

Physics based Pathline Clustering and Exploration – Supplemental Document

Duong B. Nguyen^{†1}, Lei Zhang^{‡1}, Robert S. Laramée^{§2}, David Thompson^{¶3}, Rodolfo Ostilla Monico^{||1}, and Guoning Chen^{**1}

¹University of Houston, USA

²Swansea University, Wales, UK

³Mississippi State University, USA

Abstract

In this supplemental document, we provide the extension of our framework to higher-ordered attributes measured along the individual pathlines. Specifically, we focus on the vector-valued attributes, such as velocity vectors and the gradient vectors of certain scalar attributes. We also provide some details on the AHC computation, clusters generation, and our modified edge-bundling method. For the details on the functionality of our visual analytic system and its user interaction, please refer to the accompanying video. Finally, this supplemental document details the discussion of the comparison between the AHC framework and the model-based clustering using the proposed distance metric.

1. Extension to Vector-valued Attributes

Inspired by the vector-based correlation computation [BMLC19], we extend our TSM metric to compute the similarity between two TACs with **vector-based** attributes (e.g., velocity vector and acceleration). Specifically, we use the similarity metric between two vectors from the work [BMLC19] to compute the correlation between two vector-valued attribute. To compute the distance between two vectors, we use the following approach, i.e., we replace $D_e(\bullet, \bullet)$ in Eq.(2) of the paper with the following $d(\mathbf{g}_i, \mathbf{g}_j)$.

$$d(\mathbf{g}_i, \mathbf{g}_j) = (2 - d_d(\mathbf{g}_i, \mathbf{g}_j)) d_m(\mathbf{g}_i, \mathbf{g}_j) - 1 \quad (1)$$

$$d_d(\mathbf{g}_i, \mathbf{g}_j) = \frac{\langle \mathbf{g}_i, \mathbf{g}_j \rangle}{\|\mathbf{g}_i\| \|\mathbf{g}_j\|}, \quad d_m(\mathbf{g}_i, \mathbf{g}_j) = 1 + \left| \frac{\|\mathbf{g}_i\| - \|\mathbf{g}_j\|}{\|\mathbf{g}\|_{max}} \right|$$

where $\|\mathbf{g}\|_{max}$ is the maximum magnitude of the vector-valued attribute of interest in the entire data set.

Figure 1 provides the clustering result of pathlines of the 3D vortex tube and 2D vortex ring simulations using the TACs of velocity vectors. The shapes of pathlines can be distinguished and grouped into different clusters. In particular, for the clustering result in (a), the pathlines that exhibit upward bending (yellow) dur-

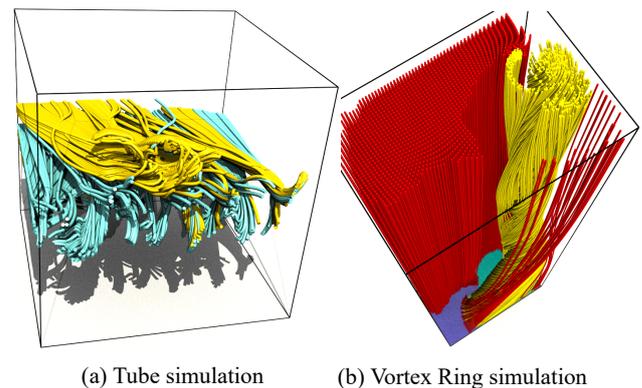


Figure 1: Pathline clustering results with velocity vector TACs for 3D vortex tube (a) and 2D vortex ring (b) simulations, respectively.

ing the vortex ring breakdown are roughly separated from those bending downward. For the result in (b), the pathlines in the two clusters are different in term of the representation. The red group corresponds to the ambient area where the vortex does not exist, while the yellow particles belong to the vortex region. Although visualizing the 3D vector-based attribute TACs is not trivial and we plan to address in the future, the results lead us to another utility of TACs which can be used to cluster pathlines.

