partial_{x_k} f = 0$ for all i, j, k and hence, (6.2) can be restated as follows:

$$\tilde{A}_{QSS} = \gamma_1 c_1, \quad \tilde{B}_{QSS} = \gamma_2 c_2, \quad (6.3)$$

and, if $\nabla f \neq 0$, the ridge tangent T from (6.1) evaluates to

$$T_{QSS} = \gamma_1 \gamma_2 c_1 \times c_2 = \mathbf{const} \cdot \nabla f, \quad (6.4)$$

conforming to the intuitive understanding of a ridge direction as stated in subsection 6.1. This makes quadratic super splines a somewhat natural choice.

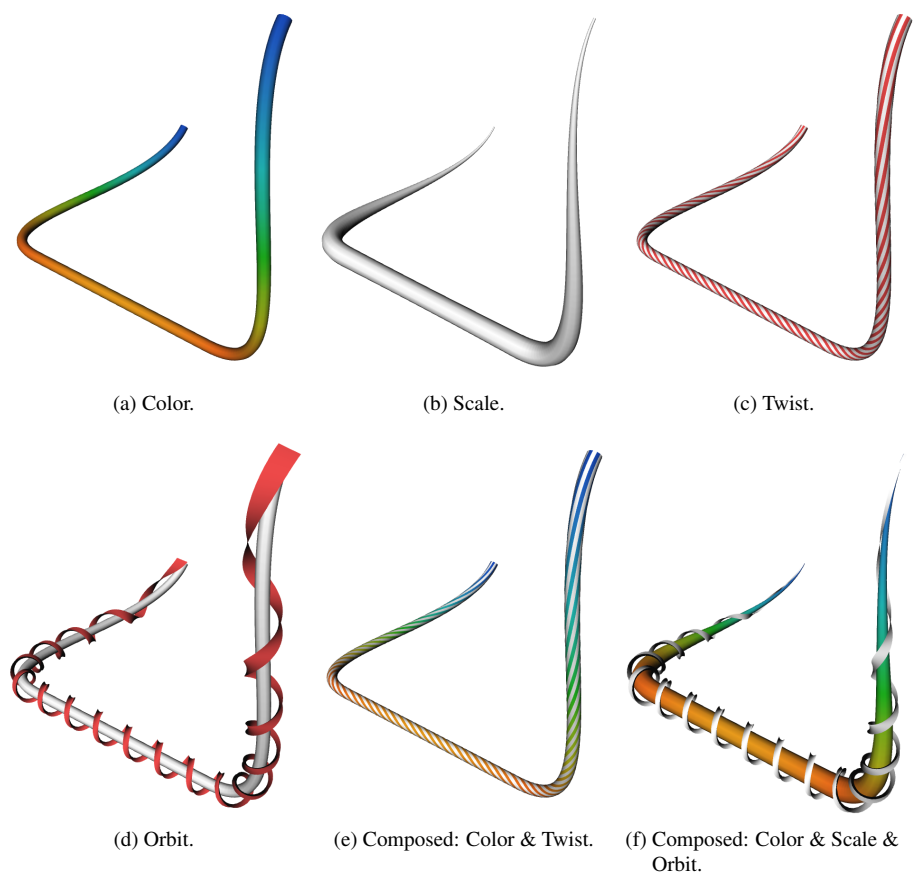


Figure 6.2: Different approaches to encoding a scalar value into the representation of a line.

6.3 Iconic Representation

To visualize vortex core lines, we use cylindrical meshes and encode different scalar values into their representation. Figure 6.2 illustrates this. In figures 6.2a-b we color or scale the cylinder according to the mapped values. Figures 6.2c-d encode sign and strength of a rotational behavior, either by using colored stripes on the cylinder itself or by placing a spiral shape around it. Our implementation allows us to combine these four variations as shown in figures 6.2e-f and 2.6b. Note, that not all possible combinations produce expressive results. Especially the usage of an orbit (figure 6.2d) tends to yield cluttered visualizations in more involved settings.

While those kinds of representing a line are quite common, we are still left with finding appropriate measures to be mapped onto our Galilean invariant vortex core lines. [JMT02] depicts spiraling stream lines around a Galilean variant vortex core line. As we aim at Galilean invariant vortex core lines, this approach is not directly applicable. [SP03] extracts and displays vortex hulls similar to isosurfaces of Δ (see section 3.1) around a vortex core line.

We propose the following measures to be used for an iconic representation of Galilean invariant vortex core lines:

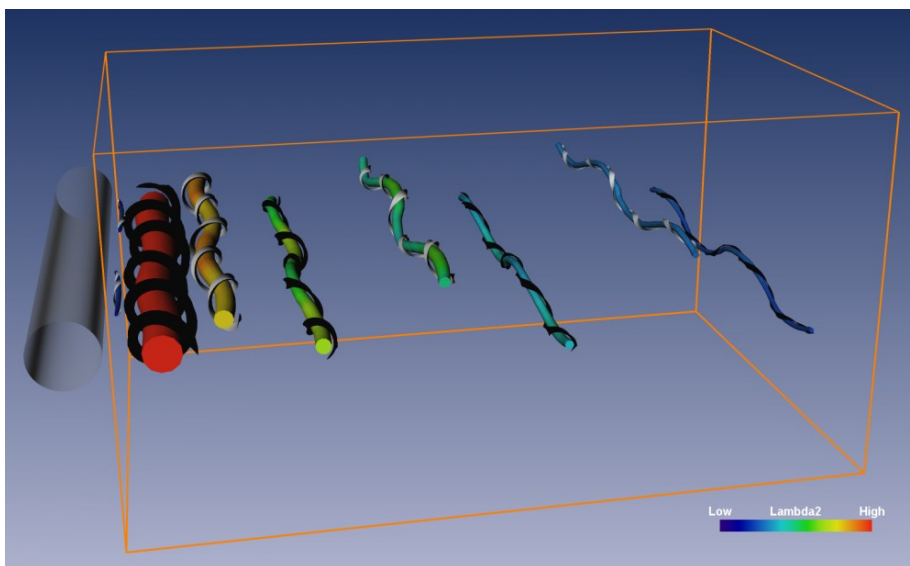


Figure 6.3: Flow behind a circular cylinder. Iconic representation of Galilean invariant vortex core lines. λ_2 was used for extraction and is encoded into color and scale of the cylindrical meshes. Red / blue color is used to indicate strong / weak vortex activity. φ is encoded into color and spiral direction of the orbits.

- *Strength/Value of vortex region quantity s* : Our vortex core lines are linked directly to a vortex region quantity s and their extremum property with respect to s ensures that no regions indicating stronger vortex activity exist away from the extracted features. Furthermore, the value of s varies along a line. To distinguish between (parts of) core lines with different vortical activity, the value of s should be encoded in the line representation. We found coloring and scaling most suitable for this.
- *Sign of rotation angle φ* : As shown in section 3.1, the rotation angle φ is derived from the Jacobian of the vector field. Its sign gives the direction of rotation of a vortex. As a visual encoding for this, the usage of color, twist or an orbit seems to be most appropriate.
- *Strength of rotation Δ* : This measure indicates the strength of spiraling patterns in the right reference frame. We found the usage of color, twist or an orbit most suitable for this.

We apply these visualization strategies in different combinations in the next section.

6.4 Applications

Figures 6.1 and 6.3 visualize a snapshot of a transitional wake behind a circular cylinder [ZFN⁺95]. This data set was derived from a direct numerical simulation of the Navier-Stokes equation by Bernd R. Noack (TU Berlin). It is given on a $88 \times 106 \times 20$ uniform grid. The data resolves the so-called ‘mode A’ of the 3D transition at a Reynolds number of 200 and at a spanwise wavelength of 4 diameters. This flow exhibits periodic

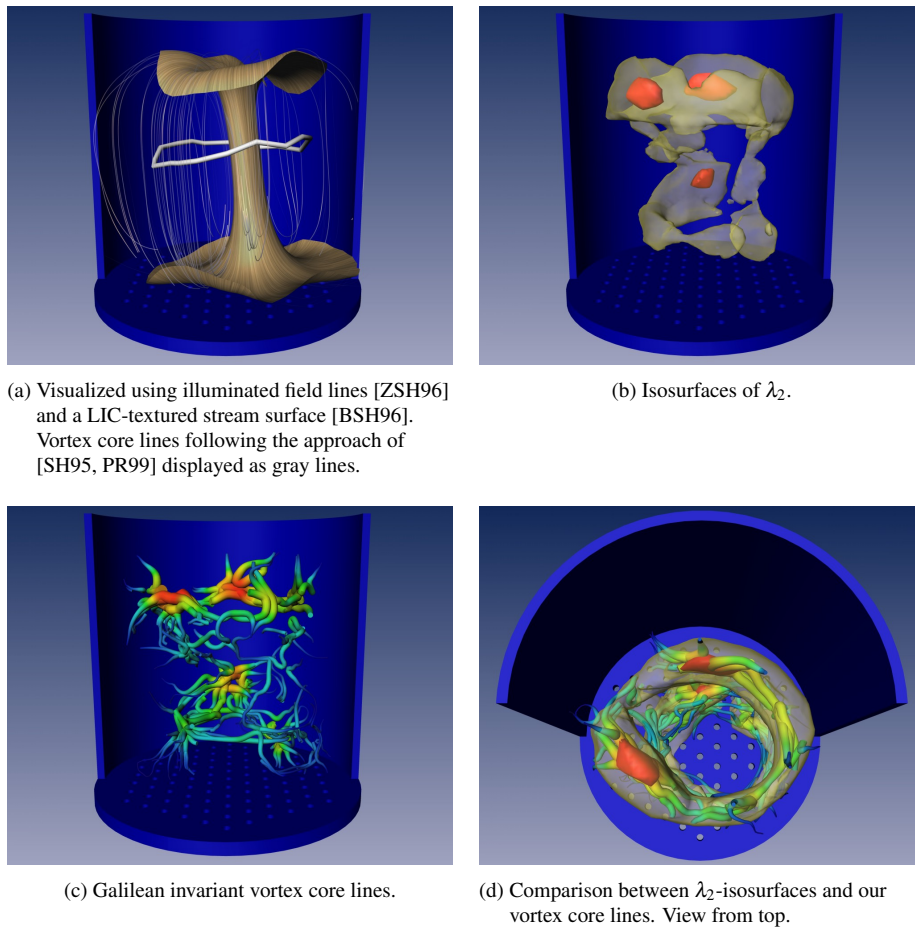


Figure 6.4: Bubble chamber. Vortex core lines extracted, colored and scaled according to λ_2 . Same colormap as in figure 6.3.

vortex shedding leading to the well known von Kármán vortex street. This phenomenon plays an important role in many industrial applications, like mixing in heat exchangers or mass flow measurements with vortex counters. However, this vortex shedding can lead to undesirable periodic forces on obstacles, like chimneys, buildings, bridges and submarine towers. The chain of vortices with their alternating orientation of rotation is clearly depicted in figure 6.3 due to the usage of spiraling orbits. This is a major property of the von Kármán vortex street. Furthermore, it can be seen that downstream the vortices lose their strength.

Figure 6.4 shows the geometry of a bubble chamber and its interior flow. The flow has been measured experimentally on a $11 \times 11 \times 10$ uniform grid by a biplanar x-ray angiography in a biofluidmechanics laboratory. The bubble chamber is used as a biochemical reactor. Air injection into the liquid through holes in the floor plate is used to improve the reaction. The dataset was provided by Axel Seeger, Biofluidmechanics Lab, Charite Berlin. Figure 6.4a shows a Galilean variant vortex core line according to [SH95, PR99] around which the flow spirals. Figure 6.4b shows isosurfaces of λ_2

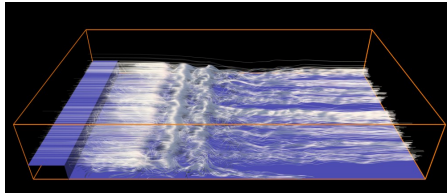
corresponding to different isovalues. In Figure 6.4c, the vortex core lines with respect to λ_2 extracted by our method are shown, sized and colored corresponding to λ_2 . Figure 6.4d is a combination of Figures 6.4b and 6.4c looking into the bubble chamber from above. This figure clearly shows that our approach yields vortex core lines in the center of the considered vortex region quantity.

Figure 6.5 shows the transitional flow around a backward-facing step. The flow field is obtained from a numerical simulation of Kaltenbach and Janke (both TU Berlin) at a Reynolds number of $Re_H=3000$ based on oncoming velocity and on step height. The corresponding boundary conditions are described in [KJ00]. The data set is given on a $266 \times 64 \times 128$ rectilinear grid. Figure 6.5a shows stream lines of the velocity field with respect to the original frame of reference. The vortex region quantity Q is visualized in figure 6.5b. This already gives an overview of the vortical structures inherent to this flow, but the visualization strongly depends on the choice of a transfer function. Figures 6.5c-d elucidate the dominant vortical structures by scaling and coloring the vortex core lines according to Q . This clearly shows that the depiction of Galilean invariant vortex core lines yields expressive visualizations even for very complex settings.

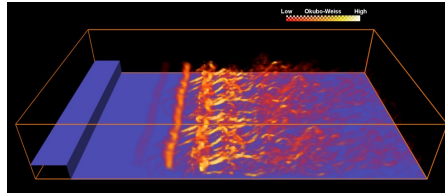
6.5 Limitations

A drawback of our method is that it requires second order derivatives. Due to the ill-conditioning of derivative computation this makes the method rather susceptible to noise. Hence, step sizes both in the zero search and in the integration must be chosen quite small for acceptable precision. Furthermore, as can be seen in Fig. 6.5d, this can result in imprecise starting points even for small step sizes. Here the spanwise Kelvin-Helmholtz vortices in front of the turbulent region are extracted as multiple parallel vortex core lines, rather than just one vortex as would be desired.

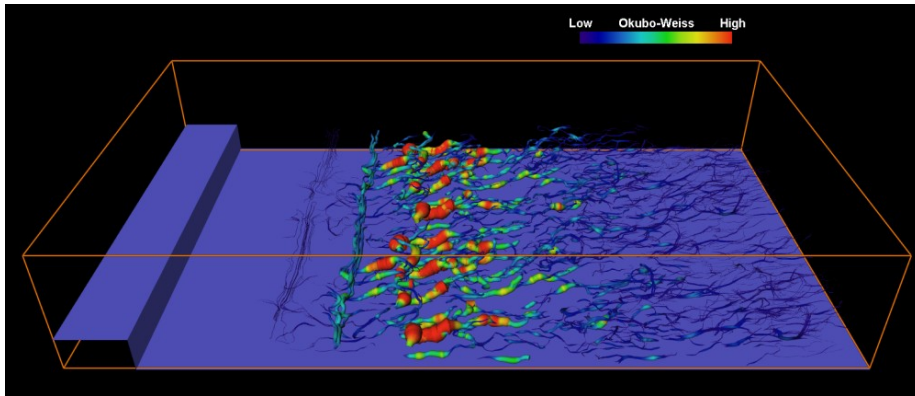
Hence, extremum line extraction method with fewer derivatives is desirable. Such a method is developed in the next chapter. It is based on scalar field topology using just first derivatives.



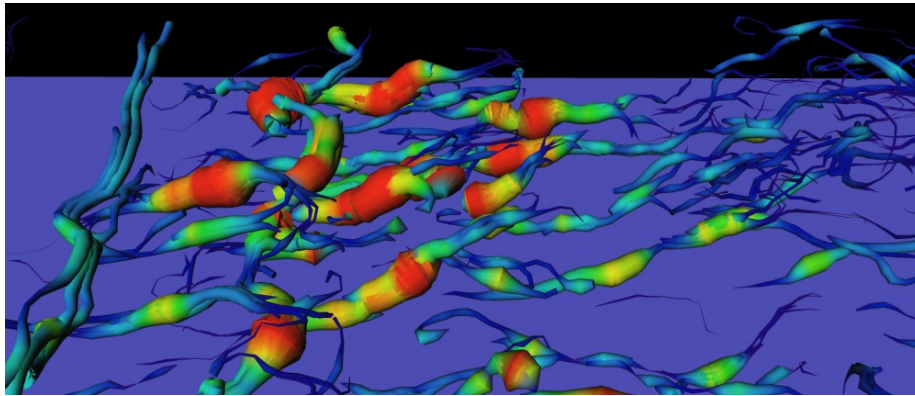
(a) Visualized using illuminated field lines with curvature-based seeding [WT02, WHN⁺03].



(b) Volume rendering of Q .



(c) Galilean invariant vortex core lines.



(d) Close up.

Figure 6.5: Flow around a backward-facing step. Vortex core lines extracted, colored and scaled according to Q .

Chapter 7

Vortex and Strain Skeletons

In this chapter, we extend the works of Miura et al. [MK97] and Sahner et al. [SWH05a] (see chapter 6) by extracting zero-, one- and two-dimensional extremal vortex features as the vortex skeleton in a Galilean invariant way, see Figure 7.1 for a motivating example. Prior to the development of this approach, all vortex core line extraction techniques aimed at the extraction of one dimensional vortex features, see chapter 3 for an introduction.

While vortex activity has been a research focus for many years, comparatively little is known about the extraction of features that can be identified with mixing properties of the flow. Mixing is actively researched in many fields, for instance in burning chambers where fuel and oxygen injection has to be synchronized for optimal combustion. Vector field topology – introduced to the visualization community by Helman et al. [HH89] – can be used for mixing detection, as saddle points are indicators for strain in the flow, as well as boundary switch and saddle connectors extracted by Theisel and Weinkauff [TWHS03, WTHS04a]. These are the intersection lines of 2D separatrices showing between which saddles and boundary regions particle transport takes place. Topological methods based on the flow field itself are not Galilean invariant, a property that in many cases is considered necessary from the physical point of view.

In this chapter we identify specific, derived scalar quantities of the flow that have a duality property: they detect vortex and strain regions simultaneously. By extracting extremal points, lines and surfaces of those properties we achieve Galilean invariant strain features that together assemble the strain skeleton. With those extracted features at hand we aim at tracking and comparing those structures in the future. However, this is beyond the scope of this work. We concentrate on the mere extraction and do not address possibilities of post processing the extracted structures. The extraction utilizes just first derivatives of the derived scalar quantities. This results in a considerable boost of robustness as compared to ridge line based approaches that require second or even third derivatives.

This chapter is organized as follows: In section 7.1 we clarify the notion of dual vortex and strain quantities and identify two criteria that meet this requirement. In section 7.2 we define vortex and strain skeletons as the collection of certain extremal structures of these quantities. We show how separation properties can be utilized to quantify the extent of vortex and strain features. Implementation issues of the extremal extraction are given in section 7.3. Finally, we apply our methods to a number of steady and unsteady data sets in section 7.4.

