



# Unsupervised extraction of rotational Lagrangian coherent structures

Marius M. Neamtu-Halic<sup>a,c,e</sup>, Stefano Brizzolara<sup>a,c,f</sup>, George Haller<sup>b</sup>, Markus Holzner<sup>c,d,e</sup>

<sup>a</sup> Institute of Environmental Engineering, ETH Zürich, CH-8039 Zürich, Switzerland

<sup>b</sup> Institute of Mechanical Systems, ETH Zürich, CH-8092, Zürich, Switzerland

<sup>c</sup> Swiss Federal Institute of Forest, Snow and Landscape Research WSL, 8903 Birmensdorf, Switzerland

<sup>d</sup> Swiss Federal Institute of Aquatic Science and Technology Eawag, 8600 Dübendorf, Switzerland

<sup>e</sup> Institute of Hydraulic Engineering and River Research, University of Natural Resources and Life Sciences, BOKU Wien, 1190 Vienna, Austria

<sup>f</sup> Institute of Science and Technology Austria (ISTA), 3400 Klosterneuburg, Austria

## ARTICLE INFO

Dataset link: <https://github.com/NeamtuMarius/Unsupervised-3D-LAVD-Extraction-Algorithm>

### Keywords:

Vortex identification methods

Lagrangian coherent structures

Turbulent flows

## ABSTRACT

Lagrangian coherent structures (LCSs) are widely recognized as playing a significant role in turbulence dynamics since they can control the transport of mass, momentum or heat. However, the methods used to identify these structures are often based on ambiguous definitions and arbitrary thresholding. While LCSs theory provides precise and frame-indifferent mathematical definitions of coherent structures, some of the commonly used extraction algorithms employed in the literature are still case-specific and involve user-defined parameters. In this study, we present a new, unsupervised extraction algorithm that enables the extraction of rotational LCSs based on Lagrangian average vorticity deviation from an arbitrary 3D velocity field. The algorithm utilizes two alternative methods for the identification of the LCS core (ridge): an unsupervised clustering method and a streamline-based method. In a subsequent step, the ridge curve is parametrized through a pruning procedure of minimum spanning tree graphs. To assess the effectiveness of the algorithm, we test it on two cases: (i) direct numerical simulations of forced homogeneous and isotropic turbulence and (ii) three-dimensional Particle Tracking Velocimetry experiments of a turbulent gravity current.

## 1. Introduction

Over the last decades, coherent flow structures have received considerable attention in the fluid dynamics community. The principal reason behind this interest stems from the crucial role that these structures play in the transport of mass and momentum in turbulent flows [1]. A typical example is that of elongated streamwise vortical structures that form in the wall-proximity of bounded or semi-bounded turbulent flows, such as boundary layers [2–4], pipe flows [5,6] and channel flows [7,8]. These vortical structures have a fundamental contribution to the wall-normal exchange of momentum, setting thereby the average streamwise velocity profile in the near-wall region and thus the turbulent drag over the solid boundary. Consequently, accurate detection of vortical structures is essential, for example, for the implementation of flow control techniques of turbulent drag reduction strategies [9–11]. Coherent flow structures also populate the boundary of turbulent jets [12–15], mixing layers [16–19] and gravity currents, where they impact the entrainment and mixing of external fluid into the turbulent flow region [20–22]. Detecting vortical structures in these flows can thus inform the spread of passive substances [23], for example. In a more fundamental setting, coherent flow structures

have been studied in homogeneous and isotropic turbulent flows as well, with the goal to understand the energy transfer across length scales [24].

A variety of extraction methods have been developed to identify coherent structures, such as the Q- [25],  $\Delta$ - [26] and  $\lambda_2$ -criterion [27]. However, most of these methods lack a fundamental requirement of mechanics for analyzing material behavior: objectivity, i.e., indifference to moving and rotating observers [28]. The theory of Lagrangian coherent structures (LCSs) theory overcomes this objectivity problem by defining hyperbolic (attracting and repelling) and elliptic (vortical) coherent structures in a material and hence frame-invariant fashion [29,30]. While these definitions and the mathematical results building on them are unambiguous, the numerical implementation of LCS-based extraction algorithms in three dimensions (3D) remains elusive. Commonly, the main approaches of 3D LCS-based algorithms assume an a priori knowledge of the structures' location and orientation to extract them [31,32].

Recently, a fully automatized and frame-indifferent algorithm based on the Lagrangian averaged vorticity deviation (LAVD) method [31,33] was introduced by Neamtu-Halic et al. [20]. The authors successfully

\* Corresponding author at: Institute of Environmental Engineering, ETH Zürich, CH-8039 Zürich, Switzerland.  
E-mail address: [nemarius@ethz.ch](mailto:nemarius@ethz.ch) (M.M. Neamtu-Halic).

extracted rotational LCSs from 3D LAVD fields without prior knowledge of the structures' orientation. However, their method is based upon several parameters that need to be provided by the user and require tuning depending on the investigated flow and data quality at hand.

A common impediment to rotational LCS algorithms is connected to finding the centerlines of the vortical structures in 3D. These 3D curves are defined as sets of local maxima in planes orthogonal to the centerlines [20]. In other words, the centerlines are codimension-two ridges of a diagnostic 3D scalar field. No algorithm for extracting such curves without user-predefined parameters is available in the literature. Here, we implement a new algorithm to determine the position of the 3D centerline of rotational LCSs based on a set of tolerances with well-defined ranges, bounded by asymptotic values of explicit connotation.

Specifically, we provide a fully automated algorithm for the education of 3D rotational LCSs based on a unambiguous set of tolerance values that eliminate the arbitrariness of the user-dependent parameters typical of other available algorithms. The input to the algorithm is a time-resolved 3D velocity field. Here we focus on rotational LCSs based on the LAVD principle as a proof of concept. The applicability of the method is, however, more general and can be implemented for other LCS definitions. For example, the recently introduced active barriers [30,34], defined as surfaces minimizing the diffusive transport of an active vector field provide a pseudo velocity field that is amenable to the same rotational LCS extraction approach. We discuss two flow examples to show the tolerance values used to educe the rotational LCSs with the intent to familiarize the user with the algorithm. Together with the details of the algorithm introduced in the next sections of the paper, we provide an open-source MATLAB code of the algorithm ([github.com/NeamtuMarius/Unsupervised-3D-LAVD-Extraction-Algorithm](https://github.com/NeamtuMarius/Unsupervised-3D-LAVD-Extraction-Algorithm)).

