Streamline Embedding for 3D Vector Field Exploration

Christian Rössl and Holger Theisel

Abstract—We propose a new technique for visual exploration of streamlines in 3D vector fields. We construct a map from the space of all streamlines to points in $\mathbb{R}^n$ based on the preservation of the Hausdorff metric in streamline space. The image of a vector field under this map is a set of 2-manifolds in $\mathbb{R}^n$ with characteristic geometry and topology. Then standard clustering methods applied to the point sets in $\mathbb{R}^n$ yield a segmentation of the original vector field. Our approach provides a global analysis of 3D vector fields which incorporates the topological segmentation but yields additional information. In addition to a pure segmentation, the established map provides a natural "parametrization" visualized by the manifolds. We test our approach on a number of synthetic and real-world data sets.

Index Terms—vector fields, streamline embedding, clustering

1 Introduction

The visual exploration of 3D vector fields is still a challenging problem for which a variety of solutions have been proposed. One class of approaches aims at finding semantic structures in the flow, i.e., flow regions which show a similar behavior concerning certain physical, geometrical, or topological criteria. Such segmentation can be achieved for example by a local analysis or by global methods applying vector field topology, as well as by hierarchical or multi-scale methods. Although topological approaches are well-established for 3D fields, there are a number of shortcomings, namely:

- Topological segmentation gives a separation of regions of different flow behavior. This segmentation is a binary information: topology tells us *if* two particular regions behave differently, it does not tell us *how* differently they behave.
- Visualization does not "scale" from 2D to 3D vector fields: given a segmentation, visualizing all streamlines even for only of a number of segments is hard because of visual clutter. This is a general problem for vector fields in three dimensions.

In this paper we propose an approach which tackles the two points mentioned above. The main idea is to consider the space of all streamlines and to equip it with a non-trivial metric. For this we use the Hausdorff distance between streamlines. Then we define a map from the set of all streamlines to points in an n-dimensional space, which preserves the Hausdorff distances as much as possible. This way, the set of all streamlines of a 3D vector field is mapped to a number of two-manifolds in the nD space. Figure 1 illustrates the main idea of the paper.

Our approach addresses the above mentioned problems: first, distances in manifolds provide a continuous measure of *how* different streamlines behave. And second, the manifolds

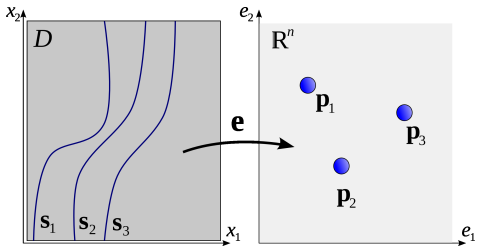

Fig. 1. Main idea: every streamline $\mathbf{s}_i \in D$ is mapped to a point $\mathbf{p}_i \in \mathbb{R}^n$ such that the Euclidean distances between $\mathbf{p}_i, \mathbf{p}_j$ correspond as much as possible to the Hausdorff distances of $\mathbf{s}_i, \mathbf{s}_j$.

can be visualized in a straightforward way, and to some extent flow field exploration reduces to surface visualization. Finally, thinking of manifolds does not only convey a quantitative view of streamline behavior but this new domain also provides a parametrization of the vector field: using the embedding, every stream line can be uniquely addressed by a point on a surface.

In this paper we describe how to find such a map from streamlines to points and explain how it can be used for interactive flow exploration.

The remainder of the paper is organized as follows. Section 2 reviews work related to ours. In Section 3 we explain the general approach and illustrate the idea with simple 2-dimensional vector fields. The implementation of this theoretical framework is reviewed in Section 5. Section 6 discusses results, Section 7 summarizes the limitations, and Section 8 concludes the paper.

2 Related work

There exist a number of approaches to find relevant structures in vector fields. Hierarchical clustering methods based on local criteria of vector fields are applied in [1], [2]. Further methods based on local analysis apply diffusion flow [3] well-known from data smoothing, or apply similarly multi-scale methods

- *C. Rössl and H. Theisel are with the Visual Computing group, University of Magdeburg, Germany.*
E-mail: {roessl,theisel}@isg.cs.uni-magdeburg.de

[4] to achieve a multilevel decomposition. Note that here the term "local" refers to a certain level of resolution and in this sense may be misleading. There are a number of approaches to detect region based vortical structures, such as [5], [6]. All methods mentioned above have in common that they act on local properties of the flow, i.e., that a streamline may enter or leave different regions of local flow behavior.

In addition there are semantic flow separations where a streamline is guaranteed to belong to exactly one segment. Among them, topological methods are perhaps the most prominent ones [7]–[11]. They aim at segmenting the flow into areas of similar asymptotic flow behavior and are widely-researched in the visualization community. We refer to [12]–[14] for state-of-the-art reports in vector field topology.

The concept of streamline predicates [15] is used in a framework aiming at segmenting streamlines by mapping one and the same predicate to every point on a streamline. This way, streamline-respecting segmentations of the flow domain are obtained for different choices of the predicates.

In DT-MRI visualization there are various approaches to treat fiber bundles instead of particular locations, for instance [16], [17]. A spectral clustering approach is taken in [18] for finding white matter fiber correspondences. Here, similarity matrices are defined using the Hausdorff distance. In [19] a hierarchical clustering scheme is applied. Various proximity measures on fiber tracts are combined in [20] for visualizing a 2D embedding.

Also related to our work are streamline placing techniques for 2D [21], [22] and 3D [23] vector fields. These methods aim at an equidistant placement of streamlines in the domain which requires a variable length of streamlines as an optimization parameter. This is in contrast to a global vector field analysis which must consider streamlines as a whole, i.e., streamlines of maximal length.

3 Streamline Embedding

Given a steady vector field $\mathbf{v}(\mathbf{x})$, over the domain D, there is exactly one streamline passing through every location $\mathbf{x} \in D$. A streamline $\mathbf{s} \subseteq D$ is a curve with the property that its tangent direction coincides with the velocity vector of $\mathbf{v}$ everywhere.

Let $\mathbf{S} \subseteq P(D)$ be the set of all streamlines of $\mathbf{v}$ where $P(D)$ denotes the power set of D. Then $\mathbf{S}$ builds a partition of D: two streamlines are either disjoint or identical, and the union of all streamlines gives D. (Note that critical points fit into this concept as well: an isolated critical point is a streamline of its own, while a streamline integration starting from the neighborhood of the critical point may come infinitely close but never reaching it.)

For our purpose we need a distance measure on $\mathbf{S}$. The only non-trivial measure we are aware of that builds a metric space is the Hausdorff distance:

$$d_H(\mathbf{s}_1,\mathbf{s}_2) = \max\{d_h(\mathbf{s}_1,\mathbf{s}_2), d_h(\mathbf{s}_2,\mathbf{s}_1)\} \ , \qquad (1)$$

with the one-sided distances

$$d_h(\mathbf{s}_i,\mathbf{s}_j) \ = \ \sup\left\{d(\mathbf{x},\mathbf{s}_j) \,|\, \mathbf{x} \in \mathbf{s}_i\right\} \ ,$$

and

$$d(\mathbf{x},\mathbf{s}) \ = \ \inf\{d(\mathbf{x},\mathbf{y}) \,|\, \mathbf{y} \in \mathbf{s}\}$$

for $\mathbf{s}_1,\mathbf{s}_2 \in \mathbf{S}$, and where $d(\mathbf{x},\mathbf{y}) = ||\mathbf{x}-\mathbf{y}||_2$ denotes the Euclidean distance in $\mathbb{R}^3$. Our goal is to establish a map

$$\mathbf{e} : \mathbf{S} \to \mathbb{R}^N$$

which preserves the Hausdorff distances as much as possible: for two streamlines $\mathbf{s}_1,\mathbf{s}_2$, the distances $d_H(\mathbf{s}_1,\mathbf{s}_2)$ and $d(\mathbf{e}(\mathbf{s}_1),\mathbf{e}(\mathbf{s}_2))$ should be as close as possible where d describes the Euclidean distance in $\mathbb{R}^N$. This way we ensure that streamlines which are close to each other in Hausdorff space are mapped to spatially close points in $\mathbb{R}^N$.

Finding $\mathbf{e}$ can be formulated in least-squares sense, i.e., as the following minimization problem: let $\{\mathbf{s}_1,...,\mathbf{s}_n\} \subseteq \mathbf{S}$ be a dense sampling of $\mathbf{S}$. We search for $\mathbf{e}(\mathbf{s}_1),...,\mathbf{e}(\mathbf{s}_n)$ such that

$$\sum_{i=1}^{n}\sum_{j=1}^{n} \left(d_H(\mathbf{s}_i,\mathbf{s}_j) - d(\mathbf{e}(\mathbf{s}_i),\mathbf{e}(\mathbf{s}_j)) \right)^2 \to \min \qquad (2)$$

This is the problem arising in the classical multidimensional scaling (MDS) method (see, e.g., [24]). Solving (2) leads to computing the spectral decomposition of a symmetric matrix. Obviously, the solution is not unique because every translation, rotation or reflection in a line remains a solution. Furthermore, the solution depends on the sampling of $\mathbf{S}$, i.e., on the number and distribution of streamlines $\mathbf{s}_i$.

Illustrating 2D examples

We illustrate the approach on a number of 2D example fields on a square domain $D = [0,1] \times [0,1]$. We decided to choose the dimension $n = 2$ and establish maps to $\mathbb{R}^2$ for the sake of illustrating the concepts. The shown embeddings are essentially orthogonal projections of embeddings in $\mathbb{R}^3$. We use a rather low number of streamlines here and obtain discrete 1-manifolds which would converge to true manifolds which appear for an infinite number of samples. The colors of streamlines match colors of respective segments/connected components. (The coordinate systems and axes for embedding are chosen as described in Section 6, see also Section 5.3 for minimization of (2).)

Figure 2 shows streamlines samples from a linear vector field describing a node sink. The resulting mapping to $\mathbb{R}^2$ shows that $\mathbf{e}(\mathbf{s}_i)$ lie on a closed 1-manifold consisting of four sharp corners corresponding to streamlines passing through the four corners of D. The sharp corners arise from the fact that the Hausdorff distances become larger towards the diagonal streamlines.

Figure 3 shows sampled streamlines of a simple saddle. Its embedding shows four unconnected 1-manifolds. They describe the four topological segments of the flow.

Figure 4 shows the sampled streamlines of a center and its corresponding embedding. Again, the separation of the 1-manifolds is induced by the topological structure of the vector fields, here the boundary switch points. This becomes even more evident when we shift the center to enforce a different flow between boundaries, see Figure 5. Finally, Figure 6 shows streamlines of a more complex symmetric vector field with multiple critical points.

