

# Multi-field Visualisation via Trait-induced Merge Trees

Jochen Jankowai\*  
Linköping University

Talha Bin Masood†  
Linköping University

Ingrid Hotz‡  
Linköping University

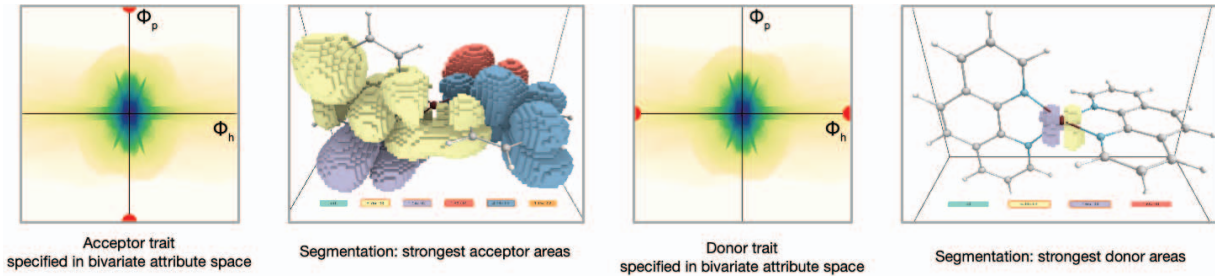


Figure 1: The charge transition of molecules can be described as a bivariate density field ( $\Phi_h, \Phi_p$ ). The image shows two different traits highlighting acceptor respective donor behavior of the molecule in a spatial segmentation. The traits are specified by a pair of points with high absolute values in  $\Phi_p$ , respective  $\Phi_h$ .

## ABSTRACT

In this work, we propose trait-based merge trees a generalization of merge trees to feature level sets, targeting the analysis of tensor field or general multi-variate data. For this, we employ the notion of traits defined in attribute space as introduced in the feature level sets framework. The resulting distance field in attribute space induces a scalar field in the spatial domain that serves as input for topological data analysis. The leaves in the merge tree represent those areas in the input data that are closest to the defined trait and thus most closely resemble the defined feature. Hence, the merge tree yields a hierarchy of features that allows for querying the most relevant and persistent features. The presented method includes different query methods for the tree which enable the highlighting of different aspects. We demonstrate the cross-application capabilities of this approach with three case studies from different domains.

**Index Terms:** Human-centered computing—Visualization—Visualization design and evaluation methods; Human-centered computing—Visualization—Visualization application domains—Scientific visualization

## 1 INTRODUCTION

Typically, natural phenomena are complex and their simulations need to fit this complexity in the model and output. Oftentimes, this results in a multitude of output variables. Examples include meteorology data containing fields for pressure, temperature, and precipitation, data from computational chemistry simulations often yield a set of density fields, or simulations within mechanical engineering usually include tensor data whose scalar derivatives may be interpreted as multi-variate data.

The visualization of multi-field data has, however, long been an enigma. Methods for scalar data may be applied to each field individually but usually do not extend to a joint analysis of all fields. A typical solution is to simultaneously visualize several fields in a display matrix, which means a heavy cognitive load for the users as they need to combine the separate images mentally into one.

\*e-mail: jochen.jankowai@liu.se

†e-mail: talha.bin.masood@liu.se

‡e-mail: ingrid.hotz@liu.se

Including interactive methods in a space of reduced dimensionality can alleviate this mental strain only partially.

The situation is similar when it comes to topological data analysis (TDA). TDA has become a fundamental analysis tool for scalar fields in scientific visualization due to its great potential for data abstraction and aggregation [16]. A large number of topological descriptors exist that are used to generate automatic visualizations or guide interactive exploration. In contrast, the use of topological methods in multi-field visualization is still a largely unexplored domain [43]. There are some interesting ideas to extend concepts from TDA to multi-variate fields. Examples are Reeb spaces a high-dimensional analog of Reeb graphs [14], joint contour nets (JCN) [8] a discrete approximation of the Reeb space, or Jacobi sets, a concept that highlight points where the gradients of the individual scalar functions align [13]. However many of the proposed solutions involve complex theoretical concepts that are still challenging to use in practical applications.

Recent advancements address these issues by opening up key scalar field visualization methods, such as iso-surface and volume rendering [19], to multi-variate data. Namely, fiber surfaces and feature level sets (FLS) [18]. Initially, fiber surfaces have been introduced by Carr et al. [9] as a method for bi-variate data and have later been extended by Blecha et al. [3] to general multi-variate data. Fiber surfaces rely on the intersection of mesh cells and user-given polygons in attribute space. If there is no intersection, the method will not yield a fiber surface. FLS remedies this restriction by treating the case of intersection as the zero level-set of a distance field. The distance field is computed with respect to a geometry and/or range in attribute space. Similar to fiber surfaces, this geometry is defined by the user. It is called a *trait* and defines the parameter configurations the user wishes to find in the data. As stated earlier, the zero level-set of this distance field will yield the same surface as a fiber surface. However, if the zero level set is empty, the user is able to investigate how close the field gets to the trait by rendering different level sets of the distance field.

With the feature definition via traits and their extraction in place, a useful next step for multi-field visualization is a topological analysis of feature level sets. Even though a trait defines and extracts a parameter setting of interest, little is known about the neighborhood of features and the overall structure of the resulting distance field. Depending on the problem statement, the occurrence of highly local features vs. larger features might be of interest. Noise may generate data points fulfilling the trait criteria and clutter the distance field.

Topological analysis, in particular the use of merge trees, addresses these concerns in a robust manner. The proposed method makes the features extracted through FLS computation browsable through an interface to interact with the corresponding merge tree. The combination of FLS and scalar field topology provides a simple concept for topological analysis of multi-variate fields.

**Contributions** Our approach combines two straightforward and established methods which, in combination, allow for topological analysis of multi-variate data and tensor fields. We first compute a distance field using the FLS method and use it as input to compute a merge tree, see Figure 2. The choice of merge trees comes naturally as the leaves in the tree correspond to those areas in the data that are closest to the trait(s). The trait-induced merge tree opens up an array of topology-based simplification and query methods that can be used to gain further insight into the underlying structure of the data. The method is flexible with respect to chosen features of interest represented by traits in attribute space and the results are easy to interpret.

## 2 RELATED WORK

The method presented combines recent advancements in multi-field visualization as well as topological data analysis for visualization, namely merge trees. In the following, we will briefly put our work into the context of existing contributions.

### Attribute space interactions for multi-field visualization

Coordinated multiple linked views are often used to cope with multi-fields or other complex data, see e.g. the state-of-the-art report by Roberts et al. [32]. In most of this work, representations in attribute space play an as important role as spatial representations. In this context, multi-dimensional transfer function (TF) design plays an important role to generate spatial representations [27]. To reduce the complexity of the task a common approach is to use attribute space clustering or segmentation. Wang et al. [39] propose to segment a 2D density plot in attribute space by facilitating a Morse decomposition to automatically generate a TF. With a similar goal, Cai et al. [7] suggest a two-level approach, starting with a topology-preserving dimensionality reduction step followed by a clustering step. Another cluster-hierarchy-based method used for interactive transfer function generation was proposed by Dobrev et al. [12]. Here a cluster tree visualization in combination with parallel coordinates serves as an interactive interface.

Given the multidimensional nature of tensors, visualization methods for multi-variate data have also been developed in context with tensor field visualization. Kindlmann et al. [24] introduced direct volume rendering (DVR) for tensor fields using a barycentric shape space for opacity assignment. Jankowai et al. [19] present an interface using glyph widgets to design a transfer function for rendering tensor features. They augment the volume rendering with a texture for directional information. The widgets employ characteristic glyph representatives for intuitive navigation through the attribute space. Refer to [17, 26] for respective overview articles.