This work is organized as follows. In Section 2, we describe the different steps of the algorithm. In Section 3, we show the applicability of the algorithm to three different data sets, and we draw our conclusions in Section 4.

## 2. Algorithm

In the following, we describe a procedure to educe elliptic LCSs, hereinafter referred to as ‘‘rotational’’ LCSs, from 3D flow data. First, we briefly introduce the computation of the LAVD field from 3D Eulerian data. Then, we describe and compare two alternative procedures to identify points that belong to ridges of the LAVD field. The first of these methods is based on a modified gradient ascend algorithm applied to the gradient of the LAVD field, while the second method uses a fast computation of streamlines of the same field. Subsequently, we apply an unsupervised clustering algorithm on the ridge point cloud to distinguish between different LAVD ridges. Eventually, through a procedure that we call *pruning*, we select and sort ridge points, in order to parametrize each LAVD ridge as a 1D curve in 3D. We then apply a spline interpolation to this curve to obtain the tangent vectors along LAVD ridges.

### 2.1. LAVD calculation

To compute the LCSs, as a first step, the LAVD field is computed from 3D time-resolved Eulerian velocity data [31]. To this end, fluid particle trajectories are computed by integrating the equation of motion

$$\frac{d\mathbf{x}}{dt} = \mathbf{v}(\mathbf{x}, t), \quad (1)$$

with  $\mathbf{x}$  denoting the fluid particles position and  $\mathbf{v}$  the velocity field. Once fluid trajectories are computed, the LAVD field introduced by [31] is obtained as

$$\text{LAVD}_{t_0}^T(\mathbf{x}_0) := \frac{1}{T} \int_{t_0}^{t_0+T} \left| \boldsymbol{\omega}(\mathbf{x}(s; \mathbf{x}_0), s) - \bar{\boldsymbol{\omega}}(s) \right| ds \quad (2)$$

by integrating the norm of the vorticity deviation  $\left| \boldsymbol{\omega}(\mathbf{x}(t; \mathbf{x}_0), t) - \bar{\boldsymbol{\omega}}(t) \right|$  over the extraction time  $T$  using a fourth-order Runge–Kutta method [31]. The vorticity deviation is obtained as the difference between the vorticity along trajectories  $\boldsymbol{\omega}$  and the instantaneous spatial average of the vorticity  $\bar{\boldsymbol{\omega}}$ .

For a given initial time  $t_0$  and integration time  $T$ , the Lagrangian vorticity field is computed and stored in three (one for each component of the vorticity vector) 4D matrices. The first three dimensions represent the spatial distribution of the vorticity field, while the fourth represents time. For each time, the spatial average of the vorticity vector  $(\bar{\boldsymbol{\omega}}(s))$  in Eq. (2) is calculated by averaging each of the 4D matrices over the first three dimensions. Then, one unique 4D matrix corresponding to the norm of the vorticity deviation  $\left| \boldsymbol{\omega}(\mathbf{x}(t; \mathbf{x}_0), t) - \bar{\boldsymbol{\omega}}(t) \right|$  is computed. The integration procedure is performed by summing up  $\left| \boldsymbol{\omega}(\mathbf{x}(t; \mathbf{x}_0), t) - \bar{\boldsymbol{\omega}}(t) \right| ds$  and dividing by the integration time. By storing the cumulative sum of the LAVD matrix and dividing by the current time, one can obtain all the LAVD fields with extraction time smaller than or equal to  $T$ . This latter procedure is useful for studying the sensitivity of the results to the extraction time.

As an example, Fig. 1(a) shows the LAVD field computed from 128<sup>3</sup> fluid trajectories released at a given instant from a cubic grid mesh of a sub-volume in a turbulent flow. The data is from the forced, homogeneous and isotropic turbulent flow case taken from the John Hopkins Turbulence (JHT) database [35]. The grid size corresponds to  $2\eta$ , where  $\eta = (\nu^3/\epsilon)^{1/4}$  is the Kolmogorov length scale with  $\nu$  denoting the fluid kinematic viscosity and  $\epsilon$  the average dissipation rate of the turbulent kinetic energy. The integration time  $T$  is equal to  $0.85\tau_\eta$ , where  $\tau_\eta = (\nu/\epsilon)^{1/2}$  is the Kolmogorov time scale. To have an impression of the intricate structure of the 3D LAVD field, in Fig. 1(b) we show isosurfaces corresponding to the 99th percentile of the LAVD magnitude inside the cube. Fig. 1(c) illustrates how the integration time affects the resulting LAVD field. With an increasing integration time window, specific regions with strong LAVD magnitude retain their coherence (right panel), while other areas experience a greater level of diffusion.

### 2.2. Ridge detection

The core of the procedure we propose here to identify vortical structures in 3D is the identification of the center of the vortices. The set of points constituting these centers is a curve in the 3D space, which Haller et al. [31] identified with the innermost (maximum) member of an LAVD level surface family. Although the center of the structure has a precise definition [31], its identification is a complex problem in turbulent data sets. Indeed, unlike in some idealized flows, the LAVD is generally not constant along the center line of vortical LCSs in turbulent flows [20].

As a consequence, the requirement that a vortex core is an LAVD level set must be relaxed and a different property of vortex center has to be employed. Similarly to Neamtu et al. [20], we define the core of the vortex as a codimension-two ridge of the LAVD field. By the definition of a height ridge, points along a ridge of a scalar field are local maxima within planes normal to the ridge at those points. In other words, the points along ridges have a relatively low rate of change along the ridge curve compared to the rapid decrease in the radial direction along planes locally normal to ridge curve. Due to its implicit nature, this property is challenging to utilize as it depends on local properties of a priori unknown curves. To overcome this challenge, in the following, we introduce two different algorithms.