The examples above show us some properties of the proposed mappings: they consist of a number of unconnected 1-manifolds where every manifold corresponds to a topological

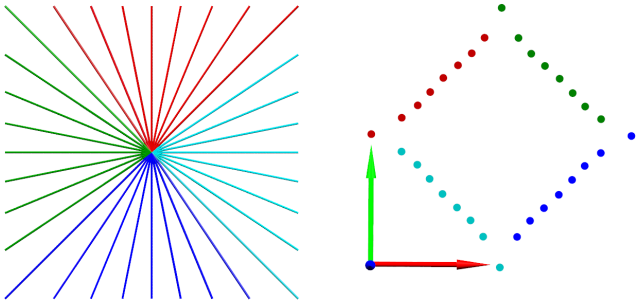

Fig. 2. Linear 2D vector field - a sink node and its embedding.

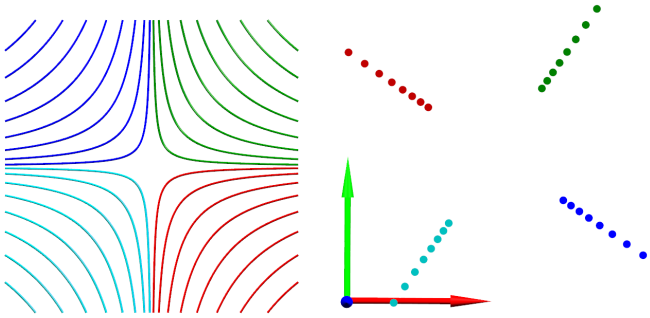

Fig. 3. Linear 2D vector field - a saddle and its embedding.

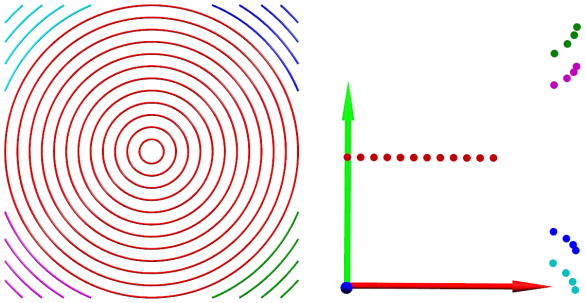

Fig. 4. Linear 2D vector field - a center and its embedding.

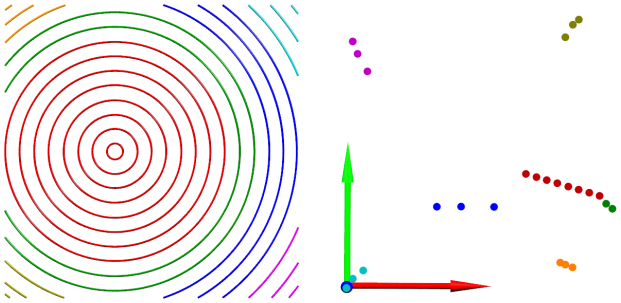

Fig. 5. Shifting the center within the same domain bounds results in a different global behavior.

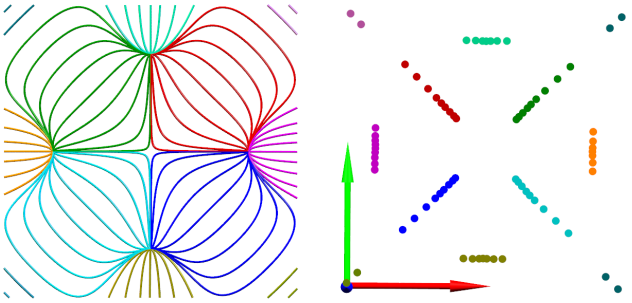

Fig. 6. A more complex symmetric 2D vector field with multiple critical points and its embedding.

sector. The manifolds have a shape and a position relative to each other, giving information on how different the streamlines in differently topological sectors behave.

4 Analysis of Streamline Embedding

In this section we provide a theoretic analysis of streamline embedding. We reveal the relation to vector field topology which leads to the conclusion that topological segments are mapped to manifolds. Furthermore, we discuss alternative distance measures other than the Hausdorff distance.

4.1 Relation to vector field topology

In this section we show how the Hausdorff distance of streamlines is related to vector field topology. The essential proposition is that the Hausdorff distance is discontinuous only between topological segments. We take this further in the subsequent section to conclude that the map $\mathbf{e}$ maps each topological segment to a manifold.

Topology-based flow visualization has emerged as a standard tool for analyzing vector fields. We refer to the state-of-the-art reports [12], [13] for an overview of the topic and definition of vector field topology. In the following theorem we require the concept of a *topological skeleton* which partitions the domain D into regions of similar flow behavior, e.g., streamlines seeded in a certain segment all end in the same critical point or boundary region. For consistency, we assume that separation curves and surfaces and critical points, as well as boundary switch points and curves are considered segments.

Theorem 1 (Continuity of Hausdorff distance): Given is a vector field $\mathbf{v} : D \to \mathbb{R}^n$, $n \in \{2,3\}$, and the partition $\mathscr{S} = \{S_1, \ldots, S_N\}$, $\bigcup_i S_i = D$ and $S_i \cap S_j = \emptyset$ for $j \neq i$, induced by its topological skeleton.

Let $\mathbf{x}_0 \in S_i \in \mathscr{S}$ and $\mathbf{x} \in D$ with $||\mathbf{x} - \mathbf{x}_0||_2 < \delta$. Let s_0 and s be the streamlines passing through $\mathbf{x}_0$ and $\mathbf{x}$, respectively: $\mathbf{x}_0 \in \mathbf{s}_0$ and $\mathbf{x} \in \mathbf{s}$. Then

$$\lim_{\delta \to 0} d_H(\mathbf{s}_0, \mathbf{s}) = \begin{cases} 0 & \text{iff } \mathbf{x} \in S_i \\ c > 0 & \text{iff } \mathbf{x} \notin S_i \end{cases}.$$

The above theorem states that the Hausdorff distance is *continuous within* topological segments, and d_H is *discontinuous between* topological segments.

We provide a proof for unbounded 2D vector fields ($n = 2, D = \mathbb{R}^2$). The same arguments apply to the 3D case, and similarly the proof can be extended to bounded domains.

Proof: Let $\mathbf{v} : \mathbb{R}^2 \to \mathbb{R}^2$ be a sufficiently smooth vector field. Let $\mathbf{x}_1^c \in \mathbb{R}^2$ be a critical point – w.l.o.g. a source node – of $\mathbf{v}$, i.e., $\mathbf{v}(\mathbf{x}_1^c) = \mathbf{0}$. Let α denote an angle w.r.t. a local coordinate system at $\mathbf{x}_1^c$. Let $\mathbf{s}_\alpha$ be the streamline emanating from the close vicinity of $\mathbf{x}_1^c$ in direction α which ends in another critical point $\mathbf{x}_2^c \neq \mathbf{x}_1^c$.

Consider a second streamline $\mathbf{s}_\beta$ emanating from $\mathbf{x}_1^c$ in direction β. We are interested in the limit $\alpha \to \beta$. There are two cases:

(i) $\mathbf{s}_\beta$ ends in $\mathbf{x}_2^c$ (Fig 7 (a)):
Then $\lim_{\alpha\to\beta} d_H(\mathbf{s}_\alpha, \mathbf{s}_\beta) = 0$ because with $\alpha \to \beta$ the pointwise distance between $\mathbf{s}_\alpha, \mathbf{s}_\beta$ becomes arbitrarily small as $\mathbf{v}$ does not vanish along $\mathbf{s}_\beta$ until $\mathbf{x}_2^c$ is reached.

(ii) $\mathbf{s}_\beta$ ends in another critical point $\mathbf{x}_3^c \neq \mathbf{x}_2^c$ (Fig 7 (b)):
Then $\lim_{\alpha\to\beta} d_H(\mathbf{s}_\alpha, \mathbf{s}_\beta) = c > 0$.

The constant $c > 0$ depends on $\mathbf{v}$ in the neighborhood of $\mathbf{x}_3$. Assume w.l.o.g. $\mathbf{x}_3^c$ is a sink node[1]. There exists a neighborhood $\mathcal{N}$ of $\mathbf{x}_3^c$ such that a streamline seeded in any location within $\mathcal{N}$ will end in $\mathbf{x}_3^c$. Then we define $c > 0$ as the maximum radius of a sphere inscribed in $\mathcal{N}$.

With the definition of the topological segmentation $\mathcal{S}$ we can interpret the condition for case (i) as $\mathbf{s}_\alpha, \mathbf{s}_\beta \subset S_i$, i.e., the streamlines belong to the same topological segment. And likewise for case (ii) we have $\mathbf{s}_\alpha \subset S_i$, $\mathbf{s}_\beta \subset S_j$ with $S_j \in \mathcal{S}$ and $i \neq j$, i.e., the streamlines belong to different topological segments. This concludes one direction of the proof. The reverse direction can be shown with the same arguments. □

We restricted ourselves to the domain $D = \mathbb{R}^2$ and made assumptions on critical points for the sake of simplicity and a concise proof. The extension to unbounded 2D or 3D domains is straightforward, and we provide only few remarks. In 2D, separation curves start from and end in saddle points and/or boundary switch points for bounded domains in structurally stable configurations. Figure 7 (c-d) illustrates the situation from a source node to the boundary. While this yields an extra case, the arguments of the proof remain the same. The situation is only slightly different for center nodes, which are regarded unstable and hardly appear in practice[2]. In 3D, generally the same arguments apply. However, there are additional topological structures like, e.g., separation surfaces.

4.2 Embedding manifolds

In this section we apply the above theorem to show that the map $\mathbf{e}$ maps topological segments of vector fields to manifolds in $\mathbb{R}^N$.

The segmentation $\mathcal{S}$ partitions $D \subseteq \mathbb{R}^n$, i.e., topological segments $S_i \in \mathcal{S}$ are m-dimensional objects, $0 \leq m \leq n$. Generally, we have $m = n$, e.g., segments are 2-dimensional regions for $D \subseteq \mathbb{R}^2$, however, for special cases like critical points ($m = 0$) and separation curves ($m = 1$) we have $m < n$. Furthermore in 3D there may exist 2-dimensional segments S such that $P(S)$ is a 2-dimensional subspace of $\mathbb{R}^3$.