[†] duongnguyenbinh@gmail.com

[‡] zhanglei.later@gmail.com

[§] R.S.Laramée@swansea.ac.uk

[¶] dst@cavs.msstate.edu

^{||} rostilla@central.uh.edu

^{**} chengu@cs.uh.edu

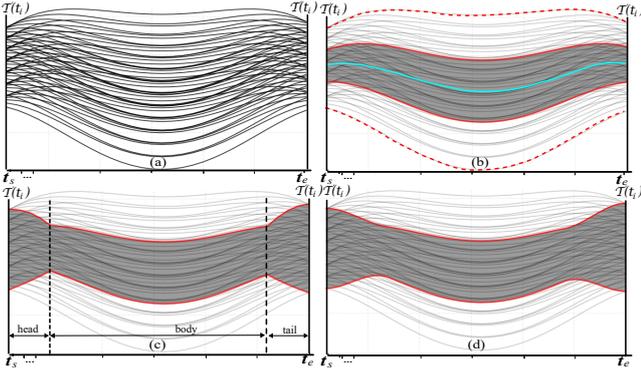


Figure 2: Illustration of edge-bundling method for TACs cluster visualization. (a) individual TACs, (b) edge-bundling rendering of a group of TACs by thinning toward the representative TAC, (c) preservation of the range of a group of TACs by adding head and tail segments, (d) the final result.

2. AHC Implementation and Clustering Result Generation

The traditional AHC runs in $O(N^3)$ time where N is the number of TACs. This running time makes the AHC impractical for large datasets. We implement the parallel, locally-ordered AHC proposed by Walter et al. [BWP08], which runs in sub-quadratic time. This greedy algorithm is based on the following observation: if two clusters, A and B, are nearest neighbors, they will eventually be merged together. Thus, it is better to find all possible A and B pairs, and cluster them immediately. To quickly identify the closest clusters for merging, we store clusters in a K-d tree which requires $O(\log n)$ running time.

3. Modified Edge Bundling

Traditionally, visualizing clusters is achieved by assigning each cluster a specific color. However, showing all TACs with colors assigned based on their cluster IDs will result in clutter, making it difficult to recognize the behavior encoded in TACs as demonstrated in Figure 2(a). To address this issue, we adapt the edge bundling technique for parallel coordinate plot (PCP) visualization developed by Palmas, et al [PBO*14]. Given a cluster of TACs, the average (cyan curve) and boundary TACs (dotted red curves) are first derived (Figure 2(b)). Then we offset the two boundaries towards the centroid. The offset operation does not change the overall behavior of the TACs in the cluster, while the range of the cluster, i.e., the coverage of the attribute values at the two ends of TACs t_s and t_e is changed. To preserve the coverage information of the cluster, we create a head and a tail for the edge-bundling by keeping the maximum and minimum of the attribute values at the two ends (Figure 2(c)). After we apply the edge-bundling method, the clusters in Figure 2(a) are shown as Figure 2(d), which greatly reduces the clutter.

4. AHC vs Model-based Clustering

For time-series data (e.g., TACs) clustering, similarity-based methods and model-based methods are usually applied. AHC is a rep-

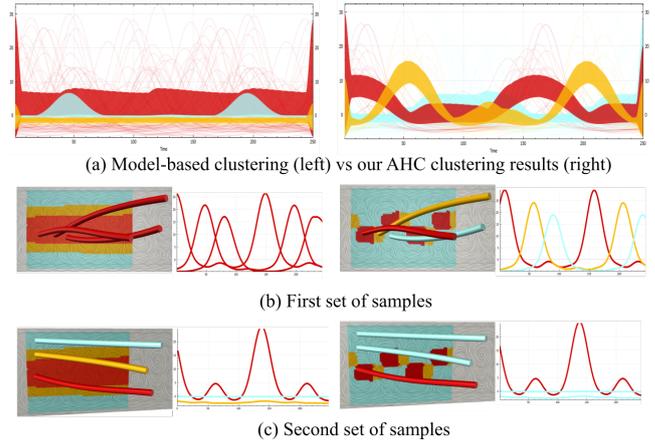


Figure 3: Comparison between AHC and model-based clustering. Model-based method classifies TACs only based on the magnitude while AHC considers both magnitude and shapes of TACs.

resentative of the similarity-based method in which a customized distance metric needs to be designed to measure the similarity between two TACs. In contrast to similarity-based methods, model-based methods assume that the data are generated by a mixture of underlying probability distributions. The probabilistic representation enables the derivation of consistent expectation-maximization (EM) learning algorithms for the clustering problem, which eliminates the need for a similarity metric. Here, we compare the clustering results obtained using the proposed AHC and new metric with those produced by the model-based methods for our TAC data.