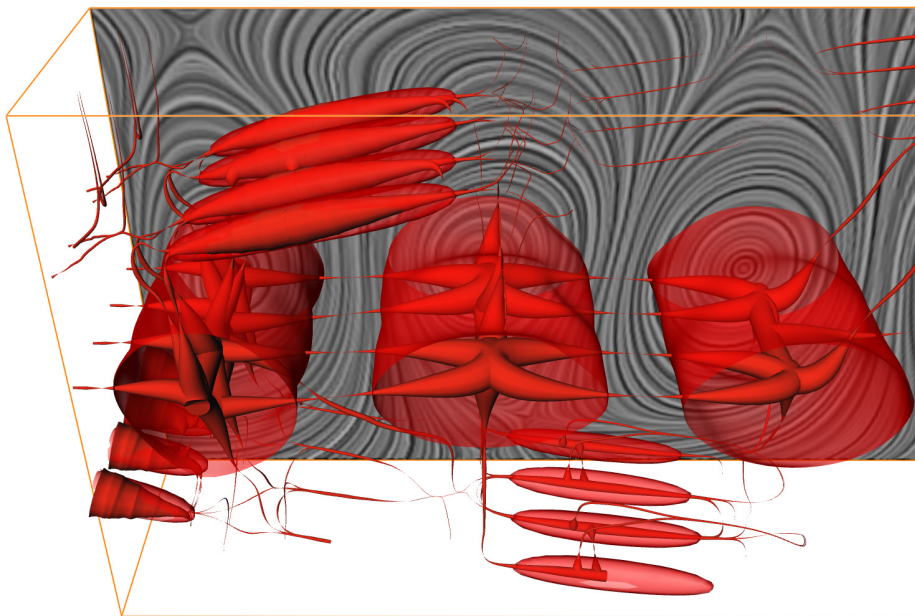


Figure 7.1: A closeup of the cylinder dataset showing vortex regions with transparent isosurfaces of Q . The approach presented in this chapter aims at extracting 1d and 2d extrema of duality quantities like Q . The lines shown here are the maximum lines of Q where $Q > 0$ extracted by our methods. The lines are scaled by their Q -value and can be regarded as centers of isosurfaces. The correspondence of the lines in the center of the swirling motion shown in the LIC plane justifies the notion of vortex core lines.

7.1 Dual Vortex and Strain Quantities

Inherent to the decomposition of the flow field gradient ∇v into \mathbf{S} and Ω from (3.1) is the following duality: vortical activity is high in regions where Ω dominates \mathbf{S} , whereas strain is characterized by \mathbf{S} dominating Ω .

In order to identify vortical activity, Jeong et al. used this decomposition in [JH95] to define the vortex region quantity λ_2 as the second largest eigenvalue of the symmetric tensor $\mathbf{S}^2 + \Omega^2$. Vortex regions are identified by $\lambda_2 < 0$, whereas $\lambda_2 > 0$ lacks physical interpretation. λ_2 does not capture stretching and folding of fluid particles and hence does not capture the vorticity-strain duality detailed above.

In the following subsections we discuss two quantities which utilize the decomposition of ∇v to identify not only vortices but also strain regions. This *duality property* of those quantities will later be used in section 7.2 to define 0D, 1D and 2D vortex and strain features that together assemble the corresponding feature skeletons.

7.1.1 The Okubo-Weiss Criterion

Whenever the Okubo-Weiss criterion $Q := \frac{1}{2}(\|\Omega\|^2 - \|\mathbf{S}\|^2)$ (see Sec. 3.1.1) is positive, the vorticity magnitude dominates the rate of strain, indicating a vortical region. Likewise, negative values indicate the dominance of strain over vorticity. We will use this duality property of the Okubo-Weiss criterion to extract vortex and mixing features.

See Figure 3.1 on page 32 for an example at the cylinder dataset. In this figure

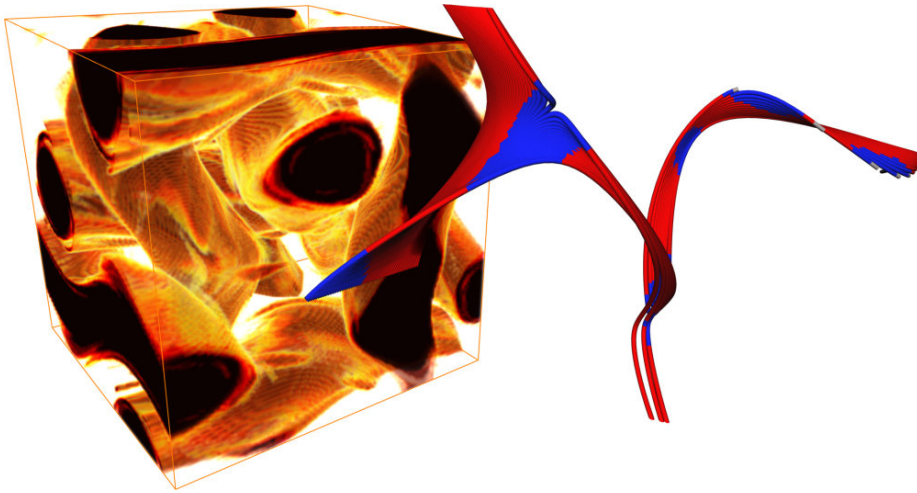


Figure 7.2: \mathbf{M}_Z -criterion of the periodic ABC-flow. Darker colors indicate higher ellipticity times. Some path lines used for computing \mathbf{M}_Z are shown: hyperbolic points are colored blue, elliptic points are red.

and throughout this chapter, we use red colors to denote vortex features. Blue color will stand for strain features. The data set was derived by Bernd R. Noack (TU Berlin) from a direct numerical Navier Stokes simulation by Gerd Mutschke (FZ Rossendorf). It is a 3D time-dependent Galerkin approximation in the time range $[0, 2\pi]$. It will be explained in detail in section 7.4. We use this data set throughout the next sections to illustrate our techniques.

7.1.2 The \mathbf{M}_Z -Criterion

The qualitative property stated in the end of section 3.1.3 is a duality property of \mathbf{M}_Z similar to that of the Okubo-Weiss criterion. Note that the duality of vortex and strain activity is intrinsic to the \mathbf{M}_Z -criterion, as \mathbf{M}_Z is solely based on a strain analysis. Hence, Haller defines a vortex as *lack of strain*.

The computation of \mathbf{M}_Z relies on the computability of the positivity of $\langle \xi, \mathbf{M}\xi \rangle$ on Z . We address the implementation issues next.

7.1.3 Implementation of \mathbf{M}_Z

For 3D time-dependent flows, the \mathbf{M}_Z -criterion defines a 3D time-dependent scalar field which can be computed as follows:

Figure 7.2 shows a volume rendering of \mathbf{M}_Z of the analytic ABC-flow as used by Haller in [Hal05]. Note how the saddle-like behavior of the path lines corresponds to hyperbolic (blue) points. However, it also shows a drawback of the method, as the path lines might leave the domain in non-analytic fields before the maximum integration time is reached. Further challenges can be seen in finding a suitable seeding set for path line integration and the determination of the maximum integration time T_p . Both aspects are open research issues regarding \mathbf{M}_Z .

For an implementation of the above algorithm, the computation of the positivity of $\langle \xi, \mathbf{M}\xi \rangle$ on the zero strain cone Z in step 3a remains to be clarified. Haller argues

At each time step t_i

1. Generate a set of seeding points for path line integration. This may be the original grid at t_i , a subset thereof, or the grid points of a uniform grid defined in a region of interest.
 2. Integrate path lines for T_P seconds. We use a 4th-order Runge-Kutta integration with adaptive step size.
 3. For each path line $x(t)$ started at seeding point (x_0, t_i)
 - (a) Decide for each point on the path line if it is elliptic or hyperbolic (Figure 7.2).
 - (b) Add up all times where the path line is elliptic and associate this total time value to x_0 at time t_i .
-

that Z can be described by an elliptic cone using the eigenvectors e_1, e_2, e_3 of the strain tensor \mathbf{S} and its corresponding eigenvalues s_1, s_2, s_3 , ordered such that

$$\text{sign} s_1 = \text{sign} s_2 \neq \text{sign} s_3, \quad |s_1| \geq |s_2|. \quad (7.1)$$

Then writing M in strain basis

$$\hat{M} = (e_1 \ e_2 \ e_3)^t M (e_1 \ e_2 \ e_3), \quad (7.2)$$

due to the symmetry of Z , the positivity of $\langle \xi, M\xi \rangle$ is equivalent to positivity of the one-parameter function

$$\begin{aligned} m(\alpha) = & \hat{M}_{11} b \cos^2 \alpha + \hat{M}_{22} a \sin^2 \alpha + \hat{M}_{33} ab \\ & + \sqrt{ab} \left(2\hat{M}_{13} \sqrt{b} \cos \alpha \right. \\ & \left. + 2\hat{M}_{23} \sqrt{a} \sin \alpha + \hat{M}_{12} \sin 2\alpha \right) \end{aligned} \quad (7.3)$$

for all $\alpha \in [0, 2\pi]$, where

$$a = -\frac{s_1}{s_3}, \quad b = 1 - a, \quad (7.4)$$

see [Hal05] for details. The performance of the positivity check of m on the interval $[0, 2\pi]$ is of crucial importance for the overall performance of the \mathbf{M}_Z -computation, as this check has to be performed $dT \cdot nS \cdot nT$ times, where dT is the number of time steps, nS the number of seeds per time step, and nT is the average number of sample points on a path line. We found that checking for zeros of m using bisection combined with a first-order derivative estimation speeds up the computation about 3 times compared to equidistantly spaced sign checks. However, compared to the Eulerian Q -criterion which can be computed in seconds, the Lagrangian approach of the \mathbf{M}_Z -criterion is much more time-consuming. For 128^3 seeding points the computation time can be up to two hours per time step on modern hardware.

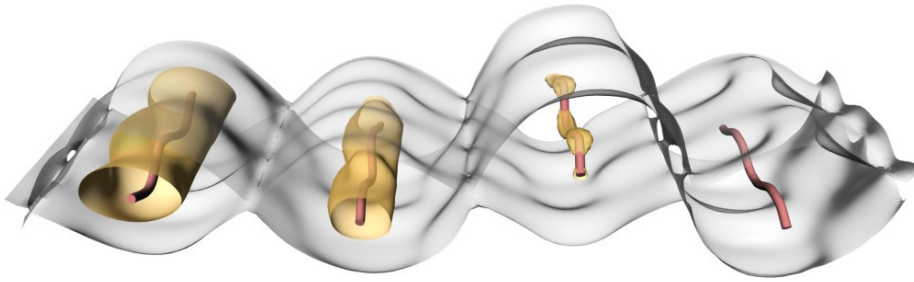


Figure 7.3: Isosurfaces are not well suited for higher dimensional extremum extraction: Gray isosurface $Q = 0$ is too far away from strongest vortex activity indicated by the shown extremum lines (red). Yellow isosurface $Q = 2.7$ splits up and misses some regions at all. The maximum lines of Q (red) show location and extent of the vortices correctly.

7.2 Vortex and Strain Skeletons

We aim at the identification of structures of high strain and vortical activity utilizing the criterions discussed in section 7.1. Common parameter-dependent visualization techniques like volume rendering or extraction of isosurfaces are not best suited for this due to the following reasons:

- These approaches require the choice of isovalues or transfer functions, which raises the question of how to choose these parameters appropriately.
- Isosurfaces tend to give wrong answers when it comes to examining the extent of vortices or strain regions since for certain isovalues they split up even inside such regions. For a visualization of the Q -criterion one may choose an isovalue of $Q = 0$ since this separates vortex and strain regions. However, from the resulting visualizations one can usually not infer the regions of strongest activity since typically the surfaces are too far away from those centers (Figure 7.3).
- Volume rendering is a purely qualitative technique which lacks the availability of sharp geometric features that can be used e.g. to measure distances between vortices.

To avoid these difficulties we choose to extract *extremal* features of Q and \mathbf{M}_Z . Due to their duality we identify the following features:

- The 0D, 1D and 2D *minimal* features of those quantities are points, lines and surfaces of maximal strain.
- The 0D, 1D and 2D *maximal* features of those quantities are points, lines and surfaces of maximal vortex activity.

One further property must be regarded for Q : only maximal features for which $Q > 0$ should be regarded as vortical features. On the other hand, only minimal features for which $Q < 0$ should be regarded as strain features. It is not clear if such a natural border exists for the \mathbf{M}_Z -criterion. The most natural choice might be $\frac{1}{2}T_P$, the half path line integration time, as there the hyperbolic and elliptic times on the path line balance. However, the actual integration time of each path line may be smaller than T_P since they can leave the domain earlier.

7.2.1 Extremal Feature Definition using Scalar Topology

We choose to compute these extremal features by means of a continuous extraction of scalar topology, see Sec. 2.5.1. The utilization of scalar field topology for extremal feature extraction has various advantages: The computation of separatrices as stream lines and stream surfaces of the gradient field is well understood, and stable tools exist for this purpose, see [GTS⁺04a, KOD⁺05] and section 7.3. No higher than first derivatives of the scalar field are being used, resulting in more stable algorithms as opposed to curvature based methods. Also, recent advances in scalar topology allow topological simplification of three-dimensional scalar fields using the Morse-Smale-Complex [GNP⁺05], answering the question, which topological features are most persistent.

All topological separatrices are of global nature. E.g., it is impossible to decide if a given point lies on a watershed by a local analysis.

By the separation property of watersheds and watercourses, watercourses of Q still have a meaning where $Q > 0$: although they can not be regarded as strain structures there, they still separate two regions of maximal vortex behavior (and analogously for maximal structures where $Q < 0$). By using this separation property of minimum surfaces and neglecting their meaning as a strain structure, we can separate vortex regions from each other, and analogously for strain regions, see Figure 7.5. Using the notion of watersheds we define

Definition 4 (*Vortex and Strain Skeletons*)

1. *The strain skeleton is the collection of minima, minimum lines and minimum surfaces of the duality quantities Q (where $Q < 0$) and \mathbf{M}_Z . The minimum surfaces partition the flow into vortex domains.*
2. *The vortex skeleton is the collection of maxima, maximum lines and maximum surfaces of the duality quantities Q (where $Q > 0$) and \mathbf{M}_Z . Following the notion of [SWH05a], lines in the vortex skeleton of Q are Galilean invariant vortex core lines with respect to Q . The maximal surfaces partition the flow into strain domains.*

We argue to call the surfaces in the strain skeleton strain surfaces. Accordingly, surfaces in the vortex skeleton are called vortex surfaces as direct generalization of the approach in [SWH05a] where Galilean invariant vortex core lines are extracted as extremum lines of vortex region quantities. By intuition, a vortex is a line with spiralling streamlines around it, but this is not necessarily the case for all vortices, see [JH95].

Figure 7.4 shows how the vortex and strain skeletons of Q can be used for hierarchical feature display, considering a subregion of the cylinder dataset as an example. Throughout this chapter, all strain structures are colored blue. Vortex structures are colored red. In 7.4a, minima and minimum lines scaled according to the scalar value of Q give a powerful overview of the strain structures, showing the most prominent features in one view. In 7.4b, the complete strain skeleton is shown. The additionally displayed surfaces show that the lines of extremal strain lie inside extremal strain surfaces. Analogously, Figure 7.4c shows the vortex core lines with respect to Q . The complete vortex skeleton is shown in 7.4d where the vortex core lines are complemented by the maximum vortex surfaces. Both for the vortex and for the strain skeleton, just those parts of the separatrices are shown where $Q > 0$ and $Q < 0$, respectively.

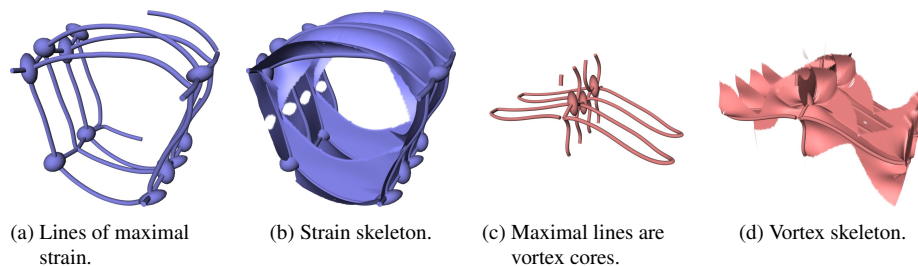


Figure 7.4: Strain and vortex skeletons in a subregion of the cylinder dataset. 7.4a shows lines of maximal strain. 7.4b shows the complete strain skeleton with the extremal strain surfaces displayed only where $Q < 0$. 7.4c shows the maximal vortex lines regarded as vortex core lines. 7.4d shows the complete vortex skeleton, adding the maximal vortex surfaces that lie between the vortex core lines. Again, vortex surfaces are displayed only where $Q > 0$.

Figure 7.5 gives an example where the minimum surfaces are used to subdivide a vortex in further regions. Only parts are shown where $Q < 0$, so an isosurface of $Q = 0$ would label all shown features as belonging to the same vortex. By showing the minimum surfaces (blue) we see how the vortex core line is subdivided into three parts, corresponding to the three maxima of Q along the line.

The extremal structures give a complete overview of the topology of the scalar quantities and hence of the vortex and strain activities in the flow. Note that this includes primary and secondary vortex structures. Primary vortex structures, e.g. Kelvin-Helmholtz vortices, can be observed in the cylinder flow as spanwise vortex core lines (Figure 7.1) – in this example they correspond to patterns of swirling stream lines in a certain frame of reference. Secondary vortex structures, e.g. rib vortices, are streamwise vortex core lines in the cylinder flow connecting neighboring Kelvin-Helmholtz vortices. This pattern is typical of shear flows and can be observed in other data sets as well (see Section 7.4). Note that secondary vortex structures can not be described as patterns of swirling motion using an obvious frame of reference, i.e., their extraction has to be based on measures independent of a certain reference frame. While our technique is able to extract those features, it does not allow to distinguish between primary and secondary vortices. This is an open research issue.

Before we give details about our extraction scheme in section 7.3, we motivate our choice of topological separatrices as features.

7.2.2 Separatrices vs. Height Ridges

While there is just one reasonable definition for a local 0D-extremum of a scalar quantity f , there is no canonical generalization to higher-dimensional features. Topological separatrices are just one choice, see [Ebe96] and [Soi99]. Another prominent approach is the height ridge definition (see e.g. Eberly [Ebe96] and Sec. 2.6) that is based on a convexity-analysis of the graph of f . We give a short introduction here and discuss advantages and disadvantages of both approaches to motivate our choice for the watershed definition. We refer to [Ebe96] for a deeper discussion.

Basically, a height ridge line follows the least-convexity-direction of the graph. A more formal definition is given in Definition 1 on page 29.