### Topology guided visualisation

Concepts from topological data analysis can be found in many visualization applications for scalar fields. Especially structures like the contour tree or merge tree are frequently used to guide visualizations in an increasing number of applications. In an early paper, van Kreveld et al. [38] augment a contour tree with seed sets for fast iso-contour computation. Weber et al. [40] present an approach for volume rendering of topologically segmented scalar fields, assigning a different transfer function to each segment. Takahashi et al. [36] follow a similar goal proposing automatic transfer functions accentuating topological changes in scalar fields. Takeshima et al. [37] have extended this work by introducing a set of topological attributes that serve as auxiliary variables for the design of multi-dimensional

transfer functions. Especially successful are methods that integrate results from the topological analysis into interactive frameworks. Bremer et al. [5] offer a linked-view interface for efficient analysis of burning cells from turbulent combustion simulations. They augment the computed merge tree with global and local statistical information, supporting extensive feature extraction and analysis. Bock et al. [4] also used a combination of merge tree analysis and an interactive user interface for efficient segmentation of micro-CT scan data of fishes. Besides the merge tree, other topological structures have also been used in visual frameworks. An example is the work by Shivashankar et al. [35] who present a queryable hierarchy of Morse-Smale complexes that allows astronomers to examine filamentary structures of the cosmic web on different scales. A survey of topology-based methods in visualization can be found in the report by Heine et al. [16].

### Extension of iso-surfaces and topological concepts to multi-fields

A core feature of the presented method is the rendering of relevant iso-surfaces for multi-fields. Multi-field iso-surfaces have initially become accessible through the work of Carr et al. [9]. They have introduced an extension of iso-surfaces to bi-variate data using sets of fibers. Fibers can be understood as the bi-variate equivalent to iso-lines which allows for the generation of fiber surfaces based on polygons defined in attribute space.

Different extraction and rendering methods for fiber surfaces have since been presented. Wu et al. [42] offers a system for interactive exploration of bi-variate data through real-time pixel-perfect fiber surface rendering. This is achieved through intersection tests in range space to calculate fiber surfaces on the fly. Klacansky et al. [25] implemented a topology-agnostic and exact calculation of fiber surfaces with a speedup of up to two orders of magnitude compared to the initial article using a range of acceleration methods.

Fiber surfaces have then been generalized to general multi-variate data by Raith and Blecha et al. [3, 31]. In their framework, user-defined geometries in attribute space are called interactors. Intersections of input cells with this geometry in attribute space define the set of points in the spatial domain that form the iso-surface.

The latest addition to multi-field iso-surface visualisation has been feature level-sets (FLS) [18]. They are an approach to generalising iso-surfaces to multi-variate data via distance field computation in attribute space. For a given set of points in attribute space, the distance towards a user-defined geometry or range is calculated. The geometry or range is called a trait and defines the parameters of interest. Once the distance field computation is finalised, the resulting field is pulled back into domain space and can be used to render iso-surfaces.

FLS have been applied by Nguyen et al. [29] and Athawale et al. [1] to separate and visualise large- and small-scale structures in Taylor-Couette flow simulation data and to examine the correlation between velocity magnitude and temperature attributes in numerical simulations of idealised solar farms, respectively.

## 3 DATA, TRAITS, AND MERGE TREES

The proposed method is based on three main concepts. Multi-variate data, feature level-sets and traits, and merge trees. A summary of these concepts is given below. For a more extensive definition, we refer the reader to [18] and [10], respectively.

### 3.1 Multi-variate data

We assume a multi-variate data as input, given as a set of continuous fields  $F_1, F_2, \dots, F_m : \mathbb{X} \rightarrow \mathbb{R}$  defined on the *data domain*  $\mathbb{X}$ . In the case of tensor fields, these functions are the components of the tensor field. We then assemble an *attribute space*  $\mathbb{A} \subset \mathbb{R}^n$  by combining a set of (selected) field values and possibly some derived quantities.

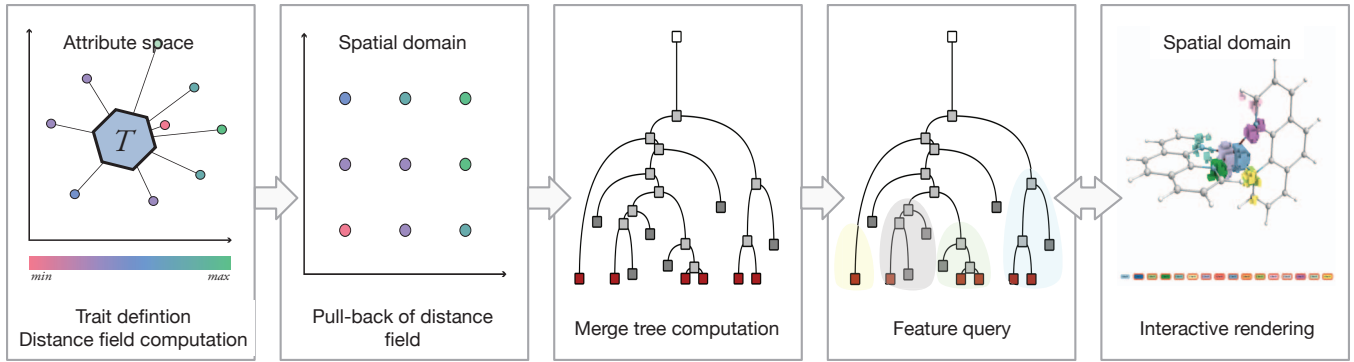


Figure 2: Pipeline schematic. The user-defined trait corresponds to the parameter settings that are of interest. For every vertex, the distance to this trait is calculated in attribute space, whereafter the distance value is assigned to the vertex in the domain (pull-back). This concludes the feature level-set computation. The resulting distance field serves as input for the computation of a merge tree. The tree undergoes simplification and can then be queried in several ways, see Sec. 4.1. Finally, the user can interact with the resulting domain segmentation via a legend or a slice through the data.

Such a data set is then transformed and summarised by a multi-variate mapping,  $f : \mathbb{X} \rightarrow \mathbb{A}$ , where  $n$  is the number of selected field values and their derived quantities. The attribute space  $\mathbb{A}$  is equipped with a metric (e.g., a Euclidean metric), denoted as  $d_{\mathbb{A}}$ .

### 3.2 Feature level sets

We define a *trait*  $T$  as a subset in the attribute space  $T \subset \mathbb{A}$ .  $T$  may, e.g., be a convex polygon, a point, a collection of points, or a line segment. A *feature* in the data domain is the pre-image of a trait from the attribute space,  $f^{-1}(T) = \{x \in \mathbb{X} \mid f(x) \in T \subset \mathbb{A}\}$ . For an arbitrary trait in an attribute space of arbitrary dimension, its corresponding feature in the data domain may be empty. Therefore, for a fixed trait  $T \subset \mathbb{A}$ , we define the *trait distance field*,  $d_T : \mathbb{A} \rightarrow \mathbb{R}$ , where  $d_T(a) = \min_{t \in T} d_{\mathbb{A}}(a, t)$ . The *trait-induced distance field* (or *feature distance field*) is a scalar function defined as  $h_T = d_T \circ f : \mathbb{X} \rightarrow \mathbb{R}$ . Finally, the *trait-induced level sets* are the level sets of  $h_T$ ,  $h_T^{-1}(c) = \{x \in \mathbb{X} \mid h_T(x) = c\}$ .

