Streamline Variability Plots for Characterizing the Uncertainty in Vector Field Ensembles

Florian Ferstl, Kai Bürger and Rüdiger Westermann

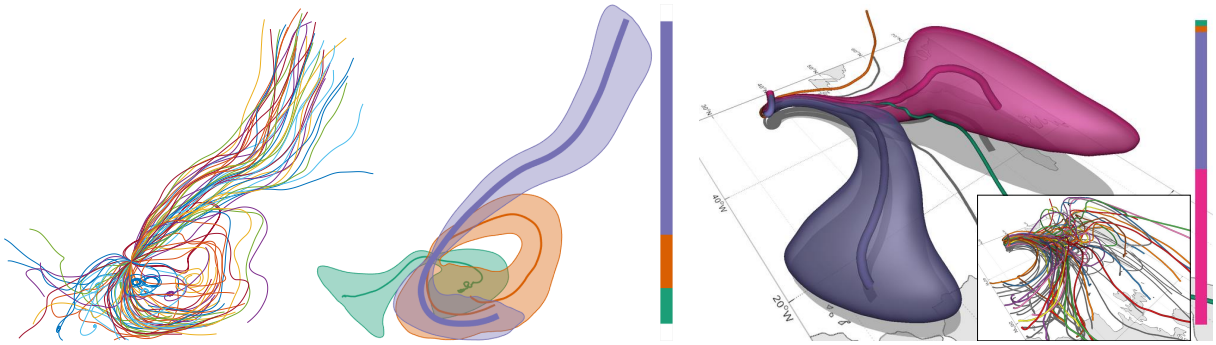


Fig. 1. From a set of streamlines in an ensemble of vector fields (left), our method generates an abstract visualization of the major trends in this set (middle). For each trend, a region of high confidence and a representative streamline-median is extracted. The relative strength of a trend is indicated by the thickness of its median line and by the bar plot on the right. Our method works in 2D and 3D (right), as well as for particle trajectories in time-dependent fields.

Abstract—We present a new method to visualize from an ensemble of flow fields the statistical properties of streamlines passing through a selected location. We use principal component analysis to transform the set of streamlines into a low-dimensional Euclidean space. In this space the streamlines are clustered into major trends, and each cluster is in turn approximated by a multivariate Gaussian distribution. This yields a probabilistic mixture model for the streamline distribution, from which confidence regions can be derived in which the streamlines are most likely to reside. This is achieved by transforming the Gaussian random distributions from the low-dimensional Euclidean space into a streamline distribution that follows the statistical model, and by visualizing confidence regions in this distribution via iso-contours. We further make use of the principal component representation to introduce a new concept of streamline-median, based on existing median concepts in multidimensional Euclidean spaces. We demonstrate the potential of our method in a number of real-world examples, and we compare our results to alternative clustering approaches for particle trajectories as well as curve boxplots.

Index Terms—Ensemble visualization, uncertainty visualization, flow visualization, streamlines, statistical modeling.

1 Introduction

Exponentially increasing computing power has led to continuous improvements in computational field dynamics over many years, but nonetheless numerical simulations are sometimes strikingly poor. One reason is the highly non-linear and often chaotic nature of the governed physical processes, which makes some situations intrinsically hard to predict. One great challenge today is to identify the limits of predictability in different situations and produce the best estimations that are physically possible, often with only limited knowledge about the initial conditions and the dynamical laws governing the field's temporal evolution.

In many scientific fields, the recognition that predictability is limited has led to a paradigm shift in how predictions of dynamic processes are created. Instead of making a single deterministic computation of the future field state, ensembles of many numerical simulations are computed—based on a set of possible initial states and random variations to account for model uncertainty—and predictions take the form of probabilities of occurrence of specific features derived from the simulated fields. For instance, the North American Ensemble Forecast System and the ensemble prediction systems of the European Centre for Medium-Range Weather Forecasts (ECMWF) routinely generate between 25 and 51 forecast runs.

To analyze the uncertainty that is represented by an ensemble, the variability of the ensemble members need to be characterized, and the major trends and outliers in the shape and spatial location of relevant features need to be determined. A well-known visual analysis technique for ensembles of features are so-called spaghetti plots, overlaid plots—typically in 2D—of features like particle trajectories or iso-contours in individual members of an ensemble field. Especially in the atmospheric domain, spaghetti plots are one of the most popular means for variability analysis. Spaghetti plots, on the other hand, can produce visual clutter when many features overlap, and they cannot easily convey major trends, outliers, and statistical properties of the feature distribution. To overcome some of these limitations, Whitaker et al. [45] and Mirzargar et al. [23] have recently introduced contour and curve box-plots, respectively, for the visualization of curve-like features based on the concept of statistical data depth.

Our contribution: Our work extends on previous approaches in that it introduces a new method to statistically model the variability of specific features among the ensemble members, so that the probability of a particular feature situation can be estimated from the ensemble. We concentrate on the visual analysis of streamlines in 2D and 3D flow fields, and visualize the statistical properties of streamlines passing through a selected location. We first transform the feature representation into a low-dimensional Euclidean space, in which a distance metric as well as an ordering of the features along each dimension is

• *Florian Ferstl, Kai Bürger and Rüdiger Westermann are with the Computer Graphics and Visualization Group, Technische Universität München. E-mail: ferstlf@in.tum.de, buerger|westermann@tum.de*

given. We employ this to derive a statistical model of the distributions of clusters of similar streamlines in the Euclidean space, and we propose a method to transform the resulting distributions into confidence regions—so called *lobes*—of the streamlines in the spatial domain. We call the resulting visualizations *streamline variability plots*. Our particular contributions are

- the use of principal component analysis (PCA) to convert streamlines into a structure preserving Euclidean space (PCA-space), and the clustering in this space to detect trends and outliers in the set of streamlines,
- a new concept of streamline-median, which is based on the existing concept of the (multidimensional) geometric median,
- a non-parametric probabilistic model of the clustered streamline distributions in PCA-space, and a new approach to transform the probabilistic model in this space into a streamline distribution in the spatial domain,
- a visualization method for confidence lobes in 2D and 3D to indicate an estimated range of locations which includes all streamlines within a prescribed standard deviation.

In particular we will demonstrate, that this new approach adheres to the requirements on uncertainty visualization techniques proposed by Whitaker et al. [45], as it conveys statistical properties of the shapes of streamlines, provides a qualitative abstract and quantitative statistical interpretations of streamlines, and reveals major trends and outliers in the initial data.

We demonstrate the potential of our method in a number of real-world examples, and we compare our results to alternative line clustering approaches as well as curve boxplots [23]. Moreover, we show that all involved operations can be computed fast enough to allow for an interactive exploration of even 3D vector-valued ensembles to identify the sources and evolution of uncertainty in streamlines.

2 Related Work

Our technique uses streamline clustering and probabilistic streamline estimation for ensemble visualization, taking into account the similarity and frequency of occurrence of streamlines over multidimensional intervals. The technique has overlap with techniques in uncertainty and ensemble visualization, and curve clustering:

Uncertainty and Ensemble Visualization: The visualization of uncertainty belongs to the top challenges in scientific visualization. For the most recent survey on the topic let us refer to the book by Bonneau et al. [3]. In uncertainty visualization one usually assumes a stochastic uncertainty model, instead of a set of possible data occurrences as in ensemble visualization.

To visualize the effect of uncertainty on the position and structure of relevant features such as iso-contours, previous works have used confidence envelopes [29, 48], surface displacements [14], and made use of the concept of animation [5, 20]. The concept of numerical condition was introduced to extract level-set features in uncertain scalar fields [32], and it was further extended to account for correlations in the data [34, 31], as well as to also consider non-parametric models for uncertainty [33].

The concepts of stream lines and critical points has been generalized to uncertain (Gaussian) vector field topology, in order to segment the topology by integrating particle density functions [28]. Probabilistic local features, such as critical points, have been extracted from Gaussian distributed vector fields using Monte Carlo sampling [30]. In a fuzzy topology, the topological decomposition is performed by growing streamwaves, based on a representation for vector fields called edge maps [2].

Obermaier and Joy [26] have classified ensemble visualization techniques into feature-based and location-based approaches. The latter analyze and compare data properties at fixed locations in the domain using descriptive statistics. Feature-based approaches analyze domain-specific features which are first extracted from the individual ensemble members. The visualization of feature variability in ensemble fields is often performed via spaghetti plots of selected contour lines or threshold probabilities of 2D fields such as surface wind speed [35, 46]. Glyphs and confidence ribbons were introduced to emphasize the Euclidean spread of contour ensembles [40]. Recently, Whitaker et al. [45] and Mirzargar et al. [23] built upon the ordering of multivariate data using the concept of statistical band depth to enable an improved visualization of the uncertainty in spaghetti plots of ensembles of curves. Locations in 3D flow fields were clustered based on the divergence of transport patterns to analyze trends in flow ensembles [16].