To facilitate the discussion, we include comparison figure used in the paper here (see Figure 3). From the comparison, we see that the model-based method tends to classify TACs only based on the magnitude while AHC consider both magnitude and shapes of TACs. As in our settings both the trend (or shape) and the magnitude of the TACs are important in characterizing their similarity, we opt for AHC over the model-based method.

Other reason for choosing the AHC in our case is due to the performance of the clustering during the user interaction. Although the classic AHC has large computational complexity as the similarity has to be measured between every pair, the performance of model-based clustering method will decrease (i.e it will run slower) if we increase the number of clusters. This is because AHC produces a hierarchy of the clusters, which enables the interactive selection of different numbers of clusters to show. In contrast, model-based method has to recompute the clusters if the number of the clusters is changed during user interaction. In addition, the model-based method requires computation of a probability to assign each TAC to a cluster for each iteration during the expectation-maximization (EM) computation. Therefore, the numbers of computations will increase with the increase of the number of clusters.

One thing we wish to point out is that, in contrast to the model-based method, AHC is sensitive to TACs of varying length, which can have consequences in our clustering. To address this, we only perform clustering on a group of TACs that have similar lengths.

Our assumption is that if two TACs have rather different lengths, they do not have similar overall behaviors considering the importance of their life span in unsteady flow. Nonetheless, we can still focus on the portions where the two TACs have similar lengths using the proposed temporal clustering.

5. Results with 2D flow behind a cylinder

Figure 4 provides the comparison between the clustering results of TACs computed from the two attributes λ_2 and Q . Although these two attributes have different temporal trends, they result in similar clustering results for the vortex structure analysis task. In practice, the user should explore different attributes to identify the one that can reveal most effectively the flow behaviors of interest (e.g., vortices in this example).

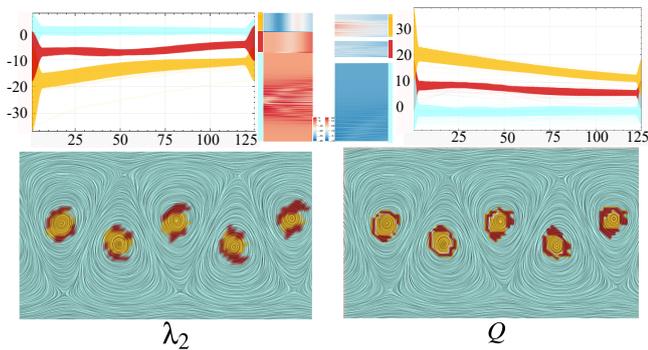


Figure 4: The comparison between the clustering results of TACs computed from the two attributes λ_2 (left) and Q (right). Although the two attributes have opposite temporal trends, they result in similar clustering results, i.e., both reveal the two-layered configuration of the vortices.

6. Results with 3D Tube Simulation

Figures 5 provide detailed results and visualizations for the global clustering of pathlines based on vorticity for the 3D Tube simulation.

References

- [BMLC19] BERENJKOUB M., MONICO R. O., LARAMEE R. S., CHEN G.: Visual analysis of spatio-temporal relations of pairwise attributes in unsteady flow. *IEEE transactions on visualization and computer graphics* 25, 1 (2019), 1246–1256. [1](#)
- [BWP08] B. WALTER K. BALA M. K., PINGALI K.: Fast Agglomerative Clustering for Rendering. *IEEE Symposium on Interactive Ray Tracing* (2008). [2](#)
- [PBO*14] PALMAS G., BACHYNSKYI M., OULASVIRTA A., SEIDEL H. P., WEINKAUF T.: An edge-bundling layout for interactive parallel coordinates. In *Visualization Symposium (PacificVis), 2014 IEEE Pacific* (2014), IEEE, pp. 57–64. [2](#)