### 3.3 Merge trees

Let  $g : \mathbb{X} \rightarrow \mathbb{R}$  be a continuous scalar field. For the computational purpose, assume  $g$  is defined on a simply connected compact simplicial complex  $\mathbb{X}$  and is linearly interpolated on the interiors of its simplices. Two points  $x, y \in \mathbb{X}$  are considered *equivalent*, denoted by  $x \sim y$ , if  $g(x) = g(y)$  and  $x$  and  $y$  are a part of the same connected component of the sub-level set, i.e.  $g^{-1}(-\infty, g(x)] = g^{-1}(-\infty, g(y)]$ . The quotient space  $\mathbb{X}/\sim$  is called a *merge tree* of  $g$ . The merge tree records birth, death, and merge events of sub-level set components during a sweep of  $g$  from  $-\infty$  to  $\infty$ . Typically, merge trees are computed using algorithms based on the work by Carr et al. [10]. It is based on a *sub-level set filtration* of  $g$ , which is observing changes in a sequence of nested sub-level sets connected by inclusions.

### 3.4 Merge tree simplification

The original tree may contain many leaves coming from noisy data, imprecision, or irrelevant structures. Two common metrics, that we consider in our pipeline, for determining which leaves and arcs should be simplified (merged into their parent arc) are persistence and hypervolume.

**Persistence** In context with merge trees, persistence arises from a *sub-level set filtration* generating a pairing of critical points  $(x_i, y_i)$ . During filtration, a feature is generated in one critical point and disappears in the other. The interval spanned by the function values of the critical points  $[g(x_i) = g(y_i)]$  represents the feature lifetime interval. To each pair of critical points, one can then assign

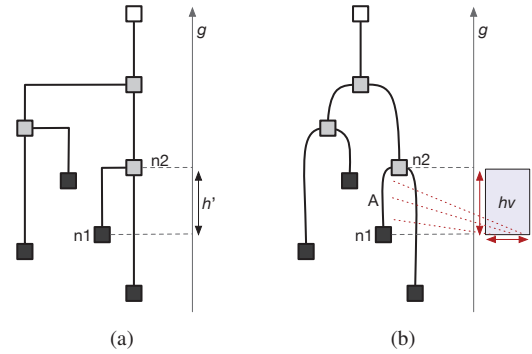


Figure 3: Simplification metrics: (a) Persistence. Persistence  $h' = g(n_2) - g(n_1)$  where  $g(x)$  is the scalar function over the domain. Where  $n_1$  and  $n_2$  are two paired critical points. (b) Hypervolume. For any arc  $A$ , the volume of all associated vertices is accumulated and multiplied by the height of the corresponding arc. Hence, hypervolume is a metric that takes the spatial embedding of the data into account while persistence is only related to the data range.

a *persistence value* which is the difference of the scalar values in the two critical points  $g(x_i) - g(y_i)$ . It gives some notion feature stability [11]. This critical point pairing can be used for a controlled simplification of the data removing features ordered by their persistence value. A *branch decomposition tree* derived from a merge tree is a hierarchical representation of these pairs of critical points [30], see Fig. 4b.

The geometric interpretation of a low persistence feature in a two-dimensional example would be a shallow valley on a height-field map. Likewise, a high persistence value equates to a deep intrusion in a field. Low persistence values often occur in noise where small differences in function values create irrelevant extremal points. To retain the relevant features while eliminating irrelevant ones, low-persistent branches, leaves, and their incident arcs are merged into their parent branch. As shown in Fig. 3a, persistence directly correlates to tree height.

**Hypervolume** Hypervolume [4] is a metric that takes local geometric measures of the underlying data domain into account. As shown in Fig. 3b, hypervolume is the product of arc height and the volume contained by the region corresponding to the arc. In practice,



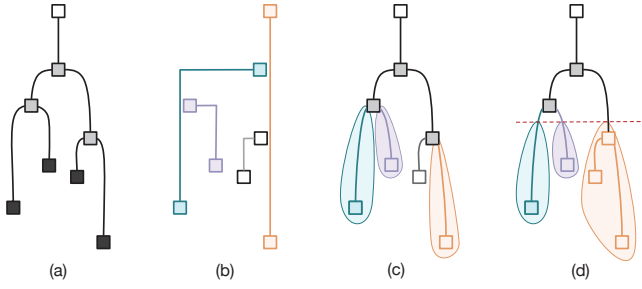


Figure 4: Query methods. (a) shows the original tree. The user may choose between segmentation of the tree based on (b) branch decomposition, (c) leaf nodes, (d) sub-trees. Every query method has its merits and drawbacks, see Sec. 4.1.

this is the number of voxels contained in a segment multiplied by the difference in function values at the minimum and the saddle connected by the arc. Unlike persistence, the hypervolume metric requires additional information about vertex-segment association for computation.

#### 4 TRAIT-INDUCED MERGE TREES

Feature level sets (FLS) enable the user to extract features as iso-surfaces according to a certain feature distance interest. They do not, however, offer a way to perform any sort of selection or filtering which may result in a cluttered rendering or the inclusion of noise in the rendering. Combining FLS with scalar field topology, merge trees in particular, adds the option to select and filter surfaces based on some metric, see Sec. 3.4. The use of merge trees comes naturally as their leaves correlate to those data points in the data which are closest to the defined trait.

For a fixed trait  $T \subset \mathbb{A}$ , given its trait-induced distance field  $h = h_T : \mathbb{X} \rightarrow \mathbb{R}$ , we can define a *trait-induced merge tree* by tracking how the level set components  $h^{-1}(c)$  merge as we vary the distance parameter  $c$ . Similarly, we obtain a *trait-induced merge tree* by tracking the evolution of the sub-level sets  $h^{-1}(\infty, c]$ . Such a trait-induced merge tree may be used directly (or indirectly) as an interface to guide the exploration of multivariate data.

##### 4.1 Merge tree-based features

Data sets as well as research questions are different in nature. Therefore, we provide a number of different simplification and query methods for the merge tree, each of which emphasizes different aspects of the data.

**Branch decomposition** The branch decomposition is a common representation for merge trees (see Fig. 4b). It allows for hierarchical simplification and querying. When using this representation as a basis for segmentation, the user specifies a threshold for the simplification (persistence or hypervolume) and the method returns a segmentation of the domain according to the simplification of the tree at that threshold. This method always contains one branch which connects the global minimum to the maximum. In terms of visualization, this branch is often problematic since its vertices usually enclose all others. When rendering the segment of this branch, all other segments as well as the global minimum may not be visible.

**Extremal points and their incident arcs** This method extracts the leaf nodes and their neighboring vertices and segments the domain accordingly (see Fig. 4c). Here too, the user first specifies a simplification threshold. Unlike in the first method, it is now the merge tree itself that is simplified and then queried. This method has the advantage of highlighting every minimum separately which gives a detailed overview of the spatial distribution of minima.

**Sub-trees** Sub-trees are extracted by first simplifying the tree as above and then cutting it at a user-specified level (see Fig. 4d). The segments are then given by the vertices whose function value is below the threshold and which are contained within the branches that are directly affected by the cut. The result is something similar to contour forests where each segment in the domain is specified by a sub-tree originating from the cut downward.

#### 5 INTERACTION AND RENDERING

The method has been implemented in an interactive visualization framework using Inviwo [21] providing a large variety of rendering options. Besides specifying the rendering options the main interaction possibilities concern the trait specification and the feature selection.

**Trait specification** For the trait specification, we provide similar options as the ones introduced in the original paper about feature level sets [18]. In the current implementation, we support point traits which consist of one or a few points in attribute space; cubical traits, defined by intervals in all attribute space dimensions; or a few further explicit geometries. As the main interaction panel, we use a parallel coordinates plot where one line crossing all parallel axis corresponds to a point trait and a set of intervals for the cubical traits. For tensor field, we additionally support picking of tensors in a glyph rendering of the data set.