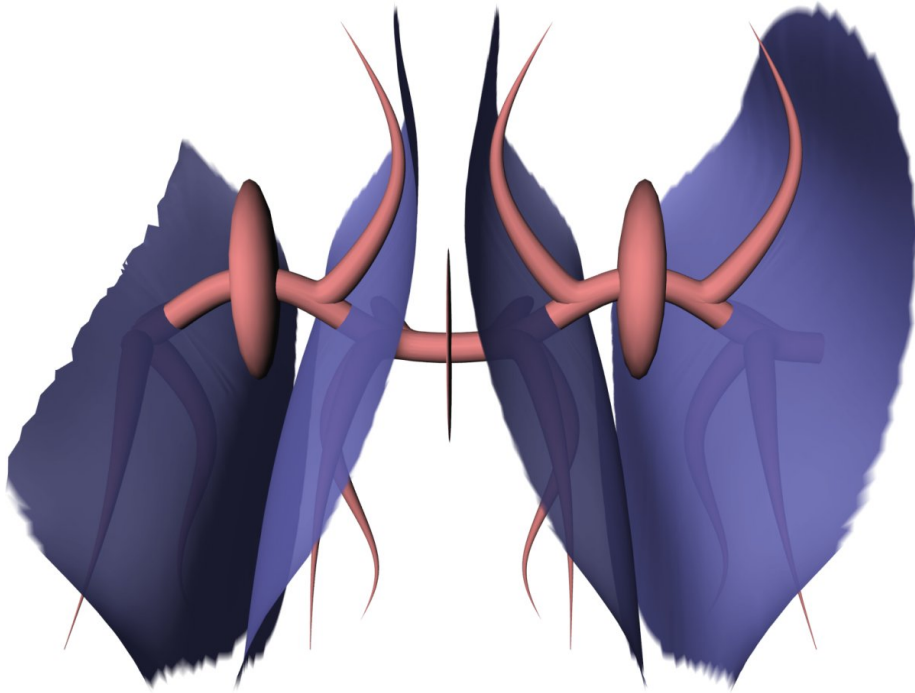


Figure 7.5: A close-up of the cylinder dataset. Three maxima of Q (red ellipsoids) are separated from each other by minimum surfaces (blue). This divides the domain into three different vortex regions. The maxima are connected by a maximum line (red), i.e., a vortex core line defined by Q .

While implementational details on height ridge extraction can be found in [Ebe96], we just sketch the procedure here: With the terminology as in Definition 1, $d_r = \nabla P_1 \times \nabla P_2$ is the direction in which a person on a height ridge line can walk along it. Given one point on each height ridge, it is sufficient to integrate d_r from each point to obtain the complete set of height ridges. Accordingly, a two dimensional height ridge is given implicitly by the surface normal ∇P_1 .

With those two different extrema definitions at hand the question arises, how they are related, and if the features of the one definition are possibly a subset of the other feature set. To this end, we note that at each saddle point with a Hessian matrix of full rank $2a$) holds for $i = 1, 2, 3$ in the definition of height ridges above. It is a consequence of the inverse function theorem [Str95] that depending on the eigenvalue setting, either a 2D height ridge and a 1D height valley or a 2D height valley and a 1D height ridge emanates from the saddle point.

So for every topological separatrix there exists a height ridge or valley counterpart. Note that although they have the saddle point in common, they do not necessarily have to coincide.

In contrast, there usually exists a variety of height ridges that are not topological separatrices. Figure 7.6 gives an example. Here we see a circular watershed on the crater rib, separating the local minimum inside the crater from the global minimum outside the crater. Additionally, a small perturbation of the symmetry of the crater rib creates a height ridge without separation property. This can be regarded as a conse-

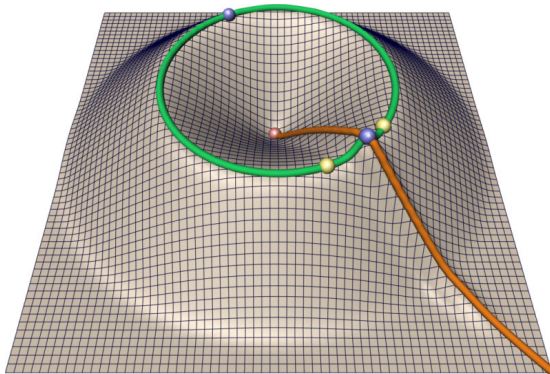


Figure 7.6: 2D-terrain exemplifying different extrema definitions. The closed green line is a watershed separatrix. Both the green and the orange line conform to the height ridge definition.

quence of the fact that watersheds are of global nature, whereas the definition of height ridges is local.

This locality is the main advantage of the height ridge definition. By this property it is possible to track a height ridge point in time using the feature flow field approach [TS03], but it is impossible to do so with a point on a watershed – simply due to its global nature. Although the time tracking of extremal features is beyond the scope of this thesis, we want to sketch how this disadvantage of topological separatrices can be overcome: A 2D-separatrix of some saddle can be followed in time by tracking the corresponding saddle point first [TS03, WTHS06] and extracting the separation surface again afterwards.

We see two disadvantages of the height ridge definition for our purposes: Firstly, height ridge extraction is per se less stable than watershed extraction, as the height ridge definition is based on the 2nd derivative of the scalar, and the scalar topology definition above uses first derivatives only. See Section 7.3 for comments on a discrete, derivative-free approximation. Secondly, topological separatrices can be sorted by the hierarchical computation of Morse-Smale complexes, for instance by persistence sorting, see 2.5.2 for details. We do not know of comparable considerations for height ridges.

7.3 Extremal Feature Extraction

In this section we provide implementational details for the extraction of topological features required for the vortex and strain skeletons. Here we discuss a discrete approximation method only. For the continuous extraction, please refer to section 2.5.1. The provided discrete methods work for arbitrary grids. The continuous versions require an interpolated gradient of the quantities.

7.3.1 Morse-Smale Approximation by Watershed Segmentation

The watershed segmentation yields an incomplete approximation [Soi99] of the Morse-Smale Complex (see Sec. 2.5.2) in the following way: Based on discretely extracted minima, water is shed in the following way: At start, each minimum gets its own label and is put in a priority queue. Now a region growing is performed. In each step, the unprocessed grid node with the smallest scalar value is grown by one grid node in each direction. The priority queue decides which grid node is processed next. As a result, we obtain a segmentation of the domain by labeled regions with time complexity

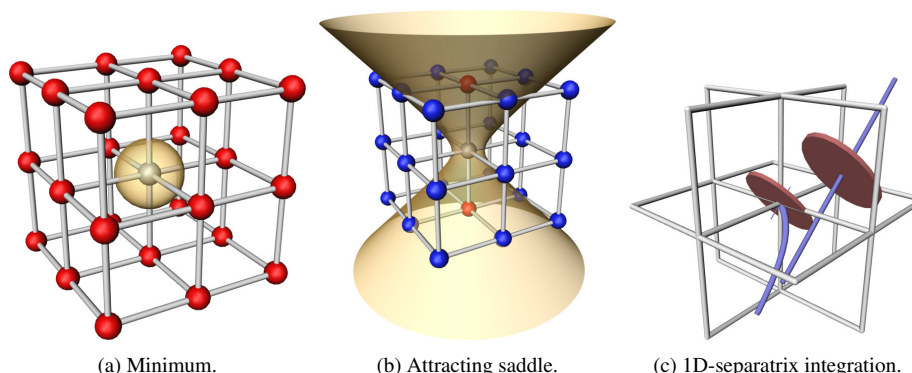


Figure 7.7: Discrete critical point extraction (exemplified using a regular grid). Red nodes are larger than the central node, blue nodes are smaller. Saddles can be extracted by counting the number of connected components of the 26 neighbors (7.7b). 7.7c shows that for separatrix integration, the continuously extracted critical points must be used, because the zeros of the gradient do not necessarily lie on grid nodes.

$n \log n$. Their boundary surfaces are topological 2D separatrices, and intersections of the surfaces are topological 1D separatrices. However, not all topological separatrices can be extracted this way. An obvious 2D example is the separatrix connecting the terminal maximum in Fig. 2.2d: this separatrix can not be determined by watershed segmentation, as it does not separate two different minima. Hence, using watershed segmentation, some features will be missed, but it allows for simple, robust and fast algorithms.

7.3.2 Critical points

A *discrete* method on grids with monotone interpolation works as follows: In this setting, all minima, saddles and maxima necessarily lie on the grid nodes. It clearly can be decided by looking at the direct neighbors if the point is a minimum or maximum. A saddle point can be decided by labelling the neighbors as *larger* and *smaller* respectively. If more than one connected component exists of any type, the grid node is a saddle, see Figure 7.7b for a regular grid example.

Note that the discrete extraction results in similar, but not identical sets of critical points compared to the continuous extraction. This is due to the fact that common interpolation schemes are not necessarily differentiable, and hence the continuously extracted critical points not necessarily lie on the grid nodes, see Figure 7.7c.

7.3.3 1D-Separatrices

(Minimum) 1D-separatrices are the lines where the (minimum) 2D-separatrices join in non-manifold junctions, Figure 7.4. This property can be used for a *discrete* extraction.

Note that for the seeding of continuous separatrix extraction (Sec. 2.5.1), the continuously extracted critical points must be used. Using the discretely extracted points would cause missing parts of the separatrix, because the two seeding points do not necessarily lie on opposite sides of the critical point(Figure 7.7c).

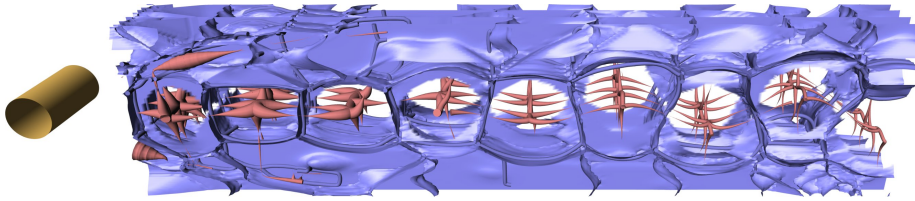


Figure 7.8: Strain skeleton of the cylinder data set. It partitions the domain in vortex regions. Inside the vortex regions, lines of maximal vortical behavior are shown scaled by Q . Close-ups of the structures can be seen in Figure 7.4.

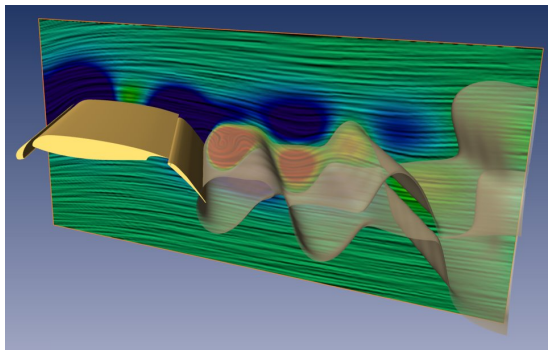


Figure 7.9: SCCH airfoil visualized using isosurface $Q = 0$ and a LIC plane colored by Q .

7.3.4 2D-Separatrices

The watersheds surfaces can now be extracted as the border surfaces between label regions. We use the generalized marching cubes algorithm due to Hege et al. [HSSZ97] for this purpose.

In complex settings we use this discrete surface extraction for 2D separatrices as it is faster and more robust.

7.4 Applications

Figures 3.1, 3.2, 7.3, 7.4, 7.5 and 7.8 show the flow behind a circular cylinder, already introduced in section 4.5. Figure 7.8 shows the complete strain skeleton of the cylinder dataset in blue, partitioning the flow into compartments that correspond to a single vortex each. Inside the compartments, the line structures of the vortex skeleton are shown.

Figures 7.9 and 7.10 show the flow around a Swept-Constant-Chord-Half-model (SCCH) of an airfoil that was simulated by Bert Günther (TU Berlin) at a Reynolds number of 10^6 on a curvilinear block structured grid with 1.3 million cells. Due to the constant chord and periodic boundary conditions this is a 2.5D configuration (the step from 2.5D to a realistic 3D configuration would mean to switch from constant to varying chord and to non-periodic boundary conditions). The sweep angle of the airfoil to the flow direction is 30° and the angle of attack is 6° . The turbulence was simulated by a combined URANS and DES approach. Figure 7.10b shows the line type structures of the vortex and strain skeletons of Q . Note that by our method, the collection of all extremal strain and vortex lines provide a good overview over the dataset, while the isosurfaces in Figure 7.10a miss the smaller features downstream.

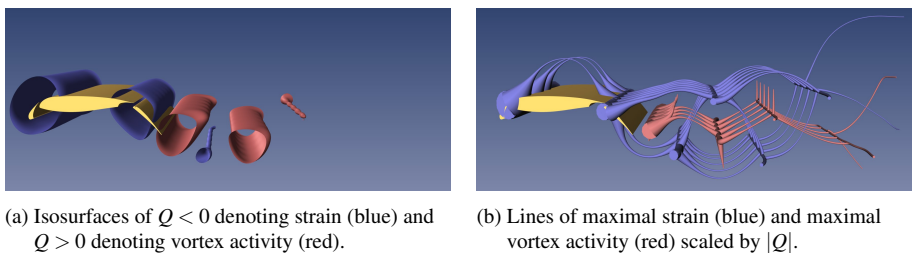


Figure 7.10: In the flow around an airfoil, isosurfaces of the Okubo-Weiss criterion Q are shown to the left. To the right, the line structures in the vortex and strain skeletons extracted by our method are displayed, showing lines of maximal strain (blue) and lines of maximal vortex activity (red) that are vortex core lines. Our method gives a complete overview of the location and extent of vortex and strain features in the flow, whereas the isosurfaces miss the smaller features downstream and give only a rough location for the larger features upstream.

In Figure 7.11 we applied our methods to a 3D time-dependent turbulent mixing layer. The velocity field has been computed with a pseudo-spectral direct numerical simulation by Pierre Comte, employing the computational domain and boundary conditions of [CSB98]. The Reynolds number is 100 based on the initial shear-layer thickness and convection velocity. The velocity ratio between the upper and lower stream is 3 : 1 (Figure 7.11a). The data consists of 500 time steps of a $480 \times 48 \times 96$ uniform grid. Figure 7.11b shows the minimum lines of \mathbf{M}_z which correspond to lines of maximal strain. It can clearly be seen that those structures lie in the shear layer which corresponds to intuition. In 7.11c isosurfaces of Q display the spatial evolution of Kelvin-Helmholtz vortices (primary vortex structures), vortex pairing, and the spanwise formation of streamwise rib vortices (secondary vortex structures). In 7.11d the vortex skeleton of Q is shown with lines scaled by the value of Q . The whole vortex structure of the flow can be seen at one view. In particular, our method is capable of resolving secondary vortex structures as well as the less vortical structures further upstream that are hidden by the isosurfaces in 7.11c.

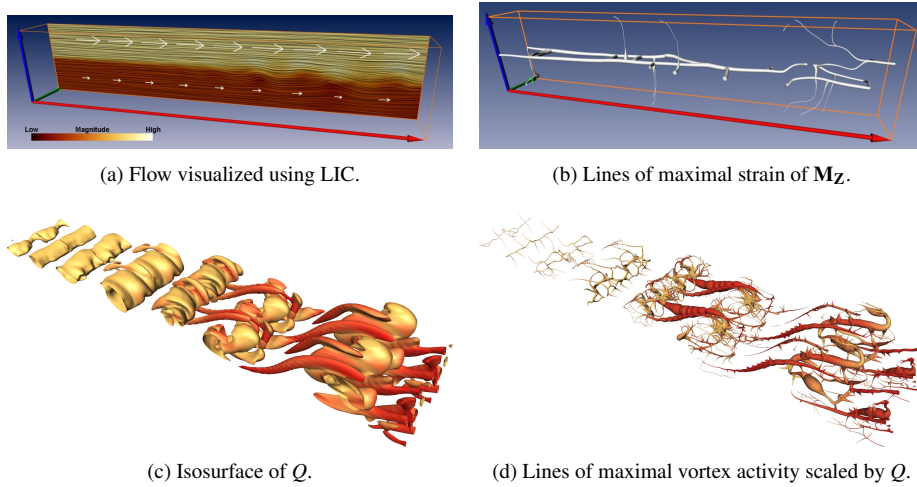


Figure 7.11: Turbulent mixing layer. The lines of maximal strain as indicated by \mathbf{M}_Z perfectly match up with the shear layer. The vortex skeleton of the Q -criterion elucidates the spatial evolution of Kelvin-Helmholtz vortices, vortex pairing, and the spanwise formation of streamwise rib vortices.

7.5 Outlook

Having extracted the vortex and strain skeletons using topology tools, topologically persistent simplification of those skeletons is desirable. Also, having extracted extremum lines using first derivatives by topological means, a derivative free extraction method is within reach due to the work on discrete topology by Forman [For98]. In the next chapter, we introduce such a derivative free approach to feature extraction including topological simplification. Due to the intricate nature of this task, we develop this technique for scalar fields defined on surfaces, focusing on curvature measures.

Chapter 8

Application: Feature Based Multiparameter Analysis of a Wing Model Flow

We apply the vortex skeleton approach developed in chapter 7 to analyze a flow data set which incorporates two additional parameters. These parameters are introduced by a periodic excitation of the flow around an airfoil which aims at increasing the lift. This active flow control technique employs an unsteady wall jet located near the leading edge of the flap to excite the flow and thus provoke either delay of separation or reattachment. Important parameters of this excitation are *frequency* and *intensity* of air injection. We study simulated datasets for 23 different parameter combinations. While finding the one with the highest lift is a simple computational task, we want to understand the underlying physics and explain *why* the lift is higher for a certain parameter set than for another. This can guide in finding new excitation strategies.

Each parameter combination is a 3D time-dependent flow field with 200 time steps. The grid consists of 1.3 million cells. This totals in over 200 GB of data. In this setting, an automatic feature extraction scheme is necessary due to the sheer size of the data set. Using the technique developed in chapter 7, we extract the vortex structures of the flow since they indicate the success of the induced perturbation.

8.1 Active Flow Control Simulation Data

Engineers aim at controlling flow with active and passive mechanisms. Passive methods contain effects caused by adequate profiling or by self-triggered processes of the flow. In contrast, active methods are characterized by excitation mechanisms that insert external energy into the flow.

Passive methods have been well-investigated in the last decades and have been integrated in scores of technical applications in the form of vortex generators as well as Gurney flaps [SGT04]. Today, the majority of research is focused on concepts with active flow control.

A large number of experimental and numerical studies have shown the general effectiveness of active flow control for single airfoils. In most investigations, leading edge suction is applied to delay transition [MCML89]; nonetheless, jet flaps are also

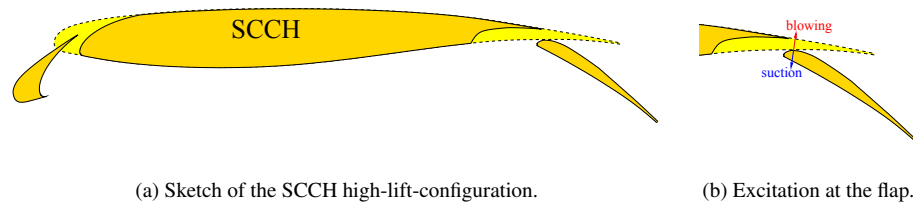


Figure 8.1: SCCH Wing Model and periodic excitation by blowing and suction.

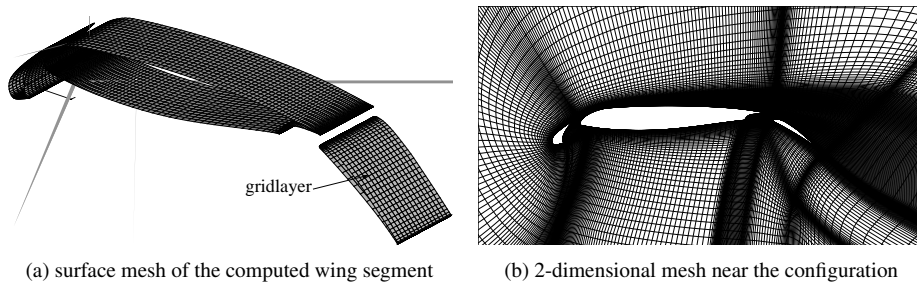


Figure 8.2: Computational mesh

employed for lift increase and manoeuvring. Surface suction/blowing can be used to rapidly change lift and drag on rotary wing aircraft [HJ97].

