Eurographics Conference on Visualization (EuroVis) 2013
B. Preim, P. Rheingans, and H. Theisel
(Guest Editors)

Volume 32 (2013), Number 3

Visualizing Robustness of Critical Points for 2D Time-Varying Vector Fields

B. Wang[1], P. Rosen[1], P. Skraba[2], H. Bhatia[1] and V. Pascucci[1]

[1]Scientific Computing and Imaging Institute, University of Utah, USA
[2]Jozef Stefan Institute, Slovenia

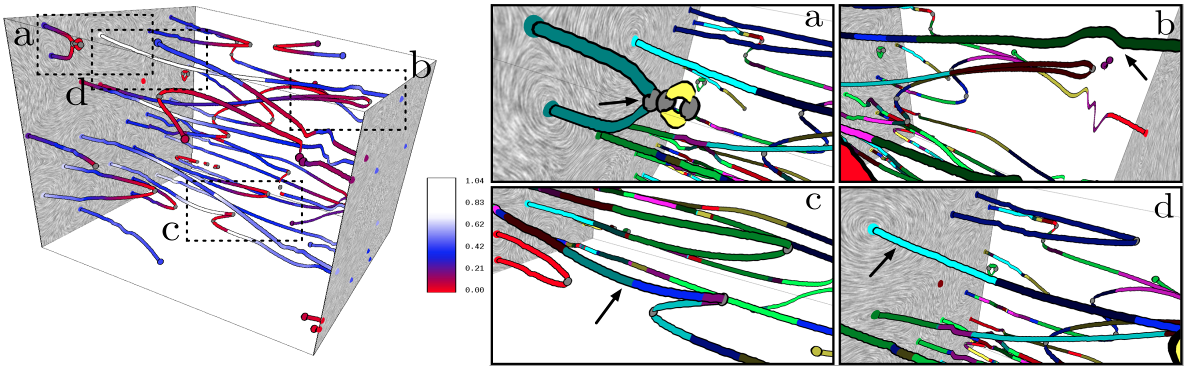


Figure 1: *Robustness assignment for critical point trajectories for a 2D time-varying vector field from a combustion simulation. The trajectories are mapped to colors based on their (dynamic) robustness values. The zoomed-in versions show robustness pairings among the trajectories: same color segments are paired to each other. (a) and (c) involve fold and blue-sky bifurcations. (b) and (d) are part of a long trajectory with high robustness values but different partners.*

Abstract

Analyzing critical points and their temporal evolutions plays a crucial role in understanding the behavior of vector fields. A key challenge is to quantify the stability of critical points: more stable points may represent more important phenomena or vice versa. The topological notion of robustness is a tool which allows us to quantify rigorously the stability of each critical point. Intuitively, the robustness of a critical point is the minimum amount of perturbation necessary to cancel it within a local neighborhood, measured under an appropriate metric. In this paper, we introduce a new analysis and visualization framework which enables interactive exploration of robustness of critical points for both stationary and time-varying 2D vector fields. This framework allows the end-users, for the first time, to investigate how the stability of a critical point evolves over time. We show that this depends heavily on the global properties of the vector field and that structural changes can correspond to interesting behavior. We demonstrate the practicality of our theories and techniques on several datasets involving combustion and oceanic eddy simulations and obtain some key insights regarding their stable and unstable features.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric algorithms, languages, and systems

1. Introduction

Vector fields represent an important and ubiquitous class of data that arise in many scientific disciplines. Vector field topology, introduced into the visualization community by Helman and Hesselink [HH91], has inspired much research in the analysis and visualization of vector fields, e.g. see surveys [LHD*04, PVH*03, WE05, PL09, MLP*10, BCP*12]. A large body of work focuses on the study of critical (fixed) points [PP03, SKMR98, TSH01, TSH00a, Wei08], locations where the vector value is zero. A key challenge is to quantify the stability or significance of a critical point, for both stationary and time-varying vector fields.

In this paper, we use a relative of *persistence* [ELZ02, DW07,RKG*11], called *robustness*, to represent the stability of critical points with respect to perturbations in a rigorous manner. Understanding how stable points are is crucial to assessing their significance. The notion of persistence has been used successfully for scalar field visualization and analysis but is ill-suited to vector-valued data. On the other hand, robustness, introduced in [EMP11] through the algebraic concept of well diagrams [EMP10], is specifically designed for such data. Recently, an algorithm was given to compute well diagrams for vector fields on $\mathbb{R}^n$ [CPS12]. We relate and extend these results to perform analysis and visualization of both stationary and time-varying 2D vector fields. Another closely related work by Kasten el. al. [KHNH11] introduced the notion of long-lived features. Such long-lived features are different from robust features as the former focuses on lifetime and the latter relies on the amount of perturbation necessary to eliminate the feature. However, as shown in Section 6 in the case of ocean eddies, these two concepts are correlated. Thus maybe the most interesting features are those with high robustness values across a long period of time. The work in [RLH11] also strongly advocated the need for importance measures for critical points and proposed such a measure under the setting of stationary vector fields.

In [EMP11], the authors first proposed the question regarding how different robustness is from persistence and whether there is a reduction of one to another. They gave a setting in which the two are almost the same under the one-dimensional notions of robustness. Such relations were further explored in [BEMP13] by describing how the robustness of features in level and interlevel sets, quantified through well groups, can be read off the persistence diagram of the function. However in more general settings the reduction from robustness to persistence is not known and the authors in [EMP11] have conjectured that robustness may sit somewhere between the 1-parameter notion of persistence and its multi-parameter generalization [CZ07]. The notion of robustness offers twofold benefits. First, it helps us differentiate features from noise, although what is considered noise may be application-driven and data-dependent. Second, it motivates us to seek compact representations of the datasets by maintaining essential features. Such representations could potentially speed up analysis and reduce visual clutter. Our main contributions can be summarized as:

1. We give two different notions of robustness, static and dynamic. The notion of *static robustness* and its computation for stationary vector field is not new. It was first introduced in [CPS11] as *robustness*. Intuitively, static robustness of a critical point is the minimum amount of local perturbation necessary to cancel it, measured under a well-defined metric. Our contribution regarding static robustness is to describe explicitly, for the first time, its provable properties associated with quantifying the stability of a critical point with respect to perturbations of the vector field. On the other hand, we introduce the new concept of *dynamic robustness*, motivated by establishing an explicit connection between the stability of critical points with the stability of well diagrams. It offers a different view of critical point stability by describing the global correspondence of a set of critical points with respect to bounded perturbations. The properties associated with both types of robustness give a theoretical basis for our analysis and visualization.

2. Using existing [CPS11] algorithms for static robustness and new algorithms for dynamic robustness, we map robustness values to critical point trajectories for 2D time-varying vector fields. The mapping gives us a structural description of the vector field. In particular, they help us understand the *temporal stability* of critical points and how they migrate and interact with one another.

