Volume 33 (2014), Number 8 pp. 28–37 COMPUTER GRAPHICS *forum*

Structural analysis of multivariate point clouds using simplicial chains

B. Rieck* and H. Leitte*

*Interdisciplinary Center for Scientific Computing, Heidelberg University, Germany

Abstract

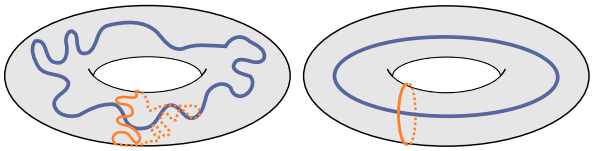
Topological and geometrical methods constitute common tools for the analysis of high-dimensional scientific data sets. Geometrical methods such as projection algorithms focus on preserving distances in the data set. Topological methods such as contour trees, by contrast, focus on preserving structural and connectivity information. By combining both types of methods, we want to benefit from their individual advantages. To this end, we describe an algorithm that uses persistent homology to analyse the topology of a data set. Persistent homology identifies high-dimensional holes in data sets, describing them as simplicial chains. We localize these chains using geometrical information of the data set, which we obtain from geodesic distances on a neighbourhood graph. The localized chains describe the structure of point clouds. We represent them using an interactive graph, in which each node describes a single chain and its geometrical properties. This graph yields a more intuitive understanding of multivariate point clouds and simplifies comparisons of time-varying data. Our method focuses on detecting and analysing inhomogeneous regions, i.e. holes, in a data set because these regions characterize data in a different manner, thereby leading to new insights. We demonstrate the potential of our method on data sets from particle physics, political science, and meteorology.

Categories and Subject Descriptors (according to ACM CCS): I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques E.1 [Data]: Data Structures—Graphs and networks

1. Introduction

Multivariate data analysis is a complex task. Projection methods, for example, provide valuable assistance in identifying structures in data sets. Higher dimensions often require special algorithms for dimensionality reduction, such as *multidimensional scaling* (MDS). The challenge with large dimensions is readily apparent when different data sets from the same source need to be compared. Given a set of experimental measurement data, for example, an analyst might be interested in finding out whether parts of a multi-run experiment contain anomalies caused by faulty equipment. Similar issues occur when multivariate time-varying point clouds need to be compared. Is it possible to detect whether two high-dimensional point clouds exhibit the same characteristics? How can differences be found and quantified? These questions are especially relevant when an analyst is looking for initially unknown patterns.

Aiming to support users in answering these questions, topological methods have gained much momentum over the last few years. In contrast to purely geometrical methods, topological algorithms focus on delivering a dimension-independent structural description of multivariate point clouds. Topological data analysis assumes that an input data set has been sampled from an unknown manifold in a high-dimensional space. This unknown manifold is then described in terms of its invariants, i.e. characteristic features (such as the genus of a surface in $\mathbb{R}^3$). *Persistent homology* is a popular method in topological data analysis that characterizes data sets through a multi-scale description of their simplicial homology: Each multi-scale feature detected by persistent homology corresponds to a high-dimensional "hole" in the data set. A "hole" can be considered an inhomogeneous region containing few or no data points. The basic visualizations for persistent homology [ELZ02, Ghr08, RML12] essentially count the number of holes in a data set. While this helps in distinguishing data sets, these visualizations remain very abstract and do not offer any links to the actual points in the point cloud. Recently, the calculation of persistent ho-



(a) *Non-localized loops* **(b)** *Perfectly-localized loops*

Figure 1: *An idealized torus data set and its two loops in dimension 1. The non-localized loops typically occur as the results of the standard persistence calculation algorithm.*

mology was modified to incorporate a more precise description of each "hole". This description is obtained by calculating *simplicial chains*, i.e. subsets of points that are supposed to be situated near the boundary of the detected hole. However, as persistent homology does not employ any geometrical information, the insight offered by simplicial chains is algebraically correct, but does not always correspond to a tangible feature in the point cloud (see Fig. 1a). This oversight can be rectified by employing geometrical information to localize the simplicial chains, which results in a very concise description of the boundary of an inhomogeneous region in the data set (see Fig. 1b).

Our method calculates the persistent homology of a data set and uses geometrical information to arrive at a localized description of simplicial chains. We then represent these high-dimensional chains in a graph structure in order to make them accessible for analysts. Through this process, we can qualitatively describe a multivariate point cloud through its inhomogeneous structures. Informally put, we focus on things that are *missing* from a data set, which results in a different characterization of multivariate point clouds. The **contributions** of our paper are:

- We introduce *simplicial chain graphs*, a new visual metaphor for the inhomogeneous topological structure of a data set.
- We identify key properties of simplicial chain graphs and demonstrate how these properties describe a data set.
- We analyse multiple data sets in order to explain how to make use of the new visual metaphor. More precisely, we show how localized topological information can support data exploration using established methods for multivariate data visualization.

2. Related work

There are two common classes of algorithms for exploring multivariate point clouds. Algorithms of the first class aim to identify relevant dimensions in the point cloud, while algorithms of the second class project all dimensions to 2D or 3D. *Isomap* [TdSL00] and *projection pursuit* [FT74] are well-established algorithms of the first class. Typical representatives of the second class are given by *scatterplot matrices* [Mar09, pp. 238–239] and *parallel coordinates* [Mar09, pp. 247–249]. Projections methods such as *multidimensional scaling* [Mar09, pp. 242–245], *principal component analysis* [Mar09, pp. 226–234], and *linear discriminant analysis* [Mar09, Chapter 2] are also employed very often.

A different approach for the analysis of multivariate point clouds aims to describe their topological structure. Because topological methods depend on intrinsic properties of a point cloud, they are less sensitive to the choice of metrics [Car09]. Thus, the number of topological methods increased over the past few years: Singh et al. [SMC07] present the Mapper algorithm, which obtains a simplicial complex from a data set via clustering algorithms. This complex is then displayed as a graph. The *generalized contour tree* introduced by Carr et al. [CSA02] is commonly used for the analysis of high-dimensional scalar fields. Duke et al. [DCK*12] recently generalized contour trees and Reeb graphs to obtain the *joint contour net* for the analysis of multivariate data sets from nuclear science. Likewise, *persistent homology* is often employed to analyse multivariate data sets. Several visualizations for its topological structures have already been suggested: Edelsbrunner et al. [ELZ02] introduce a compact visualization, called the *persistence diagram*, to show topological attributes on multiple scales. Ghrist [Ghr08] attempts to visualize the topological structure in a more accessible way using *persistence barcodes*. Adams and Carlsson [AC09] use persistent homology to identify submanifolds in the space of range image patches. Rieck et al. [RML12] demonstrate how to analyse general multivariate data sets using *persistence rings*, a radial visualization that balances compactness and accessibility.

