A

Paper A

The State of the Art in Topology-based Visualization of Unsteady Flow

Armin Pobitzer[1], Ronald Peikert[2], Raphael Fuchs[2], Benjamin Schindler[2], Alexander Kuhn[3], Holger Theisel[3], Krešimir Matković, and Helwig Hauser[1]

[1]Department of Informatics, University of Bergen, Norway
[2]ETH Zürich, Switzerland
[3]University of Magdeburg, Germany
[4]VRVis Research Center in Vienna, Austria

Abstract

Vector fields are a common concept for the representation of many different kinds of flow phenomena in science and engineering. Methods based on vector field topology are known for their convenience for visualizing and analyzing steady flows, but a counterpart for unsteady flows is still missing. However, a lot of good and relevant work aiming at such a solution is available. We give an overview of previous research leading towards topology-based and topology-inspired visualization of unsteady flow, pointing out the different approaches and methodologies involved as well as their relation to each other, taking classical (i.e., steady) vector field topology as our starting point. Particularly, we focus on Lagrangian methods, space-time domain approaches, local methods, and stochastic and multi-field approaches. Furthermore, we illustrate our review with practical examples for the different approaches.

This article was published in Computer Graphics Forum, 30(6):1789–1811, 2011.

1 Introduction

The concept of *flow* plays a central role in many fields of science. Classical application fields are, for example, the automotive and aviation industry, where the investigation of air flow around vehicles is an important task. However, the same concepts are used in the simulation and analysis of water flow in turbines of power plants, of blood flow in vessels, the propagation of smoke in buildings, and weather simulations, to mention just a few. The visualization of data gained from the simulation/measurement of such processes is relevant for the domain users as visualization has the potential to ease the understanding of such complex flow phenomena. In this context, topological flow visualization methods have been developed, with the aim to give insight into the overall behavior of the flow. A characteristic of this class of methods is the segmentation of the flow domain into regions of substantially different flow behavior, providing a topology of the flow domain.

Topological methods for flow visualization have been researched over recent decades and a specific conference, called *Topological Methods in Visualization* (TopoInVis), has recently been established [66, 72].

The overall setting for topological methods is more general than described above. Namely, any vector field , interpreting it as the rate of change of a certain quantity, might be visualized using such methods. Then, the vector field represents the states of a dynamical system, governed by differential equations. In such a setting the evolution of certain points/configurations can be described mathematically as solutions of the differential equation

$$\dot{\mathbf{x}}(\mathbf{t}) = \mathbf{v}(\mathbf{x}(\mathbf{t}), \mathbf{t}).$$

Because of the tight relation of this model to fluid dynamics the vector field $\mathbf{v}$ is often referred to as *flow*. Notice, however, that in that case the vector field needs to fulfill additional equations (e.g., the Navier-Stokes equation) in order to represent a flow in a fluid-dynamical sense.

If the vector field $\mathbf{v}$ does not depend on the variable t the system is said to be autonomous, otherwise non-autonomous. Equivalently, the expressions *steady* and *unsteady* (or simply *time-dependent*) flow are used.

In the study of steady flow / autonomous dynamical systems certain features such as critical points, separatrices and closed orbits play an important role. In 1989, Helman and Hesselink introduced these concepts to the visualization community under the name of *vector field topology* [74]. Methods for visualizing steady flow fields, especially planar flow fields, have achieved a high level of proficiency, while the unsteady case is still challenging and by no means complete [107, 106, 162, 161, 55, 176].

Vector field topology (VFT) and feature extraction build a solid base for understanding and visualizing a given steady flow field, and there is a considerable amount of work available on possible direct extensions of VFT for unsteady flows. Although this may seem a canonical research direction, both theoretical considerations [151] and practical demonstrations [184, 227] show clear limitations of this approach to unsteady flow.

Taking one step back, the overall goal is to find methods that can give comparable answers for unsteady flow as VFT for steady flow, namely to segment the flow domain into parts with coherent properties in terms of their temporal evolution. Consequently, we consider the term *topology-based* visualization as slightly more openly defined and may read it out to *yielding analogous results as topological methods* for the purpose of this survey. Such a segmentation reduces drastically the information to be displayed in order to convey a holistic understanding of the flow on a more semantic level.

In the remainder of this introduction we give a short overview of the field and attempt to structure it. A detailed discussion with many additional references is then left to the respective sections.

Classical vector field topology (i.e., for steady flows) segments the flow domain in regions where trajectories show the same behavior when looking at the temporal (t) limits at $\pm\infty$. This fact needs special attention when taking the step from steady to unsteady flow: in a steady field a finite amount of data can be used to determine the flow behavior at an arbitrary instance of time. For unsteady fields, this is not true: the information available is usually restricted to a certain time-window. This means that, in general, no statement about the asymptotic behavior of the trajectories is possible. Visualizing time-dependent flow essentially poses different research challenges as compared to visualizing steady flow.

Despite this, the first attempts at approaching a topology-based visualization of unsteady flow interpreted the unsteady field as a stack of steady flow fields. This induced the idea that a VFT-like segmentation of unsteady flow can be achieved using the methods already known for discrete time slices and identifying corresponding structures in subsequent time steps. Methods for the topology-based visualization of unsteady flow based on trajectories in individual time steps can be classified as tracking methods (tracking in time). In Section 3 we give an overview of the research done in this direction. The trajectories in a fixed time step $t = t_0$ are solutions of the following first-order ordinary differential equation

$$\dot{\mathbf{x}}(\mathbf{s}) = \mathbf{v}(\mathbf{x}(\mathbf{s}), \mathbf{t_0}), \quad \mathbf{x}(\mathbf{t_0}) = \mathbf{x_0}. \tag{1}$$

These solutions are called *streamlines*. Notice that the integration time s is not related to the time t on which the vector field $\mathbf{v}$ depends. The t-time becomes in that case a parameter of the system. Even though this is no issue from a purely mathematical point of view, the s-time still lacks physical interpretation. Following a streamline means 'freezing' the flow at some instance of time t and integrating (along a 'virtual' time s) to $\pm\infty$. Only in special cases do particles follow streamlines in realistic scenarios (and usually for a while only, if at all).

A promising approach is to investigate the behavior of *pathlines*, i.e., the solutions of

$$\dot{\mathbf{x}}(\mathbf{t}) = \mathbf{v}(\mathbf{x}(\mathbf{t}), \mathbf{t}), \quad \mathbf{x}(\mathbf{t_0}) = \mathbf{x_0}. \tag{2}$$

The solutions of this equation describe the theoretical path of massless particles through the flow.

Another approach that uses the path of massless particles is the investigation of so-called *streaklines*, defined as

$$\mathbf{x_t}(\tau) = \mathbf{x}_\tau(\mathbf{t}) \tag{3}$$

where $\mathbf{x}_\tau$ is the solution for the initial value problem

$$\dot{\mathbf{x}}(\mathbf{s}) = \mathbf{v}(\mathbf{x}(\mathbf{s}), \mathbf{s}), \quad \mathbf{x}(\tau) = \mathbf{x_0} \tag{4}$$

evaluated at $s = t$. This describes mathematically the common experimental setup of injecting a marker (say dye) in a flow at a fixed spatial location $\mathbf{x_0}$ for the time interval $[t_0, t]$. The function $\mathbf{x_t}$ is then a parameterization of the curve consisting of the injected particles at time t, more precisely, $\mathbf{x_t}(\tau)$ is the position of the particle seeded at $\tau \in [t_0, t]$ at time t.

The concepts of path- and streakline are essentially different from the concept of streamlines in unsteady flow. Their focus is the behavior of one or more moving particles. Therefore they can be classified as Lagrangian methods. We discuss these methods in Section 4. However, applied to steady flow, which is of course a special case of unsteady flow, all three definitions yield the same trajectories.