3. We have applied existing visualization methods to create a tool that enables interactive exploration of robustness of critical points for 2D time-varying vector fields. To demonstrate the practicality of our theories and techniques, we use this tool on several real world datasets involving combustion and ocean eddy simulations. We obtain some key insights regarding their stable and unstable features.

2. Preliminary

In this section, we review previous work [EMP11, CPS11, CPS12] by providing its relevant background in degree theory, merge tree and well group theory. We provide minimal algebraic definitions and illustrate the related concepts by examples whenever possible. For non-specialists, the examples towards the end of this section may be sufficient to convey a basic understanding of these concepts.

Degrees. The first concept we need is the *degree* of a continuous mapping. Here we avoid its algebraic definition (see [Hat02] page 134 and [CPS12]), but point out that in a 2D vector field, the degree of a critical point x (that is, the degree of a mapping f at x), denoted as $\deg(x)$, equals its Poincaré index. That is, a source, a sink and a saddle have degrees $+1$, $+1$ and -1, respectively. Furthermore, for a path-connected component C in the domain that encloses a set of critical points $\{x_i\}$, its degree (more precisely, the degree of f restricted to C) is $\deg(C) = \sum_i \deg(x_i)$.

Merge tree. Given a continuous 2D vector field $f : \mathbb{R}^2 \to \mathbb{R}^2$, we can define a scalar function $f_0 : \mathbb{R}^2 \to \mathbb{R}$ at each point $x \in \mathbb{R}^2$ as its Euclidean norm, $f_0(x) = ||f(x)||_2$. Let $\mathbb{F}_r = f_0^{-1}(-\infty, r]$ be the *sublevel set* of f_0 for some $r \geq 0$. A value $r > 0$ is a *regular value* of f_0 if $\mathbb{F}_r$ is a 2-manifold, and for all sufficiently small $\varepsilon > 0$, $f_0^{-1}[r-\varepsilon, r+\varepsilon]$ and $f_0^{-1}(r)$

are diffeomorphic; otherwise it is a *critical value*. We further assume f_0 has a finite number of critical values and $\mathbb{F}_0$, the set of critical points of f, is finite.

If we observe the evolution of $\mathbb{F}_r$ as r increases, we can construct a graph that tracks (connected) components of $\mathbb{F}_r$ as they appear and merge. This is called a *merge tree* (or *join tree* [CSA00]). The root represents the entire domain of f_0 while the leaf represents the creation of a component at a local minimum. An internal node represents the merging of two or more components. We further augment the merge tree with degree information at each node. That is, to each node, we record an integer that is the degree of the corresponding component in the sublevel set. Since the degree of a component is the sum of the degrees of the critical points lying in it, an initial computation of the degrees of critical points is sufficient to determine the degree of any component of any sublevel set [CPS12].

Well groups and well diagrams. Let $f, h : \mathbb{R}^2 \to \mathbb{R}^2$ be two continuous 2D vector fields. Define the distance between the two mappings as $\mathrm{d}(f,h) = \sup_{x \in \mathbb{R}^2} ||f(x) - h(x)||_2$. A continuous mapping h is an *r-perturbation* of f, if $\mathrm{d}(f,h) \leq r$.

If h is an r-perturbation of f, then $\mathbb{H}_0 = h^{-1}(0)$ is a subspace of $\mathbb{F}_r$. The components of each space generate a vector space. These vector spaces are called the *0-dimensional homology* groups, $\mathsf{H}(\mathbb{H}_0)$ and $\mathsf{H}(\mathbb{F}_r)$, of $\mathbb{H}_0$ and $\mathbb{F}_r$ respectively, whose ranks equal the number of components. Since we only deal with dimension 0 homology we drop the dimension from the homology functor notation. The subspace relation induces a linear map $j_h : \mathsf{H}(\mathbb{H}_0) \to \mathsf{H}(\mathbb{F}_r)$ between the two vector spaces. The *well group*, $\mathsf{U}(r)$, is the subgroup of $\mathsf{H}(\mathbb{F}_r)$, whose elements belong to the image of each j_h, for all r-perturbation h of f [CPS12]. That is, $\mathsf{U}(r) = \bigcap_h \operatorname{im} j_h$. Intuitively, an element in $\mathsf{U}(r)$ is considered a stable element in $\mathsf{H}(\mathbb{F}_r)$ if it does not disappear with respect to any perturbation. Assuming a finite number of critical points, the rank of $\mathsf{U}(0)$ is the number of critical points of f.

For values $r \leq s$, $\mathbb{F}_r \subseteq \mathbb{F}_s$ induces a linear map $\mathsf{f}_r^s : \mathsf{H}(\mathbb{F}_r) \to \mathsf{H}(\mathbb{F}_s)$ between the two homology groups. It can be shown that $\mathsf{U}(s) \subseteq \mathsf{f}_r^s(\mathsf{U}(r))$, for $r \leq s$. Therefore, the rank of the well group decreases monotonically as r increases. We define the *terminal critical value* of f_0 as the radius r at which the rank of $\mathsf{U}(r)$ decreases. A point $r \in (0, \infty)$ belongs to the *well diagram* of f_0, $\mathrm{Dgm}(f_0)$, with multiplicity k if the rank of the well group drops by k at r [CPS12]. For reasons of stability, the point 0 is counted with infinite multiplicity. The point ∞ is counted with multiplicity k if for all sufficiently large values of r, the rank of $\mathsf{U}(r)$ is k.

The Equivalence Theorem [CPS12] suggests an algorithm to compute the well diagram. It states that, if r is a regular value of f_0, then the rank of the well group $\mathsf{U}(r)$ is the number of components C of $\mathbb{F}_r$ such that $\deg(C) \neq 0$. That is, the information within a well diagram can be read off from an augmented merge tree with degree information, as illustrated by the examples below.

Stability of well diagrams. The well diagram is stable in the following sense. Let $g : \mathbb{R}^2 \to \mathbb{R}^2$ be another vector field with a finite number of critical points. Construct a bijection $\mu : \mathrm{Dgm}(f_0) \to \mathrm{Dgm}(g_0)$ that sends the kth highest point in $\mathrm{Dgm}(f_0)$ to the kth highest point in $\mathrm{Dgm}(g_0)$. Recall, the point 0 in each well diagram has an arbitrary multiplicity. By choosing the appropriate multiplicities for 0, μ is a bijection. The *bottleneck distance* between $\mathrm{Dgm}(f_0)$ and $\mathrm{Dgm}(g_0)$ is $W_\infty(\mathrm{Dgm}(f_0), \mathrm{Dgm}(g_0)) = \sup_{a \in \mathrm{Dgm}(f_0)} |a - \mu(a)|$. The Stability Theorem of Well Diagrams [EMP11] states that the bottleneck distance between two well diagrams is bounded by the distance between the mappings. In other words, $W_\infty(\mathrm{Dgm}(f_0), \mathrm{Dgm}(g_0)) \leq \mathrm{d}(f,g)$.

Examples. We illustrate the computation of well diagrams through the construction of augmented merge trees as shown in examples (a) and (b) in Figure 2. We do not show the non-zero minima of f_0 in the visualization since they do not correspond to critical points of the vector field.