Each segment S_i can be generated as an image $P(S_i)$, i.e., it is "spanned" by the set of all streamlines in S_i. These streamlines can in turn be generated by a continuous seeding structure $\mathbf{g}_i$ which is an $(m-1)$-dimensional object. For example, consider a 2D segment $S_i \subset \mathbb{R}^2$: one can choose arbitrary seed points for each streamline in S_i. But obviously one can also arrange the seed points along a continuous curve $\mathbf{g}_i \subset S_i$ by ensuring that for "neighboring" streamlines the distance of seed points vanishes. Hence, the seeding structure is a 1-dimensional object. Figures 2-6 (left) illustrate this: the colors indicate segments, and the seeding structures can be chosen as straight lines. Note that boundaries have to be taken into account in these examples.

Following Theorem 1, the Hausdorff distance is continuous only within topological segments. Remember that the map $\mathbf{e}$ assigns a point $\mathbf{e}(\mathbf{s}) \in \mathbb{R}^N$ to each streamline $\mathbf{s}$ such that Hausdorff distances between streamlines are preserved in least squares sense. We assume that $\mathbf{e}$ is continuous. Then a segment $S_i \subset \mathbb{R}^m$ is mapped to a $m-1$ dimensional manifold in $\mathbb{R}^N$. And likewise any other k-dimensional segment $S_k \in \mathcal{S}$ is mapped to another, distinct $(k-1)$-dimensional manifold.

We conclude this section with few remarks. The above arguments reveal that it is reasonable to choose an embedding into $\mathbb{R}^2$ and $\mathbb{R}^3$ for 2D and 3D vector fields, respectively. Choosing a larger dimension N only alleviates the error term (2), also called strain. Figure 8 illustrates this: compare the 3D embedding to the 2D embedding in Figure 4. The 3D embedding shows 1-manifolds and a smaller error which can be seen from the better preservation of symmetry in this example. Furthermore, it is reasonable to choose a distance measure on streamlines which is indeed a metric such as the Hausdorff distance. However, given a metric on streamlines, e.g., d_H, there is no guarantee that the map $\mathbf{e}$ (as constructed by the MDS method) preserves the triangle inequality. So far, we only used continuity arguments to separate manifolds. The Hausdorff distance between distinct manifolds quantifies the difference of associated topological segments. A natural and open question is how the shape of distinct manifolds, e.g., their curvature, can be interpreted.

4.3 Alternative distance measures

We choose the Hausdorff distance for embedding streamlines because it is well known that it is a *metric*. Recall that the minimum requirements for a reasonable distance measure are positive definiteness and symmetry, in addition, a metric satisfies the triangle inequality. In this section we summarize other distance measures for curves which can be used as alternatives. Such measures are applied for tracking fiber traces in diffusion tensor imaging, we refer to the state-of-the-art report [25] for an overview.

1. A similar argument as for streamlines ending in a sink can be used for streamlines converging to a closed orbit with sink-like behavior.

2. Note that for divergence-free 2D vector fields structurally stable centers do exist. Such fields do not have sources or sinks, and streamlines either are closed or leave the domain. For such fields topological methods are generally not applicable whereas our streamline embedding works, see examples in Figures 4 and 5.

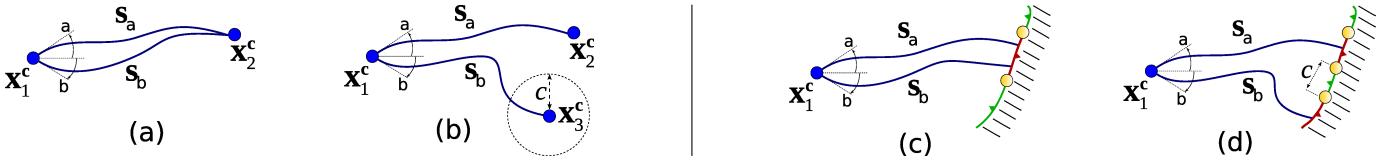

Fig. 7. Left: two streamlines start from $\mathbf{x}_1^c$. They end either in the same (a) or in different (b) critical points. Right: if the streamlines end in different boundary segments (d), there must be a non-empty segment of different (in/out) flow behavior in-between.

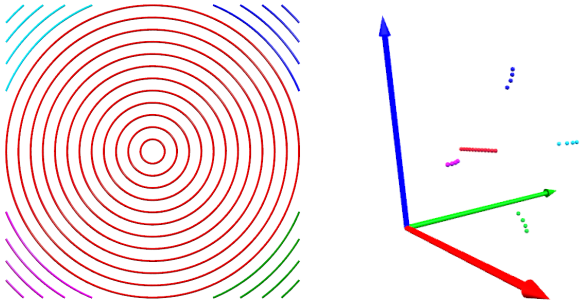

Fig. 8. 2D vector field from Figure 4 embedded in 3D.

The Hausdorff distance d_H can be considered a "worst case distance" as it uses the maximum distance from either line to the other streamline. There are two obvious modifications to obtain alternative distance measures:

First, one could restrict the maximization from whole lines to just *end points* of streamlines:

$$d_E(\mathbf{s}_1,\mathbf{s}_2) \;=\; \max\{d(\mathbf{s}_1(t_1)-\mathbf{s}_2(t_1))\,|\,t_1,t_2\in\{0,1\}\}\;,$$

assuming $\mathbf{s}_1$ and $\mathbf{s}_2$ are both parametrized over the interval $[0,1]$. This yields a much simpler distance measure [26] which can be evaluated in constant time. Despite this simplicity, the above theorem states that such measure separates topological segments (distance zero within and c between segments) and is hence considered useful [27].

Second, one could look for a "smoother" version of the Hausdorff distance which is less sensitive to extrema. A frequent choice is a measure based on *mean* distance [28] between streamlines:

$$d_M(\mathbf{s}_1,\mathbf{s}_2) \;=\; \frac{1}{2}\left(d_m(\mathbf{s}_1,\mathbf{s}_2)+d_m(\mathbf{s}_2,\mathbf{s}_1)\right)\;,$$

with

$$d_m(\mathbf{s}_1,\mathbf{s}_2) \;=\; \operatorname*{mean}_{\mathbf{x}_1\in\mathbf{s}_1}\min_{\mathbf{x}_2\in\mathbf{s}_2} d(\mathbf{x}_1,\mathbf{x}_2)\;.$$

Zhang et al. [29] modify d_m to take the mean only over points whose minimum distance is above some threshold τ, i.e., replace d_m by

$$d_m^\tau(\mathbf{s}_1,\mathbf{s}_2) \;=\; \operatorname*{mean}_{\mathbf{x}_1\in\mathbf{s}_1\wedge d(\mathbf{x}_1,\mathbf{x}_2)>\tau}\;\min_{\mathbf{x}_2\in\mathbf{s}_2} d(\mathbf{x}_1,\mathbf{x}_2)\;.$$

In addition to the symmetric average $d_M^\tau=\frac{1}{2}(d_m^\tau(\mathbf{s}_1,\mathbf{s}_2)+d_m^\tau(\mathbf{s}_2,\mathbf{s}_1))$, they consider measures using either the shorter or the longer distance of $d_m^\tau(\mathbf{s}_1,\mathbf{s}_2)$ and $d_m^\tau(\mathbf{s}_2,\mathbf{s}_1)$. The motivation is to emphasize diverging parts of streamlines $\mathbf{s}_1,\mathbf{s}_2$ more than d_M but less than d_H, in this sense the measure d_M^τ is *in-between* d_M and d_H.

Although the *minimum* distance $d_{\min}$ between curves is not considered useful in the literature [25] we include it in our experiments. Note that this measure can easily degenerate (as well as the end point distance d_E), see Figure 19.

Brun et al. [30] define *feature* descriptors from the mean and square root of covariance matrix of a sampled streamline which yields a 9-dimensional vector. They define the distance measure d_F between streamlines as the Euclidean distance between their feature vectors.

We apply and compare the above mentioned distance measures in our experiments (see Section 6.6). Note that the theoretical analysis is only valid for the Hausdorff metric.

5 Implementation

Technically, our approach can be summarized as follows. We *sample* a finite number of streamlines in D and measure Hausdorff distances for every pair of streamlines. From the distance matrix the multidimensional scaling method computes an *embedding* in $\mathbb{R}^N$ (with $N=3$), such that distances are preserved in least-squares sense. The result is a discrete map $\mathbf{e}$ which assigns every sampled streamline a point in $\mathbb{R}^3$. A *partition* of the samples into clusters is computed from the distances. We *visualize* the image of $\mathbf{e}$ which are points sampling 2-manifolds. In the remainder of this section we describe this process in detail.

5.1 Hausdorff distances between streamlines

Streamlines are integrated by a standard fourth order Runge-Kutta method with adaptive step size control. We use a simple approach to measure Hausdorff distances between stream lines: the integration scheme provides each streamline as a piecewise polynomial which we resample with respect to arc length and a uniform distance τ between samples. We then reduce the problem to computing the *discrete* Hausdorff distance between two point sets. The sampling rate τ bounds the error induced by this simplification, we choose it two orders of magnitude smaller than the minimal Hausdorff distance between streamline (see below). We remark that streamline integration takes $<3\%$ of the overall run time, computation of Hausdorff distances takes $>90\%$. We apply mostly a brute force approach of quadratic complexity testing all distances between all point pairs without using any spatial data structures in the inner loop (see (1)): we realize that hierarchical structures, such as kd-trees, don't always pay off because they have to be set up per streamline. And for

each streamline the number of points is still relatively small such that the gain from logarithmic versus linear complexity is not enough to justify the additional overhead. We made a compromise and apply a kd-tree (using the ANN library [31]) only if the number of samples of a particular streamline is above some threshold. Possible other improvements are advanced sweep-line methods for points and line sets such as [32] which apply a Voronoi diagram, or taking advantage of a parallel GPU implementation.

5.2 Sampling streamlines

Ideally we would like to achieve a regular sampling of the embedded 2-manifolds. The only reasonable way is to express the degree of regularity in terms of Hausdorff distances between streamlines and hence Euclidean distances in the embedding. We take a straightforward approach to generate a set of streamlines: we discretize the domain D with a regular grid with fixed cell size h and enumerate all grid points. For every grid point we integrate a streamline. We put this streamline in the set only if its Hausdorff distance to all streamlines already in the set is larger than a threshold $\delta_{\min}$. Otherwise the streamline is discarded. This way we can adjust the sampling rate, and we avoid local oversampling by choosing the parameters h and $\delta_{\min}$. Distance tests are executed in reverse order such that potentially near streamlines are tested first. This leads to an early reject in practice, typically only an average number of $10-300$ streamlines were considered for each test of a potential new streamline. We remark that a random sampling (with a caching strategy for early reject) generally showed inferior results in terms of streamline placement.