Overviews of active flow control are given by Wygnanski and Gad-el-Hak [Wyg04, GeH01].

8.1.1 Wing Model

The wing used for the simulations is the *SCCH* (Swept Constant Chord Half model) high-lift configuration that has already been used for several experimental studies targeting passive flow and noise control concepts [Koo05, KKE05], see Figure 8.1.1. The problem is modeled as an infinite swept wing (2.5D) in order to reduce the computational domain to a single spanwise segment of the configuration. The typical three-component setup consists of a main airfoil equipped with deployed slat and flap, see (figure 8.1a). All profiles have blunt trailing edges. The flap is situated at a fixed position underneath the trailing edge. The angle of attack is fixed at 6 degrees for the whole configuration, which is situated in the typical range of approach for civil aircraft.

In all numerical investigations the freestream velocity u_∞ corresponds to a Reynolds number of $Re = 10^6$, based on the chord of the clean configuration (with retracted high-lift devices). This high Reynolds number was chosen in the simulation to demonstrate the relevance to industrial applications.

Figure 8.2 shows the mesh around the slat, the main airfoil and the flap. The dimensions of the computational domain are 15 chords forward, above and below the configuration and 25 chords behind. Figure 8.2b shows the two-dimensional mesh around the entire configuration. The two-dimensional computational c-type mesh consists of 90,000 cells in total and is expanded in 3D in 16 layers, totalling in 1,300,000 cells (figure 8.2a) for each of the 202 time steps.

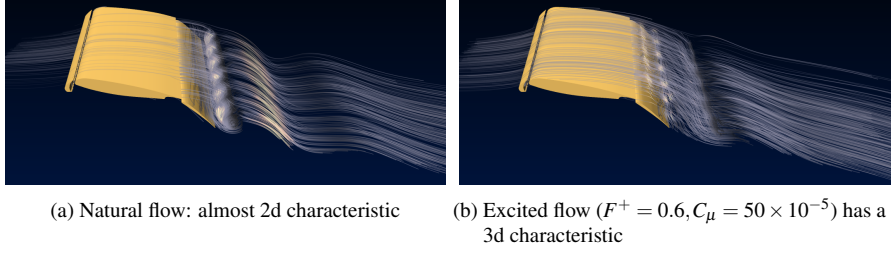


Figure 8.3: Streamlines showing the character of the flow above the flap.

8.1.2 Unexcited Flow

The flow field of the *SCCH*-configuration without excitation is characterized by massive separation above the upper surface of the flap. The mean separation point is located close behind the flap leading edge, and downstream a large recirculation region occurs. The unsteady behaviour of separated flow is mainly governed by large vortices shed from the flap trailing edge that interact with the vortices generated in the shear layer between the recirculation region and the flow passing through the slot between main airfoil and flap nose. The unsteady structures are characterized by a two-dimensional behaviour (figure 8.3a).

8.1.3 Excitation Mechanism

In the simulation the excitation was enforced by a periodic suction/blowing type boundary condition. The perturbation to the flow field is introduced through the inlet velocity on a small wall section representing the excitation slot:

$$u_{exc}(t) = u_a \cdot \sin \left[2\pi \cdot \frac{c}{c_k} \cdot F^+ \cdot \tau \right] \quad (8.1)$$

with $u_a = u_\infty \sqrt{\frac{c}{H} C_\mu}$, $F^+ = f_{per} \cdot \frac{c_k}{u_\infty}$, $\tau = t \cdot \frac{c}{u_\infty}$

where c is the clean chord length, c_k the flap length, u_a is the amplitude velocity of the perturbation oscillation, F^+ is the non-dimensional perturbation frequency, τ is the dimensionless time given in convective units of the whole configuration, H is the slot width between main airfoil and flap ($H = 0.00186 c_k$) and C_μ is the non-dimensional steady momentum blowing coefficient. The oscillating jet is emitted perpendicular to the wall segment of the excitation slot, and is located at 6% chord behind the flap leading edge (figure 8.1b).

8.1.4 Goal of the Excitation

It is the goal of the excitation to increase the lift by downsizing the recirculation zone and re-energizing it. To do so, we perturb the shear layer using a periodic excitation scheme as described in section 8.1.3. This increases the number of instabilities leading to a faster decay of the shear layer. These instabilities can be observed by means of vortex structures. A large, two-dimensional vortex binds a high amount of energy and causes less energy dissipation than smaller vortex structures with a three-dimensional

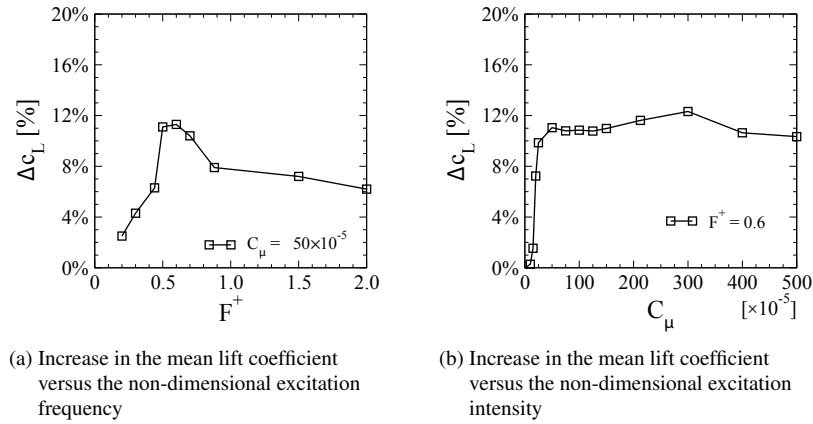


Figure 8.4: Numerical results of the excited flow.

characteristic. The smaller vortex structures help distributing the energy to the low-energy recirculation zone. Due to this the recirculation area becomes smaller.

Hence, the excitation has to influence the flow structures in the shear layer such that the vortices become smaller and have a three-dimensional characteristic. For this to happen, the parameters of the excitation have to be chosen carefully. In particular, the following parameters are of great importance:

- *Frequency*
We have to influence the flow with a frequency that interacts with the natural frequencies of the flow. Only this way we can target the natural flow structures.
- *Intensity*
The intensity of the excitation jet has to be strong enough to actually reach the shear layer.

In order to find an optimum excitation, simulations with different frequencies at $C_\mu = 50 \times 10^{-5}$ (which corresponds to the free stream impulse) were performed first. Simulations were then run at different intensities at the determined optimal frequency $F^+ = 0.6$. In the following we study the influence of these two parameters on the vortex structures and the lift. While the lift can be computed directly, the main goal for the vortex analysis is to understand *why* a certain parameter combination exhibits the highest lift.

8.2 Vortex Skeleton Parameter Study

In the following we study the influence of the actuation parameters *intensity* and *frequency* by comparing the vortex skeletons of the various parameter settings.

8.2.1 Frequency of Air Injection

Figure 8.4a presents results of the excitation in the simulation with different frequencies. The diagram shows the difference of the lift coefficient relating to the unexcited case depending on excitation frequency. The largest lift can be found at a frequency

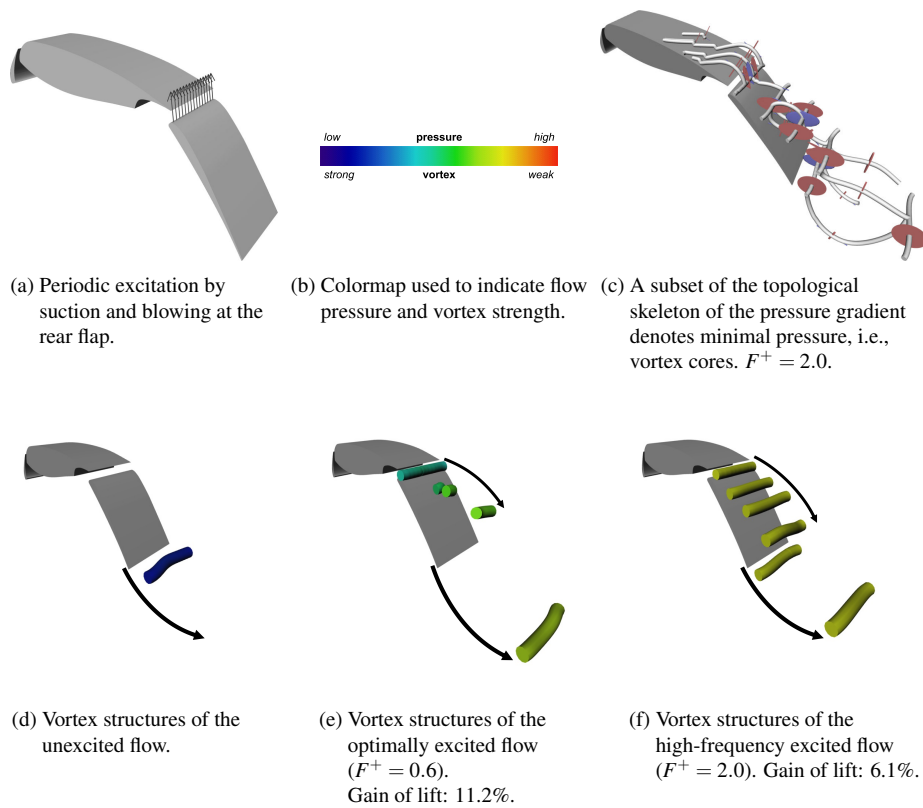


Figure 8.5: A topology-based vortex analysis of the flow around an airfoil elucidates the impact of an active flow control technique and explains why a high-frequency excitation leads to a smaller gain of lift.

of $F^+ = 0.6$. In this case the lift coefficient can be enhanced by 11% compared to the baseline simulation. In the range of strong gain in lift ($F^+ = 0.2 \dots 0.5$) the infinite wing achieved the same change in lift at higher frequencies compared to the two-dimensional flow. At frequencies slightly higher than the optimal excitation frequency ($F^+ = 0.6$) the gain in lift decreases.

In figure 8.5 we show the results of our topology-based vortex analysis which has been applied to study the influence of the frequency of air injection. While we know *which* frequency yields the best lift, our main goal is to uncover the underlying physics: we want to know *why* a certain frequency yields higher lift than others.

Figure 8.5a indicates the setup: we want to achieve performance enhancements by controlling the flow separation at the rear flap using periodic air injection. The vortex structures have been extracted as topological separatrices of the pressure gradient and denote lines of minimal pressure. Figure 8.5c shows parts of the topological skeleton of the pressure gradient. A quantification of the separation lines based on pressure and a subsequent filtering of weak vortices has been applied. This allows us to concentrate on the most important information.

The result is shown in figures 8.5d–f where the impact of the frequency of air injection onto the vortex structures can be studied. Note that this is a five-dimensional data set consisting of three spatial dimensions, time, and the frequency parameter di-

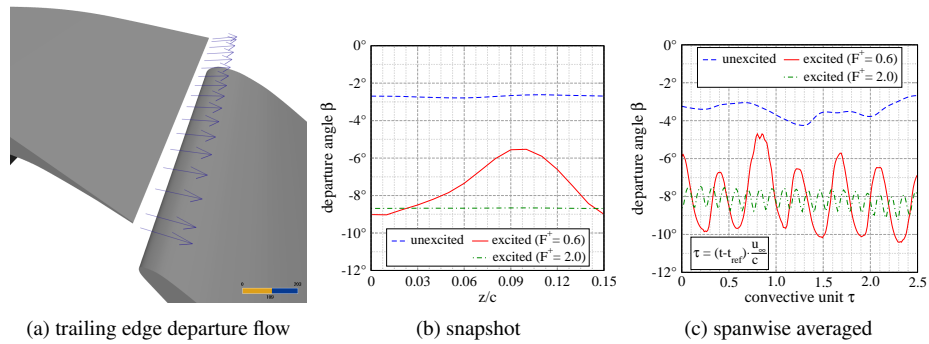


Figure 8.6: Our vortex analysis of different excitation frequencies could be evaluated successfully by examination of the trailing edge departure angle of the main airfoil (first column). The angle between these vectors and the oncoming flow direction is shown at a snapshot in the second column and spanwise averaged in the third. The dashed, solid and dotted lines denote the unexcited flow, $F^+ = 0.6$, $F^+ = 2.0$, respectively.

mension – the intensity is fixed. Raising the frequency causes a reduction of the lower vortex, which is a necessary condition for gaining lift. However, higher frequencies ($F^+ > 0.6$) are not beneficial to the lift. Using a visual comparison of the vortex structures at different frequencies, we found that new vortex structures are induced by the air injection itself. This has a negative effect on the pressure ratio and consequently on the lift. Especially at higher frequencies, the excitation dominates the natural flow structures and induces long-living, almost two-dimensional vortices in fast succession at the top of the rear flap (figure 8.5f). In contrast to this, the induced vortices at $F^+ = 0.6$ dissolve quickly and therefore, they are less influential.

We want to evaluate our analysis technique using a more traditional method. Figure 8.6 shows the trailing edge departure angle of the main airfoil in the unexcited case, which we define as the angle between the oncoming flow direction and the flow direction at the trailing edge. This angle is directly linked to the pressure distribution above the main airfoil: a larger angle indicates a higher pressure gradient which is a main trigger of increased lift. The second and third column of figure 8.6 show the trailing edge departure angle for the unexcited, optimally excited ($F^+ = 0.6$), and high-frequency excited flow ($F^+ = 2.0$). At $F^+ = 0.6$ (solid red line in 8.6b) we can observe a pronounced three-dimensionality in contrast to the other cases. This causes a higher dissemination of the excitation energy and inhibits the formation of strong vortex structures – as we already know from our vortex analysis, see figure 8.5e. At higher frequencies, the excitation dominates the natural flow and the departure angle is almost two-dimensional in spanwise direction similar to the unexcited case. This goes hand in hand with the observed vortex structures for these cases which are almost two-dimensional, too.

Our topology-based vortex analysis technique contributed to the physical understanding of the flow structures and was a substantial part of the optimal choice of parameters.

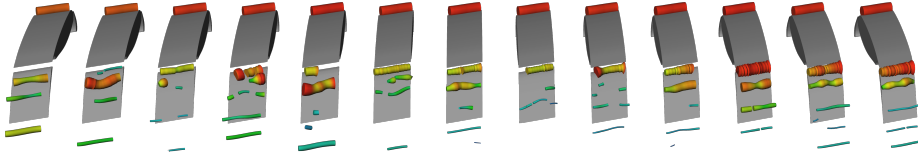


Figure 8.7: The intensity variation was performed on 13 different intensity values ranging from $C'_\mu = 10$ to $C'_\mu = 500$ (shown in increasing order from left to right). See figure 8.8 for a close-up.

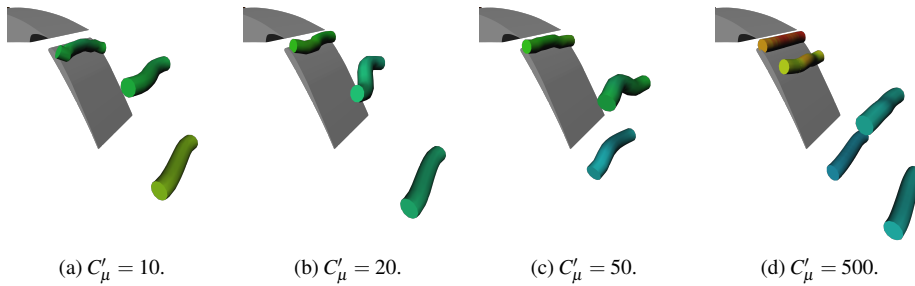


Figure 8.8: Topology-based vortex analysis of the intensity variation at four selected intensities. At intensities of $C'_\mu = 20$ and $C'_\mu = 50$ the structures are three-dimensional, whereas at lower and higher intensity, the structures have a two-dimensional characteristic. At $C'_\mu = 500$, a strong excitation-induced vortex is created at the top of the rear flap.

8.2.2 Intensity of Air Injection

At the optimal excitation frequency of $F^+ = 0.6$ we simulate 13 different excitation intensities of $C_\mu \cdot 10^5 = C'_\mu = 10, 15, 20, 25, 50, 75, 100, 125, 212, 300, 400, 500$. Figure 8.7 shows the result of our vortex analysis.

Excitation with low intensity ($C'_\mu = 10 \dots 50$) leads to a strong increase in lift (max. 11%) (figure 8.4b). However, if C'_μ becomes larger than 50, the lift ceases to increase further. In general, the results for both excitation parameters show that the lift either cease to increase or begins to decrease slightly if the frequency or intensity exceeds a certain limit ($F^+ = 0.6, C'_\mu = 50$).

As in the frequency modulation, our topology-based vortex analysis helps in understanding those effects: At a low intensity of $C'_\mu = 10$, the excitation jet is too weak to perturb the shear layer significantly, see figure 8.8a. This changes at higher intensities. At $C'_\mu = 20$ (figure 8.8b) and $C'_\mu = 50$ (figure 8.8c) three-dimensional vortex structures are produced as expected by the perturbation of the shear layer. However, at $C'_\mu = 500$, the excitation is so intense that the excitation jet penetrates through the shear layer and a significant part of the energy is absorbed by the free stream. This results in a large, excitation induced vortex with a very short life cycle (figure 8.8d). As only parts of the energy can be used to destroy the shear layer, the downstream vortices keep much of their two-dimensional characteristics. Furthermore, $C'_\mu = 500$ is 10 times higher than in the $C'_\mu = 50$. Although the effectivity is comparable, the excitation with $C'_\mu = 50$ is much more efficient.

Chapter 9

Extraction of Feature Lines on Surface Meshes

In the last chapters we developed a vortex extraction scheme based on the extraction of extremal lines, first by using third derivatives in chapter 6, then by means of just first derivatives in chapter 7. In this chapter we will show how extremal lines can be extracted without any derivative computation. Being interested in vortex extraction of 3D flow fields, we aim at extremal lines of 3D scalar fields. However, in this chapter, due to the complexity of the matter, we tackle this problem for triangular surfaces. Here, vortex extraction does not make much sense, but there are other extremal structures that are worth examining, namely salient feature lines that turn out to be extremal structures of certain curvature measures of the surface. With the work in this chapter we prove that a gradient free extraction of extremal structures is possible. It is left for future research to further develop the methods for 3D scalar fields.

However, the extraction of salient surface features is a crucial and interesting task in its own right. In non-photorealistic rendering, for instance, the goal is to depict a surface with a sparse set of characteristic features only. Another important example is the registration and fusion of a number of shapes, e.g. for atlas generation in biomedical applications, such that common features are identified correctly.