Curve Clustering: Our work is related to clustering approaches for curves in 2D and 3D space, which group a set of curves into similar sub-sets based on a given similarity measure. For a general overview of clustering techniques let us refer to the overview article by Jain [18]. For the clustering of curves, most often geometric similarity measures have been employed, for instance, based on Euclidean distances [9, 36], curvature and torsion signatures [47, 21], predicates for stream- and pathlines based on flow properties along these lines [4], or user-selected streamline predicates [39]. For a good overview of similarity measures using geometric distances between curves let us refer to the comparative study by Zhang et al. [50].

Integral curves in flow fields have been clustered using a two-stage approach, by first performing a geometry-based coarse grouping of streamlines, and then clustering in a low-dimensional Euclidean space comprising streamline properties based on shape and velocity [8]. The pairwise Hausdorff metric between streamlines has been employed to project streamlines into an Euclidean space and perform spectral graph clustering in this space [36]. For curves, Agglomerative Hierarchical Clustering (AHC) with different cluster proximity measures has shown to be effective [21, 47]. Different clustering approaches and similarity measures for fiber tracts in Diffusion Tensor Imaging (DTI) data have been evaluated [24], among them shared nearest neighbor clustering and AHC. A geometry-based similarity metric considering partial intervals for fiber tracts in DTI data has been used in AHC to cluster such tracts [49]. A reduction technique called Laplacian eigenmaps has been applied to transform fiber tracks to a low dimensional Euclidean space [6]. Recently, an evaluation of different clustering approaches for streamlines using geometry-based similarity measures has been performed [27].

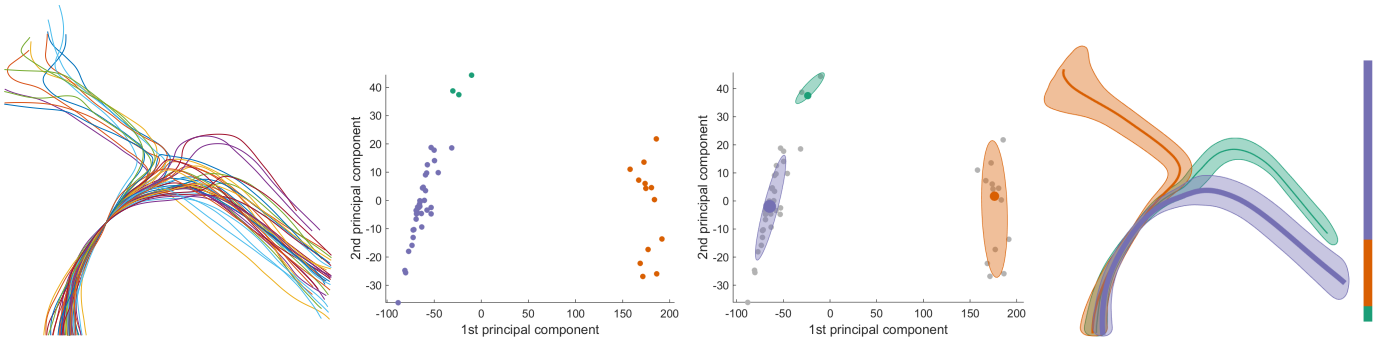
In some of the previous works, curves have been reduced to low-dimensional representations, for instance by using single measures of geometric similarity. This can result in a significant amount of information that is lost, and usually the initial data can no longer be reconstructed from the reduced representation. It is worth noting that our approach overcomes both of these limitations.

Related to the clustering of integral lines in flow fields are approaches performing clustering of vector fields based on local coherent regions, e.g., by merging locations which are similar in position and orientation of the vectors [43], by splitting regions where the differences between streamlines in these regions and streamlines in an approximated flow field are large [15], by using anisotropic diffusion to automatically cluster regions of strong correlation in the flow data [13], and by clustering trajectories into sets of vector fields [12]. For an overview of approaches for vector field clustering let us also refer to the survey by Salzbrunn et al. [38].

Mostly related to our approach is the method initially proposed by Bashir et al. [1], and later evaluated in the report by Zhang et al. [50], where PCA was used in combination with Euclidean k-means clustering to group pedestrian trajectories which were extracted from surveillance videos. Our work builds upon this approach, yet we propose a number of modifications and extensions to better reveal trends and enable the construction of probabilities of occurrence of streamlines.

3 Overview

Our method takes as input a set of n streamlines of m vertices each, which were generated by starting a particle integration at the same location in an ensemble of n vector fields (see Fig. 2(a)). We will



(a) Input (b) PCA + Clustering (c) Confidence Ellipses + Geom. Medians (d) Variability Plot

Fig. 2. Method overview: (a) The initial set of streamlines. (b) PCA transforms lines into an Euclidean space—PCA-space—in which clustering can be performed. (c) Multivariate normal distributions—represented by confidence ellipses and geometric (cluster) medians—are fitted to the points in PCA- space. (d) Medians and ellipses are transformed back to the domain space and yield the variability plot of the streamline ensemble.

also show an example where the streamlines are generated by slightly varying the start position to indicate the effect of these variations on the streamline distribution. In all of our examples, we use a fix-step numerical integration scheme for computing the streamlines, and we parameterize the streamlines via the integration time along the curves. Our method performs a number of operations on the initial streamline set, which are illustrated in Fig. 2.

Each trajectory can be seen as a point in a $(d \cdot m)$-dimensional vector space (d is the dimension of the streamline vertices), and this high dimensionality makes processing methods such as finding similarities difficult. Therefore, we first reduce the dimensionality in a statistically optimal way, by projecting the streamlines onto a low-dimensional orthogonal subspace that captures as much of the variation of the initial streamlines as possible. This is illustrated in Fig. 2(b). In a mean-square sense, the best way to do the projection is PCA, which transforms the streamlines into a simpler representation in an Euclidean space. We will subsequently call this space the *PCA-space*.

To perform a streamline PCA, streamlines are linearized into the rows of a $n \times (d \cdot m)$ matrix. Therefore, all streamlines should have the same number of vertices. However, this is not always the case, because streamlines leave the domain early or might terminate in critical points. Our approach is to fill the missing positions in the matrix by repeating the last streamline vertex. This vertex repetition doesn't introduce new information, because the additional vertices are perfectly correlated, and exactly this can be handled well by PCA. Even though more advanced possibilities exist, for instance, to continue the streamlines with the speed and direction of the last vertex, we have found that neither of them has a significant impact on the clustering and, most importantly, on the appearance of the resulting variability plot.

In the PCA-space the streamlines are clustered into major trends using an appropriate clustering scheme, and for each cluster a multivariate normal distribution—represented by a confidence ellipse and a geometric cluster median in Fig. 2(c)—is fitted to the points. Here, a confidence ellipse describes the set of points that are closer to the mean than a given amount of standard deviations.

Conceptually, the statistical distribution of each cluster is now transformed back into the high-dimensional input space, yielding the corresponding set of streamlines. This is illustrated in Fig. 2(d), where the streamlines correspond to the cluster medians, and the lobes correspond to the streamlines that are within a selected range of standard deviations. Since the operations in PCA-space are performed for each cluster separately, one lobe and one line are generated for every cluster, giving the final streamline variability plot.

3.1 PCA

PCA is a powerful technique for extracting structure from possibly high-dimensional data sets. It is performed by solving an Eigenvalue problem or, alternatively, by computing a singular value decomposition. PCA is an orthogonal transformation of the coordinate system in which the initial data is described. It is often the case that in the new coordinate system a small subset of coordinates is sufficient to account for most of the structure in the data.

PCA is a standard technique in statistics and many other fields, and we will only briefly describe the underlying principles here. We will, however, put special emphasis on the discussion of how the results of a streamline PCA can be interpreted, and how these results can be employed for streamline clustering.

In the following discussion, the i-th streamline is denoted $\mathbf{s}_i$, and every streamline comprises m vertices. Let us note that, in common PCA terminology, each streamline corresponds to an observation, and each vertex corresponds to (multiple) variables. PCA transforms the n streamlines into an equivalent $(n-1)$-dimensional representation by computing their principal component scores, i.e., the scalar values by which each principal component is weighted to obtain the streamline as a linear combination of these components.