Finally, there have also been efforts to incorporate geometrical information with topological algorithms in order to improve their results: Carr et al. [CSvdP10] show how to store and compute various geometrical properties for the contours of a contour tree. Erickson [Eri12] provides a detailed outline of multiple low-dimensional localization techniques for simplicial chains. Dey et al. [DLSCS08] introduce an algorithm for localizing handle and tunnel loops in 3D models. Erickson and Whittlesey [EW05] develop a graph-based localization scheme for the fundamental group and the first homology group of oriented 2-manifolds. Their work can be used to describe optimal cuts for the simplification of surfaces.

3. Mathematical background

In the following, we give brief accounts of the topological methods used by our method. We refer the reader to Hatcher [Hat02] and Munkres [Mun84] for a detailed introduction to algebraic topology. Edelsbrunner and Harer [EH10] give a comprehensive introduction to persistent homology and topological data analysis.

3.1. Algebraic topology

The notion of a *topological space*, i.e. a subset of $\mathbb{R}^m$ with some distance function, is a central concept in alge-

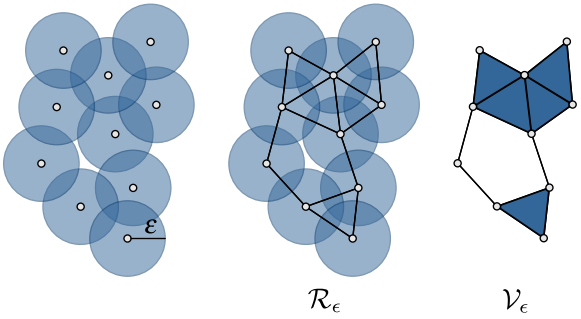


Figure 2: *Vietoris-Rips expansion procedure. From left to right: ε-neighbourhoods, the Rips graph $\mathcal{R}_\varepsilon$, and the Vietoris-Rips complex $\mathcal{V}_\varepsilon$.*

braic topology. Topologists aim to identify invariants of such spaces—properties that do not change when the space is stretched, bent, and twisted by homeomorphisms. A common invariant is given by *simplicial homology*. It assigns a set of algebraic groups, the *homology groups*, to an input space. Each group-theoretic generator of a k-dimensional homology group then describes a k-dimensional "hole" in the space. In lower dimensions, these holes can be described intuitively as *connected components* (dimension 0), *tunnels* (dimension 1), and *voids* (dimension 2). A higher-dimensional hole may be considered a part of a space where a k-dimensional sphere can be attached.

The algebraic rank of a homology group in dimension k is known as the kth *Betti number* b_k [Mun84, p. 24] of a topological space. Betti numbers are commonly used to distinguish different topological spaces from each other. For example, a 2-sphere has a single connected component, no tunnels, and encloses a void in $\mathbb{R}^3$. Its Betti numbers are thus $b_0 = 1$, $b_1 = 0$, and $b_2 = 1$ (the remaining b_k are zero). In contrast, a torus has a single connected component, two loops (namely, one around its centre, the other around its hollow tube—see Fig. 1b), and encloses a void in $\mathbb{R}^3$, as well. Its Betti numbers thus are $b_0 = 1$, $b_1 = 2$, and $b_2 = 1$ (again, the remaining b_k are zero). Betti numbers can thus discern a sphere and a torus from each other without requiring any geometrical information about the space. This signature property of Betti numbers also applies to higher dimensions, where distinguishing different spaces from each other by purely geometrical means might not be possible.

Algebraic topology has multiple equivalent descriptions for homology groups. From a computational point of view, simplicial homology is the most appealing one because it can be calculated algorithmically [Mun84, pp. 55–61]. The homology calculation is a matrix reduction scheme. It requires that the topological space is described as a *simplicial complex*, i.e. a generalized graph structure for arbitrary dimensions. The generators of each homology group are represented as a formal sum of simplices, the *simplicial chain*. Each simplicial chain describes a closed "path" (without a boundary) in a simplicial complex, i.e. a path that constitutes a hole in the complex. The chains in Fig. 1b, for example, may be seen as a closed path of 1-simplices (i.e. edges).

3.2. Persistent homology

Since real-world data sets are usually not described by simplicial complexes, their structure needs to be approximated. A common approximation uses a distance measure d (such as the Euclidean distance) and a scale parameter ε to obtain the *Rips graph* (also known as neighbourhood graph) $\mathcal{R}_\varepsilon$ of the data set. $\mathcal{R}_\varepsilon$ contains an edge between points x and y iff $d(x,y) \leq \varepsilon$. From $\mathcal{R}_\varepsilon$, the Vietoris-Rips complex $\mathcal{V}_\varepsilon$ [Vie27], a special simplicial complex, is obtained. $\mathcal{V}_\varepsilon$ contains a k-simplex iff $\mathcal{R}_\varepsilon$ contains all of its edges (see Fig. 2). It would now be possible to calculate the Betti numbers of the data set from $\mathcal{V}_\varepsilon$, but the numbers turn out to be very sensitive to noise and the scale parameter ε. Even slight variations for ε might result in a different set of homology groups being assigned to the data set. In their seminal paper, Edelsbrunner et al. [ELZ02] show how these instabilities can be amended: Their persistent homology algorithm calculates Betti numbers for a range of different values for ε. To this end, simplices in $\mathcal{V}_\varepsilon$ are partitioned into positive and negative simplices. A positive simplex creates a topological feature (i.e. a k-dimensional hole), while a negative simplex destroys one. Persistent homology thus speaks of creator and destroyer simplices. The weight of each simplex (for example, the value of ε for which it appears in $\mathcal{V}_\varepsilon$) is now used to assign each topological feature a value from $[0,\infty]$. This value is called the *persistence* of the hole. Holes with a large persistence are considered to be important, while a small persistence value often indicates (topological) noise. The *homology classes* of $\mathcal{V}_\varepsilon$, i.e. topological features that are not destroyed in $\mathcal{V}_\varepsilon$, are assigned a persistence value of ∞. There are multiple methods for assigning a weight to each simplex, for example to emphasize certain structures in a data set (see Singh et al. [SMC07] for an overview). In this paper, we use the standard *distance filtration*. Here, each 0-simplex is assigned a weight of 0 and each 1-simplex (u,v) a weight of $d(u,v)$, i.e. the distance between points u and v.