#### 2.2.1. Gradient climbing

The first algorithm for ridge detection is based on the gradient climbing of the  $\nabla \text{LAVD}_{t_0}^T$  field. Such a basis was used in the past for other algorithms [20,21], but the implementation relied on user-dependent parameters. To avoid this arbitrariness, an independent

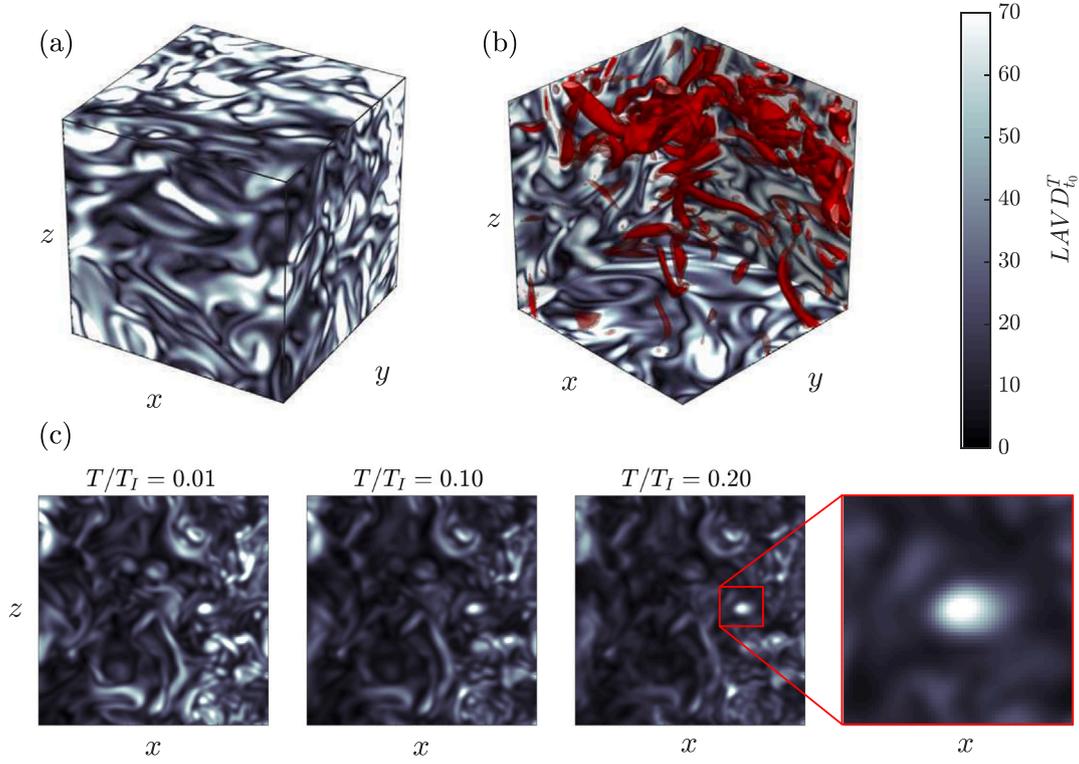


Fig. 1. LAVD fields extracted from sub-volumes of the JHT dataset. Panel (a) LAVD on the surface of a cube of 128 grid points corresponding to 256 Kolmogorov lengths; the extraction time is 0.02 integral time scales  $T_I$  corresponding to 0.85 Kolmogorov times and 20 time-steps. Panel (b) shows the isosurfaces of LAVD corresponding to the 99th percentile of the LAVD magnitude inside the cube. Panel (c): LAVD extracted on a volume slice of 128 grid points for different value of the extraction time.

stopping condition for ridge detection is introduced in this work. Once the potential ridge point candidates are identified, an unsupervised clustering algorithm is employed to classify the points in different ridges.

Let us denote a ridge point location with  $\mathbf{x}^r$ . To start the climbing algorithm along the  $\nabla \text{LAVD}_{i_0}^T$  field, trajectories are initialized at  $\mathbf{x}_0^r$ . For the initial position  $\mathbf{x}_0^r$ , all points on the Eulerian grid of  $\nabla \text{LAVD}_{i_0}^T$  can be selected. However, to save the computational cost, a threshold on the norm of  $\nabla \text{LAVD}_{i_0}^T$  was imposed to select grid points characterized by a high steepness of the LAVD field. This procedure reduces the computational costs, without considerably impacting the final results.

The ridge point locations are thus determined by the following gradient climbing routine:

$$\mathbf{x}_{n+1}^r = \mathbf{x}_n^r + \gamma \nabla \text{LAVD}_{i_0}^T(\mathbf{x}_n^r), \quad n \geq 1, \quad (3)$$

where  $\gamma > 0$  is the step size. A suitable choice for the step size is  $\gamma = 1/L$ , where  $L$  is the Lipschitz constant of the gradient, defined as

$$\left| \nabla \text{LAVD}_{i_0}^T(\mathbf{x}) - \nabla \text{LAVD}_{i_0}^T(\mathbf{x}') \right| \leq L |\mathbf{x} - \mathbf{x}'|, \quad \mathbf{x}, \mathbf{x}' \in U \quad (4)$$

to prevent oscillations of  $\mathbf{x}^r$  over the whole domain  $U$ . The constant  $L$  can be selected as the largest eigenvalue of the Hessian matrix  $\nabla^2 \text{LAVD}_{i_0}^T(\mathbf{x})$  of the LAVD field. We adopt as sufficient condition,

$$L = C_1 \max_{\mathbf{x} \in U} \left( \max_{i=1,2,3} \text{eig}_i \left( \nabla^2 \text{LAVD}_{i_0}^T(\mathbf{x}) \right) \right) \quad (5)$$

where  $\text{eig}_i$  denotes the  $i$ th eigenvalue and  $C_1 \geq 1$ .

The gradient climbing algorithm needs an appropriate stopping condition and in the case of 1D ridges, we adopt the following procedure: for each iteration step  $n$ , we evaluate the binary field  $B_{i,j,k}^n$  of the ridge's edge position by binning the 3D space with boxes of edge length  $l = \min(dx, dy, dz)$ , where  $(dx, dy, dz)$  is the velocity field resolution, i.e., the distance between adjacent grid points in  $x$ ,  $y$  and  $z$  direction, respectively.  $B_{i,j,k}^n = 1$  if there are any ridge points in the box, else it

is zero. We evaluate the mean-square error between two subsequent iterations by defining

$$\text{MSE}^n = \sum_{i,j,k} \left( B_{i,j,k}^n - B_{i,j,k}^{n+1} \right)^2. \quad (6)$$

The algorithm stops when  $\text{MSE}^n \leq C_2 \text{MSE}^{n-1}$  with  $C_2 \approx 0$ .