PCA starts by subtracting from every vertex the mean value of all vertices. In our application this has the effect that every streamline is characterized by its offset from the mean streamline $\bar{\mathbf{s}} = \frac{1}{n}\sum_i \mathbf{s}_i$. Consequently, this leads to the following representation, which we will refer to as the PCA-space representation of the streamlines:

$$\mathbf{s}_i = \bar{\mathbf{s}} + \sum_{j=1}^{n-1} c_{ij}\mathbf{u}_j. \tag{1}$$

The unit vectors $\mathbf{u}_j$ are the principal components (PC), and the coefficients c_{ij} are the principal component scores. The scores are sorted in descending order of importance, such that $\mathbf{u}_1$ is the direction in which the points $\mathbf{s}_i$ have the largest variance, $\mathbf{u}_2$ is the direction in which the points $\mathbf{s}_i$ have the second largest variance, and so on. Because all $\mathbf{s}_i$ were zero-centered beforehand, we only need up to $n-1$ basis vectors to represent all streamlines. An important property of PCA in our application is that the PCs form an orthonormal basis of the streamline space $\mathbb{R}^{d\cdot m}$. This means that the PCA-space, in which each streamline i is uniquely defined by its PC scores c_{ij}, is an Euclidean space which is equivalent to the original streamline space. I.e., many operations like clustering based on Euclidean distances give identical results in this alternative representation. Let us also note here, that through Eq. (1) it is possible to transform a point in the PCA-space into the corresponding representation in the streamline space. We will make use of this to generate streamline variability plots from a statistical model of the scores in the PCA-space.

In many situations where PCA is used for a statistical data analysis, only the PC scores are investigated and visualized. On the other hand, the PCs themselves are often helpful to analyze specific physical features in multiple spatially correlated physical fields. For instance, in fluid mechanic, where PCA is known as Proper Orthogonal Decomposition (POD) [19], periodic patterns can be extracted from turbulent flows as principal components of the time-varying velocity field [22, 25]. In meteorology, where PCA is known under the term

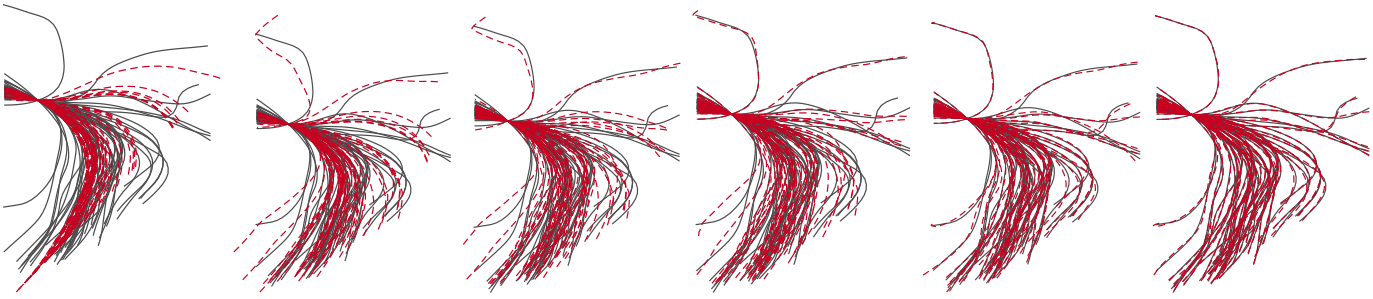


Fig. 3. Reconstruction quality: The original streamlines are shown in gray, reconstructions using an increasing numbers r of PCs are shown as dashed, red lines. The corresponding amount of explained variance $\mathrm{ex}(r)$ is given in parenthesis (see Eq. (2)).

Empirical Orthogonal Functions (EOF) [46], PCA is used to extract atmospheric phenomena as principal components of scalar field ensembles like geopotential height and temperature [44].

The mentioned applications exploit the property of PCA to capture the dominant low-frequency structures in the first PCs, while random fluctuations are expressed in the remaining modes. This effect can also be observed when decomposing streamlines into their PCs, since streamlines can also be considered a type of spatially correlated data. Once a PCA of streamlines has been computed, the streamline representation can be reduced to an optimal low-rank approximation, by using only the dominant PCs. This is demonstrated in Fig. 4, where the first three PCs of a set of 2D streamlines are shown. It can be observed that the first PCs correspond to streamlines exhibiting very low frequency variations. Furthermore, the third PC crosses over the mean line while the first two PCs do not, which indicates increasing spatial frequency in the higher modes. If more PCs were shown, ever more oscillations around the mean line could be observed.

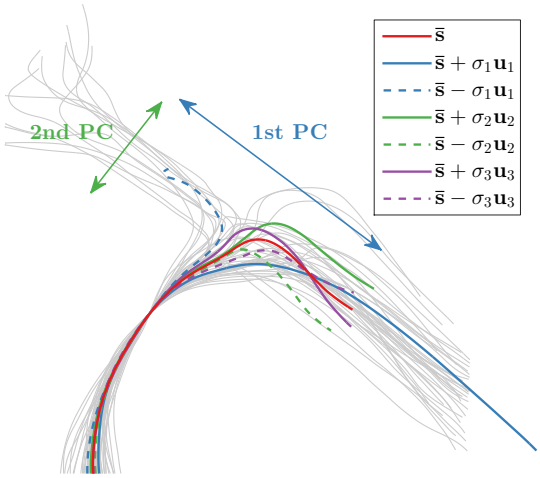


Fig. 4. Mean curve $\bar{\mathbf{s}}$ (red) and first three principal components (PC) $\mathbf{u}_1$-$\mathbf{u}_3$ of the 2D streamlines in gray. PCs have to be considered as offsets to the means curve, so $\bar{\mathbf{s}} \pm \sigma_j \mathbf{u}_j$ are shown instead of $\mathbf{u}_j$. Scaling factors σ_j are the standard deviations of the corresponding PC scores c_{ij}, which emphasize their relative importance. The first PC captures the general deviation of the streamlines in top-left/bottom-right direction, the second PC captures the less important deviation in bottom-left/top-right direction (two-sided arrows). The major trends in the set of streamlines are well represented by the first two PCs.

To obtain an optimal low-rank approximation, one just has to restrict the sum in Eq. (1) to the first r components. It can be shown, that the resulting approximations are optimal in a least squares sense, i.e., they minimize the reconstruction error

$$\sum_{i=1}^{n} \left\| \left(\bar{\mathbf{s}} + \sum_{j=1}^{r} c_{ij}\mathbf{u}_j \right) - s_i \right\|^2 .$$

This leaves the question how to determine the appropriate number of components. We use one of the most common cutoff-criteria, by looking at the amount of explained variance that is represented by different choices of r. Let $\sigma_j^2 = \mathrm{var}(c_{1j}, \ldots, c_{nj})$ denote the variance of the j-th PC (notice that the corresponding mean values are all zero). Then the amount of explained variance by the first r components is

$$\mathrm{ex}(r) = \sum_{j=1}^{r} \sigma_j^2 \Big/ \sum_{j=1}^{n-1} \sigma_j^2 . \qquad (2)$$

For a given explained variance threshold τ, the number of components is chosen as the smallest r for which $\mathrm{ex}(r)$ is greater or equal than τ. Since we perform two different tasks in rank-reduced spaces, which require different degrees of precision, we also use two different thresholds in our approach. On the one hand, for clustering (see Section 3.2), we have found $\tau_1 = 0.99$ to be sufficient. For generating the final plots via splatting of streamlines into a discrete grid (see Section 4), on the other hand, slightly more detail is often required, and we use $\tau_2 = 0.999$. In several of our experiments this leads to only three or four PCs that had to be considered, and we never used more than eight PCs in any of our experiments. The resulting approximation errors are depicted in Fig. 3.

3.2 Clustering

Once the streamlines have been transformed into the reduced PCA-space of rank r_1, each streamline is represented by a tuple $\mathbf{c}_i = (c_{i1}, \ldots, c_{ir_1})$, i.e., by a (r_1)-dimensional point in PCA-space. Our goal now is to derive a statistical model of the streamline distribution in this space. We could model our multidimensional data via a single multivariate normal (MVN) distribution, yet often the data includes significantly different trends, showing up as multiple distinct clusters in the PCA-space. A common remedy to this problem is to use a Gaussian mixture model (GMM), which represents the multi-dimensional Probability Density Function (PDF) by a weighted sum of multiple MVN distributions. The Gaussian mixture model is parameterized by the mean vectors, covariance matrices, and mixture weights from all component densities.