Recently, Zomorodian and Carlsson [ZC08] described an extension to the persistence algorithm. The new algorithm is capable of calculating a description of a topological feature along with the creator and destroyer simplices of persistent homology. The description is returned in the form of a *simplicial chain*, i.e. a formal sum of simplices. Due to the algebraic nature of the persistence algorithm, a simplicial chain in larger dimensions generally does not correspond to an intuitive description of a "hole". In fact, almost any closed "path" of simplices of the given dimension is an admissible simplicial chain for the persistence algorithm. Fig. 1a, for example, depicts two typical simplicial chains that are the result of the persistence algorithm. These non-intuitive chains inevitably occur because the persistence algorithm does not have any information about the geometry of the data

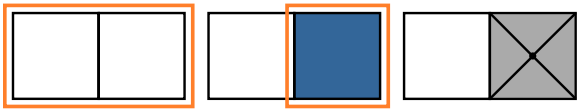


Figure 3: *Localizing a simplicial chain. The calculation of the geodesic ball uses geometrical information about the input data, while sealing the complex prepares the localization of the next chain.*

set [ZC08]. Before being of use for data analysis, simplicial chains hence need to be localized.

A short note on terminology: The modified persistence algorithm describes topological features by *simplicial non-bounding cycles*, which are a special case of simplicial chains. Because our method is capable of working with arbitrary simplicial chains, we still refer to these cycles as simplicial chains in this paper.

4. Methodology

In this section, we first explain the localization algorithm originally proposed by Chen and Freedman [CF10]. We then detail our improvements to the algorithm such that it can operate on weighted Rips graphs and obtain geometrically more meaningful localizations. Last, we describe our dimension-independent graph representation that we obtain from localized simplicial chains. This representation serves as a structural description of multivariate point clouds.

4.1. Localizing simplicial chains

Localization schemes aim to integrate some notion of geometry into the persistence calculation. This leads to simplicial chains that correspond to salient properties of a data set. Chen and Freedman [CF10] present a localization scheme that is based on geodesic graph distances. In contrast to randomized schemes such as the one described by Zomorodian and Carlsson [ZC08], their algorithm is deterministic and thus suitable for the analysis of multi-run data sets. Localization yields simplicial chains that are geometrically more meaningful: The distance of each simplicial chain to their corresponding hole in the data set is less than for the randomized localization—see Fig. 1b. Ideally, if a high-dimensional hole can be described by a set of data points, a localization algorithm should produce a simplicial chain whose distance to the set of points is as small as possible. Empirical evidence suggests that this is the case when the deterministic localization scheme is applied to real-world data sets. Given a dimension d for which all simplicial chains are to be localized, the algorithm originally consists of the following steps:

1. Calculate *discrete geodesic distances* on the Rips graph $\mathcal{R}_\varepsilon$ of the data set.
2. Extend the geodesic distance to all simplices in order to obtain a description of *discrete geodesic balls* of $\mathcal{V}_\varepsilon$.
3. Find the smallest geodesic ball that contains an essential simplicial chain of $\mathcal{V}_\varepsilon$, i.e. a homology class of $\mathcal{V}_\varepsilon$. This radius is taken to be the *size* of the simplicial chain. By using only simplices from the smallest geodesic ball, the chain is localized.
4. Seal the simplicial complex and restart the algorithm until all chains have been localized.

In the following paragraphs, we will give short descriptions for each of the steps outlined above. Fig. 3 briefly illustrates the different steps in the algorithm. We refer to Chen and Freedman [CF10] for more details.

Calculating discrete geodesic distances on $\mathcal{R}_\varepsilon$ Each vertex p of $\mathcal{R}_\varepsilon$ defines a discrete geodesic distance function f_p by setting $f_p(q) := \operatorname{dist}(p,q)$ for all vertices q, where $\operatorname{dist}(p,q)$ is the number of edges on the shortest path connecting vertices p and q. If p and q are in different connected components of $\mathcal{R}_\varepsilon$, then $f_p(q) := \infty$.

Extending geodesic distances to $\mathcal{V}_\varepsilon$ Given a fixed source vertex p, the distance function can be extended to any simplex σ of the simplicial complex by setting $f_p(\sigma) := \max_{q \in \operatorname{vert}(\sigma)} f_p(q)$, i.e. the maximum function value of the vertices of σ. This yields an ordering of $\mathcal{V}_\varepsilon$, and we define a discrete geodesic ball of radius r with a centre vertex p as the subcomplex of $\mathcal{V}_\varepsilon$ that contains all simplices σ with $f_p(\sigma) \leq r$.

Finding the smallest geodesic ball containing an essential simplicial chain The *size* of the simplicial chain is now set to be the smallest radius of the geodesic ball centred at some vertex p for which the persistence calculation returns an essential hole in dimension d. To this end, we first calculate the distance function f_p for each vertex p. We then use the values of f_p to sort $\mathcal{V}_\varepsilon$ and calculate its persistent homology. We take the minimum radius as the creation threshold of the first essential generator in dimension d. We obtain a localized simplicial chain by applying the extended persistence calculation by Zomorodian and Carlsson [ZC08] to the smallest geodesic ball.

Sealing $\mathcal{V}_\varepsilon$ After localizing the smallest chain in dimension d, we add a new vertex v to $\mathcal{V}_\varepsilon$. For each of the d-simplices of the localized chain, we add a $(d+1)$-simplex with v as an additional vertex to $\mathcal{V}_\varepsilon$. Furthermore, we add all faces of these new simplices. This augmentation "closes" the d-dimensional hole whose chain we localized. We can now rerun the algorithm until all other chains in dimension d have been localized.