**Feature selection** For the feature, we provide two ways of interacting with the segmentation of the domain. The first interface consists of a legend positioned below the 3D rendering. The legend contains a button for every segment in the data. Clicking on such a button will toggle the voxel-wise rendering of that segment. Active buttons are highlighted with a red boundary. For navigation, the colors of the buttons and the rendered segments correspond to each other. Additionally, the buttons show the value of the minimum of the segment, which is a measure of the distance to the trait. This way, the user can select segments based on their distance to the trait.

#### 6 RESULTS

We demonstrate the powerful utility and generality of our proposed framework across a range of scalar, vector, and tensor field data sets.

##### 6.1 Two-point load

Since FLS was originally developed with tensor fields in mind, our first example is based on tensor fields. Using an individual tensor as a trait, FLS have introduced a notion of “tensor iso-surfaces”. In combination with trait-induced merge trees, this opens an entirely new concept of “tensor field topology”. To verify the proposed method, the first case study involves a well-known numerical material simulation of stresses inside a solid block. Onto the top of this block, two forces are applied, one pulling and one pushing, as illustrated in Fig. 5a. Subsequently, this data set will be referred to as ‘two-point-load’.

The output of the simulation is a stress tensor-field containing symmetric tensors which have six independent degrees of freedom. As described in Fig. 2, the first step is to define traits and calculate the distance field. The expected material stress is highly anisotropic at the points of impact, unaffected in areas furthest from the impact points and planar along the midsection between the impact points.

In order to express these states, we will use the following terminology: Eigenvalues are referred to as principal stresses  $\lambda_i$ . They are ordered such that  $\lambda_1 \geq \lambda_2 \geq \lambda_3$ , named major, intermediate, and minor principal stress. Linear, planar, and spherical anisotropies are given by  $c_l = \frac{\lambda_1 - \lambda_2}{\lambda}$ ,  $c_p = \frac{2(\lambda_2 - \lambda_3)}{\lambda}$ , and  $c_s = \frac{3\lambda_3}{\lambda}$  respectively with  $\lambda = \lambda_1 + \lambda_2 + \lambda_3$  [41]. One frequently used invariant is the maximum shear stress that is defined as  $\lambda_1 - \lambda_3$ . Fig. 5a shows two level-sets of this invariant.

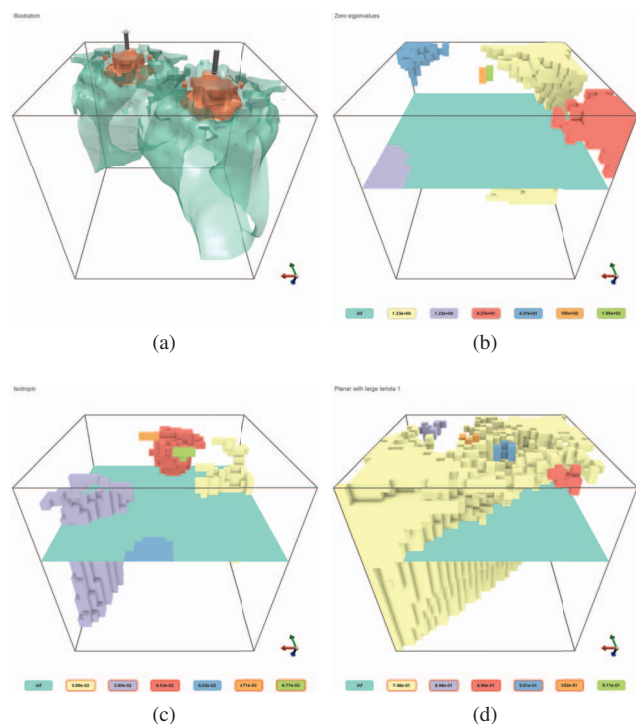


Figure 5: (a) Illustration of the two-point-load data set. The arrows indicate the forces applied to the block of metal. The iso-surfaces display the anisotropy that occurs in the material. (b) shows segmentation of distance field for regions where the eigenvalues are close to zero. (c) shows segmentation of distance field for high spherical anisotropy, i.e. isotropic behaviour. (d) shows segmentation of distance field for planar behaviour coupled with a high principal eigenvalue. The active segment buttons are highlighted with a red boundary.

The traits were defined to match these criteria to verify the resulting regions. For the first trait, all three principal stresses were set to zero. Fig. 5b shows regions that are close to this behaviour. The result is in line with our expectations since the highlighted regions are those farthest away from the impact points. Fig. 5c shows regions with isotropic behaviour. The corresponding trait was set to high spherical anisotropy ( $c_s$ ) and low linear and planar anisotropies ( $c_l$  and  $c_p$ ). Comparing it to Fig. 5b, we can see that the regions are neighbouring which makes sense. Regions that are entirely unaffected by the forces will have a small band of rather isotropic behaviour around them before a more distinctive stress distribution emerges. Lastly, the third trait was set to highly planar behaviour ( $c_p$ ) coupled with a high major principal stress value ( $\lambda_1$ ). Fig. 5d depicts the resulting regions. As expected, the middle section corresponds to this trait.

## 6.2 Extraction of acceptor and donor regions in molecular electronic transitions

In this case study, we analyse the molecular electronic transitions using feature level-sets and trait-induced merge trees. The electronic structure of a molecule undergoes a change on interaction with light. This change can be represented concisely by two scalar fields,  $\Phi_h$  and  $\Phi_p$ , denoting the spatial distribution of the electron before and after absorption of photon during the electronic transition [28]. Chemists are interested in studying how the localisation of the electronic distribution changes during the transition and how different

molecular configurations affect the transitions. In particular, it is crucial to identify which parts of the molecule lose and gain charge, and therefore, act as donor and acceptor regions within the molecule, respectively. Recently, Sharma et al. [33, 34] proposed considering the two scalar fields corresponding to an electronic transition as a single multi-field and applied bi-variate analysis on the resulting field. They suggested examining the patterns in the continuous scatter plots [2] of the bi-variate field corresponding to the complete molecule or sub-regions within the molecule can help reveal the donor and acceptor behaviour.

Here, we consider electronic transitions in two copper complexes with slightly different configurations. One of these complexes has the same two molecular groups around the central copper atom, a case of *symmetric ligands*. The other complex has two different molecular groups around the copper atom, a case of *asymmetric ligands*. The task of interest to the chemists is to identify the donor and acceptor regions within these two complexes and compare if and how these two configurations differ. To accomplish this, we first convert the notion of a donor and acceptor into well-defined traits. A donor is a subset of the molecule where there is more concentration of electronic density before the transition compared to the state of the electronic distribution after transition. That is, in general, a donor region can be characterised by the set of points satisfying the condition  $|\Phi_h| > |\Phi_p|$ . However, in the case of an ideal donor behaviour, we expect  $|\Phi_p|$  to be zero while  $|\Phi_h|$  is simultaneously very high. Therefore, we define the donor trait as a set of two points with coordinates  $(\max|\Phi_h|, 0)$  and  $(-\max|\Phi_h|, 0)$  in the bi-variate range space spanned by  $\Phi_h \times \Phi_p$ . This point set trait is indicated by two red disks in Fig. 6a and Fig. 7a. Following this idea, we define the acceptor by a trait consisting of two points,  $(0, \max|\Phi_p|)$  and  $(0, -\max|\Phi_p|)$ , see Fig. 6d and Fig. 7d.