A straight forward approach to find a GMM for our data is to fit a given number of MVN distributions to the data using the Expectation-Maximization (EM) algorithm. The EM algorithm can be interpreted as a more general version of the k-means clustering algorithm, which can be applied to MVN distributions. In our application, however, the EM algorithm leads to several problems:

i. Each fitted GMM corresponds to a clustering, yet this clustering often fails to represent the observed trends. Instead, the clusters

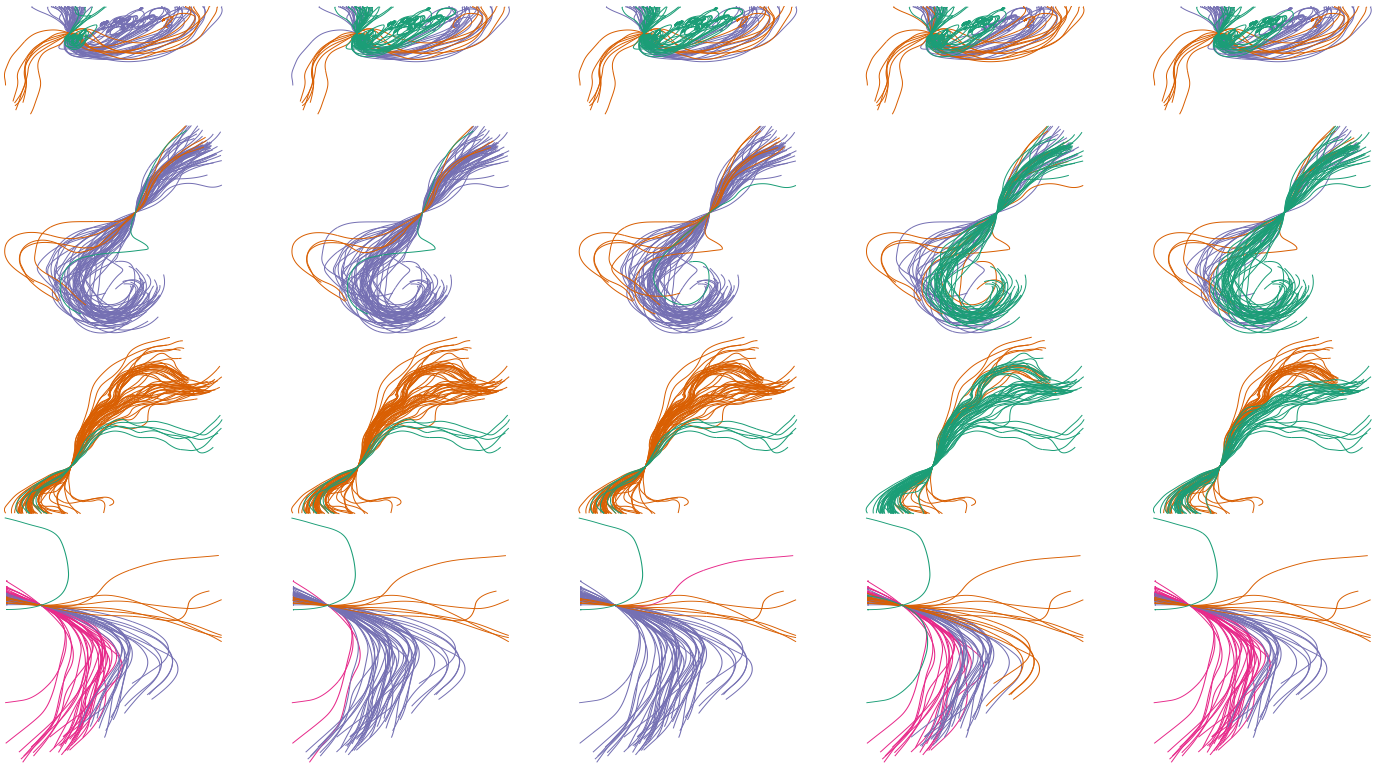


Fig. 5. Comparison of different clustering approaches using different sets of 2D streamlines. From left to right: ***PCA + AHC***, our hierarchical clustering based on the Euclidean distances in rank-reduced PCA-space. *MCPD + AHC*, hierarchical clustering using the mean-of-closest-point distance [10]. *Hausdorff + AHC*, hierarchical clustering using the Hausdorff distance. *PCA + GMM-EM*, clustering via the EM algorithm for Gaussian mixture models using the Euclidean distances in rank-reduced PCA-space. *PCA + k-means*, k-means clustering using the Euclidean distances in rank-reduced PCA-space. Average linkage was used in all AHC methods. For the examples in each row, the same number of clusters was prescribed.

often overlap in PCA-space and do not show the expected degree of separation. In addition, the multi-dimensional confidence ellipses corresponding to the clusters tend to span empty regions in the PCA-space. If we were to draw new points from the resulting PDF, some points would correspond to streamlines which have low similarity to existing ones.

ii. As mentioned earlier, when visualizing ensembles one often only has few streamlines. This number is very small so that many of the determined clusters do not fully span the reduced PCA-space. As a consequence, the EM algorithm becomes numerically unstable and requires a large regularization parameter.

iii. The need to specify the number of clusters k beforehand requires to run the EM algorithm repeatedly and then to choose the number of clusters based on a score like, e.g., the Akaike information criterion (AIC). Moreover, due to the random nature of the method, it may have to be run several times for each k. Both properties in combination can quickly let the clustering become a performance bottleneck.

In account of these reasons we decided to separate the clustering from the procedure that fits the MVN distributions: We first cluster the streamlines, and then fit an MVN distribution to each cluster. Note that, strictly speaking, we are not using a full GMM, because we do not compute weights for the individual components. Instead, we visualize the MVN distribution of each cluster individually and combine this with a separate visualization of the cluster sizes, i.e., via the thickness of the median lines and the bar plot.

For the clustering to work in combination with the MVN distributions, we have to ensure that it favors compact, elliptical clusters. Since the PCA-space is an Euclidean space, we can draw upon many existing clustering approaches. We have performed several experiments with different standard clustering algorithms, and based on the results we ultimately favor Agglomerative Hierarchical Clustering (AHC) in PCA-space.

AHC creates an unbalanced, binary clustering tree in a bottom-up manner. Starting with each point in a separate cluster (with cardinality one), pairs of clusters are merged successively until all points are contained in one large cluster. The pair of clusters that is combined in each step is determined by a similarity criterion, the so-called linkage criterion, as well as a distance metric, which defines the pairwise distances between raw data points. As distance metric we use the Euclidean distances in the reduced PCA-space. Common choices for the linkage criterion include single-linkage, complete-linkage, average linking [41] and Ward's Method [17]. We observed that single-linkage and complete-linkage, favoring connectedness and sphericalness, respectively, yield clusters which do not reveal the major trends in the streamline distribution effectively. Instead, we found average linking, which merges clusters based on their average point-to-point distances, to work best in our examples. Ward's method, which tries to minimize the total within-cluster variance, often delivers similar results. Specifically, AHC in combination with average-linkage yields clusters which very well satisfy our compactness requirement.

The clustering tree resulting from AHC can be split easily into the desired number of clusters (k), and thus allows for an intuitive adaption of the number of clusters. When the number of clusters is changed, the resulting new trend distribution changes in a very coherent and intuitive way: The clusters split and merge recursively according to the binary clustering tree, instead of re-forming completely every time. This effect is demonstrated in Fig. 6.

In general, we target a rather small number of less than five clusters, because we are looking for the major trends in the streamlines and noticed that the visualizations can become populated when more clusters

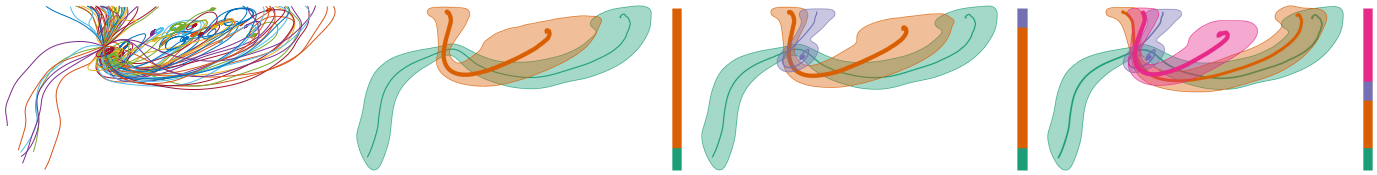


Fig. 6. Cluster refinement: For the set of streamlines in the first image, the number of clusters is incrementally increased from 2 to 4. The L-method initially guesses 3 clusters.

are used. We found that in most of our cases the L-Method [37] provides a very good initial guess for k, where we let the method choose $k \in \{2,...,10\}$. Unfortunately, due to the very low number of clusters, no automatic criterion can give perfect guesses in all possible cases. Therefore, in some cases we have to manually adjust k by ± 1 to get the most intuitive clustering.