In the context of this view on flow scenarios, structures that maintain their attracting (or repelling) nature over a relatively long time play an important role, since they influence all passing particles in a coherent manner. Along these lines, a scalar measure for the local separation behavior of the flow, the so-called finite-time Lyapunov exponents (FTLE), have gained attention in the visualization community [59]. The notion of *Lagrangian Coherent Structures* (LCS) recognizes that there are repeating patterns of motion in turbulent flows [18]. This phenomenon of repeated, similar structures has led to the assumption that understanding these coherent structures will give insight into the mechanisms of turbulence. Although, even today, there is no generally accepted definition of Lagrangian coherent structures, one important notion is

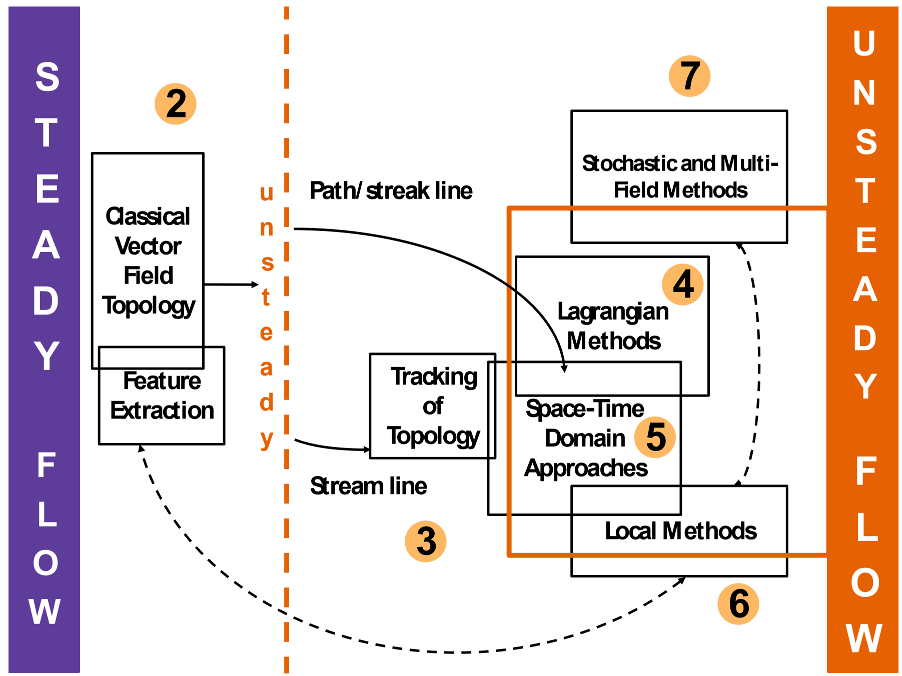


Figure 1: Approaches and methods used to achieve topology-based unsteady flow visualization and their relations. Thematic overlap is represented by intersecting building blocks. Solid arrows represent conceptual relations and dashed arrows methodological. The orange square collects the current available brick stones that come closest to the topology-based visualization of unsteady flow. The numbers indicate the sections in which the respective building blocks are discussed.

to identify them as the ridges of the FTLE field [60]. Less formally, LCS can be thought of as the boundaries between fluid regions for which injected tracer particles would behave qualitatively different [142]. Compared to VFT, there is a subtle, yet important, difference: in VFT the segmentation into different regions is point wise (Eulerian perspective) while LCS segments particles (Lagrangian perspective).

Recently, a mathematical framework called *Feature Flow Field* has been introduced which can treat the concepts of path- and streamlines in a unified way [203]. The idea behind this approach is that the unsteady flow is transformed into a higher dimensional steady flow. Then the computation of path- and streamlines reduces to the computation of streamlines of some related vector fields. Classical vector field topology is not applicable to these fields, however, since they do not contain isolated critical points. Nevertheless, it is possible to capture parts of the topological information of the original vector field, e.g., critical points, periodic orbits, and vortex axes, by constructing respective auxiliary vector fields. For different tasks different vector fields are needed. These and similar methods can be classified as space-time domain approaches and we discuss them in more detail in Section 5.

Feature extraction is an important complement to VFT in the steady case (to be precise, the extraction of some features, e.g., critical points, is an integral step in computing the topology of a steady flow). Of course, it is also desirable to extract the unsteady counterparts of the features in steady flow. Most of the methods used for this purpose are local, i.e., they use point-wise information only. The actual extraction is carried out by methods also known from image processing. In contrast to methods that involve integration, most of these techniques can be used for unsteady vector fields (at least to a certain degree – differences can show up, for example, when derivatives play into the feature specification). Currently, they focus mainly on vortex structures and separation and attachment lines. Local methods of that kind are discussed in Section 6.

One problem that feature extraction suffers from is that the definition of features involves parameters like thresholds or time windows (which is also true for FTLE) or that the definition is not unanimous (e.g., as for vortices). Often features are not detected in the actual vector field but in a field derived from the original one and the detection of multiple features (or various definitions of the same feature-type) has, consequently, to deal with multiple fields.

Since dealing with multiple feature specifications at once can be interpreted as dealing with multivariate data, the use of *Interactive Visual Analysis* (IVA) has been suggested [8]. The idea is to combine several feature detectors in order to investigate combinations of them. This is valuable both for extracting those features and for understanding the parameters that determine behavior that might be intuitively clear but not precisely defined. Another opportunity offered by IVA is to detect correlations between different feature definitions. Furthermore, this method offers the possibility to meet the needs of the user domain more flexibly.

An engineer, for instance, might be interested in additional properties (e.g., pressure, temperature, ...) of the medium, apart from the actual flow. On the other hand, engineers may use different models for the same situation, according to different tasks. IVA gives the opportunity to interactively investigate the relations between different variables/models using multiple views and linking+brushing [25].

One prerequisite regarding feature extraction is that the user has to be aware of which feature should be searched for. Recently, information theory based approaches were presented that are capable of automatically detecting regions in which something extraordinary is likely to happen [86].

Finally, one may be interested in displaying both flow topology and features. Unfortunately, it is known that separatrices may cross features (e.g. vortices) and therefore split them. *Stream-* and also *pathline predicates* offer a possibility to combine several feature detectors and flow topology in order to refine the latter, while keeping features intact [178, 174].

IVA and the above mentioned methods addressing similar problems will be discussed in Section 7.

In accordance with this brief overview of the building blocks available on the way towards topology-based visualization of unsteady flow, the rest of the paper is structured as follows: (2) Classical Vector Field Topology, (3) Tracking of Topology, (4) Lagrangian methods, (5) Space-Time Domain Approaches, (6) Local Methods, (7) Stochastic and Multi-Field Approaches, and (8) Discussion and Conclusions.

Figure 1 gives a graphical overview of the classes of approaches and methods and how they are related to each other as well as a graphical table of content of this article.

As mentioned before, this state of the art report uses the term *topology-based* in a broadened way, since it targets time-dependent vector fields. For such fields, a definition of 'topology' is not yet available, unless adopting a streamline-based view. As explained, this topology is hard to interpret in a physically meaningful manner. For a detailed overview over strictly topology-based methods for flow visualization and vortex extraction in unsteady flows we refer to Scheuermann and Tricoche [182] and Laramee et al. [68]. Salzbrunn et al. [175] present a survey of partition-based methods in flow visualization, covering also flow topology. Again, the main focus is on methods related to the tracking of steady topology, but it touches also upon some of the Lagrangian methods and stochastic and multi-field approaches that we discuss in Sec. 4 and Sec. 7, respectively.

Besides flow visualization by means of topology-based methods, there exist other approaches not covered in this article, such as *dense and texture based* and *feature based flow visualization*. For surveys on these areas of flow visualization, we refer to Laramee et al.[106] and Erlebacher et al. [34], respectively Post et al. [162, 106]. Yet another class of approaches are so-called *integration-based methods*. Since topology-based methods are usually based on integrational objects, a fair share of approaches presented there are contained in this class. If no topology is extracted, but the integrational objects are directly used for visualization, we refer to *integration-based geometric methods* for better distinction. For further discussion of this subclass of methods we refer to McLoughlin et al. [133].

2 Classical vector field topology

This section gives a brief overview on both historical and theoretical aspects of classical, i.e., steady, vector field topology as well as its application in visualization and further applications.

2.1 History

The theory of *dynamical systems* goes back to the 19^{th} century work of Henri Poincaré [159]. A modern introduction can be found, e.g., in Guckenheimer and Holmes [56]. In our context, the case of deterministic, continuous, and autonomous dynamical systems is most interesting, since such systems can be used to formulate velocity fields of a steady fluid flow. Many patterns in a flow can be described and analyzed by concepts from dynamical systems theory, such as *critical points*, *separatrices* and *periodic orbits*. Perry and Chong [150] give a comprehensive overview of such 2D and 3D flow patterns. Helman and Hesselink introduced these methods to the scientific visualization community, and used them under the notion of *vector field topology* for the visualization of computed and measured velocity fields, first in 2D [74] and later in 3D [75]. Vector field topology was further popularized both by Asimov's excellent tutorial [2] and by Globus et al.'s TOPO module [49] for NASA's FAST visualization software. Over two decades, topologically-based flow visualization has been an active research topic. A related state-of-the-art report [107] was published in 2007.