Most commonly, feature lines are defined as ridges and ravines of the surface, a notion of differential geometry including third orders of surface derivatives. Several highly evolved algorithms exist to extract feature lines based on this definition [Ebe96, OBS04, YBS05, YBYS07, HPW05, SF04]. As the stability of third derivatives is difficult to control, those approaches require the adaptation of a number of parameters on a per surface basis. This is especially unsatisfactory in registration applications, where a large number of surfaces has to be processed to establish an anatomic atlas.

In this chapter we will undertake an alternative approach based on Morse Smale (MS) theory [Mil63, Ban70, EHZ03, Pas07, For98], which is an algebraic tool for measuring topological features of a function $f : M \rightarrow \mathbf{R}$ on a manifold M . A simplification strategy can be used to obtain a hierarchy of MS complexes giving rise to a natural discrimination of feature scales. We will present a modification of the original persistence simplification strategy that is capable of extracting *extremal* feature lines in the sense that the value of f is significantly larger or smaller on the feature line than in its vicinity. This extremality property is satisfied by the 1-cells of the initial MS complex that contain the watersheds and watercourses [Soi99] as special cases.

Our main contributions are:

- We identify suitable feature indicator functions for extracting extremal feature lines (Sect. 9.1.1).
- We show that persistence simplification is not useful for extracting extremal feature lines (Sect. 9.1.3).
- We introduce a novel simplification strategy that preserves extremal feature lines (Sect. 9.1.4).
- MS based feature detection necessarily produces connected lines, which is not generally desired. We propose a method to alleviate this problem by discarding unwanted lines in a post-processing step (Sect. 9.1.5).

The main benefits of our method are:

- By using *discrete* Morse Theory, the extraction method does *not* require any derivative estimates.
- There is *only one* user-defined parameter in the method defining the level of significance of the resulting features.
- The method is insensitive to noise in the input fields (Sect. 9.2.1).

9.0.3 Previous Work

Several approaches for ridge line extraction on surfaces exist. Ohtake et al [OBS04], Yoshizawa et al [YBS05, YBYS07], Hildebrandt et al [HPW05] and Stylianou et al [SF04] present different approaches for controlling the involved third derivatives of the surface. While all those approaches yield impressive results, they suffer from the fact that a number of parameters has to be adjusted.

The present approach is based on discrete Morse-Smale theory as defined by Forman [For98, For02, Lew05]. See section 1.1 for related work in this area.

9.1 Discrete Feature Line Extraction Method

9.1.1 Scalar Feature Indicators

On a smooth surface, the two principal curvatures $\kappa_1 \geq \kappa_2$ measure the bent of the surface along the corresponding curvature directions. Therefore, the principal curvatures are indicators for surface specific feature lines. Whereas κ_1 is maximal along convex feature lines, κ_2 is minimal along concave feature lines. *Curvedness*, as introduced by Koenderink et al [KvD92], is defined by $C = \sqrt{\frac{1}{2}(\kappa_1^2 + \kappa_2^2)}$. This measure is large whenever any principal curvature has a large absolute value, and hence measures convex and concave regions simultaneously. It is maximal along both convex and concave feature lines.

On a triangulated surface, κ_1 and κ_2 , can be defined discretely for each vertex of the surface using the method described by Hildebrandt et al [HPW06].

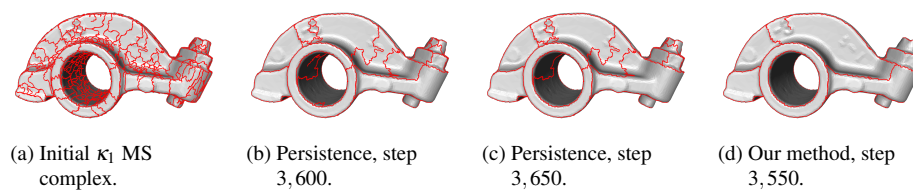


Figure 9.1: Ascending 1-cells of the initial MS complex of κ_1 on the rocker arm are a candidate set of convex feature lines in (a). 3,600 persistence cancellations lead to the line set depicted in (b). Clearly, all feature lines are still contained. After 3,650 cancellations, one of the main features is broken by persistence cancellation in (c). In contrast, feature line preserving cancellation as proposed here maintains the main features in (d).

9.1.2 Morse-Smale Complex and Feature Lines

Ascending 1-cells are maximal lines of scalar feature indicators, as they are lines of steepest ascent separating two valleys if the 1-cell is a watershed, or one valley, if the 1-cell is connected to a terminal maximum, see Fig. 2.2f, and analogously for descending 1-cells. Depending on the scalar feature indicator, just the ascending (κ_1, C), or descending (κ_2) 1-cells are of interest. Not all of these 1-cells are necessarily features. This is due to the fact that noise adds additional saddles of small difference in function value, an effect known as oversegmentation in the field of watershed transformation [Soi99]. We aim at differentiating the 1-cells that are salient features from noise by applying a cancellation criterion that favors the concentration of scalar indicator value on 1-cells opposed to their vicinity.

9.1.3 Persistence Cancellation

The *persistence* [ELZ00] of a saddle s is the least difference in function values compared to the cancellable extrema it is linked to. Cancelling saddles by increasing persistence realizes topological simplification by means of the least geometric change, see Bremer et al [BEHP04].

Persistence cancellation does not preserve feature lines as seen at the rocker arm surface in Fig. 9.1 and the ascending 1-cells of the initial MS complex of κ_1 containing the convex feature lines of interest. Persistence cancellation initially keeps this property in Fig. 9.1b. However, further cancellation destroys one of the most characteristic convex feature lines, and a line traversing a concave region is preferred, see Fig. 9.1c. The obvious reason for this is that the persistence measure only considers the difference in function value between the minimum and the saddle, and hence regards only the deepest point of the feature line.

9.1.4 Feature Line Preserving Cancellation

A feature line preserving cancellation criterion should quantify the whole feature line that is deleted by a cancellation. We want to keep just salient 1-cells from the initial MS complex which means that the quantification of the feature line should be compared to the vicinity of the line. The MS complex allows us both. An ascending 1-cell lies

between two valleys which we use as vicinity. In the following, I stands for an arbitrary scalar feature indicator.

The proposed quantification $\sigma_r(s, m)$ of a cancellation of saddle s and minimum m deleting the ascending 1-cell r is the difference of the average of I on the 1-cell and the average of I on the valley $U(m)$ containing m :

$$\sigma_{max}(r, m) = \frac{1}{area(r)} \int_r I - \frac{1}{area(U(m))} \int_{U(m)} I \quad (9.1)$$

and analogously for cancellations with a maximum M on the hill $S(M)$ deleting the descending 1-cell v ,

$$\sigma_{min}(v, M) = \frac{1}{area(S(M))} \int_{S(M)} I - \frac{1}{length(v)} \int_v I. \quad (9.2)$$

A line can not be regarded as a feature if the accumulated I average does not differ significantly compared to both incident patches. Hence, the *feature line significance* $\sigma_{max}(r)$ and $\sigma_{min}(v)$ is defined as the minimum of both values:

$$\sigma_{max}(r) = \min(\sigma_{max}(r, m_1), \sigma_{max}(r, m_2)), \quad (9.3)$$

$$\sigma_{min}(v) = \min(\sigma_{min}(v, M_1), \sigma_{min}(v, M_2)), \quad (9.4)$$

where r lies between the minima m_1, m_2 , and v between the maxima M_1, M_2 . In words, a 1-cell is regarded as salient if on the line more average I -value is accumulated than in the patches that the 1-cell bounds.

Quantifying 1-cell cancellation as in (9.3), (9.4) preserves feature lines, but also, short offshoots of the feature lines are kept that fork off orthogonally from the requested lines. To punish such offshoots, $\sigma_{max}, \sigma_{min}$ are adjusted *at terminal 1-cells only*. These emanate from those saddles, where one of the incident extrema is connected to just one saddle. For an ascending 1-cell r with r_1, r_2 being the part of the 1-cell ascending to the maxima M_1, M_2 , and a minimal 1-cell v with v_1, v_2 being the part of the 1-cell descending to the minima m_1, m_2 , respectively, the quantifications

$$\Gamma_{max}(r_i, M_i) = \frac{1}{area(r_i)} \int_{r_i} I - \frac{1}{area(S(M_i))} \int_{S(M_i)} I \quad (9.5)$$

$$\Gamma_{min}(v_i, m_i) = \frac{1}{area(U(m_i))} \int_{U(m_i)} I - \frac{1}{length(v_i)} \int_{v_i} I \quad (9.6)$$

are considered additionally by updating $\sigma_{max}(r)$ and $\sigma_{min}(v)$

$$\sigma_{max}(r) \leftarrow \min(\sigma_{max}(r), \Gamma_{max}(r_1, M_1), \Gamma_{max}(r, M_2)), \quad (9.7)$$

$$\sigma_{min}(v) \leftarrow \min(\sigma_{min}(v), \Gamma_{min}(v, m_1), \Gamma_{min}(v, m_2)). \quad (9.8)$$

In words, terminal feature lines that are not salient compared to the patches they lie in, are discarded as well. See Fig. 9.2 for an example of a 1-cell forking off a feature line.

Only valid cancellations may be performed to keep the alternating quad property of the MS complex. This means, a saddle/maximum cancellation is allowed only if the saddle connects two different maxima. Therefore we define the topological adaptations $t_{max}(r) = \sigma_{max}(r)$, if the 1-cell r can be deleted by a valid cancellation, ∞ otherwise, and $t_{min}(v) = \sigma_{min}(v)$, if a valid cancellation exists, ∞ otherwise.

The cancellation criterion t quantifying a saddle s by the least significant cancellable 1-cell passing through the saddle is then defined by

$$t(s) = \min(t_{max}(r), t_{min}(v)). \quad (9.9)$$

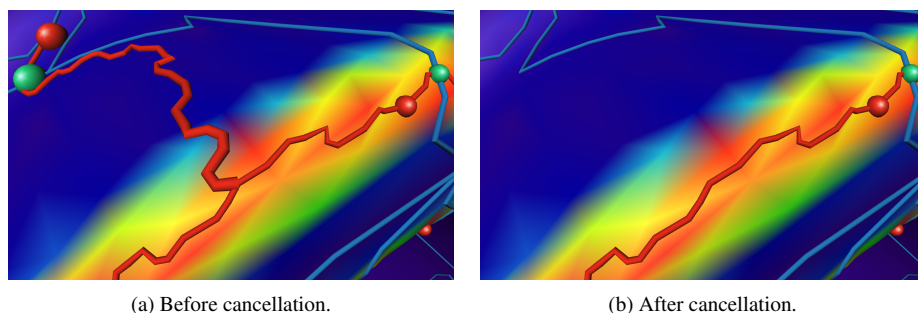


Figure 9.2: Cancellation of a terminal 1-cell forking off the outer ring feature of the rocker arm colored by κ_1 . The 1-cell emanating from the left saddle ends at the maximum on the high curvature area, but only the part up to the branching point is being deleted by the cancellation.

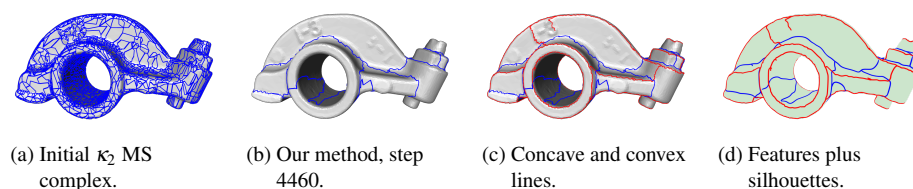


Figure 9.3: a) The initial MS complex of κ_2 containing 4476 descending 1-cell is taken as a starting point. b) 4460 feature preserving cancellations yield the visually best concave feature lines. c,d) Feature lines plus silhouette represent the geometry.

The reduction of lines in the initial complex now proceeds by canceling saddles in increasing order of t . This ensures that the lines with strongest feature line significance are kept. The saddle is canceled with the extremum for which equality is obtained in (9.9). See the result of using *feature line significance* as cancellation criterion in Fig. 9.1d. The deviation of the 1-cell as for *persistence* cancellation does not occur here; it stays on the feature line.

Two notes are important here:

1. Only one type of 1-cell is a meaningful feature line for most scalar indicators. We handle both types of 1-cells equally anyway for each quantity as this ensures a homogeneous cancellation of minima and maxima.
2. The 1-cell r used to evaluate (9.1) is restricted to the path uniquely associated with the saddle under consideration. Paths from different saddles can merge before reaching the maximum as depicted in Fig. 9.2a. This is different from the smooth setting where different 1-cells never meet. In Fig. 9.2b, the left saddle was canceled with the terminal maximum. Only the unique part of the 1-cell was removed. The remaining path to the maximum connects to other saddles and is therefore retained. Hence, 1-cells are quantified up to the first branching point only.

9.1.5 Discarding Topologically Forced Lines

1-cells r for which no valid cancellation exist might have a smaller feature line significance than the last canceled 1-cell. In this case, r is marked invisible and not displayed afterwards. Examples are the two concentric rings around the hole of the rockerarm in Fig. 9.1d that are not connected as forced by topology, and the cow's head in Fig. 9.6f, where the eye area is detached from all other lines.

9.1.6 Optional Line Smoothing

Our method results in lines that either consist of vertices or triangle centers of the mesh. While this is topologically correct, it might not be visually appealing. We applied a simple averaging with fixed filter length. The adapted Laplacian line smoothing technique introduced by Hildebrandt et al [HP04] gives appropriate guarantees when needed.

9.1.7 Implementational Issues

For the computation of the MS complex hierarchy we use the approach introduced by Cazals et al [CCL03]. Instead of using the Union Find structure detailed there, we work directly on the primal and dual spanning tree. A saddle / minimum cancellation is performed by adding the saddle as edge into the primal spanning tree and flipping all edges on the path from the saddle to the cancelled minimum. This results in simple algorithms but affects the performance as for each cancellation a number of edge flips have to be performed.

For computing feature line significance t (9.9), the involved quantities are stored in the primal and dual spanning trees. Each minimum and each maximum is being initialized with the area of their unstable and stable manifolds respectively, and with the integral of the scalar indicator over the manifold. Any saddle stores the integral and area of the stable manifolds (triangle strips), and the integral and length of the unstable manifolds (edge sequence) emanating from the saddle. Upon cancellation, those values are propagated from the canceled extrema to its uncanceled analogon, so computation of feature line significance requires $O(1)$ time.

All saddles are stored in an AVL tree that is used as priority queue. When a saddle is canceled, the saddles incident to its incident extrema have to be reevaluated, as t can be reduced by a cancellation. In theory, a single extremum can be incident to a majority of the saddles in the complex which means $O(n \log n)$ for a single update of the queue in the worst case. In practice, t is designed to cancel minima and maxima on the same priority, and hence our cancellation criterion is not significantly slower than persistence cancellation.

9.2 Results

For all results, the used curvature estimates are computed with the discrete method introduced by Hildebrandt et al [HPW06], without smoothing. All feature lines are shown as extracted by the method. We use red lines to display convex features (maximal lines of κ_1), blue lines for concave features (κ_2 minimal), black lines for *curvedness* C . Smoothed lines are displayed in the unshaded images only, all others contain the unsmoothed originals.

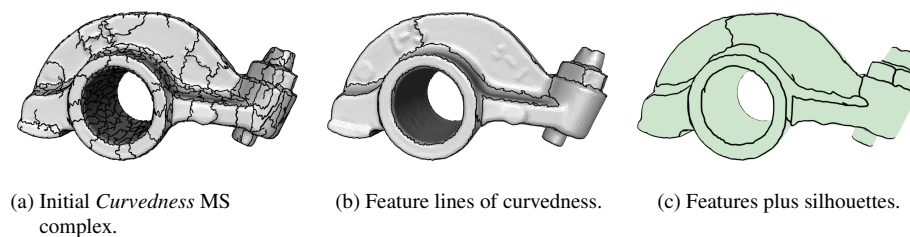


Figure 9.4: The extraction of maximal features of the *Curvedness* indicator yields both convex and concave feature lines in one. It depends on just one parameter, compared to two for the linked treatment of κ_1 and κ_2 and is therefore our preferred technique.

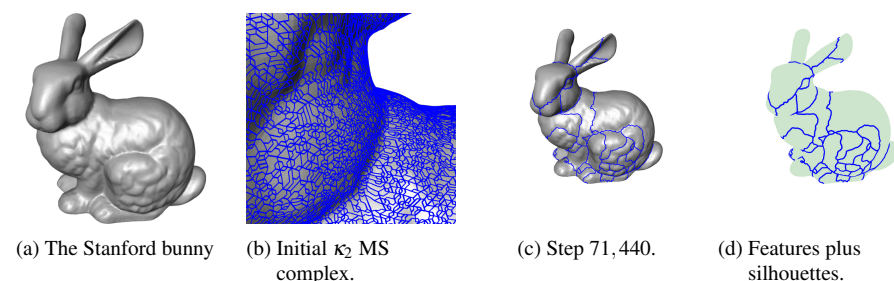


Figure 9.5: Concave feature lines are extracted on the Stanford bunny with 280K triangles. The initial MS complex of κ_2 consists of 71,577 cancellations, and its descending 1-cells are densely spread over the bunny. The kept concave features identify it.

Fig. 9.1 shows the rockerarm surface with 20K triangles. The initial ascending 1-cells are shown together with the result of our method after 3,550 (of total 3,875) cancellations. The extraction of the concave features is displayed analogously in Figure 9.3. Displaying both sets of feature lines in Figure 9.3d represents the geometry of the rockerarm well.

Treating concave and convex features separately results in two parameters. *Curvedness* allows us to reduce this to one parameter, as the maximal structures of curvedness contain both convex and concave features lines, as displayed in Fig. 9.4. The ascending 1-cells of the MS complex corresponding to the optimal cancellation value 3,388 (out of total 3,777) in Figures 9.4b-c represent the geometry of the surface, dependent on just one parameter. Due to this, curvedness is particularly suitable for our approach and we favor it over the linked usage of κ_1 and κ_2 , unless either convex or concave features are of particular interest for a specific surface, as for the Stanford bunny with 280K triangles in Fig. 9.5, where the concave features alone reflect the geometry well. In this case, the initial descending 1-cells of κ_2 are densely spread over the whole surface. The most significant concave feature lines are displayed in Figure 9.5d. In Fig. 9.6 the feature lines on the cow's head consisting of 93K triangles are displayed based on curvedness. Three different feature scales corresponding to increasing cancellation values are being displayed, ranging from fine scale in Figures 9.6b,f) to low scale in Figures 9.6d,h). It is interesting to see that the head can be recognized at quite coarse scale, and how the addition of fine scale features adds expression to the cow's face.