In Fig. 5, we compare our clustering results with those obtained by alternative clustering approaches for streamlines or other types of line data. The comparison indicates that k-means clustering and the EM algorithm for GMMs (both performed in the Euclidean PCA-space) often prefer equally sized clusters over separating trends of diverging streamline sets. On the other hand, AHC based on mean-of-closest-point-distances [10] and Hausdorff distances often tends to misclassify individual streamlines. Our clustering seems to best extract the dominant *trends* in the data, and it is able to robustly handle complex situations. This can be seen, for instance, in the top row of Fig. 5, where our approach is the only one that can separate the small set of highly curved streamlines colored in green in the left image. It is also important to note that, irrespectively of the quality of the individual clustering approaches, most of them are not suited for the construction of variability plots as proposed in our work. As we will explain next, these plots are constructed by using the multivariate distribution of streamlines in some (rank-reduced) space, i.e., the PCA-space, excluding those clustering approaches not working in such a space.

4 Variability Plots

Up to this point, the set of streamlines has been partitioned in PCA-space, and the distribution of each cluster has been approximated with an MVN distribution. Then, our principle idea is to transform a geometric representation of these distributions in the PCA-space back to the domain space in which the original streamlines reside, in order to obtain for each cluster a confidence lobe illustrating the variance and spread of the respective trend. This transformation is possible through Eq. (1), which tells us that an arbitrary point in the (rank-reduced) PCA-space can be transformed to a corresponding streamline. If the point was taken in agreement with one of the MWN distributions, then the resulting streamline follows the statistics of the corresponding cluster, even though no such streamline existed in the initial set. By this, we can, in principle, generate arbitrary many new streamlines whose shapes follow the statistical properties encoded by the different clusters in PCA-space.

In the generation of these streamlines a certain approximation error is introduced, because we truncate the PCs used for reconstruction (c.f. Fig. 3), yet it is important to recall that this error is bounded through Eq. (2) and restricted to the high-frequency details that are captured by the higher PCs. This means that all major trends will be captured if we use a "sufficient" number of PCs. On the other hand, the restriction to the first r_2 PCs yields new streamlines exhibiting a certain amount of smoothness, providing a visually appealing cluster representation.

From a statistical point of view, the number of samples that are used to fit the MVN distributions—i.e., the cardinality of the clusters—is relatively small compared to the number of dimensions (r_2) of the rank-reduced PCA-space. Therefore, small variations in the samples can significantly change the geometric representations of the MVN distributions in PCA-space. On the other hand, all of our experiments have shown that these changes have only a minor effect on the geometry of the corresponding lobes.

4.1 Confidence Lobes

To represent the MVN distributions in PCA-space, we use confidence ellipses (or contours). Let μ_k and Σ_k denote the mean and covariance matrix of cluster k, respectively. Then the confidence ellipses are defined as (filled) iso-contours of the so-called Mahalanobis distance

$$d_k = \sqrt{(\mathbf{x}-\mu_k)\Sigma_k^{-1}(\mathbf{x}-\mu_k)},$$

where $\mathbf{x}$ denotes an arbitrary point in $\mathbb{R}^{r_2}$. Intuitively, the Mahalanobis distance d_k indicates for the point $\mathbf{x}$ how many standard deviations it is away from μ_k. A confidence level, i.e., a level-set in the distance field, is specified via an iso-value in numbers of standard deviation.

In principle, it would be very appealing to do so by specifying a quantile, for instance, so that $d_k \leq \alpha$ corresponds to the 50% innermost points. Unfortunately, this leads to a very counter-intuitive behavior of the generated lobes, because it makes the threshold α sensitive to the dimension r_2 of the reduced PCA-space. In particular, increasing r_2 to make the line approximation more accurate causes the corresponding confidence regions to grow, since the threshold α has to be increased [1]. Therefore, we threshold d_k against a fixed α, which can be chosen and varied when creating the confidence lobes.

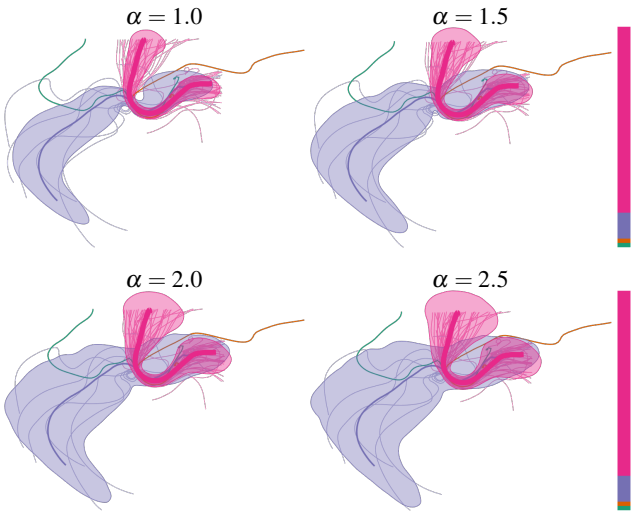


Fig. 7. Effect of the Mahalanobis threshold α on the confidence lobes: The larger α, the larger the lobes become. For $\alpha >= 2.0$ the lobes contain almost all associated lines but also tend to "overshoot". Orange and green trends contain single outliers, and hence no lobes are generated.

If we create a lobe with, e.g., $\alpha = 1.0$, it will cover the range of locations containing all streamlines that are within one standard deviation of each trend. While small thresholds like $\alpha = 1.0$ lead to very tight confidence lobes, higher thresholds with $\alpha \geq 2.0$ lead to convex-hull like shapes which sometimes "overshoot". This effect is demonstrated in Fig. 7, where for a set of streamlines the confidence lobes for different thresholds α are shown. We use a default value of $\alpha = 1.5$ in all of our experiments, if not stated otherwise.

[1] The Mahalanobis distance d_k is distributed with a χ^2_K-distribution, which changes depending on the degrees of freedom K. Here, K corresponds to r_2.

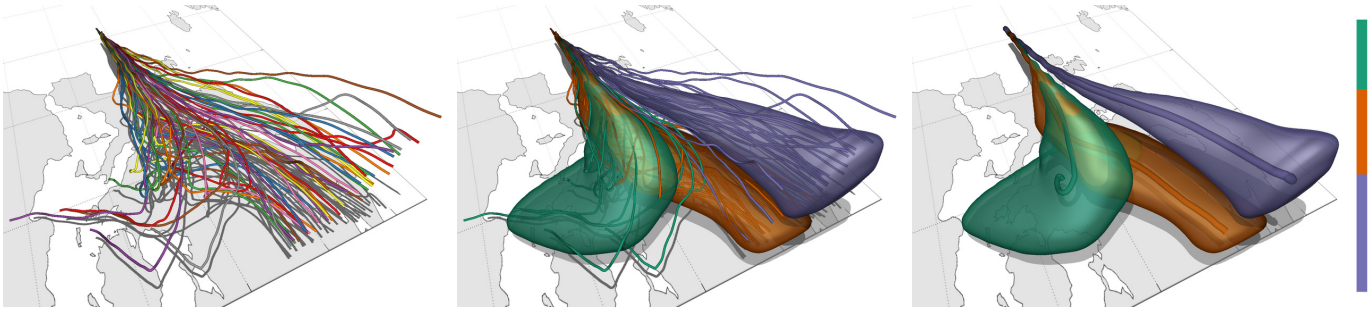


Fig. 8. Variability plot of pathlines in a time-varying 3D ensemble comprising 51 members. Left: Spaghetti plot. Middle: Pathlines colored by cluster membership, and confidence lobes. Right: Variability plot with confidence lobes and pathline-medians.