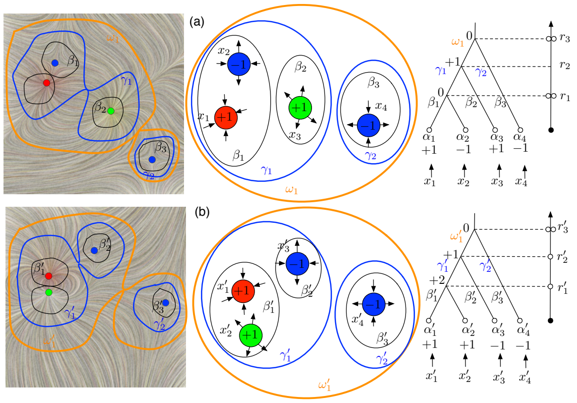


Figure 2: *Example (a) and (b): from left to right, vector fields f, relations among components of $\mathbb{F}_r$, augmented merge trees and well diagrams.*

Here the augmented merge trees are sufficient to derive its corresponding well diagram. We use α, β, γ etc. to represent components of certain sublevel sets. In example (a), the illustrative vector field $f : \mathbb{R}^2 \to \mathbb{R}^2$ on the left contains four critical points, a sink x_1 (red), a source x_3 (green), and two saddles x_2 and x_4 (blue). The mapping $f_0 : \mathbb{R}^2 \to \mathbb{R}$ has three critical values, which we denote by $0 < r_1 < r_2 < r_3$. The merge tree on the right shows how the components of the sublevel sets $\mathbb{F}_r$ evolve. At $r = 0$ there are four components $\alpha_1, \alpha_2, \alpha_3$, and α_4 that contain the four critical points, each with non-zero degree. At $r = r_1$ there are two components, β_2 and β_3 in $\mathbb{F}_r$ with non-zero degree; that is, the number of components with non-zero degree drops from four to two; therefore there are two points in $\mathrm{Dgm}(f_0)$ with value r_1. Similarly, at $r = r_3$ the number of components with non-zero degree drop from two to zero; therefore there are two points in $\mathrm{Dgm}(f_0)$ with value r_3.

In example (b), vector field f' contains four critical points: x'_1 (red sink), x'_2 (green source) , x'_3 and x'_4 (blue saddles). Following the previous argument, e.g. at r'_1 the number of components with non-zero degree drops from four to three, therefore there is a point in $\mathrm{Dgm}(f'_0)$ with value r'_1.

3. Static Robustness and Its Properties

The *static robustness* of a critical point is the height of its lowest degree zero ancestor in the merge tree, as first defined in [CPS11]. In this section, we describe explicitly how the static robustness quantifies the stability of a critical point with respect to perturbations of the vector fields through the following lemmas, whose algebraic proofs are included in the supplementary material.

Lemma 3.1 (Critical Point Cancellation) Suppose a critical point x of f has static robustness r. Let C be the connected component of $\mathbb{F}_{r+\delta}$ containing x, for an arbitrarily small $\delta > 0$. Then, there exists an $(r+\delta)$-perturbation h of f, such that $h^{-1}(0) \cap C = \emptyset$ and $h = f$ except possibly within the interior of C.

Lemma 3.2 (Degree and Critical Point Preservation) Suppose a critical point x of f has static robustness r. Let C be the connected component of $\mathbb{F}_{r-\delta}$ containing x, for some $0 < \delta < r$ and $r - \delta$ being a regular value. Then for any ε-perturbation h of f where $\varepsilon \leq r - \delta$, the sum of the degrees of the critical points in $h^{-1}(0) \cap C$ is $\deg(C)$. Furthermore, if C contains only one critical point x, we have $\deg(h^{-1}(0) \cap C) = \deg(x)$. In other words, there is no ε-perturbation ($\varepsilon \leq r - \delta$) that could cancel the critical point in C; that is, x is preserved.

In example (a), the static robustness of the critical points x_1, x_2, x_3, and x_4 are r_1, r_1, r_3, and r_3, respectively. Since the static robustness of x_1 is r_1, Lemma 3.1 implies that there exists an $(r_1 + \delta)$ perturbation for arbitrarily small δ that cancels x_1 by locally modifying the component $C \subseteq \mathbb{F}_{r_1+\delta}$ containing it. In example (b), the static robustness of all four critical points are r'_3. Let $C \subseteq \mathbb{F}_{r'}$ be a component containing x'_1 and $r'_2 < r' < r'_3$. By construction, $\deg(C) = \sum_{i=1}^{3} \deg(x'_i) = +1$. Lemma 3.2 states that any r'-perturbation preserves the degree of C. In addition, since x'_4 also has robustness r'_3, let $C' \subseteq \mathbb{F}_{r'}$ be a component containing only x'_4 for $r'_2 < r' < r'_3$. Lemma 3.2 implies that any r'-perturbation preserves x'_4.

The algorithm that assigns static robustness to each critical point involves traversing an augmented merge tree. The time complexity to build such a tree is based on the 2D contour tree construction $O(n \log n)$ [dBvK93], where n is the number of vertices in the triangulated domain.

Static robustness for a stationary vector field can be extended trivially to its time-varying setting. Suppose we have properly resolved the critical point correspondence problem and we are given the trajectories of critical points over time (so-called *critical paths*). We could obtain a robustness assignment along each critical path by first computing robustness at each discrete time slice and linearly interpolate the robustness values between time slices. This enables us, for the first time, the opportunity to study *temporal stability* of critical points.

4. Dynamic Robustness and Its Properties

In addition to static robustness, we now introduce a different stability quantifier called dynamic robustness. Given a continuous vector field $f : \mathbb{R}^2 \to \mathbb{R}^2$ and its corresponding well diagram $\mathrm{Dgm}(f_0)$, let X be the set of critical points of f and $\mathrm{Dgm}(f_0)$ the set of point values in the well diagram (including multiplicity). Since the rank of well group $\mathsf{U}(0)$ is equal to the number of critical points, we have $|X| = |\mathrm{Dgm}(f_0)|$. The *dynamic robustness* of a critical point $x \in X$ is the image $\rho(x)$ of a set bijection $\rho : X \to \mathrm{Dgm}(f_0)$, such that $\rho(x)$ is chosen among the terminal critical values of its ancestors in the merge tree which are no greater than its static robustness. The bijection ρ is referred to as a *dynamic robustness assignment*.

While static robustness quantifies the stability of a critical point by measuring the amount of perturbation required to cancel it independently, dynamic robustness quantifies the stability of a critical point by describing the global correspondence of a set of critical points with respect to bounded perturbations. The latter is motivated by establishing an explicit connection between the stability of critical points with the stability of well diagrams. Algebraically, static robustness has properties rooted in degree theory, in particular, the Hopf extension theorem [GP74]. Dynamic robustness, on the other hand, originates from the stability theorem of well diagrams [EMP11]. When the values of these two stability quantifiers coincide, their combined properties apply, implying simple relations among critical points; Otherwise, it represents more complex interactions between critical points.