Fig. 9.6h shows another property of our approach: The terminal 1-cell building the

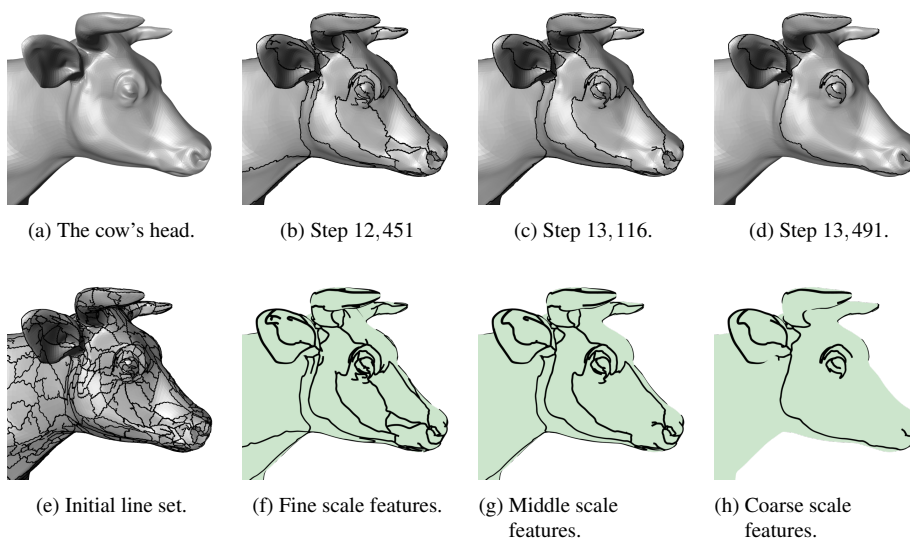


Figure 9.6: Feature lines on the cow at fine, medium and coarse scale. The coarse scale suffices to reflect the geometry, and fine scale features add more and more expression to the cow's face.

| surface | indicator | #cancels | $t_{construct}$ | t_{cancel} |
|------------|------------|----------|-----------------|--------------|
| rocker arm | κ_1 | 3,678 | 0.21 | 0.50 |
| | κ_2 | 4,476 | 0.23 | 0.55 |
| | C | 3,777 | 0.20 | 0.44 |
| bunny | κ_2 | 71,577 | 3.8 | 30 |
| cow | C | 13,676 | 1.2 | 7.6 |
| vase-lion | C | 17,613 | 2.6 | 10.4 |

Table 9.1: Timings of our method for various datasets and scalar feature indicators. In rows 3–5, the number of cancellations, the time in seconds to construct the initial MS complex, and for performing all cancellations are shown.

cow's brow is no watershed (as it does not separate two regions) and could only be extracted, as the whole MS complex is used for feature extraction.

The computational complexity of our approach is $n \log n$ for the extraction of the initial MS complex (see [CCL03]), where n is the number of vertices of the mesh. The current implementation requires the 1-cells to be updated directly on the graphs, which is of order n for a single cancellation in the worst case. Table 9.1 gives timings of our method measured on a workstation with an Intel Xeon CPU with 3.00GHz and 16 GB Ram without parallelization. We have not put much emphasis on efficiency, and it can certainly be improved.

9.2.1 Insensitivity to Noise

Our method is insensitive to noise in the surface mesh. To show this, we add Gaussian noise to the vertex positions of the rockerarm mesh in direction to the mesh normal. We used $\sigma = \frac{1}{5}$, $\sigma = \frac{1}{2}$, $\sigma = 1$ of the mean edge length. Those values correspond to the significant deformations seen in Fig. 9.7, but the extracted *Curvedness* feature lines are

still visually close to the *Curvedness* feature lines extracted from the undistorted mesh in Figure 9.4.

9.2.2 User Guided Parameter Traversal

When using *Curvedness* as feature indicator, our method depends only on one parameter. This parameter is easy to control, as it allows for binary search. As the initial feature set usually contains far too many lines, we proceed by performing two thirds of the remaining cancellations successively, until visually too few feature lines are contained. In most cases, the previous line set is then very close to the optimum. At most $\log n$ values of the cancellation parameter have to be tried, where n is the number of cancellations. An example can be seen in Fig. 9.8 for the lion vase dataset consisting of about 78K triangles allowing for 17,613 cancellations.

9.2.3 Limitations

The proposed method has two limitations. Firstly, it results in long connected lines. Whereas this is advantageous in some settings, it is too rigid in others, as it is the case on the rocker arm surface, where our technique fails to resolve the “1-3”-Feature on the side of the rocker arm both in the κ_1 case (Fig. 9.1), and in the C case (Figure 9.4). Secondly, as mentioned in the last paragraph of Sect. 9.1.7, our feature preserving cancellation criterion t from (9.9) is not monotone, i.e., it can be smaller at later cancellation stages. This makes it impossible to specify a cancellation threshold rather than a number of cancellations.

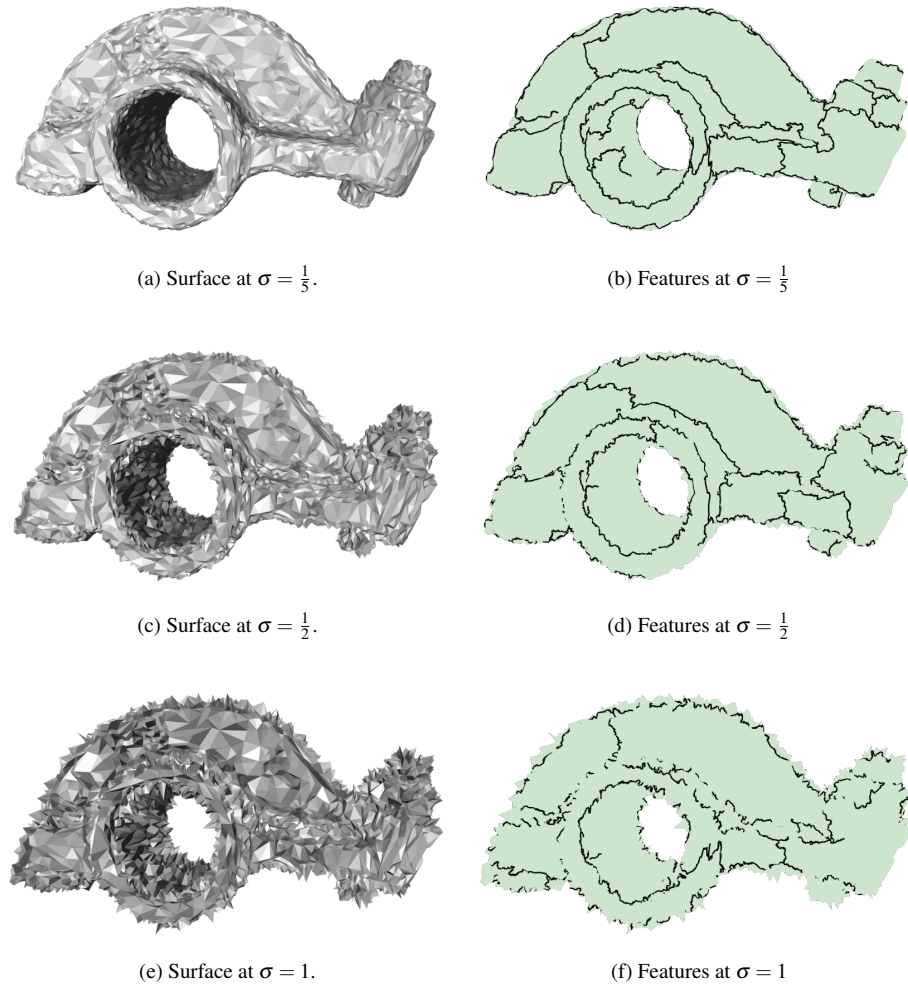


Figure 9.7: Adding different Gaussian noise scales to the rockerarm shows that our method is very stable. The feature lines just *appear* disconnected due to occlusion by surface spikes.

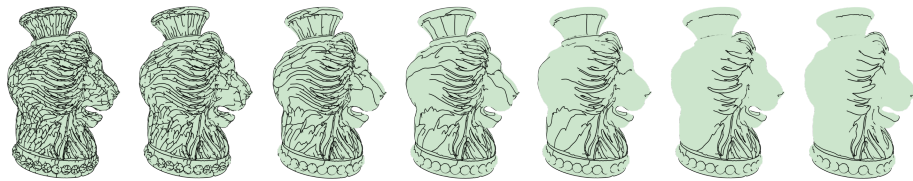


Figure 9.8: Feature lines on the lion vase. The one parameter our method depends on is traversed logarithmically. A user has to decide for *less lines* until the coarse level of detail he is interested in is reached. One further step removes a detail regarded as interesting. This way a good overview of the various level of details contained in the dataset is acquired in a few steps.

Part IV

Conclusions and Future Work

Chapter 10

Conclusions

In this thesis we developed algorithms for the extraction of vortices from 3D flow fields, both by the examination of swirling motion, and by the extraction of extremal structures of derived scalar feature indicators. A large emphasis has been put both on the invariances the extraction methods should obey, and Galilean invariance could be achieved in all but one cases.

As the first of two swirling motion based approaches in this thesis, we presented feature flow fields which are equivalent to the PV operator. Based on the FFF's, we achieved a complete classification of stable local bifurcations of tracked PV lines in saddle bifurcations, closed collapse bifurcations, inflow and outflow boundary bifurcations. A new algorithm to extract and track PV lines as a repeated stream line/surface integration of the FFF's. This way, the algorithm is independent of a particular underlying grid of the data. In fact, the accuracy of our method does not depend on the grid resolution but exclusively on the chosen technique and step size for the stream surface integration. By applying this approach to the method of Sujudi/Haimes, a novel approach for vortex core line extraction and tracking was given that, by the bifurcation analysis, gave insight into the creation, merging and collapse of vortex core lines.

While this method is not Galilean invariant, we developed a mathematical characterization of swirling particle cores that *is* Galilean invariant. By this, we addressed the identification of cores of swirling motion of path lines in unsteady flows for the first time. We introduced the Coplanar Vectors operator for deducing the characterization for 3D unsteady flows. We showed how to re-formulate and extract swirling particle cores in 3D unsteady fields using the Parallel Vectors operator – a common tool for feature extraction in the visualization community. This eases the implementation of our approach in other visualization systems. We presented a unified notation of swirling motion in 2D and 3D flows.

This method is a generalization of the approach by Sujudi/Haimes and clearly inherits its limitations. In particular, the method can result in false positives, i.e., lines that actually lie off the desired core line. Also when noise is present the method may extract a variety of short lines. Both issues have been treated in the literature [PR99] by filtering the output by length and certain angle criteria as discussed in section 5.3.2. However, cores of swirling motion have an intuitive interpretation, accompany the behavior of integral curves and their extraction is comparatively simple and fast. Also, Roth and Peikert pointed out in [RP98] that the method of Sujudi/Haimes has its limitations in settings where curved boundaries are involved. Our method might have comparable limitations and it is an interesting point for further studies to see if the

higher-order methods in [RP98] can be extended to path lines in an analogous way.

As another way of defining Galilean invariant vortex core lines, we introduced extracting ridge or valley lines of vortex region quantities like the Okubo-Weiss criterion and λ_2 and proposed an iconic representation of such lines.

A drawback of this method is that it requires second order derivatives. Due to the ill-conditioning of derivative computation this makes the method to be rather susceptible to noise. In particular, an appropriate approximation scheme has had to be chosen to make the method feasible. Still, step sizes both in the zero search and in the integration had to be chosen quite small for acceptable precision.

As a consequence, we developed an extremum line extraction method based on scalar field topology using just first derivatives. In this course, we discussed and clarified the duality of vortex and strain measurement and introduced the notion of vortex and strain skeletons. Due to the separation properties of the chosen extraction, we were able to separate vortices as well as strain regions and quantify their extent. We extracted Galilean invariant strain features for the first time, resulting in 0D, 1D and 2D features as well as Galilean invariant vortex structures.

This progress made the application to a number of complex data sets feasible. In particular, we could examine a high-dimensional flow data set around an airfoil which totals in over 200GB of data. This 3D time-dependent flow depends on two additional parameters introduced by the periodic blowing and suction: frequency and intensity of air injection. By the vortex and strain analysis we were able to identify the influence of parameter variations on the flow field and elucidated the underlying physics.

By utilizing the discrete topology by Forman [For98], we could proceed even further and reduced the number of derivatives for extremum line extraction to zero, restricting the research to feature lines on surface meshes. We showed how the Morse-Smale complex of scalar feature indicators and an appropriately adapted cancellation criterion can be used to extract meaningful feature lines on surfaces in a stable way, dependent on just one parameter. In this setting, we identified *Curvedness* as the most suitable feature indicator for the extraction of surface features.

We believe that our developments will lead to a parameter and derivative free, fast and robust vortex extraction technique based on extremal structures of derived feature indicators in the near future.

10.1 Future Work

Of course, for the vortex structures based on extremum lines of scalar feature indicators, some open research topics remain. In particular, it is very interesting to compare the feature lines of our topology-based techniques to ridge lines, both on a theoretical and on a practical level. Furthermore, extending those techniques to 3D scalar fields is a thriving challenge. Such a tool could be used for the extraction of extremal structures in scalar feature indicators from 3D flow fields, and hence for the extraction of vortex features in 3D flow fields that depends only on one parameter. Furthermore, after having reduced the number of necessary parameters for feature line extraction to one, parameter free feature line extraction seems within reach, e.g. by examining cancellation statistics to provide the user with an initial guess for an optimal cancellation parameter.

While we showed how the vortex core lines based on swirling motion can be tracked in time, no such tracking approach is known for the topology-based extremum line approaches presented in part III of this thesis. As the separatrices of scalar field

topology are global features, tracking them with a feature flow field is not possible. A possible solution would be to track the critical points and deduce a correspondence for the separatrices (and hence for the vortex core lines) from the critical point correspondence. A very challenging alternative would be to compute the Morse-Smale Complex and its cancellation hierarchy of the time dependent scalar feature indicator field, which is a 4D function. While the 3D case can be dealt with efficiently by the method of Gyulassy et al. [GNPH07], the general nD case is NP hard as shown by Joswig et al. [JP06]. They also provide an integer programming formulation for this problem, but at the current stage, only data sets of small size can be processed by this method.

Acknowledgements

Special thanks go to Tino Weinkauff who taught me his very concentrated way of doing research. I hope I could learn from him. I thank Hans-Christian Hege for the great opportunity of writing my Phd thesis at the department of Visualization and Data Analysis at Zuse institute Berlin. I thank Holger Theisel for supervising my thesis, his constant support and many good ideas. I thank Steffen Prohaska for introducing me into the world of good programming (which I still try to join). I thank Nathalie Teuber for all discussions and implementations, and Jan Reininghaus and David Günther for long discussions about Morse Theory and even longer late night card gaming. I thank Bernd R. Noack for fruitful discussions and providing the cylinder data set, Mo Samimy and Edgar Caraballo for providing the cavity data set, Bert Günther for the airfoil and Pierre Comte for the mixing layer data sets. I thank Bill Kuo, Wei Wang, Cindy Bruyere, Tim Scheitlin, and Don Middleton of the U.S. National Center for Atmospheric Research (NCAR) and the U.S. National Science Foundation (NSF) for providing the Weather Research and Forecasting (WRF) Model simulation data of Hurricane Isabel. We thank George Haller and Jon Scouten for their thoughtful comments.

All visualizations in this thesis have been created using AMIRA – a system for advanced visual data analysis [SWH05b] (see <http://amira.zib.de/>).

Bibliography

- [Ban70] T. Banchoff. Critical points and curvature for embedded polyhedral surfaces. *Am. Math. Monthly*, pages 475–485, 1970.
- [Bat67] G. K. Batchelor. *An Introduction to Fluid Mechanics*. Cambridge University Press, 1967.
- [BEHP04] P. Bremer, H. Edelsbrunner, B. Hamann, and V. Pascucci. A topological hierarchy for functions on triangulated surfaces. *IEEE TVCG*, 10(4):385–396, 2004.
- [BP02] D. Bauer and R. Peikert. Vortex tracking in scale space. In *Data Visualization 2002. Proc. VisSym 02*, pages 233–240, 2002.
- [BS94] D. Banks and B. Singer. Vortex tubes in turbulent flows: Identification, representation, reconstruction. In *Proc. IEEE Visualization 1994*, pages 132–139, 1994.
- [BS95] D. Banks and B. Singer. A predictor-corrector technique for visualizing unsteady flow. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):151–163, 1995.
- [BSH96] H. Battke, D. Stalling, and H.-C. Hege. *Visualization and Mathematics 1997*. Springer, Heidelberg, 1996.
- [Bul99] T. Bulow. Hypercomplex spectral signal representations for image processing and analysis, 1999.
- [CCL03] F. Cazals, F. Chazal, and T. Lewiner. Molecular shape analysis based upon the morse-smale complex and the connolly function. In *SCG: Proc. of the 19. Ann. Symp. on Comp. Geom.*, pages 351–3605, 2003.
- [CPC90] M. S. Chong, A. E. Perry, and B. J. Cantwell. A general classification of three-dimensional flow fields. *Physics of Fluids A*, 2(5):765–777, 1990.
- [CSB98] P. Comte, J. Silvestrini, and P. Bégou. Streamwise vortices in Large-Eddy Simulations of mixing layer. *Eur. J. Mech. B*, 17:615–637, 1998.
- [CSJ03] E. Caraballo, M. Samimy, and D. J. Low dimensional modeling of flow for closed-loop flow control, 2003. AIAA Paper 2003-0059.
- [DBG⁺06] S. Dong, P.-T. Bremer, M. Garland, V. Pascucci, and J. C. Hart. Spectral surface quadrangulation. In *SIGGRAPH*, pages 1057–1066. ACM Press, 2006.

- [dLvL99] W. de Leeuw and R. van Liere. Collapsing flow topology using area metrics. In *Proc. IEEE Visualization '99*, pages 149–354, 1999.
- [Dol07] H. Doleisch. Simvis: interactive visual analysis of large and time-dependent 3d simulation data. In *WSC '07: Proceedings of the 39th conference on Winter simulation*, pages 712–720, Piscataway, NJ, USA, 2007. IEEE Press.
- [Ebe96] D. Eberly. *Ridges in Image and Data Analysis*. Kluwer Academic Publishers, Dordrecht, 1996.
- [EGM⁺94] D. Eberly, R. Gardner, B. Morse, S. Pizer, and C. Scharlach. Ridges for image analysis. *Journal of Mathematical Imaging and Vision*, 4(4):353–373, 1994.
- [EHZ03] H. Edelsbrunner, J. Harer, and A. Zomorodian. Hierarchical morse-smale complexes for piecewise linear 2-manifolds. In *Disc. Comput. Geom.*, volume 30, pages 87–107, 2003.
- [ELZ00] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. In *FOCS '00: Proc. of the 41st Ann. Symp. on Foundations of Computer Science*, page 454, Washington, DC, USA, 2000. IEEE Computer Society.
- [ES03] J. Ebling and G. Scheuermann. Clifford convolution and pattern matching on vector fields. In *Proc. IEEE Visualization 2003*, pages 193–200, 2003.
- [ES05] J. Ebling and G. Scheuermann. Clifford fourier transform on vector fields. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):469–479, 2005.
- [ES06] J. Ebling and G. Scheuermann. Segmentation of flow fields using pattern matching. In *EuroVis*, pages 147–154, 2006.
- [FG82] P. Firby and C. Gardiner. *Surface Topology*, chapter 7, pages 115–135. Ellis Horwood Ltd., 1982. Vector Fields on Surfaces.
- [For98] R. Forman. Morse theory for cell-complexes. *Advances in Mathematics*, 134(1):90–145, 1998.
- [For02] R. Forman. A user's guide to discrete morse theory. *Sémin. Lothringien de Combinatoire*, B48c, 2002.
- [GeH01] M. Gad-el Hak. Flow control: The future. *Journal of Aircraft*, 38(3), 2001.
- [Gel01] A. V. Gelder. Stream surface generation for fluid flow solutions on curvilinear grids. In *Data Visualization 2001. Proc. VisSym 01*, 2001.
- [GGTH07] C. Garth, F. Gerhardt, X. Tricoche, and H. Hagen. Efficient computation and visualization of coherent structures in fluid flow applications. *IEEE Transactions on Visualization and Computer Graphics (Proceedings Visualization 2007)*, 13(6), November - December 2007.