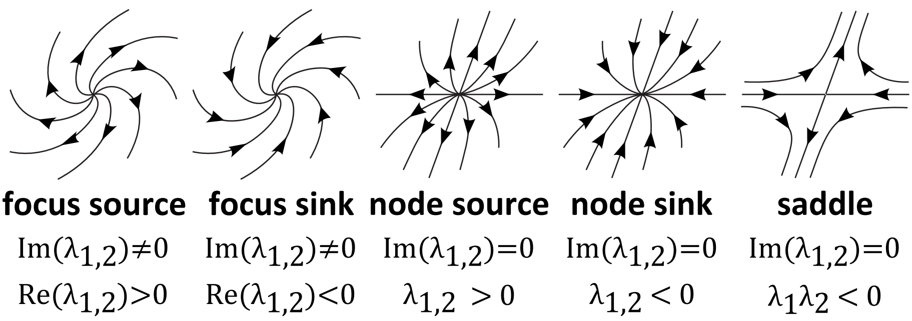


Figure 2: Types of first-order critical points in 2D

2.2 Background

Let $\mathbf{v}(\mathbf{x})$ denote a steady velocity field. Then a streamline, i.e., the solution of the initial value problem given in equation (1),

exists uniquely if $\mathbf{v}(\mathbf{x})$ is Lipschitz-continuous [56], which is the case for discrete data when interpolated with any of the popular schemes. Vector field topology now deals with the two kinds of singular streamlines, namely *stationary points* and *periodic orbits*. These singularities are of particular interest if they are *isolated*. A sufficient condition for an isolated stationary point, called a critical point, is that the velocity gradient tensor is regular at this point (while its velocity is vanishing). Similarly, a periodic orbit is isolated if the gradient tensor of the Poincaré map is regular [56]. For these first-order singularities, a type classification can be made by analyzing the eigenvalues of the gradient tensor. For 2D vector fields, there are the five possible types *saddle, node source, node sink, focus source* and *focus sink*, plus transitional types which are structurally unstable, see Fig. 2. In the special case of a divergence-free 2D vector field, there are no sources or sinks, but instead the center is a structurally stable type.

Type classifications exist also for first-order critical points in 3D fields and for first-order periodic orbits in 3D fields [2]. Finally, higher-order singularities can be further analyzed. Depending on higher-order derivatives, the singularity (critical point or periodic orbit) can still be an isolated one. A classification of higher-order critical points in 2D is given by Firby and Gardiner [36]. Scheuermann et al. [179] introduce a visualization of higher-order critical in 2D points using locally higher-order polynomial interpolations, based on the index of the singularities.

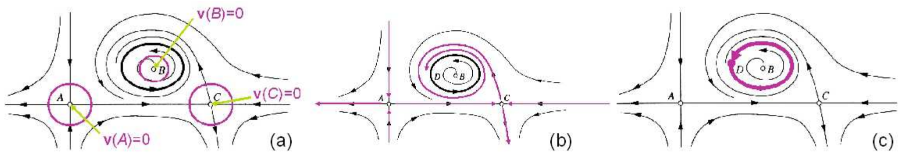


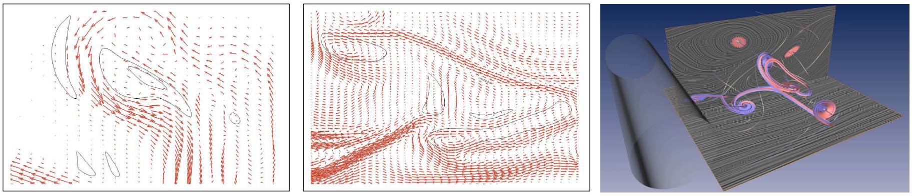
Figure 3: A topology-based visualization of a 2D vector field. In turn, the critical structures are highlighted: (a) critical points, (b) separatrices emerging from the critical points and (c) a periodic orbit. Arrowheads have been added in order to indicate attracting or repelling behavior and hence the categorization of the respective structures. A few additional trajectories enhance the perception further.

2.3 The topological skeleton of a vector field

The *topological skeleton* is obtained by computing all singularities plus their lower-dimensional invariant manifolds. In 2D fields only the saddle type critical points have 1D invariant manifolds. These are the so-called *separatrices*, i.e., the streamlines converging in either positive or negative time to a saddle point. As the topological skeleton contains most of the topological information of a (steady) vector field, it is a concise characterization of the vector field. The separatrices divide regions of different flow behavior and they often have physical relevance. In 3D velocity fields, such topological structures – then being surfaces – can indicate phenomena like flow separation or vortex axes.

Roughly speaking, the computation of the topological skeleton consists of the following steps:

1. Computation of critical points: Find all $\mathbf{x}$ such that $\mathbf{v}(\mathbf{x}) = \mathbf{0}$. Notice that this means that the right hand side of the differential equation becomes zero and the solution is consequently constant.
2. Classify the critical points: Due to $\mathbf{v}(\mathbf{x}) = \mathbf{0}$ the local behavior of the vector field is dominated by the gradient of the field (cf. Taylor series expansion). Hence, an eigenvalue analysis of the gradient can classify the flow locally. The signs of the eigenvalues are used to detect attracting, repelling, or saddle-like behavior.
3. Compute the separatrices: The invariant manifolds are computed by integrating from the critical point in the direction of the elements of a basis of the respective eigenspace (i.e., along the direction of the corresponding eigenvectors).
4. Compute higher order critical structures: Such structures are, e.g., closed orbits.
5. Classify the higher order critical structures: Analogous to critical points,



(a) (b)

Figure 4: (a) Periodic orbits detected in a turbulent 2D flow field (image courtesy of Wischgoll et al. [230] © 2001 IEEE); (b) Visualization of *saddle connectors* in a flow behind a circular cylinder (image courtesy of Theisel et al. [205] © 2004 IEEE).

higher order critical structures can be attracting, repelling, or induce saddle-like behavior.

Then, the topological skeleton is the union of critical points, respective separatrices, and higher order critical structures. Figure 3 shows the above described structures in a topology-based visualization of a 2D vector field. This description is intended to provide the reader with an intuitive understanding of how to extract the topological skeleton. For more details the reader may refer to Asimov [2].

2.4 Visualization methods based on vector field topology

A considerable amount of research has been done to extract, analyze, modify and visualize the topology of steady vector fields. Several approaches can be used to extract critical points. In piecewise linear fields, the zeros can be computed explicitly. In more general settings, one might use a Newton-Raphson approach [95]. An octree-like method is presented by Mann et al. [126]: they compute the index of the vector field (a generalization of the winding number) for each cell and a non-zero index triggers a recursive subdivision. Trotts et al. [213] introduce the notion of critical points at infinity to find new separatrices. The curvature of streamlines in the proximity of critical points is studied by Theisel and Weinkauf [196, 222] for 2D and 3D vector fields. Mahrous et al. [125] present an algorithm to extract separation surfaces to segment topologically steady 3D flow. They sample the vector field by streamlines, deriving a segmented data set from the original field and using this data set for the construction of the separation surfaces. In a later paper Mahrous et al. present an improved algorithm [124] that uses inflow/outflow matching, cell-locking, and adaptive streamline sampling to reduce computational work. Regions of different flow behavior on the boundary of 2D vector fields as well as the corresponding separatrices are considered by de Leeuw and van Liere [19] and Scheuermann et al. [180].

A first approach to detecting periodic orbits is given by Wischgoll and Scheuermann [230] which uses the underlying grid structure of a piecewise linear vector field: each grid cell is analyzed concerning the re-entering behavior of streamlines that start at its boundaries. Figure A.4(a) shows results obtained by this method. The method is extended to 3D [231] by the same authors.