Finally, we note that streamline sampling does not directly compare to "usual" manifold sampling as, e.g., in surface reconstruction, because the space of streamlines $\mathbf{S}$ is not a Euclidean vector space. All we know about this space is how to measure (unoriented) distances. In addition, each streamline (except critical points) has an infinite number of representatives, i.e., seeding points, in $\mathbb{R}^3$, and there is no straightforward way to approximate an inverse map $\mathbf{e}^{-1}$ to support seeding.

5.3 Computing the embedding

We compute the embedding induced by the discrete map $\mathbf{e}$ by multidimensional scaling (MDS) [24]. The input to MDS is a symmetric $m \times m$ distance matrix $\mathbf{D}$ with $\mathbf{D}_{ij} = d_H(\mathbf{s}_i, \mathbf{s}_j)^2$ for $i \neq j$ and $\mathbf{D}_{ii} = 0$ for $1 \leq i, j \leq m$. (The matrix $\mathbf{D}$ was set up in the sampling stage by recording all measured distances.) Constraining the mean to the origin yields the symmetric matrix $\mathbf{B} = -\frac{1}{2}\mathbf{J}\mathbf{D}\mathbf{J}$, where $\mathbf{J} = \mathbf{I} - \frac{1}{m}\mathbf{1}\mathbf{1}^\top$, and $\mathbf{1} \in \mathbb{R}^m$ is a column vector of ones. Spectral decomposition yields $\mathbf{B} = \mathbf{Q}\Lambda\mathbf{Q}^\top$ with diagonal matrix Λ holding the eigenvalues $\Lambda_{ii} = \lambda_i$ in decreasing order such that $\lambda_i > \lambda_j$ for $i < j$, and associated eigenvectors as columns of $\mathbf{Q}$. The restriction to the first N columns of $\mathbf{E} = \mathbf{Q}\Lambda^{\frac{1}{2}}$ yields coordinates in $\mathbb{R}^N$ in the rows of this matrix. The magnitude of eigenvalues "prioritize" the coordinate axis, and we are interested in the projection onto the first N eigenvectors. Thus, for a valid embedding the first N eigenvalues must be positive and should be significantly larger than the remaining eigenvalues $\lambda_{N+1}, \ldots, \lambda_m$. This is satisfied for $N = 3$ in all our experiments (see Figure 23).

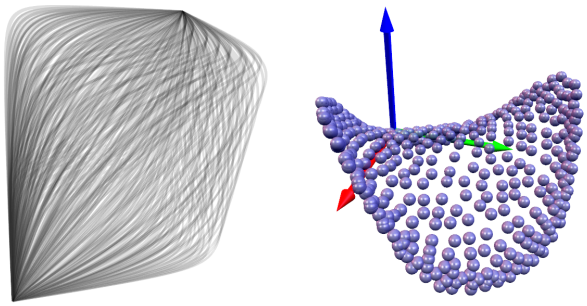

Fig. 9. Vector field (3) and its embedding.

We conclude this section with few remarks. First, an embedding into $(N-1)$-dimensional space is obtained from the embedding in N dimensions as a projection along the axis corresponding to λ_N. Figures 8 and 4 illustrate this. For all 3D examples in the subsequent sections this refers to a projection along the blue z-axis. (Note that projected manifolds may overlap in 2D.) Second, although the goal is distance preservation, the results may exhibit significant distortion as there is generally no perfect solution (in $\mathbb{R}^3$). One way to remedy this is to choose a subset of manifolds and associated streamlines and to compute a new embedding of this subset only. Finally, We remark that there are various other methods for dimensionality reduction with different goal functions. MDS is the method of choice for distance preservation. One drawback of the method as outlined above is the mandatory use of dense matrices. For large data sets it might be useful to choose another method which takes advantage of sparse matrices, e.g., Laplacian eigenmaps [33], and then choose subsets for embedding (see, e.g., Figure 14).

5.4 Clustering

We aim at a segmentation of the vector field based on Hausdorff distances (or distances under the map $\mathbf{e}$). One way to partition the field into distinct segments is using a clustering method. It is well-known that direct application of clustering methods such as k-means on the embedded points generally does not yield good results as it tries to separate point sets but does not incorporate connectivity information. Better results are obtained from spectral clustering, see, e.g., [34]. Our goal is to compute a partition into k segments *automatically* with the number $k \geq 2$ as a variable. We achieve this by normalized cuts [35], a spectral method which formulates clustering as a graph partitioning problem. (This is closely related to traditional spectral clustering as essentially the same functional is minimized.)

We briefly summarize the general idea. We assume that we want to partition the samples into two segments 0 and 1. This bi-partition can be expressed by a vector $\mathbf{b} \in \{0,1\}^m$. A normalized cut defines $\mathbf{b}$ as the minimizer of a certain cost function which expresses how well clusters are separated while taking into account compactness of clusters. The discrete

problem is now turned into a continuous problem replacing $\mathbf{b}$ by $\mathbf{y} \in \mathbb{R}^m$ in order to make optimization feasible. First, the distance ("dissimilarity") matrix $\mathbf{D} \in \mathbb{R}^{m \times m}$ is turned into a similarity matrix $\mathbf{A}$ with $\mathbf{A}_{ij} = \exp(-\frac{1}{\rho^2}\mathbf{D}_{ij})$. The parameter ρ controls how rapidly similarity falls off with the squared distance $\mathbf{D}_{ij} = d_H(\mathbf{s}_i, \mathbf{s}_j)^2$. Note that $\mathbf{A}$ can be made a sparse matrix in practice by either truncating small entries to zero, or by restricting non-zeros to a certain number of nearest neighbors using the symmetric $\mathbf{A}+\mathbf{A}^\top$. Now put the row sums of $\mathbf{A}$ as a diagonal weight matrix $\mathbf{W}$ such that $\mathbf{W}_{ii} = \sum_j \mathbf{A}_{ij}$, and compute the graph Laplacian $\mathbf{L} = \mathbf{W}^{-\frac{1}{2}}(\mathbf{W}-\mathbf{A})\mathbf{W}^{-\frac{1}{2}}$ which is symmetric and positive semidefinite. An approximate (continuous) solution to the normalized cut minimizes the Rayleigh quotient

$$\mathrm{ncut}(\mathbf{y}) \;:=\; \frac{\mathbf{y}^\top \mathbf{L}\mathbf{y}}{\mathbf{y}^\top \mathbf{y}} \;.$$

It is well known that this minimization leads to solving the eigenvalue problem $\mathbf{L}\mathbf{y} = \lambda\mathbf{y}$. (Spectral clustering as described in [34] uses a different formulation for the Laplacian matrix resulting in a shift of eigenvalues.) Let $0 = \lambda_1 < \lambda_2 \le \lambda_3 \le \cdots \le \lambda_m$ be eigenvalues of $\mathbf{L}$, and $\mathbf{y}_i$ are the associated eigenvectors. The first eigenvector $\mathbf{y}_1$ is constant and leads to $\mathrm{ncut}(\mathbf{y}_0) = 0$. The second eigenvector $\mathbf{y}_2$ provides an approximation to the normalized cut solution. Recall that the optimal cut into two segments is the discrete vector $\mathbf{b}$. Hence, we are left with computing a binary partition from $\mathbf{y}_2$, e.g., by thresholding.

Indeed, this process of computing the discrete partition is more involved as we are aiming at a k-way partition with $2 \le k \le k_{\max}$. There are various methods for computing a discrete segmentation from the eigenvectors $\mathbf{y}_2, \ldots, \mathbf{y}_{k_{\max}+1}$ including hierarchical cuts and a k-means variant. We choose a simultaneous k-way cut from multiple eigenvectors (see [35] for details). For each segmentation we obtain a cost value ncut_k of the normalized cuts, and we minimize this value by a linear search over the number of clusters k. This is efficient because the computation is dominated by computing the first $k_{\max}+1$ eigenvectors. Figure 22 shows cost (normalized for comparison) for k-way cuts plotted over the number of clusters: we determine the optimal k as the global minimum. The diagram provides a visual feedback revealing local minima which indicate "stability" of the choice of k and alternative choices. We observed that this clustering scheme tends to find reasonable numbers k and good clusters automatically. We are not restricting neighborhoods and taking advantage of sparse matrices here. This leaves the distance fall-off ρ as the only parameter of the algorithm.

5.5 Visual representation and interaction

For the visual representation of the embeddings, we have chosen small spheres to show the particular points of the embedding and to give an impression of their spatial relation. Since the embedded points are assumed to build 2-manifolds, a surface reconstruction using, e.g., alpha shapes [36], or a splat rendering after estimation and orientation of normals is an option. We prefer the discrete representation using spheres to give the user a better impression of the present point density and distribution. Furthermore, any faithful surface reconstruction has to make assumptions on sampling density relative to local feature size. This would require a careful tuning of parameters for every point cloud.

For representing the clusters, we use a simple color coding for both the streamlines and the embeddings where corresponding clusters are equipped with the same color.

Even though the embeddings show relations between different streamlines, a direct association from a point on the manifold to its corresponding streamline is lost. To cope with this, brushing and linking approaches are used: the user can interactively select a number of embedded points (or the corresponding clusters) on the manifold and get the associated streamlines, or vice versa. Similarly, the user can select a point or streamline as a representative to select the associated cluster. A new clustering can be computed any time, and the user can switch to a partial embedding of a selection of points/streamlines. Another way of interaction is *re-embedding*: the user selects a subset of points from the initial embedding. Then the corresponding subset of streamlines is considered for computing a new embedding with less distortion of distances. We refer to Section 6.4 which shows an example.

5.6 Alternative distance measures

We compare the Hausdorff distance to alternative distance measures summarized in Section 4.3. The sampling strategy remains the same as it is independent of the choice of distance measure. For our experiments, however, we computed the sampling only for the Hausdorff distance and used the same seeds for the other distance measures as this enables a better comparison of clusters on the same set of streamlines. Neither the computation of the embedding nor the clustering are affected by the choice of distance measure.

The computation of mean distance d_m, mean-of-thresholded distance d_m^τ, and minimal distance $d_{\min}$ is very similar to the computation of the Hausdorff distance d_H with similar computational cost.

The computation of the end point distance d_E is performed in constant time independent of the length of streamlines. The same is true for the feature distance d_F after feature descriptors haven been computed once for each streamline; the cost for this computation is not relevant. Both measures, d_E and d_F, are evaluated very efficiently.

6 Results and Discussion

We show 3D streamline embeddings for several synthetic and real-world data sets. We visualize the embeddings together with the streamlines, the colors denote clusters. All times were measured on a Intel Core2 CPU at 2.6GHz (using only a single core).