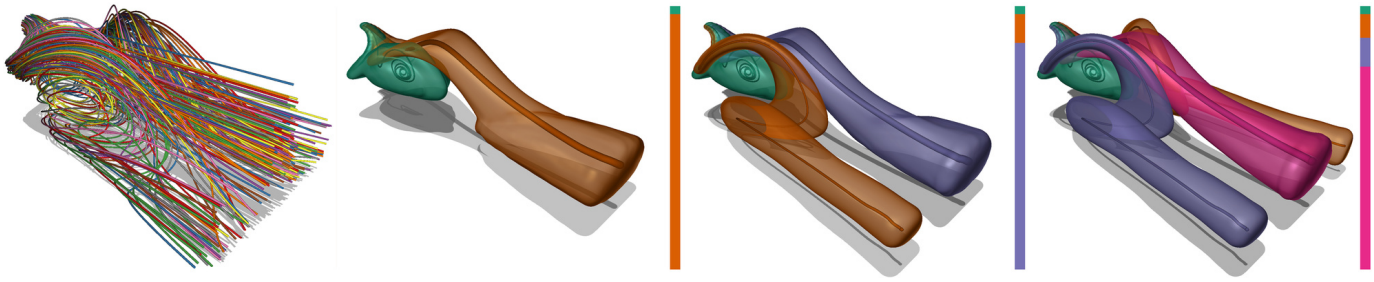


Fig. 9. Variability plots of 200 streamlines with jittered positions in an ensemble of 50 steady 3D flows around an obstacle. Left: A spaghetti plot of the streamlines. Middle-Right: Variability plots are shown, where the number of clusters increases incrementally from 2 to 4. The initial guess for the number of clusters is 3.

To transform a confidence ellipse in PCA-space to its corresponding domain space representation, we must determine the locations in domain space which are covered by at least one streamline that corresponds to a point in the interior of this ellipse in PCA-space. Since determining this is not directly possible for a given point in domain space, we follow a different approach using Monte-Carlo sampling and line drawing.

We draw random points uniformly from the confidence ellipse using a random number parameterized with the corresponding μ_k and Σ_k, and compute their streamline representations using Eq. (1). Each of these streamlines is splatted additively into the cells of a uniform grid discretizing the domain space. Note that the original streamlines are not used in this process. In this way we build a visitation map as proposed by Buerger et al. [7], and we then extract the confidence lobe using the smallest iso-value that still allows for smooth iso-contours.

Building the visitation map is performed via rasterization of the streamline vertices, which are treated as particles, or splat-kernels, of a certain diameter. This approach yields sufficient results in our application, and it is both faster and easier to implement than accurate line-splats using point-to-line distances. We use a small bi-/tri-linear splat-kernel with a support of 4 and 8 texels in 2D and 3D, respectively, and use a second pass after splatting to smooth the visitation map in order to increase the quality of the resulting iso-contours. We use visitation maps—realized as 2D and 3D accumulation textures—with a resolution of roughly 200 texels along the longest dimension. In 2D, we use a fixed number of 1000 streamlines to generate the visitation map for each cluster, and we draw for each lobe the outer contour and uniquely color its interior. In 3D, we found 5000 streamlines to be sufficient to obtain representative lobes, and we generate the visitation map directly on the GPU and render the lobes via iso-surface ray-casting. If the vertex density along the streamlines is too low, we add new vertices by linearly interpolating between consecutive vertices.

4.2 Streamline-Median

The abstract shapes of the confidence lobes are further enhanced by a single streamline representing the corresponding trend as accurate as possible. We therefore introduce a new concept of streamline-median. Since the reduced PCA-space is an Euclidean space, we build upon existing concepts here. Specifically, we use the so-called geometric median, which is an extension of the one-dimensional median to multiple dimensions. Given a set of points, the geometric median is defined as the point in space—not necessarily coinciding with one of the initial points—which minimizes the sum of Euclidean distances to all initial points. It can be calculated iteratively using Weiszfeld's algorithm.

We hence determine the geometric median for every cluster in the PCA-space and reconstruct a streamline—the streamline-median—from it. This means that the streamline-medians in our visualizations are not streamlines from the initial set, but they are artificial streamlines being closer to all initial streamlines than any other streamline. On the other hand, following the same argumentation as for constructing the confidence lobes, we know that this artificial streamline shows the general trend represented by a cluster and is free of high-frequent details which are not common to all cluster members. When drawing the streamline-median, we further use its thickness to give a qualitative visual cue indicating the relative strength of the trend. I.e., the more initial streamlines follow the trend, the thicker the streamline-median is drawn.

We further annotate each streamline variability plot to the right. The color bar shows the relative number of trajectories represented by each lobe, further enhancing the plot about qualitative information.

5 Results

In this section, we analyze the performance of our method, show additional results of our method, and perform a comparison of the proposed variability plots to curve boxplots.

5.1 Datasets

The 2D streamline examples we use in this paper were created from wind fields of numerical weather prediction data obtained from the ECMWF Ensemble Prediction System (ENS), which comprises 51 ensemble members. We use forecast runs initialized at 00:00 UTC on October 15th and 17th, 2012, and perform massless particle integration to obtain streamlines in a single steady forecast at a later time.

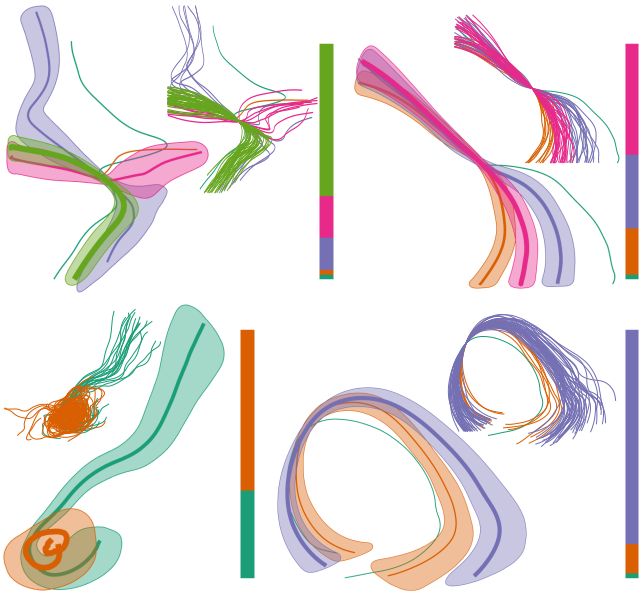


Fig. 10. 2D streamline variability plots and corresponding spaghetti plots.

Each 2D streamline is comprised of 300 vertices. Additional 2D examples are shown in Fig. 10. In the 3D examples (see Fig. 8), pathlines were computed for the first 144 hours of the forecast and for each ensemble member using the LAGRANTO model [42], which considers air masses and specific meteorological aspects rather than massless particles. We perform 200 integration steps to generate these pathlines in 3D. In addition, we further use an ensemble of 50 steady 3D flows from a simulation of a fluid flow around a cylindrical obstacle with varying Reynolds numbers (see Fig. 9). Streamlines are computed using 500 numerical integration steps.

5.2 Implementation & Performance

All the results in this paper were generated on a standard desktop PC equipped with an Intel Xeon X5675 processor with 3.0GHz×6 cores, 8GB RAM and a NVIDIA Geforce GTX 680. We use the Matlab implementations of PCA and AHC, and our own CPU implementation to fit an MVN to the data in PCA-space and generate new random variables respecting these distributions, including the streamline-median. In 2D, splatting of streamlines into a 2D grid, extracting iso-contours in this grid, and drawing the resulting filled confidence lobes are performed on the CPU. In 3D, splatting is performed on the GPU, where the confidence lobes are directly rendered via iso-surface ray-casting.

In all of our test scenarios, the time required to compute a PCA, cluster the data, and fit an MVN to this data was below 50ms, even for a bundle consisting of 200 streamlines with 500 vertices each. The most time consuming step is splatting the generated lines into the visitation map. On the CPU, roughly 10000 trajectories can be splatted into the 2D map per second, and splatting into a 3D map on the GPU can be performed at a rate of roughly 50000 trajectories per second. This performance gain is mainly due to the fast memory interface on the GPU and the possibility to process many trajectories in parallel. Overall, all variability plots shown in this paper were generated in less than 500ms, given the initial set of trajectories, and were rendered at interactive rates.

5.3 3D Variability Plots

In the following we further demonstrate the effectiveness of variability plots to depict the major trends in 3D trajectories. It should be emphasized that the generation and visualization of 3D variability plots does not require any specific algorithmic modifications of our approach, besides the use of a 3D visitation map into which the trajectories are splatted and from which the lobes are rendered.