Löffelmann et al. [121] propose visualization techniques for the Poincaré map in order to give a better understanding of the flow near periodic orbits. Peikert and Sadlo discuss periodic orbits in 3D vector fields [146]. Li et al. [113] discuss how to represent higher-order critical points on triangular surfaces using a carefully chosen triangulation and interpolation. Scheuermann et al. [179, 181] explained visualization approaches for planar flows. An algorithm for computing 2D invariant manifolds of singularities in 3D vector fields is presented by Krauskopf and Osinga [103] where the surface mesh is organized in geodesic circles. Theisel et al. [205] propose to display only pairwise intersections of such streamsurfaces, known as saddle connectors or heteroclinic orbits. Figure A.4(b) shows saddle connectors in a flow behind a circular cylinder. Peikert and Sadlo [149] present a streamsurface algorithm that robustly handles starting from and converging to singularities.

Separation and attachment lines play an important role considering the flow around and on bodies in 3D flow fields. Kenwright [96] and Kenwright et al. [97] present methods to extract attachment and separation lines. Wiebel et al. [229] present a robust method to extract separation surfaces from these lines using topology extraction in cross sections of the flow.

Sadlo and Weiskopf [171] present an approach to time-dependent 2D vector field topology based on *generalized streak lines*, i.e., streak lines with a moving instead of a fixed seed point. This allows them to give a generalized definition of saddle-type critical points for unsteady flow. This approach is inspired by *Lagrangian coherent structures*, that are treated in Sec. 4, but avoids ridge extraction.

While the topological skeleton usually provides complete information about the qualitative behavior of a flow, no quantitative information can be reconstructed from it. Löffelmann et al. [119] and Löffelmann and Gröller [120] propose the use of selected direct visualization cues in order to provide an intuitive description of the local flow near characteristic structures.

In order to account for uncertainty in vector fields, Otto et al. [139] present the concept of *uncertain* vector field topology for two dimensional fields. This approach considers the transport of local uncertainties under integration, generalizing the concept of streamlines for probability density distribution functions.

Together with the generalization of the concept of critical points, this allows for a topological analysis of the uncertain vector field. Otto et al. [140] extend the method in the previous paper to 3D. Furthermore, they improve the integration accuracy and simplify the task of saddle point detection. The topological structure found is displayed using volume rendering.

2.5 Further applications of topological features

As described by Theisel et al., topological features of vector fields have not only proved to be a valuable visualization tool, they can also be used for other tasks in processing vector fields [201].

Compressing vector fields. To simplify and compress large and complex flow data sets, methods based on topological concepts allow for more efficient computational handling and transmission. Compression in this context means to reduce the amount of data while maintaining important structures. Lodha et al. [118, 117] introduce a compression technique for 2D vector fields which prohibits strong changes of location and Jacobian matrix of the critical points. Theisel et al. [199] present an approach which guarantees that the topology of original and compressed vector field coincides both for critical points and for the connectivity of the separatrices. It is shown that even under these strong conditions high compression ratios for vector fields with complex topologies are achieved.

Topological simplification of vector fields. The topological skeleton of a vector field may become very complex due to the presence of noise. The reduction of unimportant topological features can be accomplished by simplifying the resulting topological structure. Besides smoothing of the vector field using a box filter before extracting the topology as described by de Leeuw et al. [20], more involved techniques start with the original topological skeleton and repeatedly apply local modifications of the skeleton and/or the underlying vector field in order to remove unimportant critical points. De Leeuw and van Liere [19] extract the topological skeleton and measure the importance of a critical point by computing the area from which the trajectory ends in forward or backward integration. Based on this area metric, the unimportant critical points are repeatedly collapsed to more important critical points in the neighborhood. The system described by de Leeuw et al. [20] finds couples of first order critical points in the skeleton by considering distance and connectivity of them. Then, pairs of saddle points and attracting/repelling critical points with distance less then a given threshold are collapsed. Tricoche et al. [210] use a similar approach but provide a way of consistently updating the underlying vector field instead of collapsing the extracted skeleton. Further, the simplification of the topology of a 2D vector field is accomplished by replacing clusters of first order critical points with a higher order critical point. Weinkauf et al. [226] extend this to 3D vector fields. Theisel et al. [198] solve the coupling problem of critical points by a feature flow field approach which will be explained in section 5.2 in further detail.

Topological comparison of vector fields. The definition of useful metrics on vector fields plays a crucial role in the majority of applications mentioned above. The first approaches on metrics (distance measures) of vector fields as proposed by Heckel et al. [71] and Telea et al. [195] consider local deviations of direction and magnitude of the flow vectors in a certain number of sample points. These distance functions give a fast comparison of the vector field but do not take any structural information of the vector fields into consideration. A first approach to define a topology based distance function is given by Lavin et al. [108]. Given two vector fields $\mathbf{v_1}$ and $\mathbf{v_2}$, all critical points are extracted and coupled. Then the distance of the vector fields is obtained as the sum of the distances of the corresponding critical points in $\mathbf{v_1}$ and $\mathbf{v_2}$. To compute the distance between two critical points, a number of approaches exist [108, 204]. To couple the points, Theisel et al. [200] propose the use of feature flow fields. A general demonstration of this comparison on real data sets is given by the same authors [201].

Constructing vector fields. Besides using a simulation or measurement process for data acquisition the vector field data can also be obtained by construction. Theisel et al. [197] present an approach oriented at methods from the CAGD (Computer Aided Geometric Design) context. First, a topological skeleton of a vector field is constructed by a number of control polygons. Second, a piecewise linear vector field of exactly the specified topology is automatically created. An approach for constructing 3D vector fields is presented by Weinkauf et al. [225]. There, a number of specified control polygons is used to determine location and characterization of first or higher order critical points and the saddle connectors. The resulting skeleton is used to construct a piecewise linear vector field. In application to 3D surfaces, topology-based construction and editing of vector fields can be used to enrich surfaces with additional information. Thus vector fields have been used for generating non-photorealistic visualizations, like painterly renderings or pen-and-ink visualizations, and remeshing of the underlying surface [141]. Zhang et al. [235] present a system to interactively create and edit 2D static vector field which can be applied to the limited domain of a 3D surfaces Recently, topological methods have been successfully applied to extract salient features on discrete 3D surfaces as shown by Weinkauf et al. [220].

3 First approaches towards unsteady flow fields: tracking of topology

First attempts to cope with time-dependent velocity fields were done by looking at the instantaneous velocity fields. Taking this as a starting point, some extensions to classical vector field topology are available. Newer research shows the limitations of this approach, e.g., with respect to a meaningful interpretation of the results.

3.1 Tracking of singularities

Instantaneous topology extraction can be combined with tracking of the singularities over time. Tricoche et al. [211, 212] present a method for tracking the location of critical points and detecting local bifurcations, i.e., qualitative changes in the topology of the field due to a smooth parameter change, such as fold bifurcations and Hopf bifurcations. This approach works on a piecewise linear 2D vector field and computes and connects the critical points on the faces of a prism cell structure, which is constructed from the underlying triangular grid. An extension to 3D has been given by Garth et al. [44] together with a visualization of the paths in space-time of the critical points. The framework of feature field flows allows for tracking of singularities as well. A detailed discussion of this tool is given in Section 5.2.

The consideration of bifurcations has to be handled carefully in this context. Bifurcations in the topological structure of a flow field can only happen due to changes of external flow parameters. To a certain degree, time can be seen as such a parameter when a streamlines-based view on the flow is adopted. However, due to the lack of an immediate physical interpretation of streamlines-based topology in time-dependent flow, it remains questionable how expressive the resulting structures are. In flow with a structure that only changes slowly over time it is possible that the identified 'bifurcations' indeed hint on interesting changes in the flow over time.

Wischgoll et al. [232] track closed streamlines over time by applying a contouring and connecting approach. At each time step closed streamlines are detected independently of each other, then the corresponding lines in adjacent time steps are connected.

Figure 5: Applications of FTLE to visualization. (a) In the double gyre example the critical point disjunct to the FTLE ridge separating different regimes of the flow (image created following Shadden et al. [185]). (b) Volume rendering of the FTLE field shows the regions of locally maximally attracting and repelling behavior (image courtesy of Garth et al. [42] © 2007 IEEE). (c) Extraction of ridges from the FTLE field allows additional processing and filtering to concentrate on the salient features of the flow (image courtesy of Sadlo et al. [168] © 2007 IEEE).