- [GLL91] A. Globus, C. Levit, and T. Lasinski. A tool for visualizing the topology of three-dimensional vector fields. In *Proc. IEEE Visualization '91*, pages 33–40, 1991.
- [GMDP⁺07] A. Gyulassy, V. N. M. Duchaineau, V. Pascucci, P. Bringa, A. Higginbotham, and B. Hamann. Topologically clean distance fields. In *Proc. IEEE Visualization 2007*, pages 1432–1439, 2007.
- [GNP⁺05] A. Gyulassy, V. Natarajan, V. Pascucci, P.-T. Bremer, and B. Hamann. Topology-based simplification for feature extraction from 3d scalar fields. In *Proc. IEEE Visualization 2005*, pages 535–542, 2005.
- [GNPH07] A. Gyulassy, V. Natarajan, V. Pascucci, and B. Hamann. Efficient computation of morse-smale complexes for three-dimensional scalar functions. *IEEE Transactions on Visualization and Computer Graphics (Proceedings Visualization 2007)*, 13(6), November - December 2007.
- [GTS⁺04a] C. Garth, X. Tricoche, T. Salzbrunn, T. Bobach, and G. Scheuermann. Surface techniques for vortex visualization. In *Proc. IEEE TCVG Symposium on Visualization 2004*, pages 155–164, 2004.
- [GTS04b] C. Garth, X. Tricoche, and G. Scheuermann. Tracking of vector field singularities in unstructured 3D time-dependent datasets. In *Proc. IEEE Visualization 2004*, pages 329–336, 2004.
- [Hal80] J. Hale. *Ordinary Differential Equations*. Kreiger, New York, 1980.
- [Hal05] G. Haller. An objective definition of a vortex. *J. Fluid Mech.*, 525:1–26, 2005.
- [HEWK03] E. Heiberg, T. Ebbers, L. Wigström, and M. Karlsson. Three dimensional flow characterization using vector pattern matching. *IEEE Transactions on Visualization and Computer Graphics*, 9(3):313–319, 2003.
- [HH89] J. Helman and L. Hesselink. Representation and display of vector field topology in fluid flow data sets. *IEEE Computer*, 22(8):27–36, August 1989.
- [HH91] J. Helman and L. Hesselink. Visualizing vector field topology in fluid flows. *IEEE Computer Graphics and Applications*, 11:36–46, May 1991.
- [HJ97] A. Hassan and R. Janakiram. Effects of zero-mass synthetic jets on the aerodynamics of the NACA-0012 airfoil. *AIAA Paper 97-2326*, 1997.
- [HP04] K. Hildebrandt and K. Polthier. Anisotropic filtering of non-linear surface features. *Comput. Graph. Forum*, 23(3):391–400, 2004.
- [HPW05] K. Hildebrandt, K. Polthier, and M. Wardetzky. Smooth feature lines on surface meshes. In *3rd Eurographics Symp. on Geom. Proc.*, pages 85–90, 2005.
- [HPW06] K. Hildebrandt, K. Polthier, and M. Wardetzky. On the convergence of metric and geometric properties of polyhedral surfaces. *Geometria Dedicata*, 123:89–112, 2006.

- [HSSZ97] H.-C. Hege, M. Seebass, D. Stalling, and M. Zöckler. A generalized marching cubes algorithm based on non-binary classifications. *Preprint SC-97-05*, 1997. ZIB, Berlin.
- [Hul92] J. Hultquist. Constructing stream surfaces in steady 3D vector fields. In *Proc. IEEE Visualization '92*, pages 171–177, 1992.
- [Hun87] J. Hunt. Vorticity and vortex dynamics in complex turbulent flows. *Proc CANCAM, Trans. Can. Soc. Mec. Engrs*, 11:21, 1987.
- [JBTS08] H. Jänicke, M. Böttinger, X. Tricoche, and G. Scheuermann. Automatic detection and visualization of distinctive structures in 3d unsteady multi-fields. In *Proc. Eurographics / IEEE VGTC Symposium on Visualization (EuroVis '08)*, pages 767–772, Eindhoven, Netherlands, May 26 - 28 2008.
- [JH95] J. Jeong and F. Hussain. On the identification of a vortex. *J. Fluid Mechanics*, 285:69–94, 1995.
- [JMT02] M. Jiang, R. Machiraju, and D. Thompson. Geometric verification of swirling features in flow fields. In *Proc. IEEE Visualization 2002*, pages 307–314, 2002.
- [JP06] M. Joswig and M. E. Pfetsch. Computing optimal morse matchings. *SIAM J. Discret. Math.*, 20(1):11–25, 2006.
- [JWSK07] H. Jänicke, A. Wiebel, G. Scheuermann, and W. Kollmann. Multifield visualization using local statistical complexity. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1384–1391, 2007.
- [KE07] T. Klein and T. Ertl. Scale-space tracking of critical points in 3d vector fields. In H. Hauser, H. Hagen, and H. Theisel, editors, *Topology-based Methods in Visualization*, Mathematics and Visualization, pages 35–50. Springer, 2007. Topo-In-Vis 2005, Budmerice, Slovakia, September 29 - 30.
- [KJ00] H.-J. Kaltenbach and G. Janke. Direct numerical simulation of flow separation behind a swept, rearward-facing step at $re_H=3000$. *Physics of Fluids*, 12:2320–2337, 2000.
- [KKE05] K. Kaepernick, L. Koop, and K. Ehrenfried. Investigation of the unsteady flow field inside a leading edge slat cove. In *11th AIAA/CEAS Aeroacoustics Conference (26th Aeroacoustics Conference)*, Monterey, CA, USA, 2005.
- [KKH01] J. Kniss, G. Kindlmann, and C. Hansen. Interactive volume rendering using multi-dimensional transfer functions and direct manipulation widgets. In *VIS '01: Proceedings of the conference on Visualization '01*, pages 255–262, Washington, DC, USA, 2001. IEEE Computer Society.
- [KOD⁺05] B. Krauskopf, H. M. Osinga, E. J. Doedel, M. E. Henderson, J. M. Guckenheimer, A. Vladimirovsky, M. Dellnitz, and O. Junge. A survey of methods for computing (un)stable manifolds of vector fields. *Int. J. Bifurcation & Chaos*, 15(3):763–791, 2005.

- [Koo05] L. Koop. *Aktive und Passive Strömungsbeeinflussung zur Reduzierung der Schallabstrahlung an Hinterkantenklappen von Tragflügeln*. PhD thesis, Technische Universität Berlin, 2005.
- [KvD92] J. J. Koenderink and A. J. van Doorn. Surface shape and curvature scales. *Image and Vision Computing*, 10(8):557–564, October 1992.
- [KvD93] J. J. Koenderink and A. J. van Doorn. Local features of smooth shapes: ridges and courses. In B. C. Vemuri, editor, *Proc. SPIE Int. Soc. Opt. Eng.*, volume 2031, pages 2–13, June 1993.
- [LBM⁺06] D. Laney, P.-T. Bremer, A. Mascarenhas, P. Miller, and V. Pascucci. Understanding the structure of the turbulent mixing layer in hydrodynamic instabilities. *IEEE Transactions on Visualization and Computer Graphics (Proceedings Visualization 2006)*, 12(5):1053–1060, September - October 2006.
- [Lew05] T. Lewiner. *Mesh Compression from Geometry*. Thèse de doctorat, Université Pierre et Marie Curie, France, 2005.
- [LLT03a] T. Lewiner, H. Lopes, and G. Tavares. Optimal discrete morse functions for 2-manifolds. *Comput. Geom. Theory Appl.*, 26(3):221–233, 2003.
- [LLT03b] T. Lewiner, H. Lopes, and G. Tavares. Visualizing Forman’s discrete vector field. In H.-C. Hege and K. Polthier, editors, *Visualization and Mathematics III*, pages 95–112. "Springer-Verlag, Heidelberg, 2003.
- [Lor65] E. N. Lorenz. A study of the predictability of a 28-variable atmospheric model. *Tellus*, 17:321–333, 1965.
- [MCML89] D. Maddalon, F. Collier, L. Montoya, and C. Land. Transition flight experiments on a swept wing with suction. *AIAA Paper 89-1893*, 1989.
- [Mil63] J. W. Milnor. *Morse Theory*, volume 51 of *Annals of Mathematics Studies*. Princeton University Press, Princeton, NJ, USA, 1963.
- [MK97] H. Miura and S. Kida. Identification of tubular vortices in turbulence. *J. Physical Society of Japan*, 66(5):1331–1334, 1997.
- [MR96] T. Minagawa and R. Rado. The 3D marching lines algorithm. *Graphical Models and Image Processing*, 58(6):503–509, 1996.
- [MW99] A. P. Mangan and R. T. Whitaker. Partitioning 3d surface meshes using watershed segmentation. *IEEE TVCG*, 5(4):308–321, 1999.
- [NAS90] NASA Langley Research Center. Wake vortex study at wallops island, May 1990. <http://nix.larc.nasa.gov/info?id=EL-1996-00130&orgid=1>.
- [NE94] B. Noack and H. Eckelmann. A low-dimensional Galerkin method for the three-dimensional flow around a circular cylinder. *Phys. Fluids*, 6:124–143, 1994.
- [NWB⁺06] V. Natarajan, Y. Wang, P. T. Bremer, V. Pascucci, and B. Hamann. Segmenting molecular surfaces. *Computer Aided Geometric Design (special issue on Applications of Geometric Modeling in the Life Sciences)*, 23(6):495–509, 2006.

- [OBS04] Y. Ohtake, A. Belyaev, and H.-P. Seidel. Ridge-valley lines on meshes via implicit surface fitting. In *SIGGRAPH*, pages 609–612, 2004.
- [Pas07] P.-T. B. V. Pascucci. A practical approach to two-dimensional scalar topology. In H. Hauser, H. Hagen, and H. Theisel, editors, *Topology-based Methods in Visualization*, Mathematics and Visualization. Springer, 2007.
- [Pei08] R. Peikert, 2008. private communication.
- [PFTV91] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, Cambridge, 1991.
- [PM82] J. Palis and W. D. Melo. *Geometric Theory of Dynamical Systems: An Introduction*. Springer-Verlag, 1982.
- [Pow64] M. J. D. Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The Computer Journal*, 7(2):155–162, 1964.
- [PR99] R. Peikert and M. Roth. The parallel vectors operator - a vector field visualization primitive. In *Proc. IEEE Visualization 99*, pages 263–270, 1999.
- [PVH⁺02] F. Post, B. Vrolijk, H. Hauser, R. Laramee, and H. Doleisch. Feature extraction and visualisation of flow fields. In *Proc. Eurographics 2002, State of the Art Reports*, pages 69–100, 2002.
- [PVH⁺03] F. H. Post, B. Vrolijk, H. Hauser, R. S. Laramee, and H. Doleisch. The State of the Art in Flow Visualisation: Feature Extraction and Tracking. *Computer Graphics Forum*, 22(4):775–792, 2003.
- [RP96] M. Roth and R. Peikert. Flow visualization for turbomachinery design. In *Proc. Visualization 96*, pages 381–384, 1996.
- [RP98] M. Roth and R. Peikert. A higher-order method for finding vortex core lines. In D. Ebert, H. Hagen, and H. Rushmeier, editors, *Proc. IEEE Visualization '98*, pages 143–150, Los Alamitos, 1998. IEEE Computer Society Press.
- [RZNS04] C. Roessl, F. Zeilfelder, G. Nuernberger, and H.-P. Seidel. Reconstruction of volume data with quadratic super splines. *IEEE Trans. Visualization and Computer Graphics*, 10:397–409, 2004.
- [SBM⁺01] G. Scheuermann, T. Bobach, H. H. K. Mahrous, B. Hamann, K. Joy, and W. Kollmann. A tetrahedra-based stream surface algorithm. In *Proc. IEEE Visualization 01*, pages 151 – 158, 2001.
- [SF04] G. Stylianou and G. Farin. Crest lines for surface segmentation and flattening. *IEEE Transactions on Visualization and Computer Graphics*, 10(5):536–544, 2004.
- [SGT04] M. Schatz, B. Günther, and F. Thiele. Computational modelling of the unsteady wake behind gurney-flaps. *AIAA Paper 2004-2417*, 2004.

- [SH95] D. Sujudi and R. Haimes. Identification of swirling flow in 3D vector fields. Technical report, Department of Aeronautics and Astronautics, MIT, 1995. AIAA Paper 95-1715.
- [SHJK00] G. Scheuermann, B. Hamann, K. Joy, and W. Kollman. Visualizing local vector field topology. *SPIE Journal of Electronic Imaging*, 9(4):356–367, 2000.
- [SHK⁺97] G. Scheuermann, H. Hagen, H. Krüger, M. Menzel, and A. Rockwood. Visualization of higher order singularities in vector fields. In R. Yagel and H. Hagen, editors, *Proc. IEEE Visualization '97*, pages 67–74, 1997.
- [SHR⁺06] C. R. Shalizi, R. Haslinger, J.-B. Rouquier, K. L. Klinkner, and C. Moore. Automatic filters for the detection of coherent structure in spatiotemporal systems. *Physical Review E*, 73:036104, 2006.
- [SKMR98] G. Scheuermann, H. Krüger, M. Menzel, and A. Rockwood. Visualizing non-linear vector field topology. *IEEE Transactions on Visualization and Computer Graphics*, 4(2):109–116, 1998.
- [Soi99] P. Soille. *Morphological Image Analysis*. Springer, Berlin, Heidelberg, New York, 1999.
- [SP03] M. Sato and R. Peikert. Core-line-based vortex hulls in turbomachinery flows. *J. Visualization Society of Japan*, 23(2):151–154, 2003.
- [SP07] F. Sadlo and R. Peikert. Efficient visualization of lagrangian coherent structures by filtered amr ridge extraction. *IEEE Transactions on Visualization and Computer Graphics (Proceedings Visualization 2007)*, 13(6), November - December 2007.
- [SPP04] F. Sadlo, R. Peikert, and E. Parkinson. Vorticity based flow analysis and visualization for pelton turbine design optimization. In *VIS '04: Proceedings of the conference on Visualization '04*, pages 179–186, Washington, DC, USA, 2004. IEEE Computer Society.
- [SPS06] F. Sadlo, R. Peikert, and M. Sick. Visualization tools for vorticity transport analysis in incompressible flow. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):949–956, 2006.
- [SS07] T. Salzbrunn and G. Scheuermann. Streamline predicates as flow topology generalization. In H. Hauser, H. Hagen, and H. Theisel, editors, *Topology-based Methods in Visualization*, Mathematics and Visualization, pages 65–77. Springer, 2007. Topo-In-Vis 2005, Budmerice, Slovakia, September 29 - 30.
- [Str95] R. S. Strichartz. *The Way of Analysis*. Jones & Bartlett Publishers, corr. 3rd printing edition, 1995.
- [SWH05a] J. Sahner, T. Weinkauff, and H.-C. Hege. Galilean invariant extraction and iconic representation of vortex core lines. In *Proc. EuroVis 2005*, pages 151–160, 2005.

- [SWH05b] D. Stalling, M. Westerhoff, and H.-C. Hege. Amira: A highly interactive system for visual data analysis. *The Visualization Handbook*, pages 749–767, 2005.
- [SWLP08] J. Sahner, B. Weber, H. Lamecker, and S. Prohaska. Extraction of feature lines on surface meshes based on discrete morse theory. In *Proc. Eurographics / IEEE VGTC Symposium on Visualization (EuroVis '08)*, pages 735–742, Eindhoven, Netherlands, May 26 - 28 2008.
- [SWTH07] J. Sahner, T. Weinkauff, N. Teuber, and H.-C. Hege. Vortex and strain skeletons in eulerian and lagrangian frames. *IEEE Transactions on Visualization and Computer Graphics*, 13(5):980–990, September - October 2007.
- [TGK⁺04] X. Tricoche, C. Garth, G. Kindlmann, E. Deines, G. Scheuermann, M. Ruetten, and C. Hansen. Visualization of intricate flow structures for vortex breakdown analysis. In *Proc. IEEE Visualization 2004*, pages 187–194, 2004.
- [The02] H. Theisel. Designing 2D vector fields of arbitrary topology. *Computer Graphics Forum (Eurographics 2002)*, 21(3):595–604, 2002.
- [TM98] C. Tang and G. Medoni. Extremal feature extraction from 3-D vector and noisy scalar fields. In *Proc. IEEE Visualization 1998*, pages 95–102, 1998.
- [TS03] H. Theisel and H.-P. Seidel. Feature flow fields. In *Data Visualization 2003. Proc. VisSym 03*, pages 141–148, 2003.
- [TSH00] X. Tricoche, G. Scheuermann, and H. Hagen. A topology simplification method for 2D vector fields. In *Proc. IEEE Visualization 2000*, pages 359–366, 2000.
- [TSH01] X. Tricoche, G. Scheuermann, and H. Hagen. Topology-based visualization of time-dependent 2D vector fields. In *Data Visualization 2001. Proc. VisSym 01*, pages 117–126, 2001.
- [TSW⁺05] H. Theisel, J. Sahner, T. Weinkauff, H.-C. Hege, and H.-P. Seidel. Extraction of parallel vector surfaces in 3d time-dependent fields and application to vortex core line tracking. In *Proc. IEEE Visualization 2005*, pages 631–638, 2005.
- [TWHS03] H. Theisel, T. Weinkauff, H.-C. Hege, and H.-P. Seidel. Saddle connectors - an approach to visualizing the topological skeleton of complex 3D vector fields. In *Proc. IEEE Visualization 2003*, pages 225–232, 2003.
- [TWHS04a] H. Theisel, T. Weinkauff, H.-C. Hege, and H.-P. Seidel. Grid-independent detection of closed stream lines in 2D vector fields. In *Proc. Vision, Modeling and Visualization 2004*, 2004.
- [TWHS04b] H. Theisel, T. Weinkauff, H.-C. Hege, and H.-P. Seidel. Stream line and path line oriented topology for 2D time-dependent vector fields. In *Proc. IEEE Visualization 2004*, pages 321–328, 2004.