This localization scheme is applicable in all dimensions, but has an output-sensitive computational complexity of approximately $\mathcal{O}(b_d n^4)$, where b_d is the Betti number in dimension d and n is the number of simplices of $\mathcal{V}_\varepsilon$. In a follow-up publication, Chen and Freedman [CF11] show that localizing simplicial chains to optimality is NP-hard in general.

In its original description, the algorithm assumes that $\mathcal{R}_\varepsilon$ uses unit weights. For the analysis of multivariate point

clouds, we prefer using the distances between data points as edge weights in the neighbourhood graph. Keeping these distances results in a more precise localization than the original algorithm. We observed that for most data sets, the Euclidean distance is a reasonable default choice. If available, domain-specific distance functions such as the Pearson distance or the Jensen-Shannon divergence can be used as well. The increased precision of our improved localization algorithm requires a brute-force search over all vertices of $\mathcal{R}_\varepsilon$ in order to determine the smallest geodesic ball. While Chen and Freedman [CF10] outline an optimized algorithm, it is only applicable for graphs with unit weights. To enhance performance, we instead employ *parallelization* in the search for the smallest geodesic ball: For smaller simplicial complexes ($< 10^6$ simplices), the geodesic distance calculations for multiple source vertices can be performed in parallel on the CPU. This can increase performance by a factor of 1.5–2 (given 4 cores), depending on whether the persistence calculation or the number of vertices is the bottleneck. For a further speed-up of the calculations, we employ a strategy that resembles *branch-and-bound*: When calculating f_p for some vertex p, we may skip the calculation if all updated weights of $\mathcal{V}_\varepsilon$ are larger than the currently determined minimum radius. By randomizing vertex traversal, our algorithm has a chance of finding the minimum radius early, thereby saving needless calculations.

4.2. Simplicial chain graphs

The localized simplicial chains serve to describe both the topological and the geometrical structure of the data set. We thus want to visualize them in order to obtain structural information about the multivariate input data set. The common challenge for all visualizations is that even low-dimensional simplicial chains cannot readily be drawn for higher-dimensional data sets, as each simplex of the chain corresponds to a subset of high-dimensional points.

Instead of trying to use the positional information of the data points, we therefore decided to focus on their relations. To this end, we display the simplicial chains as a graph, using the deterministic *neato* [EGK*04] layout algorithm to obtain reproducible graph layouts. We refer the reader to the supplementary materials for a more detailed description of the complete layout process. Our first attempt used simplices as the graph nodes and connected them whenever they were part of the same simplicial chain. Figure 4 depicts a typical output (see Figure 5 for both the compressed and uncompressed improved graph for the same input data). Although the graph serves to give an overview of all available simplicial chains, we ultimately considered it to be too abstract: (i) The relation between a simplex shown in the graph and the corresponding subset of the input data is not apparent. (ii) Different chains cannot be distinguished if they share the same simplex. (iii) Using nodes to encode information about each simplicial chain is difficult because a node can be shared by multiple chains.

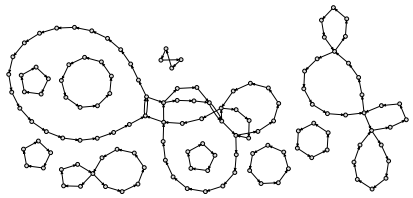


Figure 4: *Graph showing the connections between simplices that belong to the same simplicial chain. See Sec. 4.2 for more details.*

To highlight the *structural description* of a multivariate point cloud through simplicial chains, we thus integrated information about both the chains and the data points. To this end, we first decompose each simplicial chain into the data points it contains. This is done by inserting each vertex of each simplex of the chain into a set. The set of indices is then associated with the corresponding high-dimensional coordinates. From this *coordinate decomposition* we, create a graph with two types of nodes: (i) *Chain nodes*, which correspond to a simplicial chain in the data set. (ii) *Data nodes*, which correspond to a data point in the input point cloud. Only data points that occur in a simplicial chain, i.e. that are part of topological feature, will be represented as data nodes. We create an edge between a chain node and a data node whenever the corresponding simplicial chain contains the corresponding data point. The valency of each data node thus shows the number of simplicial chains it is a part of. Note that there are no edges between chain nodes. We use only the data nodes to show relations in the data set.

To maintain a sense of distances between different substructures in the graph, we calculate the medoid of the coordinate decomposition of each simplicial chain We then adjust the graph layout algorithm to place chain nodes such that their medoid distances are respected. This ensures that chains that are close in the input space will be placed in close proximity in the graph layout.

Creating a graph with an embedded hierarchy of node types proved to be advantageous for encoding different topological attributes. We first colour-coded all chain nodes by their localized radius, using a diverging red-green colour map (see left part of Figure 5 for a typical example). The radius of a simplicial chain determines the amount of space a topological feature encompasses in the input data set. A chain with a large radius thus describes a structure that is spread out over a large part of the data set. Chains with a small radius, on the other hand, describe structures that are more local. The colour-coding can be changed to represent different attributes, such as the diameter or the (un-)weighted volume [CF11]. Analysing these attributes is useful for the comparison of different data sets.

While the colour-coded chain graph already serves to highlight structures in a point cloud, we found that the graph layout becomes increasingly cluttered for larger point clouds. We thus *compressed* the chain graph: We remove all

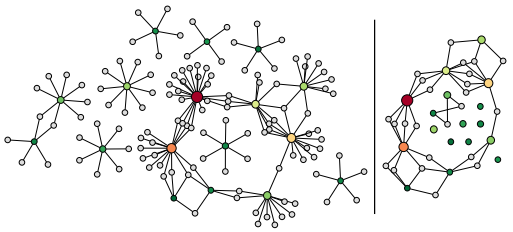


Figure 5: *Uncompressed (left) and compressed (right) simplicial chain graph for a subset of the PRIM 7 data set. The compressed graph is less cluttered and simplifies analysis.*

data nodes (and their associated edges) that have a valency of 1. Hence, only data nodes that are part of multiple simplicial chains remain in the chain graph. Since we lose the information about the number of data points in a simplicial chain, we scale the chain nodes accordingly. We found that the compression effectively removes clutter and refer to the resulting graph as the *simplicial chain graph* of the input data set—see the right part of Figure 5 for an example.