3.2 Deficiency of vector field topology for unsteady flow

As previously explained, streamlines do not capture the temporal change of the flow. In the context of experimental flow visualization, researchers noted very early that a correct frame of reference is important for extracting meaningful structures. Perry and Tan [152] suggest to extract patterns as 'seen' by an observer who is moving with the eddies, i.e., the swirling and backflow induced by flow past an obstacle. They use a correlation technique to compute the velocity of an eddy and found the resulting measurements to be quasi-steady. Later, Perry and Chong [151] state clearly that topological information is only meaningful in a Galilean reference frame in which the velocity field is nearly steady. This implies that vector field topology is not applicable if such a frame does not exist.

While known in theory, practice largely ignored this problem until when Shadden et al. [184] give with the 'double gyre' an example of an unsteady flow for which a saddle type critical point substantially deviates from the actual point of flow separation. Recently, Wiebel et al. [227] demonstrate the failure of vector field topology to find moving attractors in simulation data of a rotating liquid suspension. They suggest a procedural solution based on the evolution of density of virtual particles seeded in the flow.

4 Lagrangian methods

In the Lagrangian point of view, the fluid is described by the motion of its particles. Since the analysis is based on trajectories of one or multiple particles such methods are inherently suited for unsteady flows.

4.1 The finite-time Lyapunov exponent

The finite-time Lyapunov exponent (FTLE), by some authors also referred to as the direct Lyapunov exponent (DLE) [59], is a measure for the stretching of an infinitesimal neighborhood along a finite segment of a flow trajectory.

More formally, let $\mathbf{v}(\mathbf{x}, t)$ denote the velocity field. Then, a *trajectory* $\mathbf{x}(t)$ starting from $\mathbf{x}_0$ at time t_0 is the solution of an initial value problem (see also Equation 2). The set of all trajectories provides the *flow map* $\mathbf{x}(\mathbf{x}_0, t_0, t)$ that maps the position at time t on the trajectory started at time t_0 from $\mathbf{x}_0$. By computing the flow map gradient and left-multiplying it with its transpose, the (right) *Cauchy-Green deformation tensor field* [129] is obtained as

$$\mathbf{C}_{t_0}^{t}(\mathbf{x}_0) = \left[\frac{\partial \mathbf{x}(\mathbf{x}_0, t_0, t)}{\partial \mathbf{x}_0}\right]^T \left[\frac{\partial \mathbf{x}(\mathbf{x}_0, t_0, t)}{\partial \mathbf{x}_0}\right]. \tag{5}$$

From this, the *(maximum) FTLE* is defined as

$$\mathrm{FTLE}_{t_0}^{t}(\mathbf{x}_0) = \frac{1}{2(t - t_0)} \ln \lambda_{\max}\left(\mathbf{C}_{t_0}^{t}(\mathbf{x}_0)\right), \tag{6}$$

where $\lambda_{\max}(\mathbf{M})$ denotes the maximum eigenvalue of $\mathbf{M}$ [59].

In the limit $t \to t_0$ the FTLE is the *maximum principal rate-of-strain*, i.e., the maximum eigenvalue of the rate-of-strain tensor

$$\mathbf{S} = \left[\nabla \mathbf{v}(\mathbf{x}_0, t_0)\right]^T \left[\nabla \mathbf{v}(\mathbf{x}_0, t_0)\right]. \tag{7}$$

In the limit $t \to \infty$, the FTLE is the (standard) Lyapunov exponent which is independent of t_0. Discovered by A. M. Lyapunov in the 1890's, the Lyapunov exponents became popular in the 1970's for the analysis of chaos and predictability in dynamical systems. The finite-time variant are used [50, 234] originally also for predictability of systems, especially for atmospheric models. In a seminal paper [59], Haller applies FTLE to velocity fields of fluid flow and reveals their relationship to the Lagrangian coherent structures (LCS), which can provide the information on flow separation similar to the separatrices of vector field topology, however often also correctly for strongly time-dependent

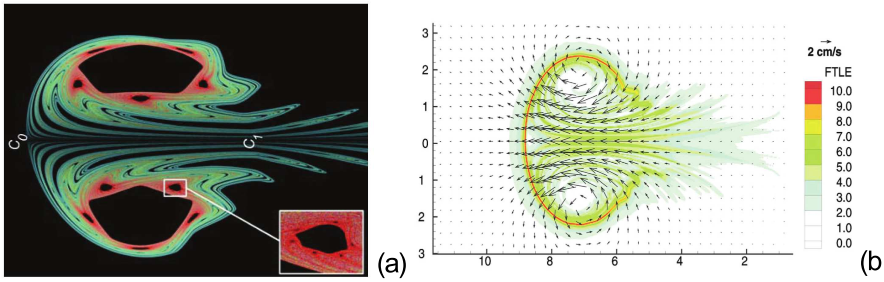


Figure 6: Analysis of a vortex ring. (a) Topological methods can benefit from the infinite integration time available and give detailed insight into regions of stability and folding structures of the flow (image courtesy of Peikert et al. [148]). (b) Even though much less integration time is available, the FTLE field can give insight into the structure of the vortex ring (image courtesy of Shadden et al. [185]).

flow. In his subsequent paper [60], he identifies the *ridges* of the FTLE as LCS. In Figure 5 we show some applications of FTLE.

Shadden et al. [184] apply FTLE to the 'double gyre' example (where vector field topology fails) and various other example flow fields in 2D. They show visually that particles seeded near the FTLE ridges do not cross them. Another counter-example for vector field topology is suggested by Wiebel et al. [227] where the FTLE peak was shown to deviate much less from the observed (moving) attractor than the topological sink.

Garth et al. [42] present an algorithm for FTLE computation in transient flow using adaptive refinement of the flow map and propose to approximate 3D FTLE by 2D FTLE computed in the orthogonal space of the velocity vector. Garth et al. propose a volume rendering approach that avoids the extraction of ridges using a 2D transfer function [43]. With a variation of this technique Garth et al. [45] compute 2D FTLE on offset surfaces of solid boundaries resulting in a visualization of flow separation and flow reattachment. Sadlo et al. address the problems of efficient computation of height ridges of FTLE [168] and of tracking FTLE ridges over time by using a grid advection technique [169]. Lipinski and Mohseni [115] present a ridge tracking algorithm for FTLE fields that uses both temporal and spatial coherency of LCS and give an error estimator for difference between advected ridge and actual LCS. Both approaches give great speed-up compared to the standard algorithm.

Apart from a more effective ridge extraction, several authors recently suggested alternative methods to speed up FTLE computation. Brunton and Rowley [6] present a fast computation scheme for the FTLE field based on a multi-stage approximation on the flow map, eliminating redundant integrations. This comes with an accuracy trade-off, for which the authors provide an error bound. This approach is especially useful for the computations of time-sequences of FTLE fields. A similar approach is presented by Hlawatsch et al. [78]. Kasten et al. [91] introduce the notation of *localized FTLE* (L-FTLE). The main idea of this approach is to exchange the deformation gradient tensor with a matrix that accumulates the separation behavior along a path line. The computation of the matrix is based on the flow gradient. The computation of the L-FTLE field allows for the reuse of values computed for previous time steps, using an idea similar to FastLIC, which results in a speed-up compared to traditional FTLE. A comparison of FTLE and L-FTLE shows few differences [91].

Comparisons of FTLE with other criteria in terms of suitability for visualization have been made by several authors. Shadden et al. [185] show that FTLE is able to reveal the fine lobes of a chaotic vortex ring while producing temporally more consistent results than an approach based on vector field topology. In Figure 6 we compare VFT and FTLE. In (a) we can see that the possibility to integrate streamlines into a chaotic region of the flow for very long integration times allows to extract sharply defined regions of stability. In Figure 6(b) we can see that the restriction to a finite time domain is alleviated using FTLE to visualize the structure of the vortex ring.

In recent studies by Green et al. [53] and Shi et al. [187], FTLE is validated against other indicators of LCS in a number of analytical and numerical flow fields, and FTLE is found to generate more detail. In a study done by Sadlo et al. [170], FTLE is shown to extract flow separation structures, but not the axes or centers of rotating flow. In comparison with vector field topology, this means that FTLE provides only partial information. In the example of a *spiral saddle critical point*, where vector field topology would give a 1D and a 2D invariant manifold that can be interpreted as a vortex axis and a separation surface, only the latter is reliably detected by FTLE.

Another current limitation of FTLE is that it requires the choice of a time window, the effect of which has not been studied sufficiently. Generally, a longer integration time will produce sharper ridges. On the other hand, this may cause a larger number of trajectories to leave the flow domain. Tang et al. [194] show that just stopping the integration at the boundaries may introduce spurious ridges and suppress true ridges. The authors develop an algorithm that extends the given flow field linearly, allowing the paths to continue at a locked separation rate and addresses the problem of particles leaving the flow domain.