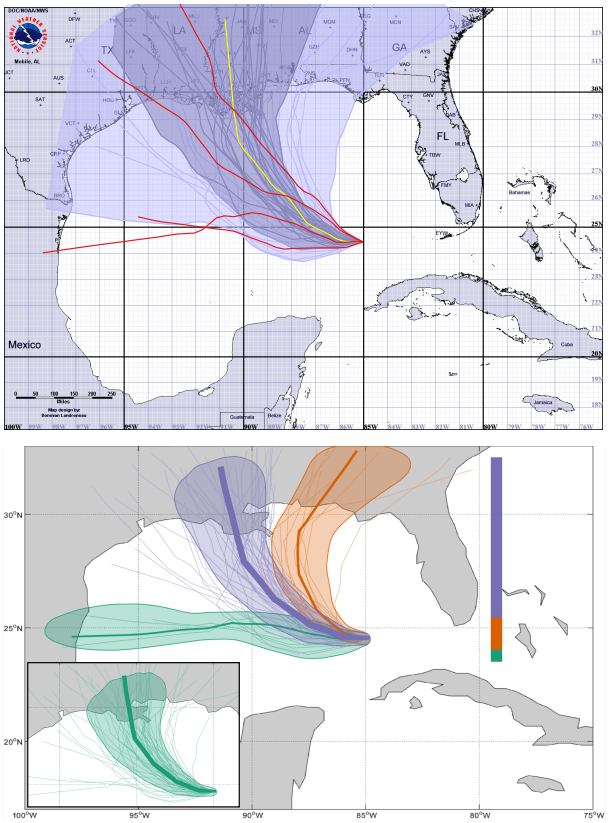


Fig. 11. Comparison of a streamline variability plot (bottom) to the curve boxplot from [23] (top) for an ensemble of 50 simulated hurricane tracks, generated by the method presented by Cox et al. [11]. The inset shows a second variability plot, where the number of clusters was manually set to one. In the boxplot, the light and dark cones are the bands which contain 50% and 100% of the curves, respectively. Red lines show outliers and the yellow line is the global median line. In both figures the original tracks are shown in the background of the plots, where the colors correspond to band-depth and clusters, respectively.

Fig. 8 shows a 3D variability plot for an ensemble of 51 pathlines in the ECMWF ensemble. It can be seen that the major trends in the data are very well separated by the variability plot, and that the abstract representation using lobes and streamline-medians provides a fairly uncluttered view compared to the spaghetti plot. The particular example demonstrates, that the variability plot can not only convey the major trends but also give a very good indication of where these trends start to separate. Especially this property is extremely helpful in meteorological applications, where the locations need to be analyzed where the divergence of predicted forecasts is going to start.

Fig. 9 shows variability plots of 200 streamlines, which were generated by seeding in each of the 50 ensemble members 4 streamlines that were randomly jittered around a common seed point. The number of clusters is adjusted incrementally from 2 to 4, while the initial guess by the L-Method is 3. Here, it can be seen how an increasing amount of clusters can lead to an increasing amount of detail in the variability plot, until a good representation of the trends is reached. The most representative plot is the last one, which reveals four significantly different trends in the streamlines and effectively conveys the symmetry in the data well.

5.4 Comparison to Curve Boxplots

Fig. 11 compares a streamline variability plot to a curve boxplot from [23] for a set of simulated hurricane tracks. The boxplot illustrates the distribution of trajectories via two nested bands containing 50% and 100% of the trajectories, respectively. In addition, a representative median trajectory, i.e., the deepest trajectory from the initial

set, and outliers are depicted. The boxplot has been generated using the entire set of trajectories, and no initial clustering was performed.

The streamline variability plot reveals three trends in the hurricane trajectories. A small number of trajectories deviates to the west and north-east—which is captured by two minor trends—while the dominant third trend in the center contains over 75% of the trajectories. The confidence lobes of the two smaller trends—in relation to the regions that are covered by the corresponding trajectories—are more widespread than the trend in the center. This means that there is a higher intra-cluster variance in the smaller trends, whereas the trajectory distribution of the dominant trend is more focused towards the region that is indicated by its confidence lobe and median line. A second variability plot is also shown, for which the number of clusters was manually set to one. This results in a single, wider lobe which is similar to the 50% band of the boxplot. The difference is, that the boxplot cone shows the region that contains 50% of the innermost curves, while our lobe shows the region that contains all curves that are within the range of $\alpha = 1.5$ standard-deviations. Furthermore, the length of the curves is conveyed differently in the two visualizations. On the one hand, the "length" of the boxplot cone corresponds to a maximum curve length, because it is constructed as a hull around all represented curves, and the yellow global median line shows the length of a single representative line. On the other hand, in the variability plot, the "length" of a lobe is typically similar to the length of its streamline-median, which is approximately equal to the median length of all represented curves.

The comparison of boxplots and variability plots clearly indicates the different use cases of both approaches. The boxplot intends to visualize the entire spread, or enclosed spatial band, of a set of trajectories, in addition indicating the percentage of trajectories being contained in sub-bands as well as outliers. The variability plot intends to detect and reveal the major trends in the data, and then plots these trends via confidence lobes to depict the probability of occurrence of the trajectories. Thus, the variability plots support a probabilistic analysis of the shapes of the major trends, rather than emphasizing the overall spread of the data. The clustering of the initial trajectories into major trends, on the other hand, can be used as a preprocess to curve boxplots, so that each trend could be represented by a separate boxplot. Thus, we could even combine both approaches, by replacing individual confidence lobes in our plots with curve boxplots.

6 Conclusion

In this work we have presented a new approach for the visual exploration of the major trends in sets of streamlines extracted from ensemble flow fields. Our approach is specifically tailored to the visualization of trends in rather small sets of streamlines, as it is typically the case when dealing with routinely simulated meteorological ensembles. Even from such sets we can faithfully reconstruct confidence lobes showing the probability of occurrence of streamlines over the domain. By using stochastic models of clusters in PCA-space, we can generate new streamlines exhibiting the statistical properties of the shapes and positions of the major trends. The method is applicable to 2D and 3D data, and the abstract visualizations we present allow to communicate effectively salient characteristics of the data distributions even in 3D.

In the future, we intend to improve our approach in the following ways: Firstly, we aim at developing improved approaches for detecting outliers in the data. So far, outliers are detected if they show up in a separate cluster, yet no specific mechanism is used to explicitly separate them. Secondly, we will investigate the use of our approach for showing trends in streamlines which have been released at different locations. We have shown the use of streamlines which have been jittered slightly around a seed location, yet for lines seeded at different locations some prior operations will be necessary to register the streamlines to each other. Finally, we will investigate our approach for clustering vector fields hierarchically, so that local trends can be separated and hierarchically represented.

Acknowledgments

We thank the authors of [23] and [11], respectively, for providing the hurricane trajectories data and for granting permission to use the curve boxplot figure. Access to ECMWF prediction data has been kindly provided in the context of the ECMWF special project "Support Tool for HALO Missions". We are grateful to the special project members Marc Rautenhaus and Andreas Dörnbrack for providing the ECMWF ENS dataset used in this study. This work was supported by the European Union under the ERC Advanced Grant 291372 SaferVis - Uncertainty Visualization for Reliable Data Discovery.

References

[1] F. Bashir, A. Khokhar, and D. Schonfeld. Segmented trajectory based indexing and retrieval of video data. In *Proc. of the Int. Conference on Image Processing*, pages 623–629, 2003. 2

[2] H. Bhatia, S. Jadhav, P.-T. Bremer, G. Chen, J. A. Levine, L. G. Nonato, and V. Pascucci. Edge maps: Representing flow with bounded error. In *IEEE Pacific Visualization Symposium*, pages 75–82, 2011. 2

[3] G.-P. Bonneau, H.-C. Hege, C. Johnson, M. Oliveira, K. Potter, P. Rheingans, and T. Schultz. Overview and state-of-the-art of uncertainty visualization. In *Scientific Visualization*, pages 3–27. Springer, 2014. 2

[4] S. Born, M. Pfeifle, M. Markl, and G. Scheuermann. Visual 4D MRI blood flow analysis with line predicates. In *IEEE Pacific Visualization Symposium*, pages 105–112, 2012. 2

[5] R. Brown. Animated visual vibrations as an uncertainty visualisation technique. In *GRAPHITE*, pages 84–89, 2004. 2

[6] 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 2003*, pages 518–529. 2

[7] K. Bürger, R. Fraedrich, D. Merhof, and R. Westermann. Instant visitation maps for interactive visualization of uncertain particle trajectories. *Visualization and Data Analysis 2012*, pages 128–136, 2012. 4.1

[8] C.-K. Chen, S. Yan, H. Yu, N. Max, and K.-L. Ma. An illustrative visualization framework for 3D vector fields. *Computer Graphics Forum*, 30(7):1941–1951, 2011. 2