The simplicial chain graph consequently describes a part of the topological structure of a data set. We identified several key properties: (i) The number of connected components and the distances between different chain nodes in the chain graph are correlated with the homogeneity of a data set. A large number of simplicial chains sharing no data points indicates that the data set contains multiple inhomogeneous regions. For example, a data set containing two classes of measurements, one lying on a hypertorus, the other on a hypersphere, will show up as multiple connected components in the chain graph. (ii) The size distributions of the data nodes encode the size of substructures in a data set. This information can be used when comparing multiple data sets. If two data sets are created from the same experiment, for example, their simplicial chain size distributions should be approximately equal. A large difference may indicate a sampling error (i.e. insufficient measurements in one data set). (iii) The radii of simplicial chains highlight the sizes of the inhomogeneous regions in the data set. A single simplicial chain with a large radius (colour-coded in red to exploit pre-attentive processing) implies that data points bound a large empty region in the data set. In a data set containing experimental measurement data, for example, this highlights missing values. These missing values may be caused by erroneous measuring equipment.

The simplicial chain graph is not limited to visualizing chains of a single dimension. In Section 5, we analyse various data sets using a chain graph with both 1-dimensional and 2-dimensional chains. To calculate the chain graph, we only require a list of dimensions for localization and a value for ε, which is used to control the persistent homology calculation. Depending on ε, different structures can be emphasized in the data set, which in turn will cause the simplicial chain graph to change. Figure 6 depicts changes in the chain

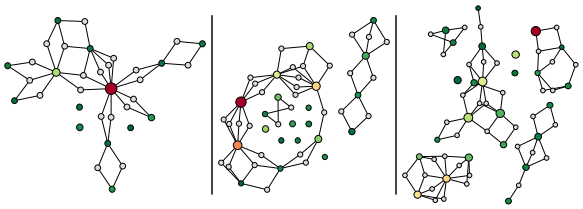


Figure 6: *Changing the graph structure by varying ε serves to emphasize structures of different sizes in a data set.*

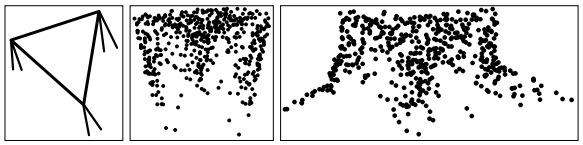


Figure 7: *Data model and projections (MDS and Isomap) of the PRIM 7 data set. Both projections only partially capture the shape of the data set.*

graph caused by large variances of ε. For small perturbations of the ε parameter, however, the *stability theorem* of Cohen-Steiner et al. [CSEH07] implies that the simplicial chains will remain stable. In our application, users can refer to persistence visualizations of the data set in order to decide whether the selected value for ε is suitable. Note that the simplicial chain graph will only show the homology classes of $\mathcal{V}_\varepsilon$, i.e. topological features with a persistence value of $+\infty$. The reason for this restriction is that the localization of arbitrary simplicial chains is still an open problem in topological data analysis. For now, the simplicial chain graph thus only focuses on the most prominent topological features of a data set. Although the chain graph can easily accommodate non-essential simplicial chains, they are not guaranteed to describe salient structures in the data set—this necessitates further research.

5. Results

In the following, we apply our method to several high-dimensional data sets from different application domains. For setting the ε parameter, we use two heuristics described by Rieck et al. [RML12]. The first one uses dendrograms and single-linkage clustering to support the user in selecting a threshold. The second one exploits distance estimates in multivariate point clouds and is suitable for for unsupervised, semiautomated analysis. As the selected data sets only exhibit few topological attributes in higher dimensions, we focus on 1- and 2-dimensional simplicial chains.

5.1. PRIM 7

The PRIM 7 data set consists of measurements from a classical experiment in high-energy particle physics. Each point in this data set corresponds to a reaction in which four different particles are created. The reaction products are characterized by 7 independent variables $X_1, \ldots, X_7$—see Friedman

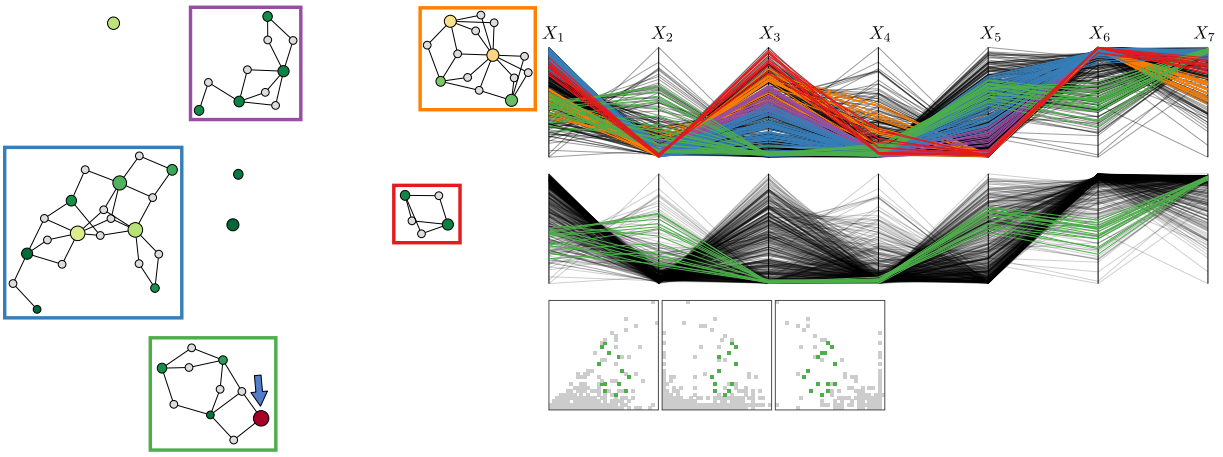


Figure 8: *Simplicial chain graph (left) and linked visualizations (right) of the PRIM 7 data set. Each connected component in the graph describes a different group of measurements in the data set. See Sec. 5.1 for a detailed discussion.*

and Tukey [FT74] for more details. The data set is interesting because it exhibits a simple low-dimensional structure: The data points are clumped near a 2-dimensional triangle, with two linear "strands" extending from each vertex of the triangle (see Fig. 7, left). The triangle is known to be characterized by large variances in variables X_3 and X_5 [CS07]. The PRIM 7 data set is known to be challenging to analyse using common projection methods or clustering algorithms [CS07]. Fig. 7 shows two projections of the data set obtained by multidimensional scaling (MDS) and Isomap. Note that both methods capture only parts of the data model. They show the central structure and some linear strands at its boundary, but the complete model can only be constructed using repeated guided projections. We thus used this data set to validate the correctness of our algorithm and show its advantages.