In recent work by Olcay et al. [138] the influence of noise and spatiotemporal resolution of the velocity field on the extracted LCS is investigated. The authors show that a coarse resolution can significantly influence the location of a LCS. Smoothing the field is shown to have the same effect. Spatial noise can have a significant effect on single realizations of the LCS, but the mean location remains near LCS extracted from the unperturbed field.

Ferstl et al. [35] present an approach for interactive investigation of 3D flows using streak lines, that uses FTLE ridges in a 2D seeding probe as seeding structures. In this manner the separation behavior detected by the FTLE can be investigated in more detail, avoiding costly computations in regions that exhibit coherent particle motion. Depending on the size of the data set, the FTLE can be computed on the fly, exploiting the GPU, or has to be precomputed.

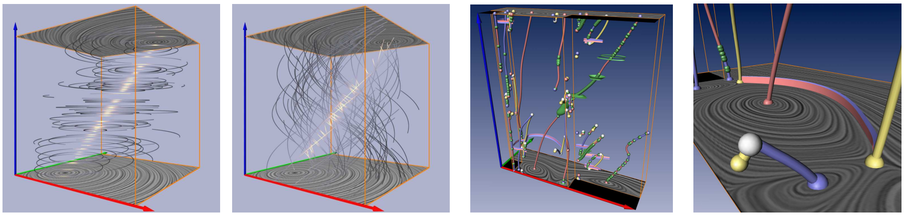
It is worth noticing that the result of LCS extraction based on FTLE is influenced by the definition of a ridge, given the choice of height ridges, watersheds, maximal curvature ridges [31] and others. Even when agreeing on the same concept of ridge, the definition may not be unanimous, as in the case of height ridges [168].

4.2 Other Lagrangian feature detectors

While FTLE, in addition to its advantages, also has the aforementioned limitation to inform only about flow separation, other calculations can be performed in the Lagrangian frame that reveal other types of flow features. Basically, by computing the Cauchy-Green deformation tensor from the flow map gradient, the rotational part is discarded. However, to detect a vortex, this information is needed. Therefore, either the flow map gradient must be used in a different way or a different type of temporal integration must be performed.

Cucitore et al.'s non-local vortex detector [16] uses a reference frame that moves with a particle to be tested. In this frame, the path of a neighbor particle is calculated for a certain time window. Then, the distance of the end point from the origin is divided by the arc length of the path. Low values of this ratio indicate a vortex center. Haller proposes another vortex detector denoted M_z [61] that is objective, i.e., invariant not only under Galilean transforms, but also for rotating frames of reference. Finally, any local vortex detector designed for steady flow can be adapted to unsteady flow by applying a Lagrangian smoothing, i.e., by computing a weighted average of the quantity obtained for the same particle at several time steps. Lagrangian smoothing has been shown to be better than a purely steady analysis by Shi et al. [186] and by Fuchs et al. [41].

Recently, several authors brought up the idea to adapt the definitions underlying vector field topology for unsteady velocity fields. Kasten et al. [90] propose minima of the acceleration magnitude, after a temporal smoothing in the Lagrangian frame, as a replacement for critical points in unsteady velocity fields. Fuchs et al. [39] present the concept of *motion compensated critical points* and



(a) Stream lines of **s** correspond to the stream lines in **v**.

(b) Stream lines of **p** correspond to the path lines in **v**.

(c) Streamline oriented topology

(d) Detail view with a saddle connection and a fold bifurcation

Figure 7: Streamlines (a) and pathlines (b) of a simple 2D time-dependent vector field obtained by linear interpolation of two steady 2D vector fields and shown as illuminated field lines. The extracted and visualized topological skeleton (c) and detailed structures (d) of the cavity data set (image courtesy of Theisel et al. [207] © 2005 IEEE).

a novel measurement for 'unsteadiness'. This allows for the identification of particles that are observing an almost steady velocity field and represent a nearly Galilean transform. In this frame of reference, classical VFT is applicable to classify the particles. This can be considered to be a first step towards a 'Lagrangian' vector field topology.

5 Space-time domain approaches

In order to be able to handle the problem of detecting features in time-dependent data sets, one way is to lift this problem into a higher dimension by interpreting the time as an additional axis and thereby assume the steady case again. This definition allows a clear definition of pathlines by means of streamlines in the lifted higher-dimensional case.

5.1 Streamlines and pathlines

When dealing with a time-dependent vector field $\mathbf{v}(\mathbf{x}, t)$, we are usually interested in its spatiotemporal characteristics. As discussed in the introduction, several concepts can be used to explore those characteristics. In a specified space-time point $(\mathbf{x}_0, t_0) \in D$ we can start a *streamline* (cf. eq.(1)) or a *pathline*. The defining ODE system (2) 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:

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

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

$$\mathbf{p}(\mathbf{x}, t) = \begin{pmatrix} \mathbf{v}(\mathbf{x}, t) \\ 1 \end{pmatrix} \tag{8}$$

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

$$\mathbf{s}(\mathbf{x}, t) = \begin{pmatrix} \mathbf{v}(\mathbf{x}, t) \\ 0 \end{pmatrix}. \tag{9}$$

This is valid for arbitrary space dimensions.

Figure 7 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 bilinear vector fields.

Now the problem of finding a streamline or pathline oriented topology is reduced to finding the topological skeletons of $\mathbf{s}$ and $\mathbf{p}$. Unfortunately, the classical vector field topology extraction techniques for 3D vector fields are not applicable for the fields $\mathbf{s}$ or $\mathbf{p}$: $\mathbf{s}$ consists of critical lines (i.e., for every critical point $\mathbf{x}^*$ of the original vector field $\mathbf{v}$ any point $(\mathbf{x}^*, t)$ in the time-space domain will become a non-isolated critical point of $\mathbf{s}$), while $\mathbf{p}$ does not have any critical points at all.

5.2 Feature flow fields

In the *feature flow field* (FFF) approach [203], a specially designed vector field in the 4D space-time domain captures parts of the topological information (critical points, periodic orbits, vortex axes) in its temporal evolution. Consider an arbitrary point $\mathbf{x}$ known to be part of a feature in a (scalar, vector, or tensor) field. A feature flow field $\mathbf{f}$ is a well-defined vector field at $\mathbf{x}$ pointing in the direction where the feature moves to. Thus, starting a streamline integration of $\mathbf{f}$ at $\mathbf{x}$ yields a curve where all points on this curve are part of the same feature as $\mathbf{x}$.

Feature flow fields are commonly used with local features, which can be described by a local analysis of the underlying field and possibly its derivatives.

Here, $\mathbf{f}$ can usually be described by an explicit formula. In the 2D case the underlying vector field is given as follows:

$$\mathbf{v}(x,y,t) = \begin{pmatrix} u(x,y,t) \\ v(x,y,t) \end{pmatrix} \tag{10}$$

Using this description, the direction of maximal change of the u and v-component of $\mathbf{v}$ is given by the gradients $grad(u)$ and $grad(v)$. In the plane perpendicular to $grad(u)$ the u component remains constant in a first order approximation of $\mathbf{v}$. A similar statement can be made for v. Thus, the only direction in which u and v remain constant is the intersection of the perpendicular planes denoted by the cross product of $grad(u)$ and $grad(v)$:

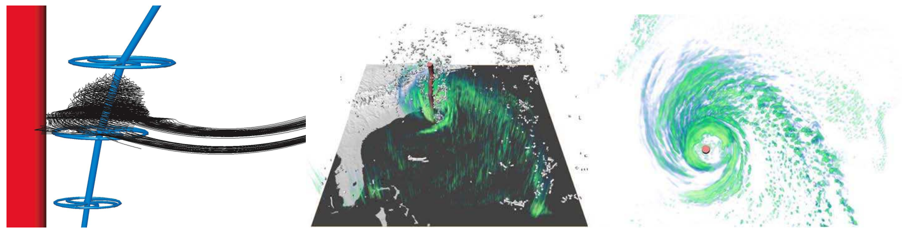
$$\mathbf{f}(x,y,t) = grad(u) \times grad(v) = \begin{pmatrix} det(\mathbf{v}_y, \mathbf{v}_t) \\ det(\mathbf{v}_t, \mathbf{v}_x) \\ det(\mathbf{v}_x, \mathbf{v}_y) \end{pmatrix} \tag{11}$$

In contrast to this, a FFF for a global feature can only be given in an implicit manner, since it can neither be decided locally whether a point belongs to a feature nor into which direction the feature evolves. Instead, the FFF approach has to be tightly coupled with a global feature detection strategy in order to assess global features.