We use a canonical coordinate system for the visualization of the embeddings. As pointed out in Section 3, the minimizer of (2) is invariant to translation and rotation. In addition, the absolute scale does not convey any relevant information. For this reason, coordinates are uniformly re-scaled to provide good views, and the vectors denoting axis have length one. The

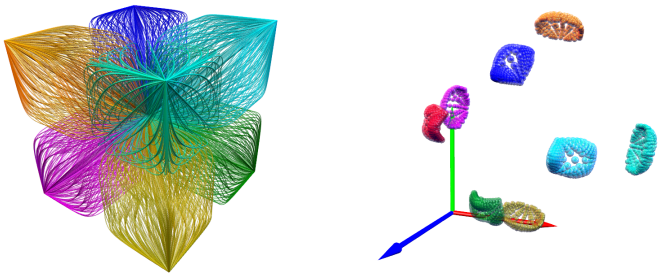

Fig. 10. Vector field (4) and its embedding.

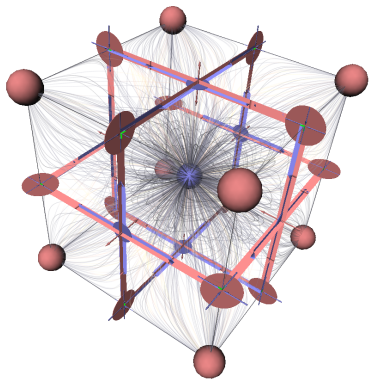

Fig. 11. Visualization of topological structure of vector field (4).

x-axis (red), y-axis (green), and z-axis (blue) refer to the first, second and third columns of the matrix $\mathbf{E}$ (see Section 5.3) and hence eigenvalues $\lambda_1 \leq \lambda_2 \leq \lambda_3$ of $\mathbf{B}$, respectively. This means the x-axis refers to the most relevant dimension, see also Section 6.7.

6.1 Synthetic examples

We start with a simple example, a flow from a source at $(0,0,0)$ into a sink at $(1,1,1)$ in the vector field

$$\mathbf{v}(\mathbf{x}) = \begin{pmatrix} \mathbf{x}_1(1-\mathbf{x}_1) \\ \mathbf{x}_2(1-\mathbf{x}_2) \\ \mathbf{x}_3(1-\mathbf{x}_3) \end{pmatrix} \quad , \quad \mathbf{x} \in [0,1]\times[0,1]\times[0,1] \qquad (3)$$

where $\mathbf{x}_i$ denote the i-th component of $\mathbf{x}$. Figure 9 shows the embedding of 403 streamlines which is essentially a saddle-like 2-manifold surface. Note that for this example we explicitly avoid seeding on planes $\mathbf{x}_i = 0$ and $\mathbf{x}_i = 1$ $(i = 1,2,3)$ because any streamline will always remain within these plane, which yields a set of 2D configurations and the respective 1-manifold embeddings (see Section 4.2). Topologically, these planes act as separatrix planes of the field. Generally seeding exactly on separatrices cannot be avoided. However, we consider such a seeding exceptional and hence unlikely for more complex synthetic fields and for measured flow fields. This example was computed on a $16\times16\times16$ streamline sampling grid with $\delta_{\min} = 0.05$. Sampling took approximately 8 seconds.

We extend this to a more complex example showing the vector field

$$\mathbf{v}(\mathbf{x}) = \begin{pmatrix} -\mathbf{x}_1(1-\mathbf{x}_1)(1+\mathbf{x}_1) \\ -\mathbf{x}_2(1-\mathbf{x}_2)(1+\mathbf{x}_2) \\ -\mathbf{x}_3(1-\mathbf{x}_3)(1+\mathbf{x}_3) \end{pmatrix} \quad , \quad \mathbf{x} \in [-1,1]^3 \qquad (4)$$

consisting of a sink at $(0,0,0)$, 8 sources, 18 saddles, and therefore 8 topological sectors. Figure 10 shows 1,652 sampled streamlines and their embedding. Here the colors show the clusters formed by eight connected components corresponding to the topological sectors. Again, we avoided seeding in certain planes $\mathbf{x}_i = c,\ c \in \{-1,0,1\}$ to avoid degenerate configurations (see previous example). We used the same sampling parameters as above. Sampling took approximately 35 seconds. We used normalized cuts, and the number of clusters was detected automatically. In addition, Figure 11 shows a visualization of the topological structure of this vector field: its critical points are shown as spheres (sources, sinks) and disks (saddles), separation surfaces are indicated as ribbons. The complexity of such direct visualization of topological structure even for simple data indicates limitations for more complex vector fields.

6.2 ABC flow

As a non-trivial example, Figure 12 shows the embedding of the (steady) ABC (Arnold-Beltrami-Childress) flow

$$\mathbf{v}(\mathbf{x}) = \begin{pmatrix} A\sin(x_3) + B\cos(x_2) \\ B\sin(x_1) + C\cos(x_3) \\ C\sin(x_2) + A\cos(x_1) \end{pmatrix} \quad , \quad \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \in [0,2\pi]^3 \ ,$$

with $A = \sqrt{(3)}, B = \sqrt{(2)}$, and $C = 1$. This incompressible and inviscid flow has some theoretical importance in fluid dynamics because it describes a closed-form solution of Euler's equation [37]. It has served as example data set in a number of visualization papers [38], [39]. We tested streamline samples on a regular $15\times15\times15$ grid and choose $\delta_{\min} = 0.075$. The sampling took approximately 63 minutes. Figure 12 shows the resulting 3,958 streamlines and their embedding which clearly exhibits the manifold structure. The normalized cut segmentation detected 17 clusters. We obtain a smaller version with 1,153 streamlines for $\delta_{\min} = 0.1$. The sampling time is reduced to approximately 3 minutes. Figure 20 shows this smaller version (d_H, top left) together with a comparison of distance measures. Note the different embedding due to the lower sampling rate which, however, shows the same manifolds.

6.3 Cylinder flow I

Figures 13 and 14 show a simulated flow around a 3D spherical cylinder. Here we use a single time-step of an unsteady simulation. The data was given on a $128\times128\times128$ grid. We sampled streamlines on a $20\times14\times10$ sampling grid with $\delta_{\min} = 0.05$. Sampling took approximately 33 minutes. Figure 13 shows the resulting 735 streamlines and their embedding.

Figure 14 shows a re-embedding of selected clusters of points/streamlines. With fewer constraints the manifold structure becomes more evident. We apply the same colors as for the original clusters in Figure 13 and use this partial embedding to interactively select clusters. Only streamlines in the selected cluster are displayed.

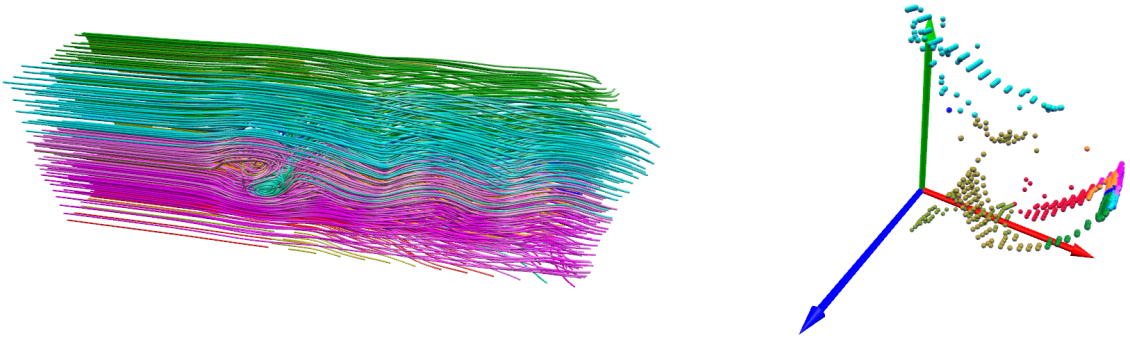

Fig. 13. Left: streamlines of the cylinder flow I and their embedding.

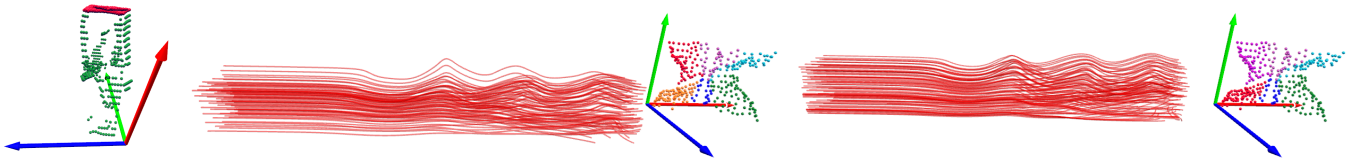

Fig. 14. We compute a new MDS embedding only for a subset of points/streamlines (left, highlighted red) and use it for selection of two clusters (center/right). We show the partial re-embedding, and only the selected streamlines, corresponding to red points, are shown. Here, the two selected clusters separate the flow in a left and a right part.

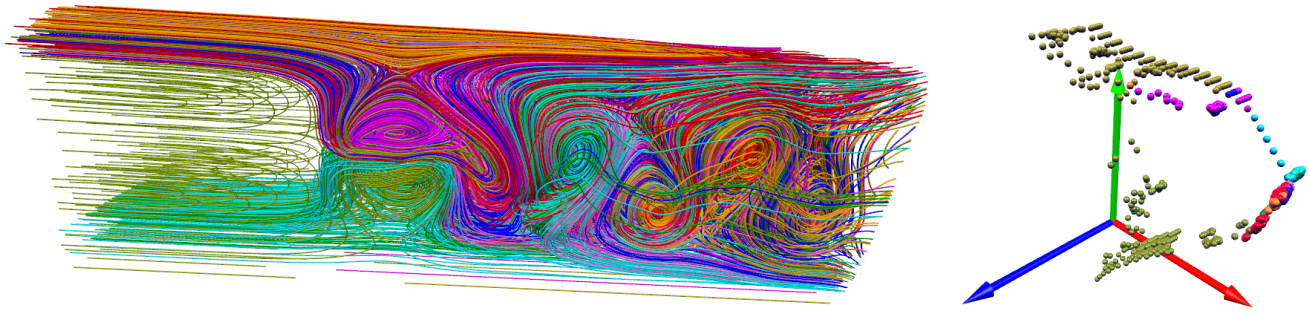

Fig. 15. Streamlines of the cylinder flow II and their embedding.