Persistent homology does not identify any prominent generators in dimensions ≥ 3, thereby confirming that the data set has a low-dimensional structure. The simplicial chain graph of the data set is depicted by Fig. 8, left. The graph has many connected components. Their distributed layout indicates that the corresponding measurements are scattered over the parameter space—otherwise, we would obtain a more densely-connected structure. A parallel coordinate plot (PCP), Fig. 8, top right, of each connected component confirms this: The variability in variables X_3 and X_5 shows that each component describes a different set of particle interactions within the centre of the parameter space. The green structure behaves exceptionally—see the analysis below. The other structures form well-separated strands in the PCP. Again, the strands are best visible in variables X_3 and X_5. We also observe that the chain radii (indicated by the node colours) do not exhibit much variance, apart from some exceptions. This indicates that all topological features corresponding to the chains have approximately the same size and thus "behave" roughly the same in terms of the particle interactions they describe. More precisely, chains of the same radius tend to bound similar holes in the data set. Each chain hence bounds a region with little or no measurements in the data set. PRIM 7 indeed does not contain any particle interactions within these regions, likely because they do not occur in nature. Information about such sparse regions is valuable for domain experts: For one thing, sparse regions may indicate physically impossible values or an experimental oversight. Moreover, the chain graph can aid in detecting anomalous structures for repeated runs of the same experiment. When data is assumed to be sampled from a high-dimensional manifold, it is reasonable to expect that its topological shape exhibits little variation over multiple runs.

We finish our analysis by taking a look at the largest simplicial chain (see highlighted node in Fig. 8, left), whose radius is strikingly different than the other radii. The size of the node indicates that the chain contains many data points, while the colour of the node indicates that its localization radius is large. In combination, these attributes suggest that the chain spans a large empty region in the parameter space—which may signify a salient feature of the data set. The PCP (Fig. 8, middle right) shows that the chain is not part of the centre of the data set as its variability in X_6 is too large. The scatterplot matrix (see Fig. 8, bottom right for some selected projections) shows that the chain describes part of the boundary of the PRIM 7 triangle. The remaining chains in this connected component of the chain graph behave similarly and exhibit variability in variables X_2 and X_6. These chains thus describe measurements that describe different particle interactions than in the rest of the data set.

Finally, we note that the isolated components in the chain graph also describe holes in the centre of the data set. Due to the sampling, they are not connected to other connected components. Their missing connections are compensated by

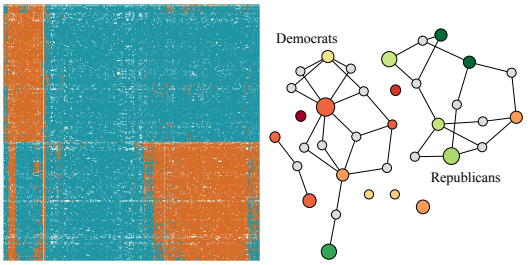


Figure 9: *Heat map and simplicial chain graph for 2008 roll call data. See Section 5.2 for a more detailed discussion.*

the graph layout algorithm, which places these chains in close proximity to the larger connected components.

5.2. Voting data

The votes in the U.S. House of Representatives are cast in the form of "Yeas" and "Nays", depending on whether a certain issue or bill is approved or disapproved of. We used http://clerk.house.gov to obtain publicly available data for previous sessions of Congress, spanning a period from 1990 to 2011. These data sets are a valuable resource for political scientists, who use it to gain insight into the voting culture of the United States. We converted the roll call results of each year into high-dimensional point clouds. For this, we assigned each representative a vector of +1, -1, 0, depending on whether the representative gave an approving vote, an opposing vote, or abstained from voting. Depending on the number of roll calls for each session of Congress, the resulting point clouds contain about 420 points with 600–900 dimensions. Statistical analysis of each data set shows that the votes have a clear separation, which makes it possible to discern the party affiliation using e.g. *principal component analysis*. Our topological analysis, in contrast, focuses on the shape of the data set. We are especially interested in substructures that define the data set in a sense, i.e. substructures without which the data set would change its topological shape.

Topological analysis uncovers a simple structure for the space of representatives: We do not find any non-trivial topological activity in dimensions ≥ 2, which suggests that the data set has a low intrinsic dimensionality. This confirms established results by Poole and Rosenthal [PR07]. We continued our analysis with voting records for 2008 and 2009 as these data sets contain the largest number of votes of the whole period. Since topological analysis requires distances between data points, the *curse of dimensionality* makes it nearly impossible to define meaningful differences between, for example, two vectors of length ≥ 600. As a preprocessing step for further analysis, we thus applied *non-metric multidimensional scaling* to embed the data set in a 25-dimensional space. While the intrinsic dimension of the data set is arguably smaller, we chose a dimension of 25 because the algorithm converged faster and reported smaller stress values. Rabinowitz [Rab75] showed that non-metric MDS is a suitable tool for the analysis of political science data. Using statistical tools, we verified that the embedded data set still exhibits the same party affiliation structure.

We then proceeded to use sorted *heat maps* (see Fig. 9 and 10) to show the general structure of the data set. The heat map uses orange to indicate opposing votes, turquoise to indicate approval, and white for abstention. Votes of each individual representative are shown in the rows of the heat map. The differently-coloured blocks indicate a clear distinction between the two parties. Some issues or bills are approved by both parties equally, apart from some dissenters. The simplicial chain graph for the 2008 data shows two larger connected components, one for the Democratic Party, the other one for the Republican Party. Each chain comprises a set of representatives with similar voting behaviour. By inspecting the representatives and their "distances" to the party line (which we considered to be the majority vote for the particular issue), we found that each chain contains those representatives whose voting behaviour differs the *most* from the party line. Hence, the dissenting representatives described by each chain constitute the boundary of the party structure. The smaller connected components describe smaller subsets of the representatives (only about 2–3 data points) whose voting behaviour is comparable, but does not coincide with the dissenters of their respective parties. The data for 2009 exhibits a similar behaviour. Since there are slightly more votes in the data set, the decomposition into exactly 2 large structures is seen more clearly. The larger connected component corresponds to the Democratic Party, which held the majority of Congressional seats in 2009. A better detection of the party boundary, resulting in more representatives *disagreeing* with the party line, is thus to be expected.