Tracking features in time-dependent fields is one of the main applications of feature flow fields [203, 206, 207]. The temporal evolution of the features of $\mathbf{v}$ is described by the streamlines of $\mathbf{f}$. In fact, tracking features over time is now carried out by tracing streamlines. The location of a feature at a certain time t_i can be obtained by intersecting the streamlines with the time plane t_i. Integrating the streamlines of FFF in the forward direction does not necessarily mean to move forward in time. In general, those directions are unrelated and the direction in time may even change along the same streamline. Those changes are always related to special events, where multiple critical points merge, split up or vanish within the underlying vector field. Hence, FFF provides a tool to localize, characterize and classify bifurcations. Notice that the notation of bifurcation implies that the flow is interpreted from the streamline-based point of view.

Besides tracking, FFF have been used for a variety of related problems. Those include topological simplification and comparison of vector fields based on critical point tracking [199], extraction of vortex core lines defined as ridges/valleys of Galilean invariant quantities [172], extraction and tracking of vortex core lines defined as centers of swirling motion [202], extraction of topological lines in tensor fields [236, 237], and identification of periodic phenomena from insufficiently time-resolved data sets measured using particle image velocimetry [22].

Weinkauf et al. [224] introduce the notion of *stable* FFF, i.e., FFF where streamlines associated with the desired feature exhibit attracting behavior. This for-



(a) (b)

Figure 8: (a) Flow past a tapered cylinder visualized using a non-local vortex detector (image courtesy of Reinders et al. [163]); (b) Visualization of the core of *swirling particle motion* in the Hurricane Isabel data set (image courtesy of Weinkauf et al. [221] © 2007 IEEE).

mulation guarantees that small numerical errors are automatically corrected during integration. The authors show the construction of such fields for two common applications of FFF: the tracking of critical points in a time-dependent 2D vector field and the extraction of parallel vector lines for 3D vector fields. This stable variant has been applied recently to identify discontinuities in multivariate data by tracking feature lines [109].

Weinkauf and Theisel [223] introduce the so-called *steak line vector field*, that allows for the formulation of streak lines as tangent curves of a derived vector field of the original field. In this manner, the costly computation of a streak line can be reduced to a simple vector field integration. Hence, this reformulation of streak lines opens up for a more extensive use of streak lines and surfaces in flow visualization.

6 Local methods

Features such as *edges* or *ridges* [64, 32, 114] of images can be extracted by methods that are *local* in the sense that they work on point-wise information, including derivatives. These methods carry over naturally from image data to scalar field data as they occur in scientific visualization problems. Height ridge extraction is applied to pressure data by Miura and Kida [134] and to vorticity magnitude by Strawn et al. [191], both times for finding vortex core lines. Ridge extraction from FTLE data is proposed by Shadden et al. [184] for finding Lagrangian coherent structures.

For the visualization of vector fields such as velocity data, adaptations or generalizations of these methods can be used. Such techniques exist for the extraction of separation and reattachment lines [97], vortex core lines [110, 193, 3, 134, 166]. Some of these vortex core line methods involve additional physical quantities, in particular the pressure gradient [3, 134], but the remaining ones, such as the classical methods by Levy et al. [110] and by Sujudi and Haimes [193] are based solely on the velocity field and its derivatives. For a detailed discussion of the topic of vortex extraction, we refer to the survey of Jiang et al. [83].

Many of these structures can be expressed with a unifying formalism, called the *parallel vectors operator* (PVO) [145]. The PVO concept is not restricted to line-like features, but can be extended to surface-like features [202]. For the case of height ridges, simplified extraction methods were recently proposed for arbitrary dimensions, together with a new class of filters for the filtering of raw features [147].

In contrast to integration-based methods, local methods are relatively unaffected by the unsteadiness of the velocity field. Therefore, most of the mentioned methods are directly applicable to unsteady flow. An exception is the recent extension of the vortex core line detector of Sujudi and Haimes to unsteady flow [221, 40]. The reason for this was that the Sujudi and Haimes method can be re-interpreted as an operation on the acceleration field. If this is computed from a given unsteady velocity field, it requires a temporal derivative term, which is not needed in the steady case.

The general approach of defining and extracting features based on local criteria for the velocity field and its derivatives is a powerful concept, due to its mathematically rigorous formulations and the simple algorithms derived from them. At first glance, it may look wrong to describe global structures of a vector field by local operators. In fact, the different behavior of height ridges and watersheds in image data led to a lively dispute [98, 31] about the correctness of local vs. global methods. However, while in steady flow one of the most interesting topological structure, the separatrix, can be computed only using global methods, there is no reason to assume that unsteady flow does not contain topologically important structures that can be found by local methods. In a related context, Ginoux and Rossetto [47] show that in 2D and 3D slow-fast autonomous dynamical systems, the *slow manifold* can be computed by finding zeros of curvature or torsion, resp., of the local trajectory. Finally, local methods can be combined with integration-based methods. An example is FTLE computation which leads to a scalar field and which has to be post-processed if sharp structures, such as height ridges, are needed.

Although the problem of detecting vortices is usually addressed using local methods as described above, there are methods that use a geometric approach. Sadarjoen and Post [167] suggest two methods detecting vortices in steady 2D flow fields detecting clusters of centers of the osculating circles and streamlines with winding number 2π and relatively close start and end point. The latter method is extended to 3D by Reinders et al. [163]. Petz et al. [154] propose a new criterion to characterize 2D vortex regions. In order to do so, they detect and cluster loops that intersect the underlying flow at a constant angle. Their algorithm is parameter-free and is not restricted to a certain type of geometry (e.g. star domains or convex domains).

Figure 8 shows visualizations of vortical flow using local (A.8(b)) and non-local (A.8(a)) detectors.

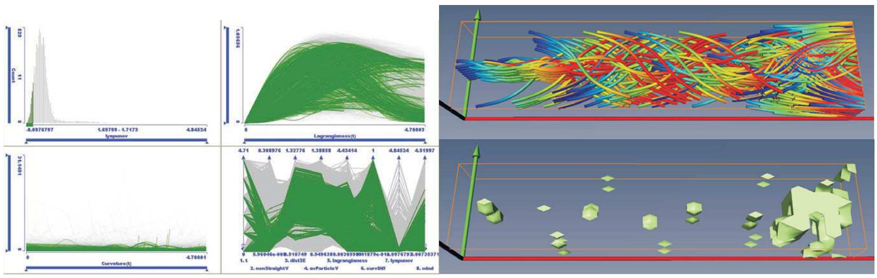
7 Stochastic and multi-field approaches

Rarely is the user just interested in one aspect (e.g., one single feature type) of a flow field. It is more common to look at multiple features, features in combination with additional measures and/or multiple definitions of the same feature at once to get an understanding of the underlying field. Recently, a number of new approaches and methods have been introduced in order to take these requirements into account.

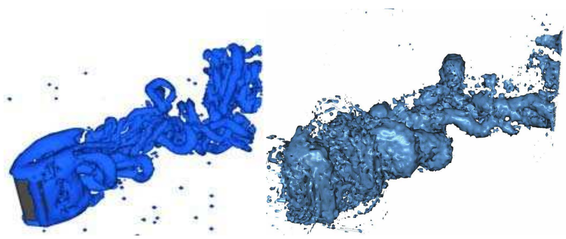
7.1 Interactive visual analysis

As the amount and complexity of data sets grows, automatic analysis methods are often not sufficient any more. In order to effectively cope with such data sets, interactive visual analysis (IVA) tries to balance human cognition and automatic analysis. The power of human perception and cognition is used to guide the analysis. The IVA approach provides an interactive discovery framework. It helps the user in getting insight, in understanding the data as well as complex, often hidden, correlations between certain data dimensions. The visual information-seeking mantra – *overview first, zoom in, details on demand* – as defined by Shneiderman [188], summarizes the main idea. Coordinated multiple views [165] are often used in this domain [130] as a proven concept. The main idea is to depict multiple dimensions using multiple views and to allow the user to interactively select (*brush*) a subset of the data in one view and all corresponding data items in all linked views will be highlighted as well [127, 25]. One of the first examples of linking and brushing with different visualization approaches in different views is a system called WEAVE [54], which was used to interactively analyze and visualize simulated data of a human heart application using focus+context visualization [67]. IVA is used in many domains [94]. In the following, however, we will focus on engineering and scientific applications.