With these point traits for donor and acceptor regions, we extract the feature level-sets for both molecules. In the case of symmetric ligands, the level-set corresponding to the donor trait concentrates around the central copper atom, as can be observed in Fig. 6b. This is as expected by the chemists, since the copper atom is known to act as a strong donor in these complexes. The feature level-sets corresponding to the acceptor trait are, however, distributed over the surrounding two molecular groups, see Fig. 6e. This behaviour is also as expected because there is no reason for the electron to prefer one molecular group over the other, both groups being the same. These feature level-sets extracted for the donor and acceptor traits were very appreciated by our collaborating partner who shared this data with us and is an expert in this domain. Since selecting the right distance threshold for feature extraction can be tricky and involves interactive visual exploration of the data, we further applied the analysis based on trait-induced merge trees to this transition. We queried the regions corresponding to the leaf nodes with lowest values in the merge tree of the donor trait and automatically found the region concentrated around the copper atom, as shown in Fig. 6c. Similarly, for the acceptor trait, the regions are distributed over the two groups around the copper atom as shown in Fig. 6f. Interestingly, these regions corresponding to leaves with lowest values closely capture the two molecular groups as separate regions in the merge tree segmentation. This automatic subdivision of the acceptor region into sub-regions matching the chemical subgroups in the molecule was also appreciated by our collaborator.

Next, we repeat the same analysis on the second copper complex with asymmetric ligands. As was the case previously, the donor region is still concentrated on the central copper atom, as shown in Fig. 7c. However, the acceptor region is now concentrated on only one of the two surrounding molecular groups as can be seen in Fig. 7f. This suggests, in the case of asymmetric ligands, electronic charge transfer can occur with preference for one molecular group over the other. As mentioned before, such change in behaviour of charge transfer in transition due to change in molecular configuration

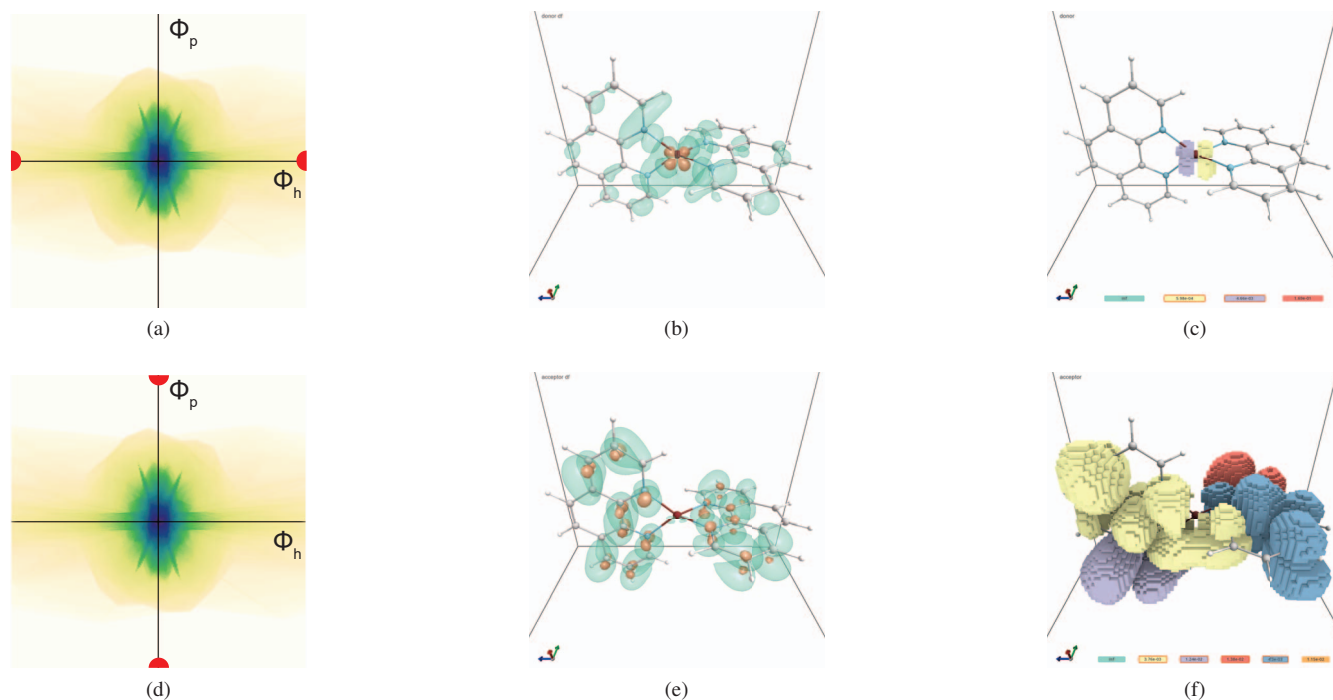


Figure 6: Molecular electronic transition data set (copper complex) with symmetric ligands. (a) and (d) CSP plots and the donor and acceptor trait, respectively (red dots). (b) donor and acceptor distance field obtained from the FLS computation. (c) shows donor regions while (f) shows acceptor regions. The merge tree has been simplified using the hypervolume metric. The extracted regions correspond to the  $n$  lowest leaves in the tree.

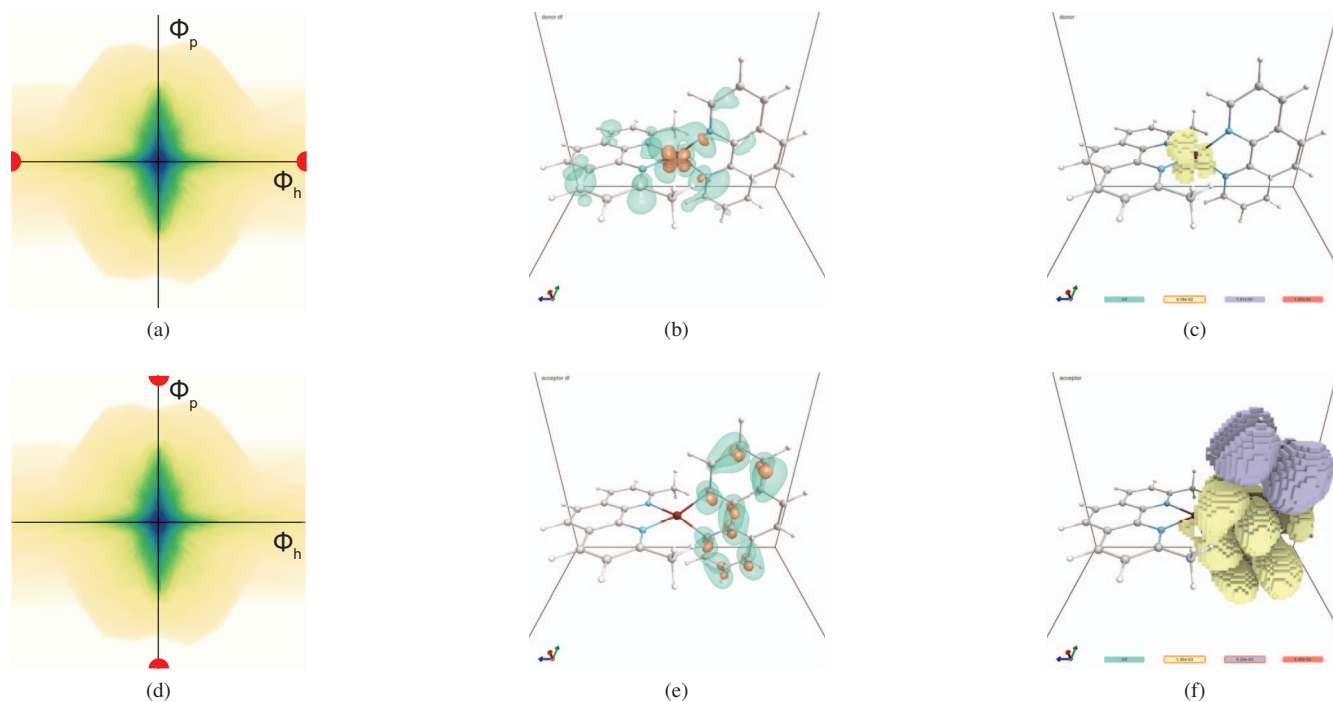


Figure 7: Molecular electronic transition data set (copper complex) with asymmetric ligands. (a) and (d) CSP plots and the donor and acceptor trait, respectively (red dots). (b) donor and acceptor distance field obtained from the FLS computation. (c) shows donor regions while (f) shows acceptor regions. The merge tree has been simplified using the hypervolume metric. The extracted regions correspond to the  $n$  lowest leaves in the tree.