- [TWHS05] H. Theisel, T. Weinkauff, H.-C. Hege, and H.-P. Seidel. Topological methods for 2D time-dependent vector fields based on stream lines and path lines. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):383–394, 2005.
- [TWSH02] X. Tricoche, T. Wischgoll, G. Scheuermann, and H. Hagen. Topology tracking for the visualization of time-dependent two-dimensional flows. *Computers & Graphics*, 26:249–257, 2002.
- [vW93] J. van Wijk. Implicit stream surfaces. In *Proc. Visualization 93*, pages 245–252, 1993.
- [WGS05] A. Wiebel, C. Garth, and G. Scheuermann. Localized flow analysis of 2d and 3d vector fields. In *Proc. EuroVis 2005*, pages 143–150, 2005.
- [WGS07] A. Wiebel, C. Garth, and G. Scheuermann. Computation of localized flow for steady and unsteady vector fields and its applications. *IEEE Transactions on Visualization and Computer Graphics*, 13(4):641–651, 2007.
- [WHN⁺03] T. Weinkauff, H.-C. Hege, B. Noack, M. Schlegel, and A. Dillmann. Coherent structures in a transitional flow around a backward-facing step. *Physics of Fluids*, 15(9):S3, September 2003. Winning Entry from the Gallery of Fluid Motion 2003.
- [WS01a] T. Wischgoll and G. Scheuermann. Detection and visualization of closed streamlines in planar flows. *IEEE Transactions on Visualization and Computer Graphics*, 7(2):165–172, 2001.
- [WS01b] T. Wischgoll and G. Scheuermann. Detection and visualization of closed streamlines in planar flows. *IEEE TVCG*, 7(2):165–172, 2001.
- [WS05] A. Wiebel and G. Scheuermann. Eyelet particle tracing - steady visualization of unsteady flow. *vis*, 00:77, 2005.
- [WSG⁺08] T. Weinkauff, J. Sahner, B. G^unther, H. Theisel, H.-C. Hege, and F. Thiele. Feature-based analysis of a multi-parameter flow simulation. In *Proc. SimVis 2008*, pages 237–251, Magdeburg, Germany, February 2008.
- [WST⁺07] T. Weinkauff, J. Sahner, H. Theisel, H.-C. Hege, and H.-P. Seidel. A unified feature extraction architecture. In R. King, editor, *Active Flow Control*, Notes on Numerical Fluid Mechanics and Multidisciplinary Design (NNFM), pages 119–133. Springer, 2007. Active Flow Control 2006, Berlin, Germany, September 27 - 29.
- [WSTH07] T. Weinkauff, J. Sahner, H. Theisel, and H.-C. Hege. Cores of swirling particle motion in unsteady flows. *IEEE Transactions on Visualization and Computer Graphics (Proceedings Visualization 2007)*, 13(6), November - December 2007.
- [WT02] T. Weinkauff and H. Theisel. Curvature measures of 3D vector fields and their applications. In *Journal of WSCG*, volume 10:2, pages 507–514, 2002.

- [WTHS04a] T. Weinkauff, H. Theisel, H.-C. Hege, and H.-P. Seidel. Boundary switch connectors for topological visualization of complex 3D vector fields. In *Data Visualization 2004. Proc. VisSym 04*, pages 183–192, 2004.
- [WTHS04b] T. Weinkauff, H. Theisel, H.-C. Hege, and H.-P. Seidel. Topological construction and visualization of higher order 3D vector fields. *Computer Graphics Forum (Eurographics 2004)*, 23(3):469–478, 2004.
- [WTHS06] T. Weinkauff, H. Theisel, H.-C. Hege, and H.-P. Seidel. Topological structures in two-parameter-dependent 2D vector fields. *Computer Graphics Forum*, 25(3):607–616, September 2006. Eurographics 2006, Vienna, Austria, September 04 - 08.
- [WTHS07] T. Weinkauff, H. Theisel, H.-C. Hege, and H.-P. Seidel. Feature flow fields in out-of-core settings. In H. Hauser, H. Hagen, and H. Theisel, editors, *Topology-based Methods in Visualization, Mathematics and Visualization*, pages 51–64. Springer, 2007. Topo-In-Vis 2005, Budmerice, Slovakia, September 29 - 30.
- [WTS⁺05] T. Weinkauff, H. Theisel, K. Shi, H.-C. Hege, and H.-P. Seidel. Extracting higher order critical points and topological simplification of 3d vector fields. In *Proc. IEEE Visualization 2005*, pages 559–566, 2005.
- [WTS⁺07] A. Wiebel, X. Tricoche, D. Schneider, H. Jaenicke, and G. Scheuermann. Generalized streak lines: Analysis and visualization of boundary induced vortices. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1735–1742, 2007.
- [Wyg04] I. Wagnarski. The variables affecting the control separation by periodic excitation. *AIAA Paper 2004-2505*, 2004.
- [YBS05] S. Yoshizawa, A. Belyaev, and H.-P. Seidel. Fast and robust detection of crest lines on meshes. In *Proc. ACM SPM*, pages 227–232, 2005.
- [YBYS07] S. Yoshizawa, A. G. Belyaev, H. Yokota, and H.-P. Seidel. Fast and faithful algorithm for detecting crest lines on meshes. In *Pacific Graph.*, 2007.
- [ZFN⁺95] H.-Q. Zhang, U. Fey, B. Noack, M. König, and H. Eckelmann. On the transition of the cylinder wake. *Phys. Fluids*, 7(4):779–795, 1995.
- [ZSH96] M. Zöckler, D. Stalling, and H. Hege. Interactive visualization of 3D-vector fields using illuminated stream lines. In *Proc. IEEE Visualization '96*, pages 107–113, 1996.

List of Figures

| | | |
|-----|--|----|
| 1.1 | Wake vortex study from NASA Langley Research Center [NAS90]. The flow around a starting agricultural plane is made visible using smoke injection. A huge pattern of swirling particle motion is created by the aircraft's wake vortex. Chapter 5 aims at extracting the cores of such areas as can be seen in 1.1b. There, the hurricane Isabel data set at $t = 33.5$ is shown with the dominating swirling particle core line (red) and a volume rendering of the cloud moisture mixing ratio. . . . | 10 |
| 1.2 | Extremal lines of scalar quantities can be regarded as center lines of iso surfaces. | 11 |
| 1.3 | Extremal lines extracted with two methods developed in this thesis. Left, the ridge lines of the Okubo-Weis criterion are shown as extracted in Chapter 6. Right, the vortex and strain skeleton of the SCCH wing is shown as extracted in Chapter 7. | 11 |
| 2.1 | Characteristic curves of a simple 2D time-dependent vector field shown as illuminated field lines. The red/green coordinate axes denote the (x,y) -domain, the blue axis shows time. | 20 |
| 2.2 | The elevation profile of a landscape as seen from above. Lines depict the partition of the landscape into hills, valleys and their intersection, the MS complex. Blue, red and green dots represent minima, maxima and saddles. Bottom row: simplifications of the MS complex by different types of cancellations and their influence on vanishing 1-cells. | 26 |
| 2.3 | Classification of critical points in scalar fields. | 27 |
| 2.4 | Separatrices originating from a repelling saddle. The line is a minimal line of steepest descent, the surface is a watershed. | 27 |
| 2.5 | Forman's discrete gradient vector field of the height function of an elevation profile is shown with red maxima, green saddles and blue minima in (a). Red lines are gradient curves ascending to maxima, blue lines are gradient curves descending to minima. In (b) its smooth analogue is shown, being discontinuous and constant on triangles. | 28 |
| 2.6 | Ridge line of a simple scalar field. | 29 |
| 3.1 | Isosurfaces of Q showing Eulerian vortex regions in a flow behind a circular cylinder. The LIC-plane is indicating corresponding vortical streamline patterns in the reference frame relative moving with the convection velocity. | 32 |
| 3.2 | Isosurfaces of M_z showing Lagrangian vortex regions. | 33 |

| | | |
|------|--|----|
| 3.3 | Flow pattern around a focus saddle. The blue plane is spanned by the eigenvectors corresponding to the complex eigenvalues. Swirling motion takes place in this plane around the real eigenvector denoted by the red arrows. | 36 |
| 4.1 | Flow behind a circular cylinder. Shown are vortex core lines in a certain frame of reference. Their evolution over time is tracked by our algorithm and depicted using transparent surfaces. Red color encodes the past while gray shows the future. | 41 |
| 4.2 | (a) two kinds of PV lines: ending in the boundary points $\mathbf{x}_1, \mathbf{x}_2$, or closed: in this case we extract two starting points $\mathbf{x}_3, \mathbf{x}_4$; (b) PV structures are surfaces in 4D. | 45 |
| 4.3 | local bifurcations of PV lines: (a) inflow boundary bifurcation; (b) outflow boundary bifurcation; (c) closed collapse bifurcation; (d) saddle bifurcation. | 47 |
| 4.4 | Saddle bifurcation. | 48 |
| 4.5 | Inflow bifurcation and closed collapse bifurcation. | 48 |
| 4.6 | Reversed outflow boundary bifurcation. | 50 |
| 4.7 | (a) PV line fold bifurcations do not exist! (b) tracking an open PV line. | 51 |
| 4.8 | seeding lines (black) for simple examples. | 52 |
| 4.9 | Saddle bifurcation of vortex core lines as defined by $\mathbf{v} \parallel \nabla \mathbf{v} \cdot \mathbf{v}$ in a quadrilinear vector field. | 54 |
| 4.10 | Stuart Vortex moving over time from left to right. | 54 |
| 4.11 | Flow behind a circular cylinder. The extracted seeding lines elucidate the alternating evolution of the vortical structures in transverse direction. | 55 |
| 5.1 | Swirling motion of stream lines in 2D steady and unsteady flows. | 58 |
| 5.2 | Swirling motion of stream lines in 3D steady and unsteady flows. | 60 |
| 5.3 | Core line (shown in red) of swirling particle motion of a simple 2D unsteady vector field \mathbf{v} (see Figure 2.1). The path lines of \mathbf{v} spiral around this core line. | 61 |
| 5.4 | As an illustration of our new technique, swirling particle motion is shown in the example of the Stuart vortex moving over time from left to right with a constant velocity. | 62 |
| 5.5 | Hurricane Isabel data set at $t = 33.5$. Shown are the dominating swirling particle core line (red) and a volume rendering of the cloud moisture mixing ratio. Orthographic view from top showing the dominant particle core (red) in the center of the clouds. | 68 |
| 5.6 | Unsteady flow over a 2D cavity. Red and green axes span the spatial domain, the blue axis denotes time. | 69 |
| 5.7 | 3D unsteady flow behind a cylinder. Existing stream line based approaches fail to capture swirling motion cores in the original frame of reference. Our new path line based method is able to extract such features without a priori knowledge. | 70 |
| 6.1 | Flow behind a circular cylinder. Vortex regions visualized as transparent isosurfaces of λ_2 . Vortex core lines displayed as cylindrical lines. | 76 |
| 6.2 | Different approaches to encoding a scalar value into the representation of a line. | 79 |

| | | |
|-----|--|----|
| 6.3 | Flow behind a circular cylinder. Iconic representation of Galilean invariant vortex core lines. λ_2 was used for extraction and is encoded into color and scale of the cylindrical meshes. Red / blue color is used to indicate strong / weak vortex activity. φ is encoded into color and spiral direction of the orbits. | 80 |
| 6.4 | Bubble chamber. Vortex core lines extracted, colored and scaled according to λ_2 . Same colormap as in figure 6.3. | 81 |
| 6.5 | Flow around a backward-facing step. Vortex core lines extracted, colored and scaled according to Q | 83 |
| 7.1 | A closeup of the cylinder dataset showing vortex regions with transparent isosurfaces of Q . The approach presented in this chapter aims at extracting 1d and 2d extrema of duality quantities like Q . The lines shown here are the maximum lines of Q where $Q > 0$ extracted by our methods. The lines are scaled by their Q -value and can be regarded as centers of isosurfaces. The correspondence of the lines in the center of the swirling motion shown in the LIC plane justifies the notion of vortex core lines. | 86 |
| 7.2 | \mathbf{M}_Z -criterion of the periodic ABC-flow. Darker colors indicate higher ellipticity times. Some path lines used for computing \mathbf{M}_Z are shown: hyperbolic points are colored blue, elliptic points are red. | 87 |
| 7.3 | Isosurfaces are not well suited for higher dimensional extremum extraction: Gray isosurface $Q = 0$ is too far away from strongest vortex activity indicated by the shown extremum lines (red). Yellow isosurface $Q = 2.7$ splits up and misses some regions at all. The maximum lines of Q (red) show location and extent of the vortices correctly. . . | 89 |
| 7.4 | Strain and vortex skeletons in a subregion of the cylinder dataset. 7.4a shows lines of maximal strain. 7.4b shows the complete strain skeleton with the extremal strain surfaces displayed only where $Q < 0$. 7.4c shows the maximal vortex lines regarded as vortex core lines. 7.4d shows the complete vortex skeleton, adding the maximal vortex surfaces that lie between the vortex core lines. Again, vortex surfaces are displayed only where $Q > 0$ | 91 |
| 7.5 | A close-up of the cylinder dataset. Three maxima of Q (red ellipsoids) are separated from each other by minimum surfaces (blue). This divides the domain into three different vortex regions. The maxima are connected by a maximum line (red), i.e., a vortex core line defined by Q | 92 |
| 7.6 | 2D-terrain exemplifying different extrema definitions. The closed green line is a watershed separatrix. Both the green and the orange line conform to the height ridge definition. | 93 |
| 7.7 | Discrete critical point extraction (exemplified using a regular grid). Red nodes are larger than the central node, blue nodes are smaller. Saddles can be extracted by counting the number of connected components of the 26 neighbors (7.7b). 7.7c shows that for separatrix integration, the continuously extracted critical points must be used, because the zeros of the gradient do not necessarily lie on grid nodes. | 94 |
| 7.8 | Strain skeleton of the cylinder data set. It partitions the domain in vortex regions. Inside the vortex regions, lines of maximal vortical behavior are shown scaled by Q . Close-ups of the structures can be seen in Figure 7.4. | 95 |

| | | |
|------|---|-----|
| 7.9 | SCCH airfoil visualized using isosurface $Q = 0$ and a LIC plane colored by Q | 95 |
| 7.10 | In the flow around an airfoil, isosurfaces of the Okubo-Weiss criterion Q are shown to the left. To the right, the line structures in the vortex and strain skeletons extracted by our method are displayed, showing lines of maximal strain (blue) and lines of maximal vortex activity (red) that are vortex core lines. Our method gives a complete overview of the location and extent of vortex and strain features in the flow, whereas the isosurfaces miss the smaller features downstream and give only a rough location for the larger features upstream. | 96 |
| 7.11 | Turbulent mixing layer. The lines of maximal strain as indicated by \mathbf{M}_Z perfectly match up with the shear layer. The vortex skeleton of the Q -criterion elucidates the spatial evolution of Kelvin-Helmholtz vortices, vortex pairing, and the spanwise formation of streamwise rib vortices. | 97 |
| 8.1 | SCCH Wing Model and periodic excitation by blowing and suction. | 100 |
| 8.2 | Computational mesh | 100 |
| 8.3 | Streamlines showing the character of the flow above the flap. | 101 |
| 8.4 | Numerical results of the excited flow. | 102 |
| 8.5 | Flow around an airfoil: topology-based vortex analysis. | 103 |
| 8.6 | Our vortex analysis of different excitation frequencies could be evaluated successfully by examination of the trailing edge departure angle of the main airfoil (first column). The angle between these vectors and the oncoming flow direction is shown at a snapshot in the second column and spanwise averaged in the third. The dashed, solid and dotted lines denote the unexcited flow, $F^+ = 0.6$, $F^+ = 2.0$, respectively. | 104 |
| 8.7 | The intensity variation was performed on 13 different intensity values ranging from $C'_\mu = 10$ to $C'_\mu = 500$ (shown in increasing order from left to right). See figure 8.8 for a close-up. | 105 |
| 8.8 | Topology-based vortex analysis of the intensity variation at four selected intensities. At intensities of $C'_\mu = 20$ and $C'_\mu = 50$ the structures are three-dimensional, whereas at lower and higher intensity, the structures have a two-dimensional characteristic. At $C'_\mu = 500$, a strong excitation-induced vortex is created at the top of the rear flap. | 105 |
| 9.1 | Ascending 1-cells of the initial MS complex of κ_1 on the rocker arm are a candidate set of convex feature lines in (a). 3,600 persistence cancellations lead to the line set depicted in (b). Clearly, all feature lines are still contained. After 3,650 cancellations, one of the main features is broken by persistence cancellation in (c). In contrast, feature line preserving cancellation as proposed here maintains the main features in (d). | 109 |
| 9.2 | Cancellation of a terminal 1-cell forking off the outer ring feature of the rocker arm colored by κ_1 . The 1-cell emanating from the left saddle ends at the maximum on the high curvature area, but only the part up to the branching point is being deleted by the cancellation. | 111 |
| 9.3 | a) The initial MS complex of κ_2 containing 4476 descending 1-cell is taken as a starting point. b) 4460 feature preserving cancellations yield the visually best concave feature lines. c,d) Feature lines plus silhouette represent the geometry. | 111 |

-
- 9.4 The extraction of maximal features of the *Curvedness* indicator yields both convex and concave feature lines in one. It depends on just one parameter, compared to two for the linked treatment of κ_1 and κ_2 and is therefore our preferred technique. 113
- 9.5 Concave feature lines are extracted on the Stanford bunny with 280K triangles. The initial MS complex of κ_2 consists of 71,577 cancellations, and its descending 1-cells are densely spread over the bunny. The kept concave features identify it. 113
- 9.6 Feature lines on the cow at fine, medium and coarse scale. The coarse scale suffices to reflect the geometry, and fine scale features add more and more expression to the cow's face. 114
- 9.7 Adding different Gaussian noise scales to the rockerarm shows that our method is very stable. The feature lines just *appear* disconnected due to occlusion by surface spikes. 116
- 9.8 Feature lines on the lion vase. The one parameter our method depends on is traversed logarithmically. A user has to decide for *less lines* until the coarse level of detail he is interested in is reached. One further step removes a detail regarded as interesting. This way a good overview of the various level of details contained in the dataset is acquired in a few steps. 116

List of Tables

| | | |
|-----|---|-----|
| 5.1 | Summary of swirling motion in 2D and 3D flows. Depending on the dimension of the autonomous system the conditions can be written using the notations of Critical Points (CP), Parallel Vectors (PV), and Coplanar Vectors (CV). | 64 |
| 6.1 | Vortex region quantities pressure p (might also be negative, as often relative pressure is considered), vorticity ω , rotation strength Δ from [SP03], Q -criterion and λ_2 criterion with the value range in which they indicate vortices. Vortex cores according to definition 3 are either ridges or valleys as shown in column 3. | 77 |
| 9.1 | Timings of our method for various datasets and scalar feature indicators. In rows 3–5, the number of cancellations, the time in seconds to construct the initial MS complex, and for performing all cancellations are shown. | 114 |

Selbstständigkeitserklärung

Ich erkläre hiermit, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.

Insbesondere habe ich nicht die Hilfe eines kommerziellen Promotionsberaters in Anspruch genommen. Dritte haben von mir weder unmittelbar noch mittelbar geldwerte Leistungen für Arbeiten erhalten, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen.

Die Arbeit wurde bisher weder im Inland noch im Ausland in gleicher oder ähnlicher Form als Dissertation eingereicht und ist als Ganzes auch noch nicht veröffentlicht.

Jan Sahner
14. Oktober 2008