Fig. 2 shows the results of the gradient climbing algorithm for the cube of the HIT flow field shown in Fig. 1. To compute the gradient of the LAVD field, a central difference scheme is adopted, though the implementation of higher-order schemes is also possible if greater accuracy is required. The upper panels show the ridge point positions for four intermediate iteration steps, including the final iteration ( $n = 2500$ ). As seen from the first panel of Fig. 2, although initially grouped due to the selection criterion discussed above, the points are rather sparse in space. For this specific case, the computational procedure was initialized at grid points with  $\nabla \text{LAVD}_{i_0}^T > \text{perc}_{80\%}(\nabla \text{LAVD}_{i_0}^T)$ , where  $\text{perc}_{80\%}(\nabla \text{LAVD}_{i_0}^T)$  is the 80th percentile of  $\nabla \text{LAVD}_{i_0}^T$ . As the algorithm progresses the points begin to accumulate on the LAVD ridges (compare the first three figures in panel a of Fig. 2), while no significant change can be seen between the last two iterations shown in panel (a).

To quantify the change between subsequent iterations, the  $\text{MSE}^n$  is shown in Fig. 2(b) against the iteration number  $n$ . Note that the  $\text{MSE}^n$  has a fast initial decrease with respect to  $n$ , while gradually decreasing more slowly. The algorithm stops when a 10% tolerance with respect to the initial value of  $\text{MSE}^n$  at  $n = 1$  is reached (dashed line in Fig. 2b). To test if an appropriate choice of the tolerance was made, the change of  $\text{MSE}^n$  between subsequent iterations,  $\Delta \text{MSE}^n$  is shown in Fig. 2(c). As seen in the figure, after approximately 1800 iterations,  $\Delta \text{MSE}^n$  fluctuates around 0, thus changes in  $\text{MSE}^n$  are relatively small.

### 2.2.2. Streamline method

We now discuss a second method to compute the centerline of the vortical structures, hereinafter called the *streamline* method. This

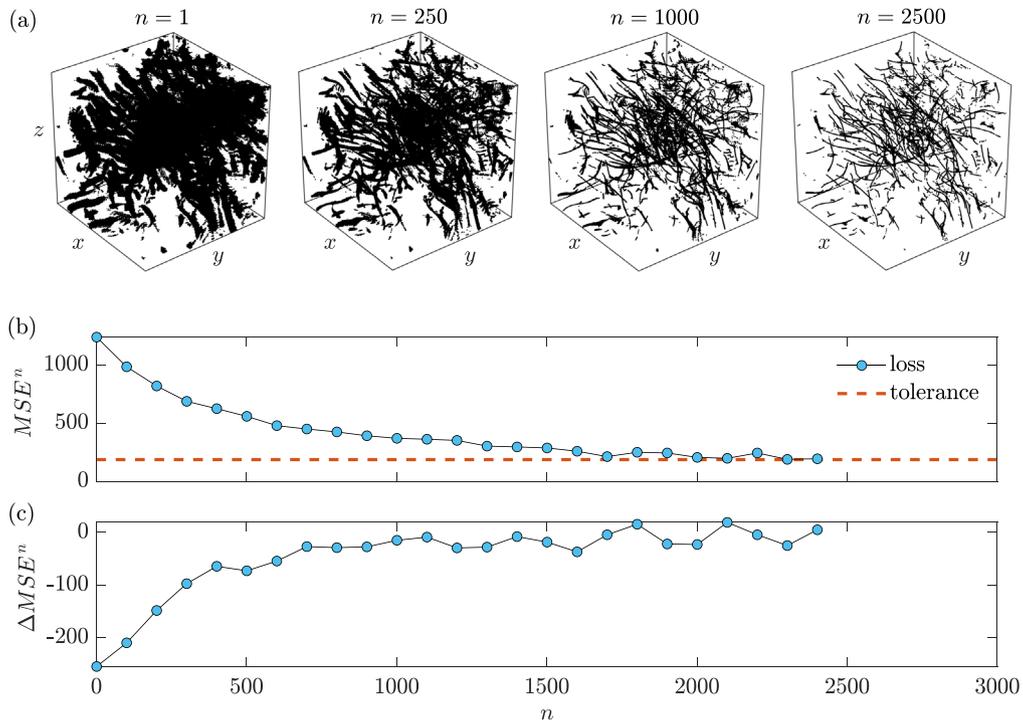


Fig. 2. Illustration of the gradient climbing routine for the ridge detection. The gradient is climbed starting from suitable starting points (panel (a),  $n = 1$ ), by iterating Eq. (3) until the desired stopping condition is met. Panel (a) shows a snapshot of the point positions across iterations; panel (b) shows the  $MSE^n$ , while panel (c) shows the increment of  $MSE^n$  between two subsequent iterations, namely the first order derivative of the loss function.