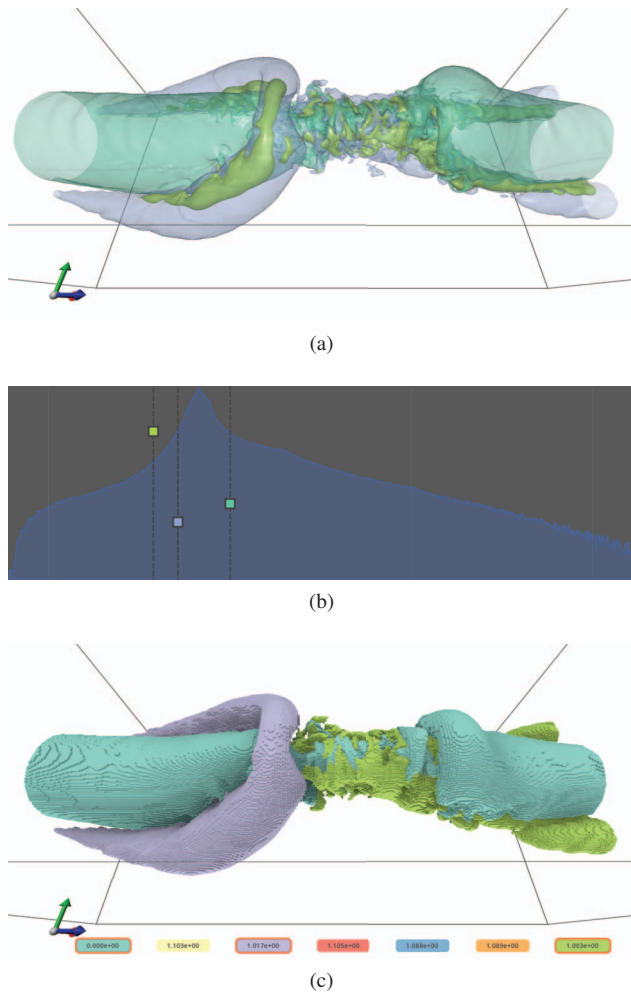


Figure 8: Vortex re-connection simulation. (a) shows volume rendering of the distance field. (b) shows histogram and iso-surface placement. The histogram of the distance field is given by the blue graph in the background. Distance from the trait is mapped to the x-axis while opacity is mapped to the y-axis. (c) shows segmentation using branch decomposition.

is of particular interest to the chemists, and this analysis is possible with our proposed techniques. Lastly, we want to point out that the continuous scatter plots for both the complexes look quite similar, as seen in Fig. 6a and Fig. 7a. Therefore, it is difficult to distinguish between the character of these two transitions just based on these plots, there is a need to explore the spatial domain as well which is facilitated by the feature level-sets and the domain segmentation based on trait-induced merge trees.

### 6.3 Vortex re-connection

Vortices are regions of rotational fluid flow around a core line defined by an arbitrary curve in space. They play a fundamental role in the study of fluid dynamics. There are numerous methods for the identification and extraction of vortex structures and their core lines [15, 23]. However, the dynamics of vortices – how they evolve and interact with each other – is less understood, and therefore, this topic is still under active research [6, 22]. The data we examine as part of this case study concerns the interaction of two parallel counter-rotating vortices. Over time, these two vortices come closer and go through a *re-connection* event. This phenomenon can be

observed, for example, in the vortices shed by the wing tips of an airplane. These vortices can sometimes be seen in the sky as condensation trails. The trails close to the airplane are initially parallel; then followed by the re-connection event, they form closed loop-like structures; before they dissipate completely in the air. Our collaborators performed a numerical simulation of this phenomenon. One of their main goals for conducting this simulation is to understand the re-connection event. It is known that after re-connection of two counter-rotating vortices, a *horseshoe* like structure emerges at the re-connection point which grows bigger in size with time and finally results in the formation of disconnected loops. Thus, the identification and analysis of this horseshoe structure is a key task in the analysis of this particular simulation data.

Unlike vortices and their cores, there is no clear definition available that we can directly apply to automatically extract these horseshoe structures. We therefore explored whether feature level sets and the trait-induced merge tree are useful for this task. Based on input from the domain experts who conducted this simulation, we chose a time step after the re-connection event where the horseshoe structure is present. One characteristic of the horseshoe is that it appears as a weaker vortex in a plane orthogonal to the two parallel vortices. Another characteristic of interest, which applies to vortices in general, is that they have lower pressure at their cores. We therefore use these two ideas to design a trait that captures the horseshoe structure. Since the two parallel vortices are along the Z-axis and are separated from each other in the Y direction, we can expect the horseshoe to lie in the XY plane with the vortex core roughly aligned with the Y-axis. Considering the velocity vector field  $\mathbf{v}$  as a multi-field consisting of three velocity components ( $v_x, v_y, v_z$ ), we can formally convert the ideas above into a trait with high absolute values of  $v_x$  and  $v_z$  combined with simultaneously low absolute values of  $v_y$ . Further, we add the pressure criterion to the trait and set it below. Fig. 8 shows the results obtained for this trait. As evident from the feature level sets shown in Fig. 8a, we can see the horseshoe-like structure on the left within the green and purple level sets. These two level sets are also closest to our trait as evident from the histogram in Fig. 8b.

For the final extraction of the horseshoe structure, we employ the branch decomposition segmentation derived from the trait-induced merge tree. As shown in Fig. 8c, the purple segment captures the horseshoe structure. This result suggests feature level sets and trait-induced merge trees can be utilized for extracting features from multi-fields, which are difficult to extract otherwise, using simple queries.

## 7 DISCUSSION

The method presented provides a novel approach of topological data analysis for multi-variate data and tensor fields and is capable of producing valuable insight into the data. However, in order to achieve these results, special attention needs to be paid in its implementation and usage.

Conceptually, the cutoff method for querying the merge tree should provide an intuitive segmentation of the domain. In practice, it proved to be hard to steer without visual aid. In order to find an adequate cutoff threshold, a persistence diagram or a tree representation with a cutoff slider could potentially make the method more usable and useful. Regarding the distance field computation, the choice of distance metric plays a significant role for the resulting distance field and will therefore influence the tree computation. The advantage of the feature level-set method is that the distance metric can be exchanged for the most appropriate one. However, determining the trait will normally depend on domain knowledge. As for the determination of the trait, this is a task that requires some domain knowledge. Additionally, interpolation artifacts may arise for non-linear data such as tensor fields. In order to compute exact level sets, the data would need to be interpolated in the spatial do-

main during the ray-marching process instead of pre-computing the distance field. For a thorough discussion of these aspects of FLS we refer the reader to [18].