The following lemma describes one of its key properties whose proof is included in the supplementary material. Intuitively, it shows the existence of a correspondence between two sets of critical points with bounded perturbations.

Lemma 4.1 (Dynamic Robustness Stability) If a critical point x of f has dynamic robustness r, then for all δ-perturbations h of f, where $\delta < r$, there exists a critical point y of h such that it has a dynamic robustness between $r \pm \delta$.

By definition, a dynamic robustness assignment is not necessarily unique. To obtain such an assignment, we find a perfect matching in the bipartite graph $G = (X, Y)$ where X is the set of critical points in f and Y is the set of points in the well diagram. G is constructed such that there is an edge between $x \in X$ and $y \in Y$ if y is among the terminal critical values of x's ancestors in the merge tree and y is no greater than the static robustness of x. If G contains m edges and n nodes, the algorithm can be derived from the max flow problem with a complexity of $O(mn)$.

In example (a) dynamic and static robustness coincide. In example (b) there are four possible dynamic robustness assignments by finding all possible perfect matchings in $G = (X, Y)$, where $X = \{x'_1, x'_2, x'_3, x'_4\}$, $Y = \{r'_1, r'_2, r'_3, r'_3\}$, and edges $E = (X \times Y) \setminus \{(x'_3, r'_1), (x'_4, r'_1), (x'_4, r'_2)\}$. The four perfection matchings with respect to critical points $\{x'_1, x'_2, x'_3, x'_4\}$, are (r'_1, r'_2, r'_3, r'_3), (r'_2, r'_1, r'_3, r'_3), (r'_1, r'_3, r'_2, r'_3) and (r'_3, r'_1, r'_2, r'_3). Here, the static and dynamic robustness assignments differ due to the fact that the degrees of the immediate ancestors of the given critical points are not necessarily zero.

Computing dynamic robustness over all time slices is nontrivial since the dynamic robustness assignment at each time slice is not necessarily unique. We are interested in a robustness assignment that is *as continuous as possible* between time t and $t+1$; that is, suppose there is a bijection $i : X \to X'$ (correspondence) between critical points X at time t and X' at time $t+1$, we find assignment $\rho : X \to \mathrm{Dgm}(f_0)$ and $\rho' : X' \to \mathrm{Dgm}(f'_0)$ at their respective time slices such that the total variation $\sum_{x \in X} |\rho(x) - \rho'(i(x))|$ is minimized. In the case where adjacent time slices have different numbers of critical points, we could map points to 0 (counted with infinite multiplicity in the well diagrams). We can find locally minimal solutions using alternating optimizations. In particular, we employ *forward* and *sandwich propagation*. The former fixes a perfect matching ρ at time t and find a min-cost perfect matching ρ' at time $t+1$ based on ρ. The latter fixes perfect matchings ρ and ρ'' at time t and $t+2$ respectively, and find a min-cost perfect matching ρ' at time $t+1$ based on ρ and ρ''. Finding min-cost perfect matching between t and $t+1$ uses the Hungarian method and takes time $O(n^3)$ where n is the number of nodes in the graph. In the datasets tested, such an optimization converges in a matter of a few seconds. We conjecture that finding the optimal assignment is NP-hard.

While we try to find matching which are continuous as possible, it not difficult to construct cases where no continuous assignment exists. This is due to global changes in the structure of the vector field (specifically surrounding bifurcations), indicating that discontinuities of dynamic robustness may highlight interesting behavior. We finally note that we use the word *dynamic* in our definition to reflect its non-uniqueness nature and its assignments across multiple time slices, it does not represent any information about the dynamic behavior of the flow.

5. Implementation

Our analysis and visualization framework relies on several implementation modules described below. Our computational pipeline also requires implementations of basic data structures such as merge tree [CSA00] and bipartite graph matching [CLRS09].

Critical point detection and critical paths tracking. The augmented merge tree algorithm is provably correct. It does, however, require correct identification of critical points. This numerical step in the computation can lead to errors and incorrect output of the well-diagram computation. In our implementation, we use a reliable critical point detection technique based on Simulation of Simplicity [EM90] that takes numerical instability into consideration [BGW*13].

Once the critical points and their degrees are computed, the next step is to track critical paths. We use a nearest neighbor construction, where each critical point identifies its nearest neighbor of the same type in the adjacent time step. Since the datasets we use have high sampling density, this simple method obtains critical point correspondences consistent with region overlapping approaches [PVH*03]. We are aware of other advanced feature tracking techniques [TWSH02, TS03, RKWH11] such as feature flow fields that could be used to construct critical paths. We are also aware of some new research that interprets feature tracking through the lens of robustness [SW13]. Our current visualization tool is designed as an illustration of how robustness could be visualized in the time-varying setting along critical paths. We would explore more advanced critical point tracking techniques in combining with the robustness framework in the near future.

Robustness assignment. For static robustness assignment, we simply traverse the augmented merge tree. For dynamic robustness assignment, we use the C++ Boost library to compute perfect matchings for bipartite graphs. For compact 2-manifolds without boundary, we expect all the robustness values to be finite. Note there is a technical condition on the domain for the algorithm to work (trivial tangent bundle), which excludes the sphere. If the domain has boundary, both static and dynamic robustness could be infinite due to boundary effects. Robustness values are then assigned to critical paths, which are visualized as explained below.

Interactive visualization. Our interactive system (as showcased in Figure 1 and the supplementary video) is capable of visualizing time-varying datasets in both 2D and 3D modes. While all robustness information is computed off-line, the interactivity allows users to adjust the visual appearances of critical points, critical paths and their associated robustness. The 2D mode shows a single time slice of the dataset. The vector field is visualized using a line integral convolution (LIC) implementation [CL93] with critical points indicated with spheres. The critical points are colored using one of a number of different transfer function modes based on their types, degrees, robustness values, or robustness partners. Critical points are considered *partners* (or they are *paired* with one another) if they share the same robustness value and belong to the same connected components at that value. We use a white, blue and red color scheme for robustness values, with white being high and red being low. We use black to indicate infinity. Users interact with the system by using a time slider. The 3D mode demonstrates the time-varying behavior of the dataset. 2D vector fields are set up as slices through 3D space with time as the third dimension.

The critical point paths are visualized using colored tubes which are stylized with black borders to be better differentiated from one another. The tubes are colored by the same transfer functions as those applied to the critical points in the 2D case. Critical point birth and death events are indicated by gray spheres. In the case that the vector field is defined on a periodic domain (i.e. a torus), its continuous boundaries need to be represented, i.e., when a critical point exists from one side and enters from the opposite side. Such events are indicated by spheres at the exits and entrances along the boundary. For a more focused analysis, our visualization tool can support a number of advanced functionalities. We enable switching on and off critical point paths so that we can focus on specific features in the data, along with the ability to chose subsets of time intervals for targeted analysis. We also support the usage of the 2D slice view within the 3D context for correlation of features between the 2D and 3D visualization modes. In addition, the system has the capability to visualize sublevel sets and their corresponding robustness. This is discussed in the next section.