In conclusion, the chain graph shows that the voting data sets have a clear topological shape. This shape is defined by all representatives whose votes disagree with their respective parties. Each connected component highlights the voting behaviour of a subset of the representatives. Each chain describes a particular set of representatives with similar voting behaviour. These smaller subsets of representatives shed light on the political climate (e.g. voting alliances). In addition, these substructures make up the boundary and the shape of the votes of each party. A larger number of chain graphs could be employed to show the evolution of voting behaviour of the U.S. House of Representatives.

5.3. Tropical Atmosphere Ocean (TAO) array data

The El Niño phenomenon describes a strong climate pattern that is defined by prolonged anomalies of sea surface temperatures in the Pacific Ocean. El Niño typically occurs at irregular intervals from 3–7 years and may last from 9 months to 2 years. The mechanisms causing this phenomenon are still under investigation. We obtained data from the *Tropical Atmosphere Ocean* (TAO) array. The array consists of approximately 70 buoy moorings in the Tropical Pacific Ocean.

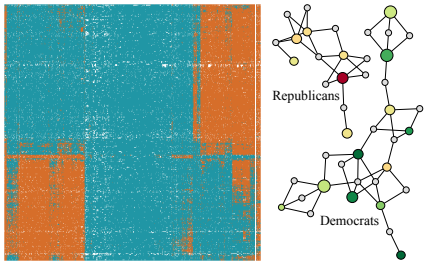


Figure 10: *Heat map and simplicial chain graph for 2009 roll call data. See Section 5.2 for a more detailed discussion.*

At regular intervals, each buoy measures the zonal wind velocity, meridional wind velocity, humidity, air temperature, and sea surface temperature. All heat map visualizations will show the attribute values in this order. We obtained combined measurements for a period from 1980–1998. Several occurrences of El Niño within this time period are known, namely 1982–1983, 1986, 1991–92, 1993, 1994–1995, and 1997–1998.

The data set has 5 dimensions and around 180000 data points (all measurements for the 18-year period). Missing values make analysing the data set very challenging: Because of technical errors and different buoy configurations, not all attribute measurements are available for the whole recording period. Fig. 11 shows a comparison of several data sets for two time periods (the remaining time periods in the data set contain similar patterns). During 1996, no El Niño occurred, while 1997–1998 saw the largest El Niño event on record. For layout reasons, we did not incorporate any distance information in the graphs.

The chain graph for the 1993 period shows a single large connected component (a), a small connected component (b), and some isolated nodes (c). Fig. 11, right, depicts heat maps for the chains in the connected components. The isolated nodes (c) turn out to correspond to measurements with missing humidity values and low values for both zonal and meridional winds. Component (a) contains chains that exhibit a distinct profile: Medium-low zonal and meridional wind velocities, high values for humidity, air temperature, and sea surface temperature. In contrast, component (b) contains chains with slightly higher zonal wind velocities, but lower meridional wind velocities. This illustrates that the space of measurements contains several topologically-distinct regions, which are captured correctly by the simplicial chain graphs. These regions cannot be captured properly by distance-based methods, as the measurements do not form significant clusters.

In contrast, the simplicial chain graph of the 1994–1995 data shows a different structure, with a new larger connected component (f). Components (d) and (e) resemble the ones for the 1993 data set. They describe a similar phenomenon: Chains with medium-low zonal and meridional wind velocities and high temperatures. Note that the measurements contain missing humidity values (shown in black) but the chain graph correctly reflects that their topological structure is similar to the 1993 measurements. Component (f) contains all measurements with extremal temperatures—these are caused by the El Niño phenomenon. Since this connected component is not present in the 1993 chain graph, its appearance indicates that the topological shape of the 1994–1995 data undergoes a drastic change. With the benefit of hindsight, we know that this change was brought on by the El Niño phenomenon.

The same anomalous changes in the measurements are reflected in the chain graphs for 1996 and 1997–1998, as well. Especially for the last data set, there are many missing values in all attributes, which results in a more fragmented space. The chains still exhibit similar characteristics as in the previous data sets. The 1997–1998 shows the limitations of the simplicial chain graph: The large amount of missing values makes finding the "real" topology of the data set very difficult. This, in turn, makes the output of the simplicial chain graph less stable. In the context of the other data sets, the chain graph output suggests that the 1997–1998 data set is more anomalous than the remaining data sets, exhibiting a different topological structure. Clearly, any such assumption obtained from the chain graph will have to be verified using a variety of different visualization techniques—the chain graph serves as one indicator here.

6. Conclusion & future work

We introduced a novel visualization method that combines topological and geometrical information to support structural analysis of multivariate point clouds. Our algorithm uses persistent homology to obtain a multi-scale description of topological attributes in a point cloud. These attributes are then assigned a simplicial chain, i.e. a set of k-dimensional simplices describing a k-dimensional hole in the data set. Since the simplicial chains initially do not contain geometrical information, we use distance information from the data set to obtain geometrically concise simplicial chains. We now combine both data points and simplicial chains from multiple dimensions into a single graph, the *simplicial chain graph*. After compressing the chain graph to gain a more succinct display, we use colours and scaling to encode information about topological attributes. We then proceeded to explain how certain patterns in the chain graph point out structures in multivariate point clouds. By analysing data sets from various disciplines, we demonstrated how the chain graph supports multivariate data analysis. For each data set, we established how the graph can be used to highlight anomalies and facilitate the comparison of time-varying point clouds.