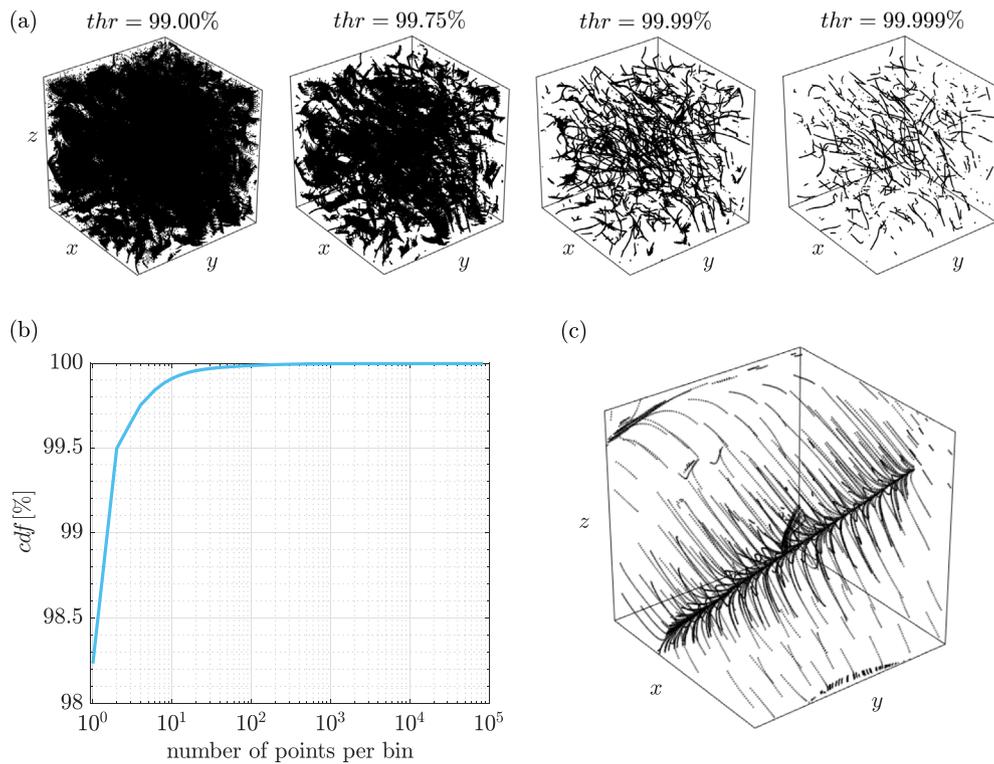
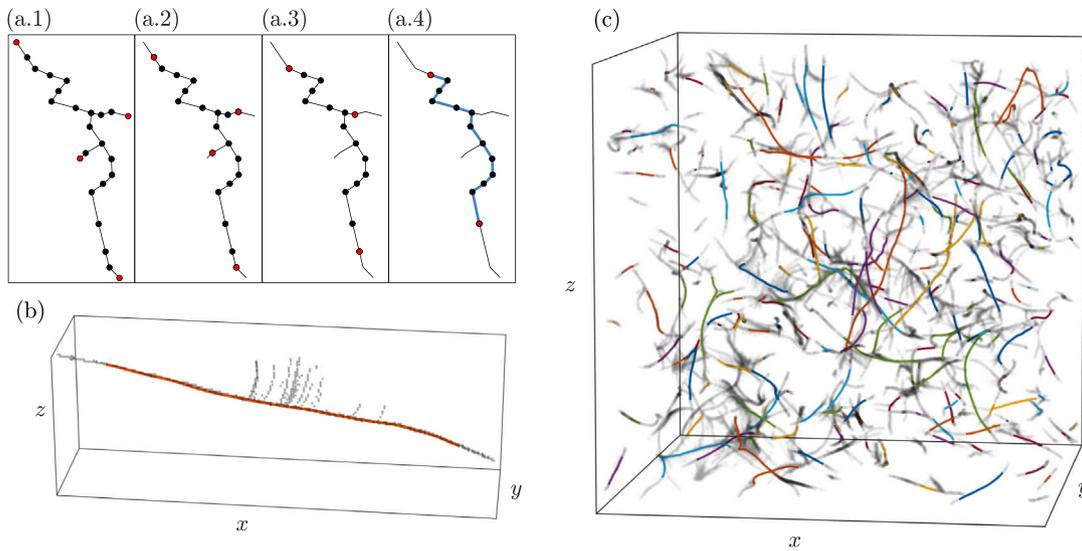


Fig. 3. Illustration of the streamline method for ridge detection. Panel (a): ridges for different values  $thr$  of the points-per-bin cumulative density function of panel (b). Panel (c): zoom-in on an accumulation region of the streamlines.

method is based on the streamlines of the  $\nabla \text{LAVD}_{t_0}^T$  vector field. While similar to the algorithm discussed in Section 2.2.1, the streamline

algorithm requires no iterations. It is based on a two-step procedure. As a first step, the streamlines are computed from the  $\nabla \text{LAVD}_{t_0}^T$  field.



**Fig. 4.** Pruning the minimum spanning tree; after building the minimum spanning tree graph (a), the vertices that have only one connection are removed (b); the process is iterated until only two vertices have only one connection, namely the first and last vertices (c).

To initialize the streamlines computing procedure, the same grid points of the  $LAVD_{t_0}^T$  vector field are employed. The integration step can be set similarly to what is described in Section 2.2.1, and the total integration time is prolonged until all the streamlines reach the domain boundaries. Subsequently, the 3D space is subdivided into bins and box-counting is applied to determine the number of streamline points per bin. The rationale behind the method stems from the accumulation of streamlines along ridges, which implies a higher number of streamline points observed along the ridge lines than elsewhere. This can be observed in Fig. 3(c).

To identify LAVD ridges, the desired 3D observation volume is subdivided into bins with a size that is comparable but not necessarily equal to the resolution of the data. Then the box-counting algorithm is applied and a cumulative density function (*cdf*) of the number of points per bin is computed. An example of a *cdf* for the  $128 \times 128 \times 128$  snapshot of the HIT flow is shown in Fig. 3(b). A threshold (*thr*) on the *cdf* allows to isolate the bins with a number of points equal to or higher than the threshold value. Examples of increasing *thr* are shown in Fig. 3(a). Note that, increasing the threshold value from 99.00 to 99.999 progressively isolates the ridge curves.

### 2.3. Parametrization of the ridge point cloud

Once the ridge point locations are determined, we classify them to distinguish between different ridges in order to locate different structures. For this purpose, we employ an unsupervised clustering algorithm that subdivides the points into classes based on their relative distances. The algorithm starts by creating as many classes as the number of points and then it collapses close points. The algorithm stops when each class is farther from each other than a certain distance  $d$ . Here,  $d$  must be comparable to the flow field resolution.

As a result of the clustering, the ridge points are now classified but ridge lines are not yet parametrized. Moreover, the points of the clusters are normally not aligned with the targeted ridge curves. Thus, to obtain 3D parametrized curves, the following procedure is applied. For each cluster, the minimum spanning tree graph is computed (Fig. 4(a)). If the graph presents more than two points with a single connection (red points in Fig. 4(a)), a *pruning* procedure is applied. As schematically shown in Fig. 4(a), this pruning consists of an iterative process in which all points with a single connection are eliminated until only two single-connection points remain, namely the two extreme points of the center line.

A typical outcome of this procedure is shown in Fig. 4(b) in which the grey points represent the initial cluster, while the orange points are the output of the pruning. While the remaining points align along a line, some points at the extremities are lost, i.e. the centerline is slightly shorter compared to the primary line of the minimum spanning tree. An example of the results from the above algorithm is given in Fig. 4(c) in which a sub-volume of the cube shown in Fig. 3 is displayed.