6. Results

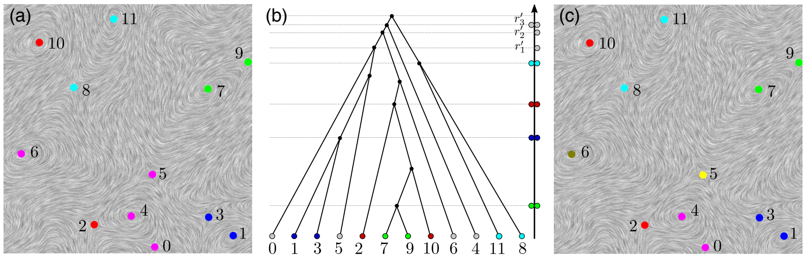
We showcase our interactive tool by focusing on different aspects of static and dynamic robustness for 2D and 3D views of time-varying vector fields. We obtain some intuitive and interesting insights by exploring the important and sometimes unexpected features of our testing datasets under the robustness setting. Although these insights are yet to be validated by the domain experts, they are still useful as they showcase the potential of the presented techniques in correlating domain-specific understanding of the flow dynamics with quantitative stability measures of the critical points.

Datasets. We demonstrate the practicality of our theories and techniques on several real-world datasets. Our first dataset (Combustion) is a taken from the simulation of homogenous charge compression ignition (HCCI) engine combustion [HSPC06]. The domain has periodic boundary, and is represented as a regular grid of resolution 640 × 640. The 2D time-varying vector field consists of 299 time-steps, each with a time interval of 10^{-5} seconds. We also explore the simulation of global oceanic eddies [MBP10] for 350 days (therefore 350 time slices) of the year 2002. We use the top layer of the 3D simulation which is represented as a 2D time-varying vector field of resolution 3600 × 2400. Our second and third datasets (CentralAtlantic and SouthAtlantic) are tiles extracted from this simulation data, representing the flow in the central Atlantic Ocean (60 × 60), and the south Atlantic Ocean (100 × 100) respectively. We construct standard triangulations on the point samples for these datasets.

In terms of performance, we use a 3.2 GHz Intel Core i7 with 8 GB of RAM, a Geforce GTX 580 graphics card. Graphics rendering for our tool is interactive, with 100+ frames per second. For Combustion, CentralAtlantic and SouthAtlantic datasets respectively (resp.), critical points computation takes 12 minutes (2.4 seconds / slice), 8 seconds and 21 seconds resp.; merge tree construction, traversal and bipartite graph computation takes 18 minutes (3.6 seconds / slice), 6 seconds and 19 seconds resp.; dynamic robustness assignment through forward and sandwich propagations takes 6.6 , 0.3 and 3 seconds to converge resp.

Combustion dataset. We give a small number of examples involving Combustion dataset here, some of which are showcased in Figure 1. We first demonstrate robustness for a single time slice of the dataset (which can be treated as a stationary vector field) in Figure 3. We compare the subtle differences in static (a) and dynamic (c) robustness assignments by showing critical points partners that are colored based on unique values. The merge tree and its corresponding well diagram are shown in (b). The gray leaf nodes in the merge tree refer to the critical points whose static and dynamic robustness values may not coincide. Their exact robustness values are shown in the table on the right. A common trend observed from our dataset is that in practice low robustness partners tend to be connected by separatrix, though not always. Another interesting feature is that the component that contains saddle point 5 merges with the component containing another saddle point 0 at a relatively high robustness value, therefore creating a degree -2 component, which later on cancels with components containing source/sink 6 and 4. This is almost identical to our synthetic example (b) (Figure 2) described earlier (by setting points 0, 5, 6, 4 to be x_1' to x_4' respectively). Notice that although points 0 and 4 are close in Euclidean distance, their robustness are quite high, implying that the vector magnitude surrounding their neighborhood remains quite large such that it requires a large perturbation to cancel these critical points.

Second we focus on time-varying vector fields and showcase global views for both static and dynamic robustness assignments along critical paths in a 3D cube, see Figure 6. We visualize both robustness assignment as well as pairing switches. Visualizing pairing switches showcases how critical points are related to each other and reflects structural changes of the vector fields. Again under a time-varying setting, there are subtle differences in how static or dynamic robustness changes along a critical path. Noticeably, in critical path a, dynamic robustness evolves more continuously due to the smaller number of partner switches. Meanwhile for the first time, we could observe highly robust critical points evolving over time, for example, along critical path c, where its corresponding critical point stays robust for more than 150 time steps, indicating an extremely stable feature of the vector field. In terms of robustness partners, we pay attention to a few interesting sets of critical paths. Notice for critical paths at e, under the static robustness setting, two critical points stay as partners the entire time as they move towards a fold bifurcation, while they switch partners midway under the dynamic robustness. Through comparisons, this might indicate an artifact as a result of alternating optimization obtaining a local minimum. On the other hand, for critical path



Pt.	S	D1	D2	D3	D4
0	r_3'	r_1'	r_2'	r_1'	r_3'
5	r_3'	r_2'	r_1'	r_3'	r_1'
6	r_3'	r_3'	r_3'	r_2'	r_2'
4	r_3'	r_3'	r_3'	r_3'	r_3'

Figure 3: *Static and dynamic robustness of critical points for a stationary vector field. Left: (a) static robustness where critical points are colored based on partners who share unique robustness values; (b) merge tree and well diagram; (c) dynamic robustness partners. Right: critical points (from left to right) whose static (S) and dynamic (D) robustness may differ.* $r_1' = 0.542685$, $r_2' = 0.550826$, $r_3' = 0.5549$. *The range of robustness is* $[0.0867794, 0.5549]$.

f, critical points partners, "remain faithful to each other until death," regardless of static or dynamic assignment.

Next, we focus on several types closely related to local bifurcations. For critical path b In Figure 6, we display the robustness values along with robustness partners, showing one blue sky bifurcation (where two critical points are suddenly created) happening between two fold bifurcations (where two critical points of opposite degree collide and annihilate each other). Notice in this case, the pairing remains stable as the two critical points move towards the bifurcation points, while robustness decreases continuously. Then we turn on and off critical paths to get unobstructed views of local features. Figure 4 shows two cases of robustness partner switches, each involving three critical paths joined by a blue sky bifurcation and a fold bifurcation. Notice that one of the critical paths with a high robustness value (white path) suddenly drops its robustness due to the partner switch near the bifurcation.