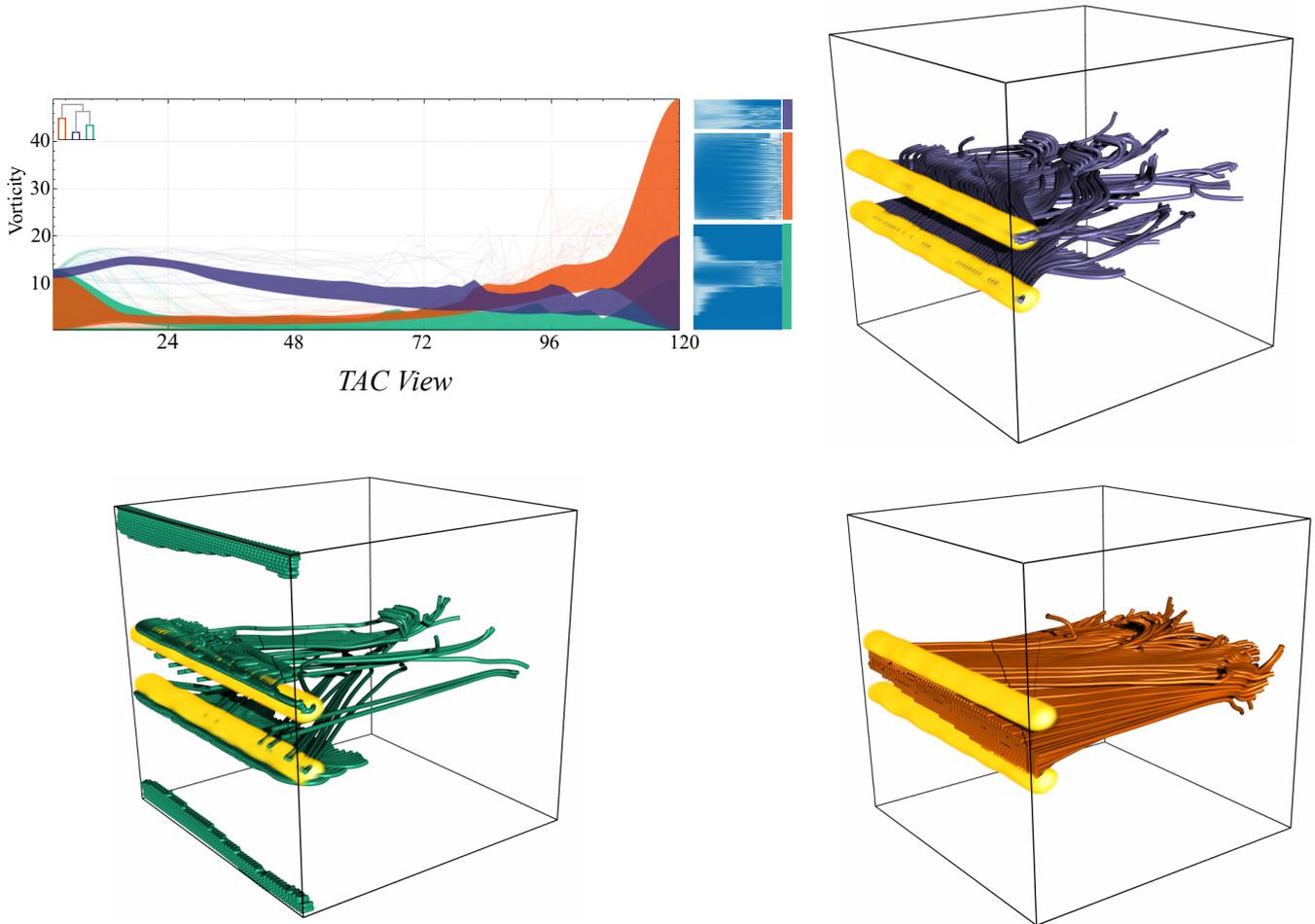


Figure 5: Global clustering result of the 3D vortex tube simulation with TACs of the vorticity attribute that reveals a three-layered structure ($p_c=1$). Grid size: $360 \times 360 \times 360$. Bounding box: $[0,2] \times [0,2] \times [0,2]$. Pathlines are seeded on a XZ plane with $Y = 0.1$. The global clustering results reveal a three-layered structure.