Despite these challenges, the method presented is conceptually simple as opposed to previously proposed methods. The computation of feature level sets is a straightforward process of calculating a distance field and merge tree computation is both well-researched and available via open-source libraries. This simple concept allows for the generalization of fundamental topological concepts with all their strength, as here demonstrated with the merge tree. Notions like persistence can be naturally extended to multi-fields or tensor fields supporting a multi-scale analysis.

On the user side, the interface consists of first defining the trait and then browsing the segmentation via a legend or slice. For a domain expert or educated user, defining the trait should be simple enough, given the nature of data analysis is targeted instead of exploratory. Once the trait is defined, the process is virtually automatic. Prior to rendering, an appropriate simplification threshold for the merge tree needs to be found in order to obtain sensible results for the segmentation. Visual aids such as persistence diagrams may be employed here. In the future, we plan to extend the supported rendering styles also including automatic transfer function design.

The implementation for FLS used in this article is GPU-based and performed at interactive frame rates for all data sets used. The computation of the merge tree took less than a minute for the selected data sets. Feature extraction based on the different methods presented in Sec. 4.1 took a comparable amount of time. Additionally, the process of feature extraction can be relayed onto the GPU and should perform at interactive frame rates. Given these numbers, we argue that the application is usable in workflows for domain experts.

## 8 CONCLUSIONS

In this paper we have introduced topology-based segmentation of multi-field data based on traits. As introduced in [18], traits are geometries defined in attribute space describing parameters of interest. After the distance from this trait has been computed for every vertex or cell, the resulting distance field is pulled back into the spatial domain. Subsequently, we perform a segmentation of this distance field based on merge trees. Finally, we allow the user to query the merge tree with a set of different methods and parameters and generate according renderings. This approach provides a entirely new link to topological analysis of multi-fields and tensor fields.

For the three case studies presented in this article, the proposed method produced the expected and desired outcomes. In particular, the extraction of donor and acceptor regions yielded results greatly appreciated by the domain scientist. Hence, we hope the field of computational chemistry may benefit from this method. The investigation of automating the segmentation process is a promising extension of this work.

## ACKNOWLEDGMENTS

This work is supported by Swedish e-Science Research Centre (SeRC), ELLIIT environment for strategic research in Sweden, and the Swedish Research Council (VR) grant 2019-05487. The application was implemented using the open-source software Inviwo [20]. The computation of contour/merge trees uses code provided by Harish Doraiswamy. The authors express their gratitude to Mathieu Linares for providing the simulation data for charge transfer and for providing very useful expert feedback on the obtained results and visualisations for this case study. The flow data set used in this paper was produced and supplied by Professor Jan Nordström, Department of Mathematics, Linköping University and Dr. Marco Kupiainen, Rossby Centre, SMHI.

## REFERENCES

- [1] T. M. Athawale, B. J. Stanislawski, S. Sane, and C. R. Johnson. Visualizing Interactions Between Solar Photovoltaic Farms and the Atmospheric Boundary Layer. In *Proceedings of the Twelfth ACM International Conference on Future Energy Systems*, e-Energy '21, p. 377–381. Association for Computing Machinery, New York, NY, USA, 2021. doi: 10.1145/3447555.3466599
- [2] S. Bachthaler and D. Weiskopf. Continuous scatterplots. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1428–1435, 2008. doi: 10.1109/TVCG.2008.119
- [3] C. Blecha, F. Raith, A. J. Präger, T. Nagel, O. Kolditz, J. Maßmann, N. Röber, M. Böttinger, and G. Scheuermann. Fiber Surfaces for Many Variables. *Computer Graphics Forum*, 39(3):317–329, 2020. doi: 10.1111/cgf.13983
- [4] A. Bock, H. Doraiswamy, A. Summers, and C. Silva. Topoangler: Interactive topology-based extraction of fishes. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):812–821, 2018. doi: 10.1109/TVCG.2017.2743980
- [5] P.-T. Bremer, G. Weber, J. Tierny, V. Pascucci, M. Day, and J. Bell. Interactive Exploration and Analysis of Large-Scale Simulations Using Topology-Based Data Segmentation. *IEEE Transactions on Visualization and Computer Graphics*, 17(9):1307–1324, 2011. doi: 10.1109/TVCG.2010.253
- [6] R. Bujack, L. Yan, I. Hotz, C. Garth, and B. Wang. State of the art in time-dependent flow topology: Interpreting physical meaningfulness through mathematical properties. *Computer Graphics Forum*, 39(3):811–835, 2020. doi: 10.1111/cgf.14037
- [7] L. Cai, B. P. Nguyen, C.-K. Chui, and S.-H. Ong. A Two-Level Clustering Approach for Multidimensional Transfer Function Specification in Volume Visualization. *The Visual Computer*, 33:163–177, 2017.
- [8] H. Carr and D. Duke. Joint Contour Nets. *IEEE Transactions on Visualization and Computer Graphics*, 20(8):1100–1113, 2014. doi: 10.1109/TVCG.2013.269
- [9] H. Carr, Z. Geng, J. Tierny, A. Chattopadhyay, and A. Knoll. Fiber Surfaces: Generalizing isosurfaces to bivariate data. *Computer Graphics Forum*, 34(3):241–250, June 2015. doi: 10.1111/cgf.12636
- [10] H. Carr, J. Snoeyink, and U. Axen. Computing Contour Trees in all Dimensions. *Computational Geometry*, 24(2):75–94, 2003. Special Issue on the Fourth CGC Workshop on Computational Geometry. doi: 10.1016/S0925-7721(02)00093-7
- [11] D. Cohen-Steiner, H. Edelsbrunner, J. Harer, and Y. Mileyko. Lipschitz Functions Have  $l_p$ -Stable Persistence. *Foundations of Computational Mathematics*, 10(2):127–139, 2010.
- [12] P. Dobrev, T. van Long, and L. Linsen. A Cluster Hierarchy-Based Volume Rendering Approach for Interactive Visual Exploration of Multi-Variate Volume Data. In *Proceedings of Vision, Modeling, and Visualization (VMV'11)*, pp. 137–144, 2011.
- [13] H. Edelsbrunner and J. Harer. Jacobi Sets of Multiple Morse Functions. *Foundations of Computational Mathematics - FoCM*, 01 2004.
- [14] H. Edelsbrunner, J. Harer, and A. K. Patel. Reeb Spaces of Piecewise Linear Mappings. In *Proceedings of the Twenty-Fourth Annual Symposium on Computational Geometry*, SCG '08, p. 242–250. Association for Computing Machinery, New York, NY, USA, 2008. doi: 10.1145/1377676.1377720
- [15] T. Günther and H. Theisel. The state of the art in vortex extraction. *Computer Graphics Forum*, 37(6):149–173, 2018. doi: 10.1111/cgf.13319
- [16] C. Heine, H. Leitte, M. Hlawitschka, F. Iuricich, L. De Floriani, G. Scheuermann, H. Hagen, and C. Garth. A Survey of Topology-Based Methods in Visualization. *Computer Graphics Forum*, 35(3):643–667, 2016. doi: 10.1111/cgf.12933
- [17] C. Hergl, C. Blecha, V. Kretschmar, F. Raith, F. Günther, M. Stommel, J. Jankowai, I. Hotz, T. Nagel, and G. Scheuermann. Visualization of Tensor Fields in Mechanics. *Computer Graphics Forum*, 40(6):135–161, 2021. doi: 10.1111/cgf.14209
- [18] J. Jankowai and I. Hotz. Feature Level-Sets: Generalizing iso-surfaces to multi-variate data. *IEEE Transactions on Visualization and Computer Graphics*, 26(2):1308–1319, Feb 2020. doi: 10.1109/TVCG.2018.2867488