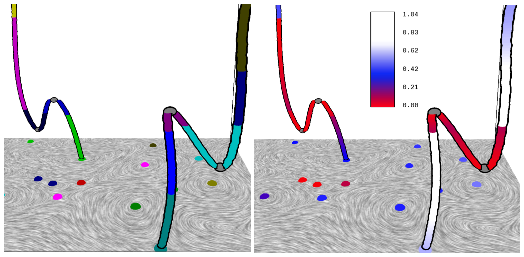


Figure 4: *A critical point trajectory highlighting the switch in static robustness pairing near birth-death points.*

For a connected component C in the sublevel set $\mathbb{F}_r$, it is possible to define the static (dynamic) robustness of C as the maximum static (dynamic) robustness of critical points it contains. We believe such a definition can help us understand the stability of a sub-domain of a stationary vector field, and for a sub-volume in the time-varying setting. For example, in Figure 5, for the Combustion dataset, we illustrate the dynamic robustness of connected components in the sublevel sets where $r = 0.1$, and observe its evolution over time. The corresponding volume is rendered with partial transparency on top of our critical point tube visualization. Here, red indicates low robustness volume which coincide with places where low robustness critical points move towards each other, such as regions surrounding fold and blue sky bifurcations. The visualization highlights situations which could be interesting for further investigation.

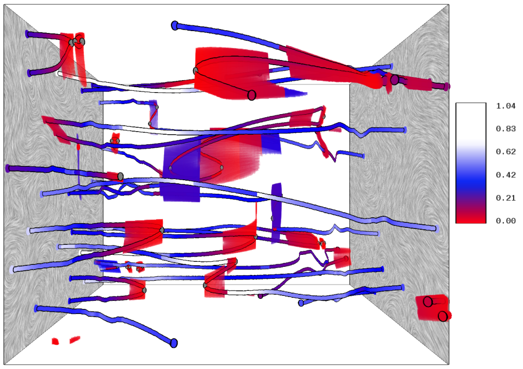


Figure 5: Combustion. *Dynamic robustness of components in the sublevel sets.*

Central Atlantic dataset. Recall that in oceanography, *eddies* [Rob83] are considered a dominant physical phenomenon in understanding ocean circulation. Eddies in the ocean can appear at different time scales, from short-lived ones (we call *small-scale* features) to long-lived ones (*large-scale* features). Eddies have closed or semi-closed circulation patterns, and in 2D they could be modeled as *vortices* whose detection, visualization and simplification [ZM95, SP99, TSH00b] play a central role in studying features of fluid flows. The eddy centers could be modeled as vortex centers, whose trajectory over time are the critical paths. In our framework, the migration and interaction between eddies over time and space are modeled as critical paths. We demonstrate that robustness assignments along critical paths for CentralAtlantic and SouthAtlantic datasets help us quantify the stability of small and large-scale eddy centers.

For the CentralAtlantic dataset, we first illustrate its global view in Figure 8 (top left) for static robustness assignments along the critical paths. There appear to be few

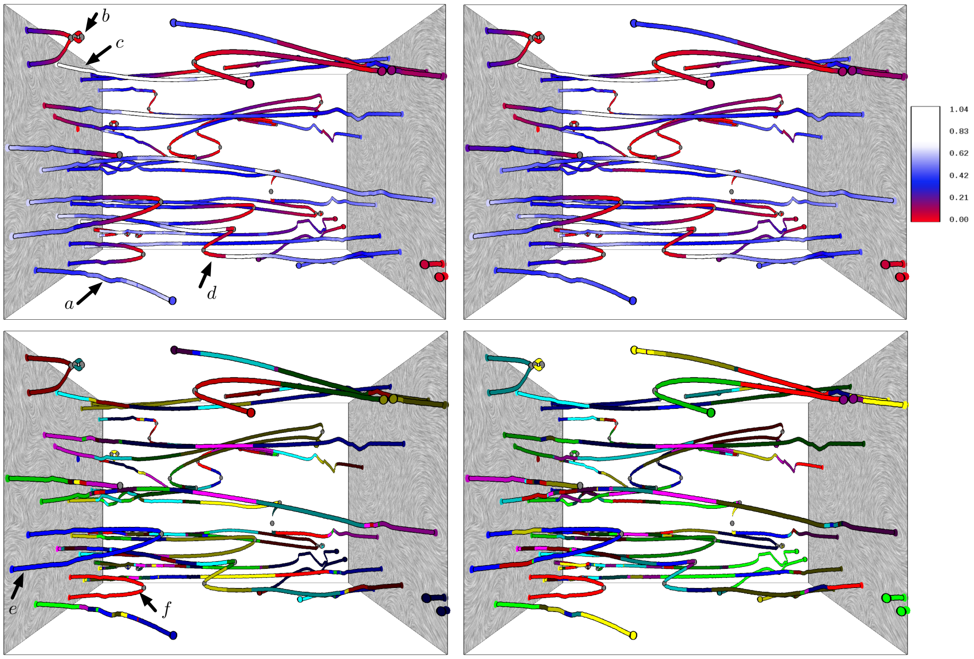


Figure 6: Combustion. *Top: robustness assignment along critical paths, for static (left) and dynamic (right) robustness. Bottom: robustness partners colored by unique values showcasing partner switches, for static (left) and dynamic (right) robustness.*

differences between static and dynamic robustness assignments. For simplicity, we illustrate our examples based on static robustness alone. The complete global view including dynamic robustness assignments are shown in the supplemental material. The critical paths in this example are not as smooth as Combustion due to their lower resolution. Also notice the dataset has a strong shearing effect in the sense that all critical paths move in roughly the same direction.

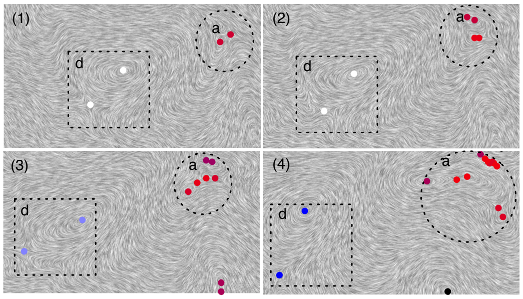


Figure 7: CentralAtlantic. *Regions (a) and (d) are in correspondence with Figure 8 top (a) and (d). From (1) to (4), time slices* 29*,* 42*,* 51 *and* 74*.*

Now we discuss several selected local features as shown in Figure 8 (top right), which have their rough boundaries illustrated in Figure 8 (top left). First, we focus on critical paths in a. Early in the time sequence, a large number of critical points appear in an isolated region with low robustness. Over a short period of time, these critical points appear, move, and disappear. They are characterized by a spaghetti-like bundle in 3D and a visual clutter of critical points in 2D (i.e., Figure 7 time slice 74). These small-scale features have low robustness indicating fluctuations of the currents that are highly unstable with respect to perturbations. As time moves forward, the fluctuations settle down as two critical paths in the flow emerge gradually over time (i.e. Figure 8 (top) b). These two critical paths show high robustness and they pair for a significant amount of time. This indicates a stable large-scale feature that is robust to perturbations. They eventually move into an area where new fluctuation is formed and disappear into a fold bifurcation. Second, critical paths e in Figure 8 (top) represent another example of how stable features evolve within unstable features. However, in this example two critical points with low robustness pair for a long period of time. They correspond to large-scale eddies residing in fluctuating regions. Over time, these critical points become stable as their robustness increases by eventually switching partners. Third, in both critical paths c and d in Figure 8 (top), we see large-scale features with high robustness, indicating stability. In addition, we observe in particular a pair of critical points in d that gradually decrease in robustness values although their Euclidean distance remains close. This indicates a gradual decrease in strength (vector magnitude of current velocity) among the neighborhood of these critical points. The 2D view of such a phenomenon is shown in Figure 7 with white color indicating high robustness.