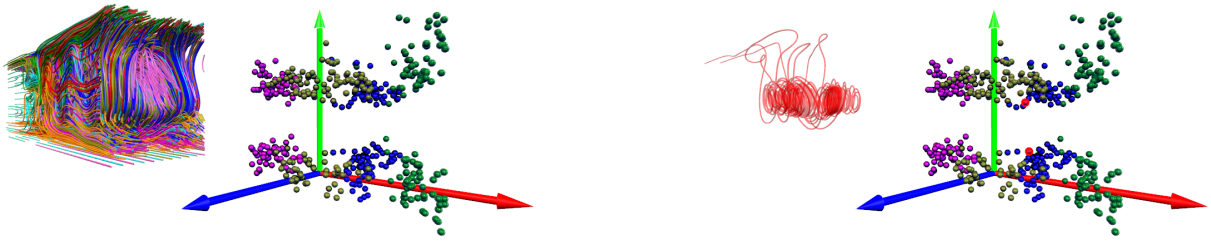

Fig. 16. We compute a re-embedding for a selection (left) and then select two points (in red) to highlight the corresponding streamlines (right)

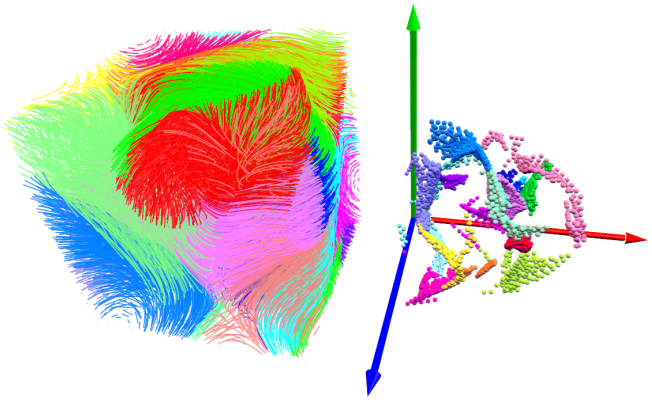

Fig. 12. 3,958 streamlines of the ABC flow and their embedding.

Fig. 18. Streamlines of the aneurysm data set and their embedding.

6.4 Cylinder flow II

This field is the same as cylinder flow I, except for the fact that we subtracted a constant average field describing the ambient part of the flow. This treatment is well-known in the fluid dynamics community. It is motivated by the idea that the observer is moving with the flow. This way we get a cascade of critical points behind the cylinder and therefore a different behavior of the streamlines. We used the same sampling parameters as for the previous data set. Sampling took approximately 52 minutes. Figure 15 shows the resulting 1,251 streamlines and their embedding (center left). For this example, the manifold structure is less intuitive as few points are far away and a large number of points clusters. To get more information, we applied a re-embedding: we selected a subset of points which appeared to collapse in a small region in the original embedding and did a re-embedding for this selection. Figure 16 shows that it unfolds to a 2-manifold.

Figure 16 shows a re-embedding of a part of the streamlines/points (chosen similar as in Figure 14 for the cylinder flow I example). We select two points (in red) and show the corresponding streamlines: as the manifolds separate the fairly symmetric flow into a left and right part, the symmetry of manifolds is reflected in symmetry of respective streamlines.

Figure 17 uses a different partial embedding of a larger subset of streamlines and shows different clusters (points in red, only selected streamlines are shown).

6.5 Aneurysm data

We conclude with a medical data set: the flow data is taken from a simulation of the blood flow in areas containing three cerebral aneurysms. Such aneurysms result from a congenital or evolved weakness of stabilizing parts of the vessel wall and potentially lead to rupture and a life-threatening bleeding [40]. Therefore, the main segmentation task here is to identify regions in which the blood flow enters the aneurysm without leaving it again. The data consists of a single time step of a time-dependent simulation. In order to compute the flow a silicone model of the vessel geometry was constructed. A CT-scan of this aneurysm phantom served as basis for the digital model for the flow simulation. This volumetric model consists of 934,266 cells: prisms and tetrahedra. There is one region of inflow and two regions of outflow. Between these regions there are three aneurysms. The size of the bounding box of the model is 43.6×187.2, we chose $\delta_{\min} = 3$.

For this example it is sufficient to seed streamlines from one or more seeding planes. We placed seeds on two planes, the first plane intersects the vessel near the inflow region, and the second plane intersects near the outflow.

We found 802 streamlines, the sampling took 566 seconds, and 11 clusters were detected. Figure 18 shows the result. One goal of the simulation is to understand which amount of the flow passes the aneurysm under which conditions such as velocity and pressure. The more important goal is to learn how this flow can be manipulated, e.g. by so-called coiling and/or inserting stents, such that flow and pressure in aneurysms become minimal. A clustering of streamlines is helpful for interpretation of the results of the simulation and to achieve these goals. A trivial clustering by inflow and outflow regions is obviously too simple. On the other hand, vector field topology is complex for this data set: it is hard to extract in a robust way and yields too much (unimportant) information. So this is not a practical option, either. Clustering based on the Hausdorff distance, or alternative distance measures, between streamlines leads to reasonable results. The run time for computing clusters is relatively low. And the streamline embedding serves as a visualization of clusters which does not suffer from visual clutter.

6.6 Comparison of distance measures

In this section we evaluate the alternative distance measures: mean distance d_m, mean-of-thresholded distance d_m^τ, minimal distance $d_{\min}$, end point distance d_E, and feature distance d_F (see Section 4.3). For all experiments we choose the threshold $\tau = 10\,\delta_{\min}$ with the same values for $\delta_{\min}$ as for the previous experiments. We use the same set of streamline samples as before (from d_H), timings are summarized in Table 2. For all results in this section we applied normalized cuts for segmentation with automatic detection of the number of clusters ($k_{\max} = 20$). Figure 22 illustrates this process: it shows cost along (y-axis) for k segments (x-axis) for the example data sets and d_H. For comparison values are scaled such that cost for $k = 2$ equals 1.

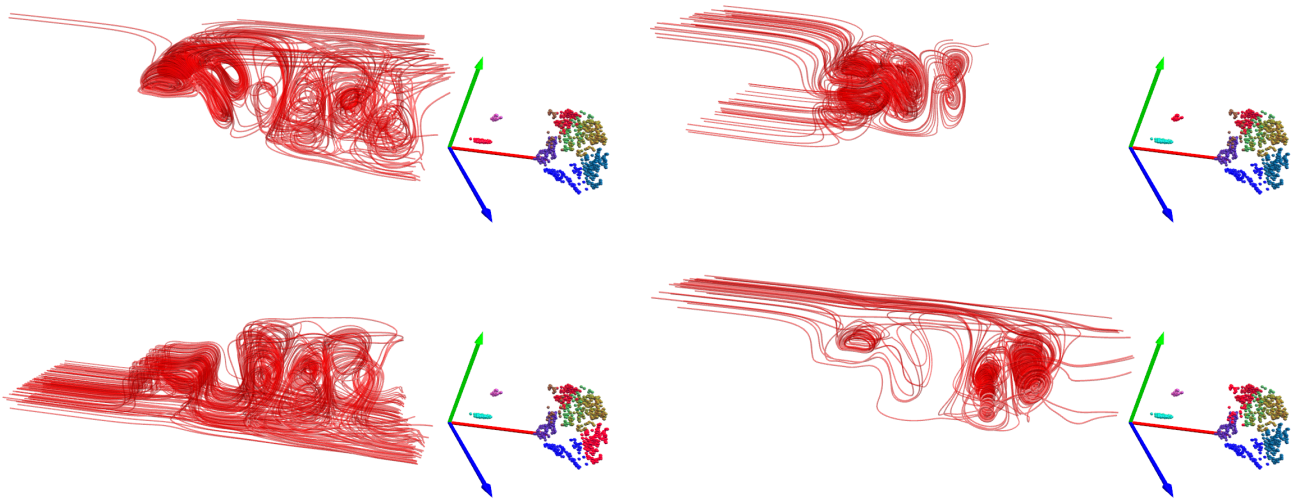

Fig. 17. Selection of some clusters the cylinder flow data. We select clusters on a partial re-embedding. Selected points are displayed in red, together with the streamlines in the selected cluster.

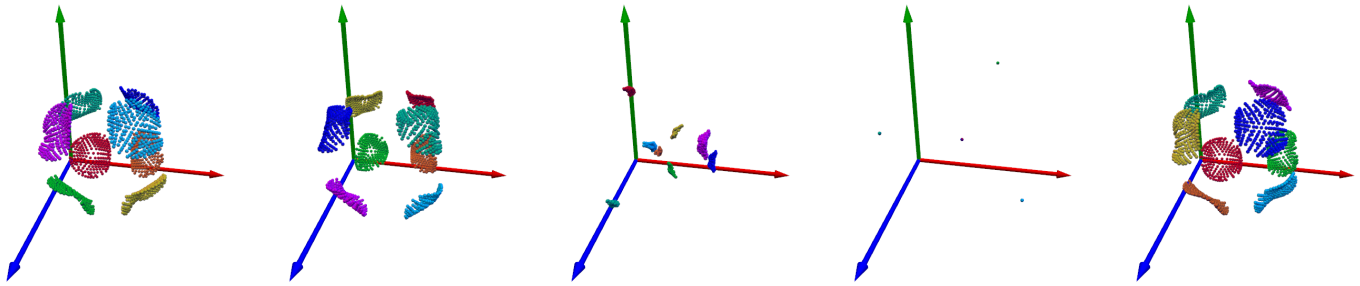

Fig. 19. Embedding using measures (from left) d_m, d_m^τ, $d_{\min}$, d_E, and d_F for synthetic vector field (4).

Figure 19 shows embeddings using different distance measures on the synthetic vector field (4). Figure 10 shows the vector field and embedding for the Hausdorff distance d_H. Note that the embeddings are degenerate for the minimal distance $d_{\min}$ and the end point distance d_E. This is obvious from their definition and the definition of this particular vector field. The same 8 clusters are recognized automatically for all measures.

Figure 20 shows the same comparison for the ABC flow. We include the Hausdorff distance, and normalized cut segmentation yields 17 (d_H), 12 (d_m), 13 (d_m^τ), 15 ($d_{\min}$), 20 (d_E), and 18 (d_F) clusters, respectively. Clustering took between 10 and 25 seconds.

For the Cylinder Flow II data (Figure 21), we obtain 9 (d_H), 11 (d_m), 16 (d_m^τ), 5 ($d_{\min}$), 17 (d_E), and 14 (d_F) clusters, respectively. Clustering took between 9 and 18 seconds.

Finally, we compare distance measures for the Aneurysm data set (without figure): here, we obtain 11 (d_H), 14 (d_m), 5 (d_m^τ), 7 ($d_{\min}$), 10 (d_E), and 12 (d_F) clusters (for all $\rho = 4$), respectively. Clustering took between 5 and 10 seconds.