### 2.4. LCSs boundary identification

As a result of the above steps, isolated parametrized 3D curves representing the center of rotational LCSs are computed. In order to identify the structure boundaries, a modified version of the algorithm developed by Neamtu-Halic et al. [20] is employed. The LCS boundaries are defined as the locally outermost convex iso-surface of LAVD bounding each ridge line. Convexity is defined locally by intersecting the iso-surface with planes orthogonal to the ridge points. To define orthogonal planes, an accurate computation of the tangent vector of the ridge is needed. For this purpose, the parametrization of the ridge point coordinates is fitted by a moving spline, from which the tangent unit vector is computed analytically. The smoothing approach is conceptually similar to the one adopted by Lüthi et al. [36] for Lagrangian particle trajectories.

For each ridge, for each ridge point, and for a given value of the LAVD threshold, the LAVD volume is sliced on planes orthogonal to the ridge line and iso-lines are computed on each plane. The iso-lines are then stored as individual objects. For each iso-line, the following properties are computed:

- **closeness:** a flag determining if the iso-line is closed, namely if its last point coincides with the first one;
- **insideness:** a flag determining if the iso-line contains the ridge point;
- **convexity:** a fraction defining the degree of convexity of the iso-line, computed as the ratio between the area of the convex hull and the area of the polygon contained by the iso-line. If the iso-line is convex, the convexity is equal to one, while is contained between 0 and 1 for a non-convex iso-line.

To determine the boundary of the LCS for each point along the ridge, the following procedure is applied: First, iso-lines are extracted from the LAVD field on a plane perpendicular to the ridge curve at the point of interest. Next, only iso-lines that are closed, enclose the ridge point,

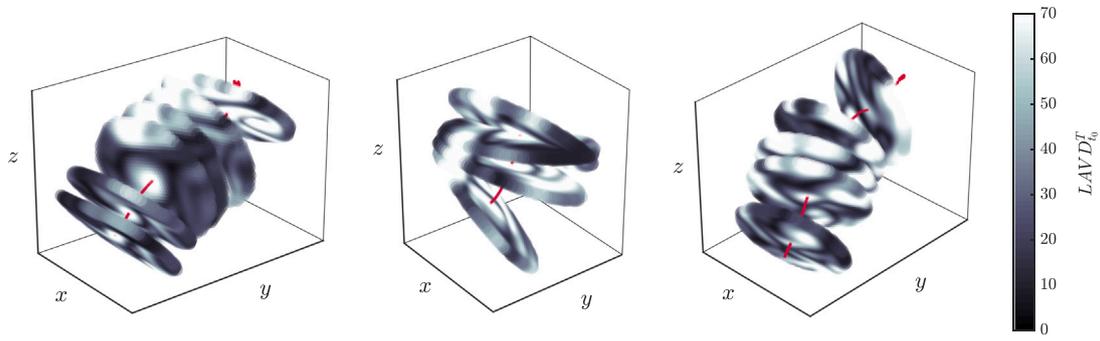


Fig. 5. Sketches of three ridges (red lines) and the surrounding LAVD fields over planes orthogonal to the ridges.

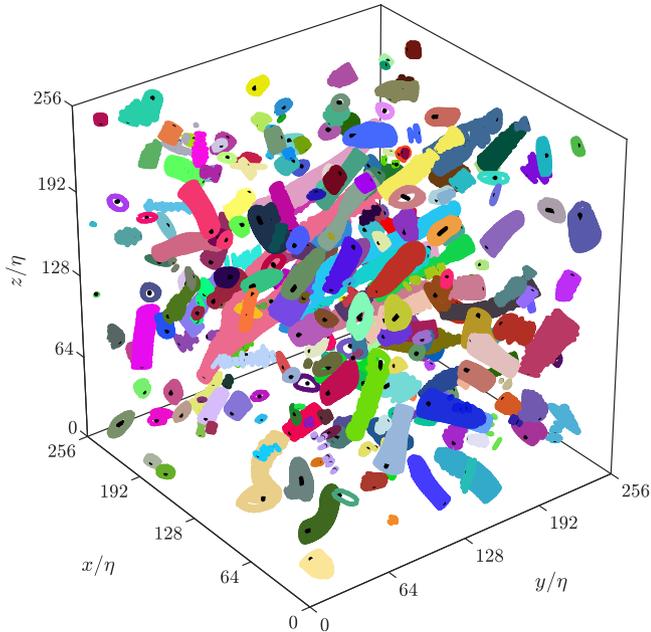


Fig. 6. Example of objectively defined vortical coherent structures extracted from an LAVD cube of 128 grid points corresponding to 256 Kolmogorov length scale  $\eta$  of homogeneous isotropic turbulence of JHT dataset. The structures in this figure are obtained with the streamlines approach.

and are convex are retained. These conditions are checked according to the criteria listed above. Among these lines, the outermost iso-line is selected as the local boundary of the LCS.

The result of the volume slicing is illustrated in Fig. 5. Note that the ridge points correspond to local 3D saddle points, where the LAVD is almost constant along the ridge and fast-decreasing in the radial direction. The final result of the extraction algorithm is the ridge center lines surrounded by the outermost convex rings.

### 3. Example flow cases

In the following, the extraction algorithm is applied to two different flow cases: the homogeneous and isotropic turbulence (HIT) data set from the John Hopkins Turbulence (JHT) database [35] and particle tracking data of a gravity current experiment [20].

In Fig. 6, the vortical structures extracted from the HIT data set are shown. The direct numerical simulation is performed on a domain of  $2\pi \times 2\pi \times 2\pi$  with periodic boundary conditions, using  $1024^3$  grid points, at a Taylor microscale Reynolds number of  $Re_\lambda = 433$ . In the example shown in Fig. 6, a subset of  $256^3$  grid points were used and the LAVD field was integrated over a time span of 200 time steps that

Table 1

Overview of the tolerance parameters used for the two data sets, with *CG* --- Gradient Climbing, *SM* --- Streamline Method and *LC* --- Line Clustering.