- [19] J. Jankowai, R. Skånberg, D. Jönsson, A. Ynnerman, and I. Hotz. Tensor Volume Exploration Using Attribute Space Representatives. In *LEVIA 2020: Leipzig Symposium on Visualization in Applications*, 2020. doi: 10.31219/osf.io/qu8rz
- [20] D. Jönsson, P. Steneteg, E. Sundén, R. Englund, S. Kottraval, M. Falk, A. Ynnerman, I. Hotz, and T. Ropinski. Inviwo - A Visualization System With Usage Abstraction Levels. *IEEE Transactions on Visualization and Computer Graphics*, 26(11):3241–3254, 2019. doi: 10.1109/TVCG.2019.2920639
- [21] D. Jönsson, P. Steneteg, E. Sundén, R. Englund, S. Kottraval, M. Falk, A. Ynnerman, I. Hotz, and T. Ropinski. Inviwo - a visualization system with usage abstraction levels. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 26(11):32–3254, 2020.
- [22] J. Kasten, I. Hotz, B. R. Noack, and H. Hege. Vortex merge graphs in two-dimensional unsteady flow fields. In M. Meyer and T. Weinkauff, eds., *14th Eurographics Conference on Visualization, EuroVis 2012 - Short Papers, Vienna, Austria, June 5-8, 2012*. Eurographics Association, 2012. doi: 10.2312/PE/EuroVisShort/EuroVisShort2012/001-005
- [23] J. Kasten, J. Reininghaus, I. Hotz, and H. Hege. Two-dimensional time-dependent vortex regions based on the acceleration magnitude. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2080–2087, 2011. doi: 10.1109/TVCG.2011.249
- [24] G. Kindlmann, D. Weinstein, and D. Hart. Strategies for Direct Volume Rendering of Diffusion Tensor Fields. *IEEE Transactions on Visualization and Computer Graphics*, 6(2):124–138, 2000. doi: 10.1109/2945.856994
- [25] P. Klacansky, J. Tierny, H. Carr, and Z. Geng. Fast and Exact Fiber Surfaces for Tetrahedral Meshes. *IEEE Transactions on Visualization and Computer Graphics*, 23(7):1782–1795, 2017. doi: 10.1109/TVCG.2016.2570215
- [26] A. Kratz, C. Auer, M. Stommel, and I. Hotz. Visualization and Analysis of Second-Order Tensors: Moving beyond the symmetric positive-definite case. *Computer Graphics Forum - State of the Art Reports*, 32(1):49–74, 2013.
- [27] P. Ljung, J. Krüger, E. Gröller, M. Hadwiger, C. D. Hansen, and A. Ynnerman. State of the Art in Transfer Functions for Direct Volume Rendering. In R. Maciejewski, T. Ropinski, and A. Vilanova, eds., *Computer Graphics Forum - STAR*, vol. 35, p. 23, 2016.
- [28] T. B. Masood, S. S. Thygesen, M. Linares, A. I. Abrikosov, V. Natarajan, and I. Hotz. Visual analysis of electronic densities and transitions in molecules. *Computer Graphics Forum*, 40(3):287–298, 2021. doi: 10.1111/cgf.14307
- [29] D. B. Nguyen, R. O. Monico, and G. Chen. A Visualization Framework for Multi-Scale Coherent Structures in Taylor-Couette Turbulence. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 27(2):902–912, 2021.
- [30] V. Pascucci, K. Cole-McLaughlin, and G. Scorzelli. Multi-Resolution Computation and Presentation of Contour Trees. In *IASTED conference on Visualization, Imaging, and Image Processing (VIIP 2004)*, pp. 452–290, 2005.
- [31] F. Raith, C. Blecha, T. Nagel, F. Parisio, O. Kolditz, F. Günther, M. Stommel, and G. Scheuermann. Tensor Field Visualization Using Fiber Surfaces of Invariant Space. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):1122–1131, jan 2019. doi: 10.1109/TVCG.2018.2864846
- [32] J. C. Roberts. State of the Art: Coordinated multiple views in exploratory visualization. In *International Conf. on Coordinated and Multiple Views in Exploratory Visualization*, pp. 61–71, 2007.
- [33] M. Sharma, T. B. Masood, S. S. Thygesen, M. Linares, I. Hotz, and V. Natarajan. Segmentation driven peeling for visual analysis of electronic transitions. In *2021 IEEE Visualization Conference, IEEE VIS 2021 - Short Papers*, pp. 96–100. IEEE, 2021. doi: 10.1109/VIS49827.2021.9623300
- [34] M. Sharma, T. B. Masood, S. S. Thygesen, M. Linares, I. Hotz, and V. Natarajan. Continuous scatterplot operators for bivariate analysis and study of electronic transitions. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–13, 2023. doi: 10.1109/TVCG.2023.3237768
- [35] N. Shivashankar, P. Pranav, V. Natarajan, R. v. d. Weygaert, E. P. Bos, and S. Rieder. Felix: A topology based framework for visual exploration of cosmic filaments. *IEEE Transactions on Visualization and Computer Graphics*, 22(6):1745–1759, Jun 2016. doi: 10.1109/tvcg.2015.2452919
- [36] S. Takahashi, Y. Takeshima, and I. Fujishiro. Topological Volume Skeletonization and its Application to Transfer Function Design. *Graphical Models*, 66(1):24–49, 2004. doi: 10.1016/j.gmod.2003.08.002
- [37] Y. Takeshima, S. Takahashi, I. Fujishiro, and G. Nielson. Introducing Topological Attributes for Objective-Based Visualization of Simulated Datasets. In *Fourth International Workshop on Volume Graphics, 2005.*, pp. 137–236, 2005. doi: 10.1109/VG.2005.194108
- [38] M. van Kreveld, R. van Oostrum, C. Bajaj, V. Pascucci, and D. Schikore. Contour Trees and Small Seed Sets for Isosurface Traversal. In *Proceedings of the Thirteenth Annual Symposium on Computational Geometry, SCG '97*, p. 212–220. Association for Computing Machinery, New York, NY, USA, 1997. doi: 10.1145/262839.269238
- [39] Y. Wang, J. Zhang, D. J. Lehmann, H. Theisel, and X. Chi. Automating Transfer Function Design With Valley Cell-Based Clustering of 2D Density Plots. *Computer Graphics Forum*, 31(3pt4):1295–1304, 2012. doi: 10.1111/j.1467-8659.2012.03122.x
- [40] G. Weber, S. Dillard, H. Carr, V. Pascucci, and B. Hamann. Topology-Controlled Volume Rendering. *IEEE transactions on visualization and computer graphics*, 13:330–41, 04 2007. doi: 10.1109/TVCG.2007.47
- [41] C.-F. Westin, S. Peled, H. Gudbjartsson, R. Kikinis, and F. A. Jolesz. Geometrical diffusion measures for MRI from tensor basis analysis. In *ISMRM '97*, p. 1742. Vancouver Canada, April 1997.
- [42] K. Wu, A. Knoll, B. J. Isaac, H. Carr, and V. Pascucci. Direct Multifield Volume Ray Casting of Fiber Surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):941–949, 2017. doi: 10.1109/TVCG.2016.2599040
- [43] L. Yan, T. B. Masood, R. Sridharamurthy, F. Rasheeda, V. Natarajan, I. Hotz, and B. Wang. Scalar field comparison with topological descriptors: Properties and applications for scientific visualization. *Computer Graphics Forum*, 40(3):599–633, 2021.