The figures show different characteristics of the measures: d_m is less sensitive to extrema than the "worst case distance" d_H. As expected, d_m^τ falls in-between d_H and d_m. The minimal distance d_{min} tends to degenerate and does not show any advantage over d_m and d_m^τ at the same computational cost[3]. The end point distance tends to yield similar results as d_H – which can be explained with arguments in Section 4, and which indicates that the Hausdorff distance is often attained for end points of streamlines – but it also tends to degenerate. Finally, the distance d_F based on feature shows good results but is hard to relate to the other measures.

6.7 Dimensionality of data

For the MDS method, the magnitudes of eigenvalues in the diagonal matrix Λ (see Section 5.3) indicate the dimensionality of the data described by the distance matrix $\mathbf{D}$. For an embedding in $\mathbb{R}^3$ we expect a significant drop in magnitude for the 4th eigenvalue λ_4. For all embeddings that are shown, the read, green, and blue axes correspond to λ_1, λ_2, and λ_3, respectively. The global scale is chosen arbitrarily. Figure 23 shows the first 16 eigenvalues for all our examples: the x-axis shows the index i of λ_i, and eigenvalues on the y-axis are normalized such that all $\lambda_1 = 1$ to enable comparison for multiple data sets. These empirical observations coincide with the theoretical analysis in Section 4.2.

3. In fact, evaluation of $d_{\min}$ is cheaper for *sampling* because the $d_{\min}$ threshold is detected immediately.

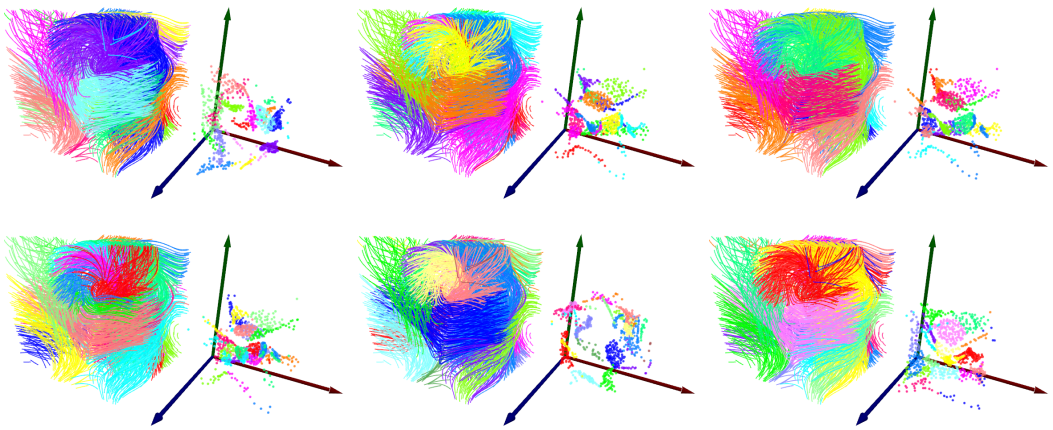

Fig. 20. Embedding using measures (from left), top row: d_H, d_m, d_m^τ; bottom row: $d_{\min}$, d_E, and d_F for the ABC flow.

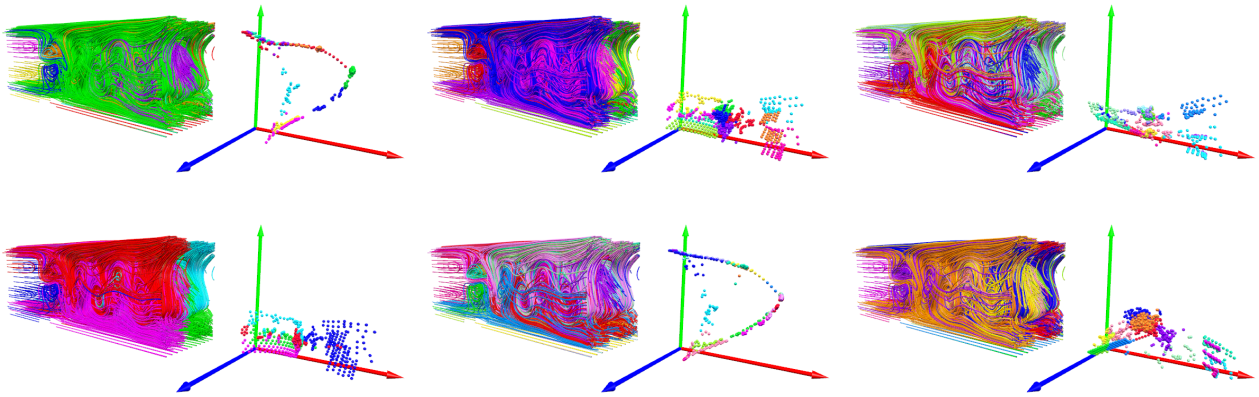

Fig. 21. Embedding using measures (from left), top row: d_H, d_m, d_m^τ; bottom row: $d_{\min}$, d_E, and d_F for the Cylinder flow II data set.

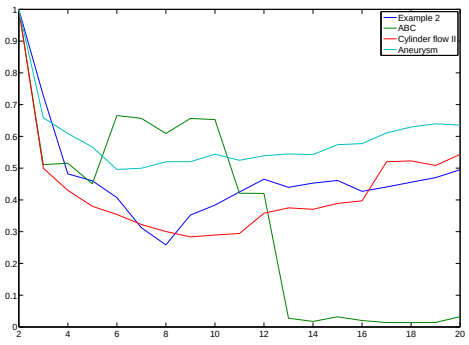

Fig. 22. Normalized cuts: the optimal number k of clusters is determined automatically as global minimum of ncut_k. The diagram shows the cost ncut_k (see [35]) plotted over the number k of clusters for d_H. Values are normalized such that for all data sets $\text{ncut}_2 = 1$.

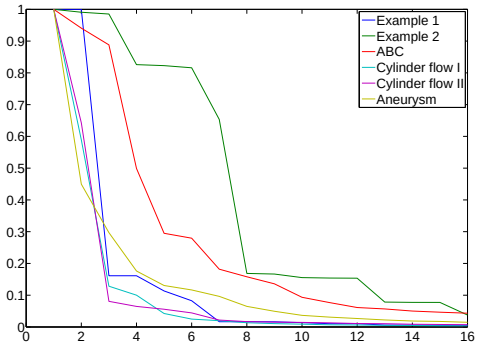

Fig. 23. MDS embedding: largest 16 eigenvalues (normalized) for our examples.

6.8 Performance

We finally report on the performance of our approach and summarize timings. The resulting number of streamlines and timings depend on the data set and the choice of sampling parameters. For all examples we sampled points on streamlines with a regular spacing of $h_s = 0.02$ (except $h_s = 0.005$ for the cylinder flows with a spatial extent of $[0, 1.5] \times [0, 1] \times [0, 0.75]$ and $h_s = 0.5$ for the aneurysm with bounding box diagonal of 289). The computation times for embedding and clustering are

data set	# samples	sampling	MDS	clustering
Example 1	403	8	0.2	–
Example 2	1,652	35	18	19
ABC flow	1,153	182	4.1	16
ABC flow	3,958	3,783	224	54
Cylinder flow I	735	1,965	1.1	12
Cylinder flow II	1,251	3,127	6.8	16
Aneurysm	802	566	1.5	3.3

TABLE 1
Timings: sampling (d_H) and embedding.

data set	#	d_H	d_m	d_m^τ	$d_{\min}$	d_E	d_F
Example 1	403	5.9	5.9	7.5	5.9	0.15	0.14
Example 2	1,652	93	92	151	80	1.6	1.5
ABC flow	1,153	763	795	864	763	1.4	0.98
ABC flow	3,958	7,230	7,323	8,027	7,165	16	14
Cyl. flow I	735	20	20	38	20	0.51	0.51
Cyl. flow II	1,251	593	599	648	595	1.5	1.6
Aneurysm	802	497	298	452	115	5.5	5.4

TABLE 2
Timings: evaluation of distance matrix for different measures.

generally much lower than those for sampling. For MDS, the run time depends on the number m of streamlines, i.e., on the size of the distance matrix.

Tables 1 and 2 summarize timings. All times are given in seconds, the first two columns denote data set and number of streamline samples.

Table 1 shows times for sampling using the Hausdorff distance d_H and times for computing the embedding. The time for clustering (Figures 19–21) is an average for the normalized cuts segmentation with automatic detection of the number of clusters. The maximum number of clusters is $k_{\max} = 20$, i.e., the first 21 eigenvectors of the dissimilarity matrix are computed. Clustering includes few iterative optimizations, we choose reasonably high limits on the maximum number of iterations that always reproduced good results. Equally good results may be obtained for considerably lower limits at lower computational cost.

Table 2 shows times for evaluation of the distance matrix $\mathbf{D}$ for all distance measures. Here, all seed points are given from the initial sampling (using d_H), only pairwise distances are evaluated.

We finally note that we have chosen all parameters very conservatively to sample manifolds faithfully, we do not try to find clever approximations. For instance, simply reducing the sampling rate would increase performance without changing the results very much. Similarly, the resolution of sampling grids could be reduced. In fact, we can always get reasonable previews within seconds to minutes.

7 Limitations

We showed the properties of our technique for a number of examples from simple to fairly complex vector fields. These tests also reveal a number of limitations: computation times for sampling are rather high. This is firstly due to a naïve evaluation of Hausdorff distances (which could be improved, see Section 5). And secondly, the number of distance evaluations is quadratic in the number of streamline samples. The last point could possibly be addressed by only considering distances to certain cluster centers for distant samples similar to multi-pole methods. These are implementation details which refer essentially to a preprocessing stage. Another important issue is that we only have coarse control over sampling. The choice of the grid size h and minimal distance $d_{\min}$ does not yield a good estimate on the resulting number of streamlines (and hence on run-time), neither does it allow for a "perfect" sampling in a sense of surface reconstruction. Furthermore, the computation of the embedding may be somewhat biased due to locally varying sampling density. This problem is hard to solve as there is not enough information available on the streamline spaces. On the other hand it is always possible to oversample data and to then filter results. Such filtering may also be necessary as the MDS embedding requires computation of eigenvalues of dense matrices. For our setup the limiting number is currently near $5,000$ samples for evaluating MDS, in terms of acceptable run-time and memory requirements. Finally, we remark that automatic clustering provides acceptable results in practice. However, clustering algorithms rely on few user-defined parameters. One way to remedy this would be to use multi-scale methods, e.g. [41]. Furthermore, the "decisions" of these algorithms are not always easy to understand. Therefore a manual inspection is recommended, possibly followed by a manual intervention, e.g., to choose a different number of clusters.