South Atlantic dataset. The global view of SouthAtlantic dataset in terms of static robustness is shown in Figure 8 (bottom left). Some individual features are highlighted in Figure 8 (bottom right) with a slightly rotated viewing angle. Here critical paths a form a blue-sky bifurcation followed by a fold-bifurcation. The two critical points involved

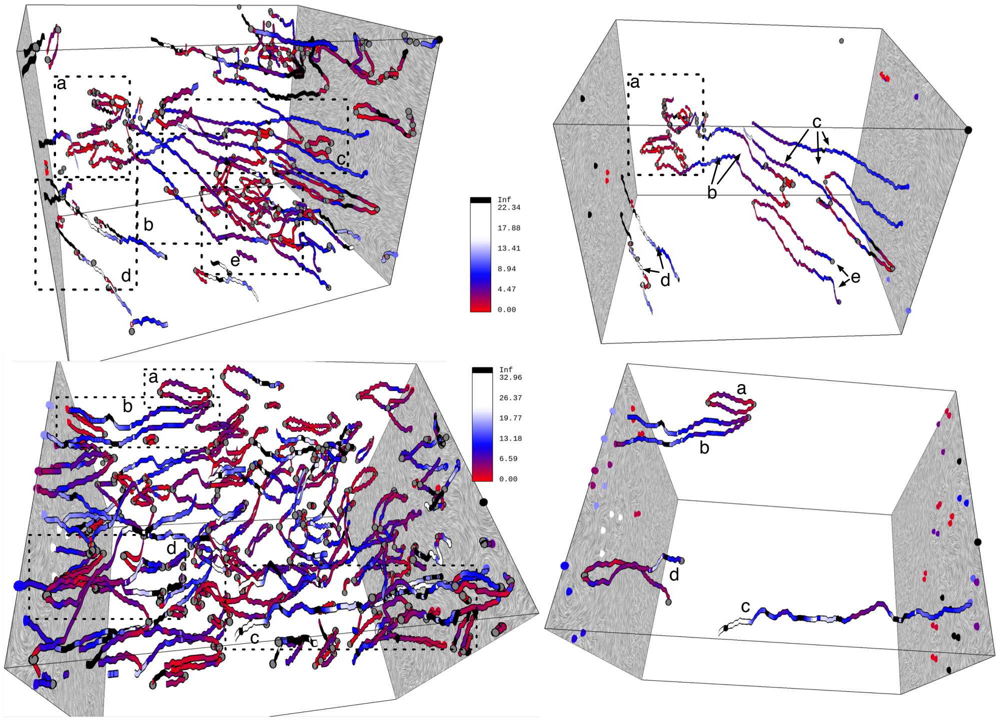


Figure 8: *Top:* CentralAtlantic. *Bottom:* SouthAtlantic. *Left: static robustness assignment along critical paths. Local features are highlighted. Right: chosen individual features.*

become stable for a relatively short period of time before beginning to fluctuate. Critical paths *b* showcase two relative large-scale features with high robustness for a long period of time before disappearing into a fold bifurcation. Similarly, critical path *d* illustrates the reverse scenario. Critical path *c* corresponds to a large-scale feature that is stable most of the time with respect to perturbations. For completeness, the global view of SouthAtlantic dataset for both static and dynamic robustness including the pairing switches are shown in the supplemental material.

7. Conclusion

We believe the robustness framework to be an important addition in quantifying the stability of critical points. Based on our experimental results, we have seen how robustness can help us separate features from noise, and explore the data in a manner meaningful in the particular application. Sometimes we may consider high robustness critical points, which are provably robust to perturbations, as features, while they may become undesirable in other situations. Based on robustness measure, we could potentially develop strategies that maintain compact representations of the datasets by keeping only the essential features for efficient analysis and uncluttered visualization. Finally we would like to note that the theory and computation of static and dynamic robustness generalize naturally to time-varying vector field in arbitrary dimensions. Visualizing robustness for higher dimensional (3D+) time-varying vectors fields remains an open problem.

Visual clutter. As shown in the result section, as the size of the data increases, we would observe increasing level of visual clutter in the visualization of the critical paths. Interaction and filtering are the natural approaches to dealing with clutter. The user can select critical paths to include/exclude from the visualization. Filtering (i.e. [BPMS12]) could also be used on the robustness of critical paths by only including those whose robustness fall below a desired range. We also believe that certain visual clutter gives insight, e.g., a group with low robustness paths indicates eddies or other instability. Addressing the general scalability issue of visual interface with respect to big data remains a challenge.

Acknowledgments

We thank Jackie Chen for the combustion dataset. We also thank Mathew Maltude from the Climate, Ocean and Sea Ice Modelling program at Los Alamos National Laboratory (LANL) and the BER Office of Science UV-CDAT team for providing us the ocean datasets. We thank Amit Patel and Guoning Chen for insightful discussions. PR was supported in part by grants from DOE NETL and KAUST award KUS-C1-016-04. PS was supported by EU project TOPOSYS (FP7-ICT-318493-STREP). This work is supported in part by NSF OCI-0906379, NSF OCI-0904631, DOE/NEUP 120341, DOE/MAPD DESC000192, DOE/LLNL B597476, DOE/Codesign P01180734, and DOE/SciDAC DESC0007446.

References

[BCP*12] BRAMBILLA A., CARNECKY R., PEIKERT R., VIOLA I., HAUSER H.: Illustrative flow visualization: State of the art, trends and challenges. *Eurographics STAR* (2012), 75–94. 2

[BEMP13] BENDICH P., EDELSBRUNNER H., MOROZOV D., PATEL A.: Homology and robustness of level and interlevel sets. *Journal of HHA* (2013), 51–72. 2

[BGW*13] BHATIA H., GYULASSY A., WANG H., BREMER P.-T., PASCUCCI V.: Robust detection of singularities in vector fields. TopoInVis (accepted), 2013. 5

[BPMS12] BORN S., PFEIFLE M., MARKL M., SCHEUERMANN G.: Visual 4D MRI blood flow analysis with line predicates. *IEEE Pacific Visualization Symposium* (2012), 105– 112. 9

[CL93] CABRAL B., LEEDOM L.: Imaging vector fields using line integral convolution. *SIGGRAPH* (1993), 263–270. 5

[CLRS09] CORMEN T. H., LEISERSON C. E., RIVEST R. L., STEIN C.: *Introduction To Algorithms*. MIT Press, 2009. 5