Data set	$C_2$ (GC)	$bin/dx$ (SM)	$P_{cdf}$ (SM)	$dd/dx$ (LC)
HIT JHT	0.075	1/5	$1 - 10^{-3}$	4
3D - PTV	0.10	1/5	$1 - 5 \cdot 10^{-3}$	5

correspond approximately to one Kolmogorov time scale. The computation time for the extraction of the ridges was approximately 60 seconds using the streamline method and approximately 110 seconds using the gradient climbing approach. Both computations were conducted on a single-core, 3.5 GHz processor.

The experimental example involves the particle tracking data of the gravity current “Ri20” from [20]. The laboratory flow is created by injecting a lighter fluid in turbulent conditions at the top of an inclined channel, which is filled with a denser fluid flowing in laminar conditions. This setup generates a turbulent gravity current that propagates along the lid of the inclined tank. The gravity current is separated from the laminar region by a sharp layer, so called turbulent non-turbulent interface. The flow is primarily driven by the buoyancy difference between the fluids, while the turbulence is sustained by shear forces between them. The relevant parameters of the flow are the Richardson number  $Ri = 0.20$  and the bulk Reynolds number  $Re = 5000$ . In this case, the data set consists of  $480 \times 120 \times 80$  grid points and the LAVD field was computed over an extraction time of one eddy turnover time. In Fig. 7, the corresponding vortical structures together with the turbulent/non-turbulent interface (red surface) are shown. In this case, a total of 23 structures were educed. The algorithm took 300 seconds to extract all ridges using the gradient climbing approach and 140 seconds using the streamline method. Both computations were again conducted on a single-core, 3.5 GHz processor.

The tolerance parameters used in these examples are listed in Table 1. These values show that the tolerances are smaller for the numerical data (higher resolution) and larger for the experimental data (lower resolution).

### 4. Summary and conclusions

In this paper, we introduced the first unsupervised algorithm for the extraction of rotational Lagrangian coherent structures in 3D and successfully applied it to velocity data from two different turbulent flows. The core of the algorithm is the computation of 3D center lines of coherent structures that correspond to one-dimensional ridges of the 3D LAVD field. As a first step, the points on the LAVD ridges were identified. To this end, two different algorithms were devised, namely a gradient climbing and a streamline-based algorithm. Subsequently, a clustering procedure was employed to group the different ridges and by pruning a minimum spanning tree built over each cluster allowed a full parametrization of the ridge lines. The algorithm concludes with

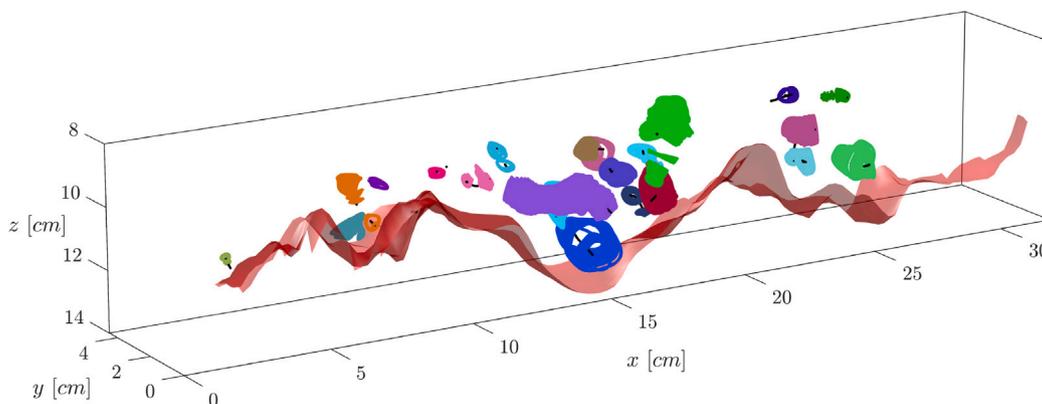


Fig. 7. Example of rotational coherent structures extracted from the 3D-PTV dataset of a gravity current. The structures shown in this figure are obtained with the gradient climbing approach.

the evaluation of the boundary of the coherent structures through the computation of local outermost close and convex contour levels in planes perpendicular to the centerline of the structures.

The novelty of the 3D algorithm introduced in this paper is twofold. On the one hand, the algorithm is independent of the orientation of the structures. Indeed, nearly all previous algorithms were specifically tailored to structures with known orientation and were difficult to generalize [37]. On the other hand, the algorithm employs tolerances that have a well-defined range, thus eliminating the arbitrariness of other user-dependent algorithms [20–22]. Indeed, the tolerances employed here have clear boundaries and well-defined “ideal” values. For example, the stopping condition for the gradient climbing would require  $C_2 = 0$ , a condition that is relaxed to  $C_2 \approx 0$  due to finite resolution and domain size.

The algorithm presented here was applied to two different data sets: a direct numerical simulation of homogeneous and isotropic turbulence and an experimental data set of a gravity current. These test cases highlighted the potential of the algorithm to handle different types of data sets of turbulent flows while showing that typical values for the parameters to be set by the users are rather similar despite the data sets presenting very different characteristics in terms of resolution, presence of experimental noise, etc.

We finally note that the algorithm specifically used the LAVD field as an input to compute the 3D ridges. However, the applicability is more general since it can in principle be applied to any vortex extraction method that employs a scalar field as an input. Examples include the trajectory rotation average (TRA) and the trajectory stretching exponent (TSE) introduced in Haller et al. [38], as well as the active transport barriers methods Haller et al. [34].

#### CRediT authorship contribution statement

**Marius M. Neamtu-Halic:** Writing – original draft, Validation, Software, Conceptualization. **Stefano Brizzolara:** Writing – review & editing, Software, Data curation, Conceptualization. **George Haller:** Writing – review & editing, Supervision, Conceptualization. **Markus Holzner:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization.

#### Declaration of competing interest

The authors report no conflict of interest.

#### Acknowledgments

M.M.N.H. and M.H. acknowledge financial support from SNSF grant number 200727. M.H. and S.B. acknowledge financial support from the DFG priority program SPP 1881 Turbulent Superstructures under Grant No. HO5519/1-2.

#### Data availability

<https://github.com/NeamtuMarius/Unsupervised-3D-LAVD-Extraction-Algorithm>.