8 Conclusions

We presented a new technique for visualizing streamlines in 3D vector fields which is based on mapping streamlines to points in 3D. This map is constrained to preserve Hausdorff distances between streamlines as Euclidean distances between points in least-squares sense. The image of the (infinite) set of streamlines covering the vector field domain is a set of 2-manifolds embedding in $\mathbb{R}^3$. We see the following advantages making our approach an alternative to existing flow visualization tools:

- Visualizing manifolds is less prone to visual clutter than visualizing stream lines directly.
- Surface topology – connected components – provides a topological segmentation of the vector field.
- Surface geometry – distance in manifolds – encodes similarity of global streamline behavior.
- Manifolds serve as 2D parametrization of the field, enabling selection of neighborhoods or mapping of additional information.

We remark that our approach does not make any assumptions on the dimensionality of the data: 2D data are handled exactly the same way as 3D data. The concept theoretically extends to higher dimensions, however, interpretation, e.g., of 4D streamlines for unsteady flows is non-trivial. Similarly, alternative distance measures can be applied. We give few examples and comparisons. We prove theoretical properties of our approach for the Hausdorff distance: streamline embedding is related to vector field topology, k-dimensional topological segments

are mapped to $(k-1)$-dimensional manifolds. This is backed by our empirical results. On the practical side, however, the current approach to manifold sampling is slow, and developing advanced sampling methods would constitute a challenging direction for future work.

ACKNOWLEDGMENTS

The authors would like to thank Thomas Schultz for helpful discussions and comments as well as the anonymous reviewers for their constructive comments. Thanks to Rocco Gasteiger and Gábor Janiga for providing the aneurysm data set.

REFERENCES

[1] B. Heckel, G. Weber, B. Hamann, and K. I. Joy, "Construction of vector field hierarchies," in *VISUALIZATION '99: Proceedings of the 10th IEEE Visualization 1999 Conference (VIS '99)*, 1999.

[2] A. Telea and J. J. van Wijk, "Simplified representation of vector fields," in *VISUALIZATION '99: Proceedings of the 10th IEEE Visualization 1999 Conference (VIS '99)*. Washington, DC, USA: IEEE Computer Society, 1999.

[3] T. Preusser and M. Rumpf, "Anisotropic nonlinear diffusion in flow visualization," in *VISUALIZATION '99: Proceedings of the 10th IEEE Visualization 1999 Conference (VIS '99)*, 1999.

[4] M. Griebel, T. Preusser, M. Rumpf, M. A. Schweitzer, and A. Telea, "Flow field clustering via algebraic multigrid," in *VIS '04: Proceedings of the conference on Visualization '04*. Washington, DC, USA: IEEE Computer Society, 2004, pp. 35–42.

[5] J. Hunt, "Vorticity and vortex dynamics in complex turbulent flows," *Proc CANCAM, Trans. Can. Soc. Mec. Engrs*, vol. 11, p. 21, 1987.

[6] J. Jeong and F. Hussain, "On the identification of a vortex," *J. Fluid Mechanics*, vol. 285, pp. 69–94, 1995.

[7] J. Helman and L. Hesselink, "Representation and display of vector field topology in fluid flow data sets," *IEEE Computer*, vol. 22, no. 8, pp. 27–36, August 1989.

[8] ——, "Visualizing vector field topology in fluid flows," *IEEE Computer Graphics and Applications*, vol. 11, pp. 36–46, May 1991.

[9] G. Scheuermann, H. Hagen, H. Krüger, M. Menzel, and A. Rockwood, "Visualization of higher order singularities in vector fields," in *Proc. IEEE Visualization '97*, R. Yagel and H. Hagen, Eds., 1997, pp. 67–74.

[10] K. Mahrous, J. Bennett, B. Hamann, and K. Joy, "Improving topological segmentation of three-dimensional vector fields," in *Data Visualization 2003. Proc. VisSym 03*, 2003, pp. 203–212.

[11] H. Theisel, T. Weinkauf, H.-C. Hege, and H.-P. Seidel, "Saddle connectors - an approach to visualizing the topological skeleton of complex 3D vector fields," in *Proc. IEEE Visualization 2003*, 2003, pp. 225–232.

[12] F. Post, B. Vrolijk, H. Hauser, R.Laramee, and H. Doleisch, "The state of the art in flow visualization: Feature extraction and tracking," in *Computer Graphics Forum*, vol. 22, no. 4, 2003, pp. 775–792.

[13] R. S. Laramee, H. Hauser, L. Zhao, and F. H. Post, "Topology-based flow visualization, the state of the art," in *Topology-based Methods in Visualization*, ser. Mathematics and Visualization, H. Hauser, H. Hagen, and H. Theisel, Eds. Springer, 2007, pp. 1–19, topo-In-Vis 2005.

[14] A. Pobitzer, R. Peikert, R. Fuchs, B. Schindler, A. Kuhn, H. Theisel, K. Matkovic, and H. Hauser, "On the way towards topology-based visualization of unsteady flow – the state of the art," in *Eurographics 2010 - State of the Art Reports*, H. and E. Reinhard, Eds., April 2010, pp. 137–154.

[15] T. Salzbrunn and G. Scheuermann, "Streamline predicates," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, pp. 1601–1612, 2006.

[16] J. Blaas, C. P. Botha, B. Peters, F. M. Vos, and F. H. Post, "Fast and reproducible fiber bundle selection in dti visualization," *Visualization Conference, IEEE*, pp. 59–64, 2005.

[17] A. Sherbondy, D. Akers, R. Mackenzie, R. Dougherty, and B. Wandell, "Exploring connectivity of the brain's white matter with dynamic queries," *IEEE Transactions on Visualization and Computer Graphics*, vol. 11, no. 4, pp. 419–430, 2005.

[18] L. O'Donnell and C.-F. Westin, "White matter tract clustering and correspondence in populations," in *Eighth International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'05)*, ser. Lecture Notes in Computer Science 3749, 2005, pp. 140–147.

[19] W. Chen, S. Zhang, S. Correia, and D. S. Ebert, "Abstractive representation and exploration of hierarchically clustered diffusion tensor fiber tracts," *Comput. Graph. Forum*, vol. 27, no. 3, pp. 1071–1078, 2008.

[20] W. Chen, Z. Ding, S. Zhang, A. MacKay-Brandt, S. Correia, H. Qu, J. A. Crow, D. F. Tate, Z. Yan, and Q. Peng, "A novel interface for interactive exploration of DTI fibers," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1433–1440, 2009.

[21] B. Jobard and W. Lefer, "Creating evenly-spaced streamlines of arbitrary density," in *Proceedings 8th Eurographics Workshop on Visualization in Scientific Computing*, 1997, pp. 57–66.

[22] G. Turk and D. Banks, "Image-guided streamline placement," in *Proc. Siggraph '96*, 1996, pp. 453–460.

[23] V. Verma, D. Kao, and A. Pang, "A flow-guided streamline seeding strategy," in *VIS '00: Proceedings of the conference on Visualization '00*, 2000, pp. 163–170.

[24] T. Cox and M. Cox, *Multidimensional Scaling*. Chapman and Hall, 2001.

[25] T. Schultz, "Feature extraction for dw-mri visualization: The state of the art and beyond," in *Proc. Schloss Dagstuhl Scientific Visualization Workshop 2009*, 2010, p. to appear.

[26] A. Brun, H.-J. Park, H. Knutsson, and C.-F. Westin, "Coloring of DT-MRI fiber traces using laplacian eigenmaps," in *Computer Aided Systems Theory - EUROCAST*, ser. Lecture Notes in Computer Science, R. Moreno-Díaz and F. Pichler, Eds., vol. 2809. Springer, 2003, pp. 518–529.

[27] B. Moberts, A. Vilanova, and J. J. van Wijk, "Evaluation of fiber clustering methods for diffusion tensor imaging," in *IEEE Visualization*. IEEE Computer Society, 2005, pp. 65–72.

[28] I. Corouge, G. Gerig, and S. Gouttard, "Towards a shape model of white matter fiber bundles using diffusion tensor MRI," in *IEEE International Symposium on Biomedical Imaging (ISBI)*, 2004, pp. 344–347.

[29] S. Zhang, S. Correia, and D. H. Laidlaw, "Identifying white-matter fiber bundles in dti data using an automated proximity-based fiber-clustering method," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, pp. 1044–1053, 2008.

[30] A. Brun, H. Knutsson, H.-J. Park, M. E. Shenton, and C.-F. Westin, "Clustering fiber traces using normalized cuts," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI*, ser. Lecture Notes in Computer Science, C. Barillot, D. R. Haynor, and P. Hellier, Eds., vol. 3216. Springer, 2004, pp. 368–375.

[31] D. M. Mount and S. Arya, "Ann: A library for approximate nearest neighbor searching," http://www.cs.umd.edu/ mount/ANN/, Tech. Rep., 2010.

[32] H. Alt, M. Godau, C. Knauer, and C. Wenk, "Computing the Hausdorff distance of geometric patterns and shapes," *Discrete and Computational Geometry – The Goodman-Pollack-Festschrift*, pp. 65–76, 2002.

[33] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Advances in Neural Information Processing Systems 14*. MIT Press, 2001, pp. 585–591.

[34] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems 14*. MIT Press, 2001, pp. 849–856.

[35] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[36] H. Edelsbrunner and E. P. Mücke, "Three-dimensional alpha shapes," *ACM Trans. Graph.*, vol. 13, no. 1, pp. 43–72, 1994.

[37] G. Haller, "An objective definition of a vortex," *J. Fluid Mech.*, vol. 525, pp. 1–26, 2005.

[38] F. Sadlo and R. Peikert, "Visualizing Lagrangian Coherent Structures and Comparison to Vector Field Topology," in *Topology-Based Methods in Visualization II*, H.-C. Hege, K. Polthier, and G. Scheuermann, Eds. Springer, 2008, pp. 15–30.

[39] K. Shi, H. Theisel, T. Weinkauf, H.-C. Hege, and H.-P. Seidel, "Visualizing transport structures of time-dependent flow fields," *IEEE Computer Graphics & Applications*, pp. 24–36, 2008.

[40] M. Neugebauer, R. Gasteiger, O. Beuing, V. Diehl, M. Skalej, and B. Preim, "Combining Map Displays and 3D Visualizations for the Analysis of Scalar Data on Cerebral Aneurysm Surfaces," in *Computer Graphics Forum (EuroVis)*, vol. 28 (3), Berlin, 10.-12. Juni 2009, pp. 895–902.

[41] T. Cour, F. Bénézit, and J. Shi, "Spectral segmentation with multiscale graph decomposition," in *CVPR*, 2005, pp. 1124–1131.