[CPS11] CHAZAL F., PATEL A., SKRABA P.: Computing the robustness of roots. http://ailab.ijs.si/primoz_skraba/papers/fp.pdf, 2011. 2, 4

[CPS12] CHAZAL F., PATEL A., SKRABA P.: Computing well diagrams for vector fields on R^n. *Applied Math. Letters 25*, 11 (2012), 1725–1728. 2, 3

[CSA00] CARR H., SNOEYINK J., AXEN U.: Computing contour trees in all dimensions. *SoDA* (2000), 918–926. 3, 5

[CZ07] CARLSSON G., ZOMORODIAN A.: The theory of multidimensional persistence. *Proc. of SoCG* (2007), 184–193. 2

[dBvK93] DE BERG M., VAN KREVELD M.: Trekking in the alps without freezing or getting tired. *LNCS 726* (1993), 121–132. 4

[DW07] DEY T. K., WENGER R.: Stability of critical points with interval persistence. *DCG 38* (2007), 479–512. 2

[ELZ02] EDELSBRUNNER H., LETSCHER D., ZOMORODIAN A.: Topological persistence and simplification. *DCG 28* (2002), 511–533. 2

[EM90] EDELSBRUNNER H., MÜCKE E. P.: Simulation of simplicity: a technique to cope with degenerate cases in geometric algorithms. *ACM Trans. on Graphics 9*, 1 (1990), 66–104. 5

[EMP10] EDELSBRUNNER H., MOROZOV D., PATEL A.: The stability of the apparent contour of an orientable 2-manifold. In *Top. Methods in Data Analysis and Vis.* 2010, pp. 27–41. 2

[EMP11] EDELSBRUNNER H., MOROZOV D., PATEL A.: Quantifying transversality by measuring the robustness of intersections. *FoCM 11* (2011), 345–361. 2, 3, 4

[GP74] GUILLEMIN V., POLLACK A.: *Differential Topology*. Prentice-Hall, Inc., Englewood Cliffs, NJ, USA, 1974. 4

[Hat02] HATCHER A.: *Algebraic Topology*. Cambridge Univ. Press, 2002. 2

[HH91] HELMAN J. L., HESSELINK L.: Visualizing vector field topology in fluid flows. *IEEE CG&A 11* (1991), 36–46. 1

[HSPC06] HAWKES E. R., SANKARAN R., PÉBAY P. P., CHEN J. H.: Direct numerical simulation of ignition front propagation in a constant volume with temperature inhomogeneities: II. Parametric study. *Combustion and Flame 145* (2006), 145–159. 6

[KHNH11] KASTEN J., HOTZ I., NOACK B. R., HEGE H.-C.: On the extraction of long-living features in unsteady fluid flows. In *Top. Meth. in Data Analysis and Vis.* (2011), pp. 115–126. 2

[LHD*04] LARAMEE R. S., HAUSER H., DOLEISCH H., VROLIJK B., POST F. H., WEISKOPF D.: The state of the art in flow visualization: Dense and texture-based techniques. *Comput. Graph. Forum 23* (2004), 203–221. 2

[MBP10] MALTRUD M., BRYAN F., PEACOCK S.: Boundary impulse response functions in a century-long eddying global ocean simulation. *Environmental Fluid Mech. 10* (2010), 275–295. 6

[MLP*10] MCLOUGHLIN T., LARAMEE R. S., PEIKERT R., POST F. H., CHEN M.: Over two decades of integration-based, geometric flow visualization. *Comput. Graph. Forum 29*, 6 (2010), 1807–1829. 2

[PL09] PENG Z., LARAMEE R. S.: Higher dimensional vector field visualization: A survey. *Proc. EG UK Theory and Practice of Comp. Graphics* (2009), 149–163. 2

[PP03] POLTHIER K., PREUSS E.: Identifying vector fields singularities using a discrete hodge decomposition. In *Visualization and Mathematics III*. 2003, pp. 112–134. 2

[PVH*03] POST F. H., VROLIJK B., HAUSER H., LARAMEE R. S., DOLEISCH H.: The state of the art in flow visualisation: Feature extraction and tracking. *Comput. Graph. Forum 22* (2003), 775–792. 2, 5

[RKG*11] REININGHAUS J., KOTAVA N., GUENTHER D., KASTEN J., HAGEN H., HOTZ I.: A scale space based persistence measure for critical points in 2d scalar fields. *IEEE TVCG* (2011). 2

[RKWH11] REININGHAUS J., KASTEN J., WEINKAUF T., HOTZ I.: Efficient computation of combinatorial feature flow fields. *IEEE TVCG* (2011). 5

[RLH11] REININGHAUS J., LOWEN C., HOTZ I.: Fast combinatorial vector field topology. *IEEE TVCG 17*, 10 (2011), 1433–1443. 2

[Rob83] ROBINSON A. R. (Ed.): *Overview and Summary of Eddy Science, Eddies in Marine Science*. Springer-Verlag, New York, NY, USA, 1983, pp. 3–15. 7

[SKMR98] SCHEUERMANN G., KRUEGER H., MENZEL M., ROCKWOOD A.: Visualizing nonlinear vector field topology. *IEEE TVCG 4* (1998), 109–116. 2

[SP99] SADARJOEN I. A., POST F. H.: Geometric methods for vortex extraction. *IEEE TVCG Symp. on Vis.* (1999), 53–62. 7

[SW13] SKRABA P., WANG B.: Interpreting feature tracking through the lens of robustness. *TopoInVis (accepted)* (2013). 5

[TS03] THEISEL H., SEIDEL H. P.: Feature flow fields. *Proceedings Symposium on Data Visualisation* (2003), 141–148. 5

[TSH00a] TRICOCHE X., SCHEUERMANN G., HAGEN H.: Higher order singularities in piecewise linear vector fields. In *The Mathematics of Surfaces IX*. Springer, 2000, pp. 99–113. 2

[TSH00b] TRICOCHE X., SCHEUERMANN G., HAGEN H.: A topology simplification method for 2d vector fields. *The Mathematics of Surfaces IX* (2000), 99–113. 7

[TSH01] TRICOCHE X., SCHEUERMANN G., HAGEN H.: Continuous topology simplification of planar vector fields. *Proceedings IEEE Visualization* (2001), 159–166. 2

[TWSH02] TRICOCHE X., WISCHGOLL T., SCHEUERMANN G., HAGEN H.: Topology tracking for the visualization of time-dependent two-dimensional flows. *Computers & Graphics 26* (2002), 249–257. 5

[WE05] WEISKOPF D., ERLEBACHER G.: Overview of flow visualization. In *The Visualization Handbook*. Elsevier, 2005, pp. 261–278. 2

[Wei08] WEINKAUF T.: *Extraction of topological structures in 2D and 3D vector fields*. PhD thesis, Univ. Magdeburg, 2008. 2

[ZM95] ZHU Z., MOORHEAD R. J.: Extracting and visualizing ocean eddies in time-varying flow fields. *Proc. of Conf. on Flow Vis.* (1995). 7