#### References

- [1] Townsend AAR. The structure of turbulent shear flow. Cambridge University Press; 1980.
- [2] Adrian RJ. Hairpin vortex organization in wall turbulence. *Phys Fluid* 2007;19(4):041301.
- [3] Ganapathisubramani B, Longmire EK, Marusic I. Characteristics of vortex packets in turbulent boundary layers. *J Fluid Mech* 2003;478:35–46.
- [4] Hutchins N, Marusic I. Evidence of very long meandering features in the logarithmic region of turbulent boundary layers. *J Fluid Mech* 2007;579:1–28.
- [5] Monty J, Stewart J, Williams R, Chong M. Large-scale features in turbulent pipe and channel flows. *J Fluid Mech* 2007;589:147–56.
- [6] Baltzer J, Adrian R, Wu X. Structural organization of large and very large scales in turbulent pipe flow simulation. *J Fluid Mech* 2013;720:236–79.
- [7] Kim J, Hussain F. Propagation velocity of perturbations in turbulent channel flow. *Phys Fluid A* 1993;5(3):695–706.
- [8] Jeong J, Hussain F, Schoppa W, Kim J. Coherent structures near the wall in a turbulent channel flow. *J Fluid Mech* 1997;332:185–214.
- [9] Choi H, Moin P, Kim J. Direct numerical simulation of turbulent flow over riblets. *J Fluid Mech* 1993;255:503–39.
- [10] Kim J. Control of turbulent boundary layers. *Phys Fluids* 2003;15(5):1093–105.
- [11] Quadrio M, Ricco P, Viotti C. Streamwise-travelling waves of spanwise wall velocity for turbulent drag reduction. *J Fluid Mech* 2009;627:161–78.
- [12] da Silva CB, dos Reis RJN. The role of coherent vortices near the turbulent/non-turbulent interface in a planar jet. *Phil Trans R Soc A* 2011;369(1937):738–53.
- [13] Da Silva CB, Métais O. On the influence of coherent structures upon interscale interactions in turbulent plane jets. *J Fluid Mech* 2002;473:103–45.
- [14] Cimarelli A, Mollicone J, Van Rieuwijk M, De Angelis E. Structure of turbulence in temporal planar jets. *Phys Fluid* 2022;34(4).
- [15] Samie M, Aparece-Scutariu V, Lavoie P, Shin D, Pollard A. Three-dimensional large-scale and very-large-scale coherent structures in a turbulent axisymmetric jet. *J Fluid Mech* 2022;948:A29.
- [16] Bernal L, Roshko A. Streamwise vortex structure in plane mixing layers. *J Fluid Mech* 1986;170:499–525.
- [17] Attili A, Bisetti F. Statistics and scaling of turbulence in a spatially developing mixing layer at  $re_\lambda=250$ . *Phys Fluid* 2012;24(3).
- [18] Attili A, Cristancho JC, Bisetti F. Statistics of the turbulent/non-turbulent interface in a spatially developing mixing layer. *J Turbul* 2014;15(9):555–68.
- [19] Schneider K, Farge M, Pellegrino G, Rogers MM. Coherent vortex simulation of three-dimensional turbulent mixing layers using orthogonal wavelets. *J Fluid Mech* 2005;534:39–66.
- [20] Neamtu-Halic MM, Krug D, Haller G, Holzner M. Lagrangian coherent structures and entrainment near the turbulent/non-turbulent interface of a gravity current. *J Fluid Mech* 2019;877:824–43.
- [21] Neamtu-Halic MM, Mollicone J-P, Van Rieuwijk M, Haller G, Holzner M. Connecting the time evolution of the turbulence interface to coherent structures. *J Fluid Mech* 2020;898.
- [22] Neamtu-Halic MM, Mollicone J-P, van Rieuwijk M, Holzner M. Role of vortical structures for enstrophy and scalar transport in flows with and without stable stratification. *J Turbul* 2021;22(7):393–412.

- [23] Haller G, Karrasch D, Kogelbauer F. Material barriers to diffusive and stochastic transport. *Proc Natl Acad Sci* 2018;115(37):9074–9.
- [24] Park D, Lozano-Duran A. The coherent structure of the energy cascade in isotropic turbulence. 2023, [arXiv:2306.08784](https://arxiv.org/abs/2306.08784).
- [25] Hunt JCR, Wray AA, Moin P. Eddies, streams, and convergence zones in turbulent flows. 1988.
- [26] Chong MS, Perry AE, Cantwell BJ. A general classification of three-dimensional flow fields. *Phys Fluids A* 1990;2(5):765–77.
- [27] Jeong J, Hussain F. On the identification of a vortex. *J Fluid Mech* 1995;285:69–94.
- [28] Gurtin ME. *An introduction to continuum mechanics*. Academic Press; 1982.
- [29] Haller G. Lagrangian coherent structures. *Annu Rev Fluid Mech* 2015;47:137–62.
- [30] Haller G. *Transport Barriers and Coherent Structures in Flow Data*. Cambridge University Press; 2023.
- [31] Haller G, Hadjighasem A, Farazmand M, Huhn F. Defining coherent vortices objectively from the vorticity. *J Fluid Mech* 2016;795:136–73.
- [32] Aksamit NO, Haller G. Objective momentum barriers in wall turbulence. *J Fluid Mech* 2022;941.
- [33] Haller G. Dynamic rotation and stretch tensors from a dynamic polar decomposition. *J Mech Phys Solids* 2016;86:70–93.
- [34] Haller G, Katsanoulis S, Holzner M, Frohnapfel B, Gatti D. Objective barriers to the transport of dynamically active vector fields. *J Fluid Mech* 2020;905:A17.
- [35] Yu H, Kanov K, Perlman E, Graham J, Frederix E, Burns R, Szalay A, Eyink G, Meneveau C. Studying Lagrangian dynamics of turbulence using on-demand fluid particle tracking in a public turbulence database. *J Turbul* 2012;(13):N12.
- [36] Luethi B, Tsinober A, Kinzelbach W. Lagrangian measurement of vorticity dynamics in turbulent flow. *J Fluid Mech* 2005;528:87–118.
- [37] Hadjighasem A, Haller G. Geodesic transport barriers in Jupiter’s atmosphere: A video-based analysis. *SIAM Rev* 2016;58(1):69–89.
- [38] Haller G, Aksamit N, Encinas-Bartos AP. Quasi-objective coherent structure diagnostics from single trajectories. *Chaos* 2021;31(4):043131.