Doleisch et al. developed a system called SimVis for interactive feature specification and localization in 3D flow data. They use simple 2D linked views, such



(a)



(b)

Figure 9: (a) Pathlines with small *Lyapunov exponents* in a flow behind a circular cylinder. The region to display is selected in the histogram (upper left window) the corresponding pathlines (upper right display) and their seeding points (lower right display) are displayed (image courtesy of Shi et al. [186]); (b) Comparison of the visualization of a flow around a cuboid using the standard λ_2-criterion (left) and *local statistical complexity* (right) (image courtesy of Jänicke et al. [85]).

as scatter plots or histograms, for the specification of flow features. Linked 3D views provide spatial information and advanced flow visualization techniques. Complex features can be described by composite brushing. The feature definitions are expressed in an XML-based feature definition language and are persistent across analysis sessions. The SimVis system has been used to analyze flows from numerous applications, such as flow through a catalytic converter, flow around a car, cooling jacket flows, etc [24, 28, 29, 105].

Another approach deals with the parameterization of pathlines in order to understand flow. The main idea is to compute various attributes from pathlines in order to understand the flow itself. Shi et al. [186] compute scalar and time series attributes of pathlines, such as: winding angle, Lyapunov exponent, direction vector, etc., and then use coordinated multiple views in order to understand the flow behavior. Figure A.9(a) shows their interface while analyzing a data set.

Bürger et al. [7] compute several local feature detectors of the same flow and use

IVA to compare them. In addition, other flow attributes (such as pressure, ...) are taken into account as well. In this way it can be intuitively decided which automatic method gives more accurate results in certain areas or time intervals. Such an approach enhances the credibility and combines the advantages of several detectors in an interactive visual analysis system.

IVA is not intended as a competitor or an alternative to the detectors described before. Instead, it is sought to be used in parallel to those methods. It offers great potential in the exploratory phase, during hypothesis generation [93]. The flow segmentation is not an isolated process, it is part of a larger work flow. Domain experts analyzing the flow have to choose detectors, and IVA can help in deciding if detectors are applicable, if a detector functions in particular case. Domain experts have to evaluate multiple detectors. Engineers, for example, compute a vortex detector first, and then check if this is an area of low pressure as well. The analysis can be refined for areas where this holds, and can be skipped for other areas. Offering multiple views, intuitive interfaces and quick selection possibilities, IVA provides a useful tool for such a complex task. It can also help to improve robustness of detectors. A filtering step is almost always necessary after a detector is evaluated. Exploiting smooth brushing [26], a method which allows non-strict brush boundaries, local characteristics of detectors can be examined much more easily. Hauser and Mlejnek [69] show how a similar approach can be efficiently applied to isosurfaces in the analysis of flows in a catalytic converter.

IVA is not really another flow segmentation method – at least not in the classical sense – but more an integrative approach which helps domain experts to understand detectors and flow behavior.

7.2 Fuzzy feature detectors

While IVA handles multi-field structures (induced by multiple features, multiple definition of features and/or additional quantities), utilizing multiple views and linking+brushing, other attempts have been made to address problems related to feature extraction and visualization in a fashion that corresponds more to the classical methods in flow visualization with respect to their outputs.

One of the drawbacks of feature extracting methods is that the user has to be aware of the type of feature which should be extracted. Additionally, the feature one is looking for may not be defined unanimously (e.g. vortices). In order to address this problem, Jänicke et al. [85] recently presented an improvement of the algorithm of Jänicke et al. [86] for an automatic extraction and visualization of regions of interest in 3D unsteady multi-flow. The authors detect space-time points that have high probability to develop into unlikely events in future using a statistics-based algorithm. As a measure for the unexpectedness of the value at a point they propose *local statistical complexity*, which is, roughly speaking, the amount of information needed to predict the future of a space-time point. Figure A.9(b) shows a visualization of a flow around a cuboid obtained by this method.

Salzbrunn and Scheuermann suggest the use of *streamline predicates* in order to combine flow topology with feature extraction [178]. The main idea is to decompose the domain into disjoint regions with coherent streamline behavior, as flow topology does, adding other distinctions than asymptotic behavior. This addresses, e.g., the problem that some features can be split up by usual flow topology. Mathematically speaking, streamline predicates are Boolean maps with disjoint support on the set of all streamlines. Flow topology is then a special case of segmentation gained through streamline predicates, called *flow structure*. Classical feature detectors can be used to refine flow topology using streamline predicates. The same ideas are applied to unsteady flow by Salzbrunn et al. [174]. In analogy to the steady case the authors introduce the notation of *pathline predicates*. Additionally, the authors present a pathline placement strategy in order to combine the structural overview provided by the partition gained by means of pathline predicates with the dynamical insight into the flow provided by tracing single particles.

In an engineering context, feature models with parameters are often used. The quantification of these parameters is obviously an important task. Ebling et al. [33] point out that topology-based methods are not capable of doing this. They show, e.g., that for an arbitrary vector field the topological skeleton of the normalized field is the same as the skeleton of the original field. This means that VFT is inherently unable to provide quantitative information on the investigated flow field. Another drawback of topological methods in this context is that superposing features may not be detected correctly. The authors suggest therefore the use of vector masks and pattern matching. This approach emphasizes the interpretation of a vector field as the superposition of many (simpler) fields.

8 Discussion and conclusions

This paper describes the current state of the art in topology-based flow visualization of unsteady vector fields. Topological methods for steady flows are used as a role model for what we expect of new methods. The terminology *topology-based*, as used in this survey, has to be interpreted accordingly, i.e., more loosely, e.g, also including *topology-inspired* methods and methods that share one major goal of topology-based methods, i.e., to achieve an expressive segmentation of a flow field. Accordingly, by surveying these approaches together, this report might contribute to approaches that aim at catalyzing convergence.

To date, the solutions for topology-based unsteady flow visualization remain incomplete, compared to the level of proficiency achieved for steady flows. Incremental extensions of methods that work well for steady flows are proven to be not able to truthfully capture the behavior of time-varying flows. Therefore new approaches and methods are examined, including both new theoretical frameworks and methodical novelties. Many of the new approaches seem to overlap to a certain extent. This suggests that a unified framework for treating unsteady flows with topology-based methods actually could be found.

The impulses that brought topological methods into the focus of the visualization community came from the application domain itself. One of the most prominent examples is Globus et al.'s TOPO module [49] for NASA's FAST visualization software. In the light of this, it may seem surprising that topology-based methods found their way into commercial solutions only in a very limited form (e.g., by Avizio, www.vsg3d.com) up to now. One possible reason for this could be that topology-based methods are still rather new in the field compared to many other techniques (especially those inspired by well known experimental setups). This means, in turn, that such methods are usually not covered by standard education curricula for simulation experts. Another reason might be that topological methods are quite advanced methods and a fair number of questions relevant to the domain expert can be answered by simpler methods as well. For more intricate questions, however, topology-based methods are able to provide insights that are not possible with other approaches, as recent publications from the fluid dynamics community show, cf. Peacock and Dabiri [142] for example. In order to be applicable by a broader community, these methods will have to be time-efficient and expressive, as well as easy to use. Finally, as many other examples show, technological advancements may also lead to the usage of advanced methods, such as topology-based methods, in contexts that would not strictly require this. For example, topology-based methods might be used for illustration purpose, in analogy to the illustrations in recent text books on dynamical systems, or in a particle seeding strategy. One first step in this direction is the use of FTLE ridges as seeding structures for interactive exploration of 3D flow using streak surfaces proposed by Ferstl et al. [35].

We perceive a strong current interest in proceeding with research on topology-based and topology-inspired visualization of unsteady flow and major attempts are being undertaken, such as the cooperative international project that the authors of this survey are involved in (www.semseg.eu).

Acknowledgements

This report has been worked out within the scope of the SemSeg project and we acknowledge the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open grant number 226042. We also thank the anonymous reviewers of our survey for their constructive input and Matthew Geoffrey Parker (Univ. of Bergen) for proofreading.