[9] Y. Chen, J. Cohen, and J. Krolik. Similarity-guided streamline placement with error evaluation. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1448–1455, 2007. 2

[10] I. Corouge, S. Gouttard, and G. Gerig. Towards a shape model of white matter fiber bundles using diffusion tensor MRI. In *IEEE International Symposium on Biomedical Imaging: Nano to Macro, 2004*, volume 1, pages 344–347, April 2004. 5, 3.2

[11] J. Cox, D. House, and M. Lindell. Visualizing uncertainty in predicted hurricane tracks. *International Journal for Uncertainty Quantification*, 3(2):143–156, 2013. 11, 6

[12] N. Ferreira, J. T. Klosowski, C. E. Scheidegger, and C. T. Silva. Vector field k-means: Clustering trajectories by fitting multiple vector fields. *Computer Graphics Forum*, 32(3pt2):201–210, 2013. 2

[13] H. Garcke, T. Preußer, M. Rumpf, A. Telea, U. Weikard, and J. van Wijk. A continuous clustering method for vector fields. In *Proc. of the Conf. on Visualization 2000*, pages 351–358. 2

[14] G. Grigoryan and P. Rheingans. Point-based probabilistic surfaces to show surface uncertainty. *IEEE Transactions on Visualization and Computer Graphics*, 10(5):564–573, 2004. 2

[15] B. Heckel, G. Weber, B. Hamann, and K. I. Joy. Construction of vector field hierarchies. In *Proceedings of the Conference on Visualization 1999*, pages 19–25, 1999. 2

[16] M. Hummel, H. Obermaier, C. Garth, and K. Joy. Comparative visual analysis of Lagrangian transport in CFD ensembles. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2743–2752, 2013. 2

[17] J. H. Ward Jr. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236–244, 1963. 3.2

[18] A. K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recogn. Lett.*, 31(8):651–666, 2010. 2

[19] J. L. Lumley. The structure of inhomogeneous turbulent flows. In *Atmospheric turbulence and radio wave propagation*, pages 166–178. Nauka, Moscow, 1967. 3.1

[20] C. Lundstrom, P. Ljung, A. Persson, and A. Ynnerman. Uncertainty visualization in medical volume rendering using probabilistic animation. *IEEE Transactions on Visualization and Computer Graphics*, 13:1648–1655, 2007. 2

[21] T. McLoughlin, M. W. Jones, R. S. Laramee, R. Malki, I. Masters, and C. D. Hansen. Similarity measures for enhancing interactive streamline seeding. *IEEE Transactions on Visualization and Computer Graphics*, 19(8):1342–1353, 2013. 2

[22] K. E. Meyer, J. M. Pedersen, and O. Özcan. A turbulent jet in crossflow analysed with proper orthogonal decomposition. *Journal of Fluid Mechanics*, 583:199–227, 2007. 3.1

[23] M. Mirzargar, R. Whitaker, and R. M. Kirby. Curve boxplot: Generalization of boxplot for ensembles of curves. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2654–2663, 2014. 1, 2, 11, 5.4, 6

[24] B. Moberts, A. Vilanova, and J. van Wijk. Evaluation of fiber clustering methods for diffusion tensor imaging. In *IEEE Visualization 2005*, pages 65–72, 2005. 2

[25] T. W. Muld, G. Efraimsson, and D. S. Henningson. Flow structures around a high-speed train extracted using proper orthogonal decomposition and dynamic mode decomposition. *Computers & Fluids*, 57(0):87 – 97, 2012. 3.1

[26] H. Obermaier and K. Joy. Future challenges for ensemble visualization. *IEEE Computer Graphics and Applications*, 34:8–11, 2014. 2

[27] S. Oeltze, D. J. Lehmann, A. Kuhn, G. Janiga, H. Theisel, and B. Preim. Blood Flow Clustering and Applications in Virtual Stenting of Intracranial Aneurysms. *IEEE Transactions on Visualization and Computer Graphics*, 20(5):686–701, 2014. 2

[28] M. Otto, T. Germer, H.-C. Hege, and H. Theisel. Uncertain 2D vector field topology. *Computer Graphics Forum*, 29(2):347–356, 2010. 2

[29] A. T. Pang, C. M. Wittenbrink, and S. K. Lodha. Approaches to uncertainty visualization. *The Visual Computer*, 13(8):370–390, 1997. 2

[30] C. Petz, K. Pöthkow, and H.-C. Hege. Probabilistic Local Features in Uncertain Vector Fields with Spatial Correlation. *Computer Graphics Forum*, 31:1045–1054, 2012. 2

[31] T. Pfaffelmoser, M. Reitinger, and R. Westermann. Visualizing the Positional and Geometrical Variability of Isosurfaces in Uncertain Scalar Fields. *Computer Graphics Forum*, 30(3):951–960, 2011. 2

[32] K. Pöthkow and H.-C. Hege. Positional uncertainty of isocontours: Condition analysis and probabilistic measures. *IEEE Transactions on Visualization and Computer Graphics*, 17(10):1393–1406, 2011. 2

[33] K. Pöthkow and H.-C. Hege. Nonparametric Models for Uncertainty Visualization. *Computer Graphics Forum*, 32:131–140, 2013. 2

[34] K. Pöthkow, B. Weber, and H.-C. Hege. Probabilistic marching cubes. *Computer Graphics Forum*, 30:931–940, 2011. 2

[35] K. Potter, A. Wilson, P.-T. Bremer, D. Williams, C. Doutriaux, V. Pascucci, and C. R. Johnson. Ensemble-Vis: A framework for the statistical visualization of ensemble data. In *Proc. IEEE Int. Conference on Data Mining Workshops*, pages 233–240, 2009. 2

[36] C. Rössl and H. Theisel. Streamline embedding for 3D vector field exploration. *IEEE Transactions on Visualization and Computer Graphics*, 18(3):407–420, 2012. 2

[37] S. Salvador and P. Chan. Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. In *Proc. of the 16th IEEE International Conference on Tools with Artificial Intelligence*, pages 576–584, 2004. 3.2

[38] T. Salzbrunn, H. Jänicke, T. Wischgoll, and G. Scheuermann. The state of the art in flow visualization: Partition-based techniques. In *SimVis*, pages 77–92, 2008. 2

[39] T. Salzbrunn and G. Scheuermann. Streamline predicates. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1601–1612, 2006. 2

[40] J. Sanyal, S. Zhang, J. Dyer, A. Mercer, P. Amburn, and R. J. Moorhead. Noodles: A Tool for Visualization of Numerical Weather Model Ensemble Uncertainty. *IEEE Transactions on Visualization and Computer Graphics*, 16:1421–1430, 2010. 2

[41] R. Sokal. A statistical method for evaluating systematic relationships. *Univ. Kansas Sci. Bull.*, 38:1409–1438, 1958. 3.2

[42] M. Sprenger and H. Wernli. The Lagrangian analysis tool LAGRANTO - version 2.0. *Geosci. Model Dev. Discussions*, 8(2):1893–1943, Feb. 2015. 5.1

[43] A. Telea and J. van Wijk. Simplified representation of vector fields. In *Proc. Conf. on Visualization*, pages 35–42, 1999. 2

[44] D. W. J. Thompson and J. M. Wallace. The Arctic oscillation signature in the wintertime geopotential height and temperature fields. *Geophys. Research Letters*, 25(9):1297–1300, 1998. 3.1

[45] R. T. Whitaker, M. Mirzargar, and R. M. Kirby. Contour Boxplots: A Method for Characterizing Uncertainty in Feature Sets from Simulation Ensembles. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2713–2722, 2013. 1, 2

[46] D. S. Wilks. *Statistical Methods in the Atmospheric Sciences*. Academic Press, 2011. 2, 3.1

[47] H. Yu, C. Wang, C.-K. Shene, and J. H. Chen. Hierarchical streamline bundles. *IEEE Transactions on Visualization and Computer Graphics*, 18(8):1353–1367, 2012. 2

[48] B. Zehner, N. Watanabe, and O. Kolditz. Visualization of gridded scalar data with uncertainty in geosciences. *Computers & Geosciences*, 36(10):1268–1275, 2010. 2

[49] 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*, 14(5):1044–1053, 2008. 2

[50] Z. Zhang, K. Huang, and T. Tan. Comparison of similarity measures for trajectory clustering in outdoor surveillance scenes. In *Proceedings of the 18th International Conference on Pattern Recognition*, pages 1135–1138, 2006. 2