Future research might focus on aspects of simplicial chain localization. First, the performance of the algorithm could be improved by following a strategy proposed by Lewis and Zomorodian [LZ13]. Second, localization should be applicable to all topological features of $\mathcal{V}_\varepsilon$ and not be restricted to

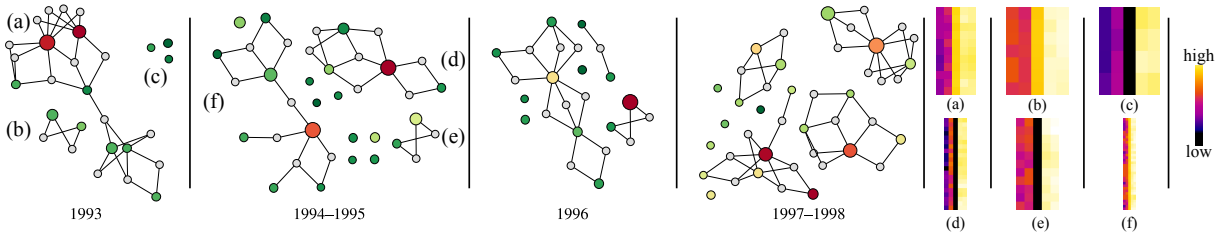


Figure 11: *Simplicial chain graphs showing measurements for periods without El Niño events (1993, 1996) and anomalous periods with larger El Niño events (1994–1995, 1997–1998) in the TAO data set. For the heat maps on the right, we adjusted the colour map to the global extremal values of each attribute in the complete data set to simplify cross-comparisons. See Section 5.3 for a detailed discussion.*

its homology classes. Another aspect for potential advancements involves the size measure for simplicial chains. Currently, there is no measure that integrates more geometrical information while still being computable in polynomial time.

References

[AC09] ADAMS H., CARLSSON G.: On the non-linear statistics of range image patches. *SIAM J. Imaging Sci. 2*, 1 (2009), 110–117. 29

[Car09] CARLSSON G.: Topology and data. *Bull. Amer. Math. Soc. 46* (2009), 255–308. 29

[CF10] CHEN C., FREEDMAN D.: Measuring and computing natural generators for homology groups. *Comput. Geom. 43*, 2 (2010), 169–181. 31, 32

[CF11] CHEN C., FREEDMAN D.: Hardness results for homology localization. *Discrete Comput. Geom. 45*, 3 (2011), 425–448. 31, 32

[CS07] COOK D., SWAYNE D. F.: *Interactive and dynamic graphics for data analysis*. Springer, 2007. 34

[CSA02] CARR H., SNOEYINK J., AXEN U.: Computing contour trees in all dimensions. *Comput. Geom. Theory Appl. 24* (2002), 75–94. 29

[CSEH07] COHEN-STEINER D., EDELSBRUNNER H., HARER J.: Stability of persistence diagrams. *Discrete Comput. Geom. 37*, 1 (2007), 103–120. 33

[CSvdP10] CARR H., SNOEYINK J., VAN DE PANNE M.: Flexible isosurfaces: Simplifying and displaying scalar topology using the contour tree. *Comput. Geom. Theory Appl. 43*, 1 (2010), 42–58. 29

[DCK*12] DUKE D., CARR H., KNOLL A., SCHUNCK N., NAM H. A., STASZCZAK A.: Visualizing nuclear scission through a multifield extension of topological analysis. *IEEE Trans. Vis. Comput. Graphics 18*, 12 (2012), 2033–2040. 29

[DLSCS08] DEY T. K., LI K., SUN J., COHEN-STEINER D.: Computing geometry-aware handle and tunnel loops in 3d models. *ACM Trans. Graphics 27* (2008), 1–9. 29

[EGK*04] ELLSON J., GANSNER E. R., KOUTSOFIOS E., NORTH S. C., WOODHULL G.: Graphviz and dynagraph – static and dynamic graph drawing tools. In *Graph Drawing Software*. Springer, 2004. 32

[EH10] EDELSBRUNNER H., HARER J. L.: *Computational topology*. American Mathematical Society, 2010. 29

[ELZ02] EDELSBRUNNER H., LETSCHER D., ZOMORODIAN A.: Topological persistence and simplification. *Discrete Comput. Geom. 28*, 4 (2002), 511–533. 28, 29, 30

[Eri12] ERICKSON J.: Combinatorial optimization of cycles and bases. In *P. Symp. Appl. Math.* (2012), vol. 70, pp. 195–228. 29

[EW05] ERICKSON J., WHITTLESEY K.: Greedy optimal homotopy and homology generators. In *Proc. ACM-SIAM Symp. Discrete Algorithms* (2005). 29

[FT74] FRIEDMAN J. H., TUKEY J. W.: A projection pursuit algorithm for exploratory data analysis. *IEEE Trans. Comput. 23* (1974), 881–890. 29, 34

[Ghr08] GHRIST R.: Barcodes: The persistent topology of data. *Bull. Amer. Math. Soc. 45* (2008), 61–75. 28, 29

[Hat02] HATCHER A.: *Algebraic topology*. Cambridge University Press, 2002. 29

[LZ13] LEWIS R. H., ZOMORODIAN A.: Multicore homology. *Preprint* (2013). 36

[Mar09] MARSLAND S.: *Machine learning - An algorithmic perspective*. Chapmann & Hall / CRC Press, 2009. 29

[Mun84] MUNKRES J. R.: *Elements of algebraic topology*. Addison-Wesley Publishing Company, Inc., 1984. 29, 30

[PR07] POOLE K. T., ROSENTHAL H. L.: *Ideology and Congress*. Transaction Publishers, 2007. 35

[Rab75] RABINOWITZ G. B.: An introduction to nonmetric multidimensional scaling. *Am. J. Polit. Sci. 19*, 2 (May 1975), 343–390. 35

[RML12] RIECK B., MARA H., LEITTE H.: Multivariate data analysis using persistence-based filtering and topological signatures. *IEEE Trans. Vis. Comput. Graphics 18*, 12 (2012), 2382–2391. 28, 29, 33

[SMC07] SINGH G., MÉMOLI F., CARLSSON G.: Topological methods for the analysis of high dimensional data sets and 3d object recognition. In *Proc. ACM Symp. Point-based Graphics* (2007), pp. 91–100. 29, 30

[TdSL00] TENENBAUM J. B., DE SILVA V., LANGFORD J. C.: A global geometric framework for nonlinear dimensionality reduction. *Science 290*, 5500 (2000), 2319–2323. 29

[Vie27] VIETORIS L.: Über den höheren Zusammenhang kompakter Räume und eine Klasse von zusammenhangstreuen Abbildungen. *Math. Ann. 97* (1927). 30

[ZC08] ZOMORODIAN A., CARLSSON G.: Localized homology. *Comput. Geom. 41*, 3 (2008), 126–148. 30, 31