TOPOLOGICAL SIMILARITY MEASURES BETWEEN MULTI-FIELD DATA

Tripti Agarwal

Master of Science by Research Thesis

June 2020

International Institute of Information Technology, Bangalore

TOPOLOGICAL SIMILARITY MEASURES BETWEEN MULTI-FIELD DATA

Submitted to International Institute of Information Technology,
Bangalore
in Partial Fulfillment of
the Requirements for the Award of
Master of Science by Research

by

Tripti Agarwal
MS2017009

International Institute of Information Technology, Bangalore
June 2020

Dedicated to my family

Thesis Certificate

This is to certify that the thesis titled **TOPOLOGICAL SIMILARITY MEASURES BETWEEN MULTI-FIELD DATA** submitted to the International Institute of Information Technology, Bangalore, for the award of the degree of **Master of Science by Research** is a bona fide record of the research work done by **Tripti Agarwal**, **MS2017009**, under my supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma. The thesis conforms to plagiarism guidelines and compliance as per UGC recommendations.

Prof. Amit Chattopadhyay

Bangalore,
The 11th of June, 2020.

TOPOLOGICAL SIMILARITY MEASURES BETWEEN MULTI-FIELD DATA

Abstract

A wide range of data that appear in scientific experiments and simulations are mostly scalar or multi-field (alternatively, multivariate - consisting of multiple scalar fields) in nature. Topological feature extraction of such data aims to reveal important properties useful to the domain scientists. For scalar data, feature extraction has been studied extensively by proposing different topological similarity (or dis-similarity) measures between two datasets and proven useful in extracting important topological features. However, for the case of multi-field data developing such topological similarity measures is still in its infancy. In the current thesis, we address the problem of computing topological similarity or dis-similarity between two multi-field data. Towards this, we propose two approaches for measuring topological similarity between multi-field data based on their Reeb spaces.

A Reeb space captures the fiber-topology of a multi-field, each point of the Reeb space representing a connected component of a fiber. Usually, the Reeb space structure is complex and finding a topological distance between two multi-fields in terms of their Reeb spaces is a non-trivial problem. Therefore, in our first approach, we project the Reeb space onto the range of the multi-field to obtain a fiber-component distribution and then propose different distance metrics between two such distributions. Given a time-varying multi-field data, the method computes a metric plot for each pair of distributions at consecutive time stamps to understand the topological changes in the data over time.

In the second approach, we propose a topological similarity measure between two multi-field data based on their multi-resolution Reeb Spaces. Our method generalizes a previous similarity measure between two multi-resolution Reeb graphs for scalar data. The method comprises of two main steps: (i) computing a multi-dimensional Reeb Space at different resolutions and (ii) defining a similarity measure between two such multi-resolution Reeb spaces.

We apply our methods in feature extraction of different multivariate real and synthetic data. The effectiveness of the proposed methods is shown by its ability to capture important topological features that are not always possible to detect using the component scalar fields.

Acknowledgements

I would like to extend my gratitude towards my thesis supervisor Professor Amit Chattopadhyay for his belief in me and for always providing a helping hand when needed. I feel fortunate enough to get a chance to work and learn from him. I am indebted to him for his constant supervision, support and motivation. I would also like to thank Professor Vijay Natarajan, at CSA of IISc Bangalore, for his contribution of time and ideas to make my work productive. I wish to express my sincere gratitude to the Science and Engineering Research Board, India (SERB/CRG/2018/000702) and IIIT-Bangalore for generous funding support. My thanks to my friends Saloni Arora, Shruti Aggarwal and Ankita Bisht for supporting me during the thesis. I would also like to thank Mrs. Reddy Rani Vangimalla for providing such a healthy working environment and sharing her invaluable experience. Discussing with you was always fun and stress relieving. I would also like to thank my lab mate R. Yashwanth for helping me and working with me. Last but not the least, I would like to thank my family for maintaining their belief and faith in me without which completing this journey would not be possible.

List of Publications

1. Agarwal, T., Chattopadhyay, A. and Natarajan, V., 2019. Topological Feature Search in Time-Varying Multi-field Data. TopoInVis 2019, Nyköping, Sweden.
2. Agarwal, T., Yashwanth, R. and Chattopadhyay, A., 2020. A Topological Similarity Measure of Multi-field Data using Multi-Resolution Reeb Spaces. Tech Report, Under review.

Contents

List of Figures

CHAPTER 1

INTRODUCTION

Most of the scientific data appear in scientific experiments and simulations are scalar or multi-field (alternatively, multivariate - consists of multiple scalar fields) in nature. For example, the simulations in fluid dynamics, combustion, molecular physics and other scientific experiments generate various scalar fields such as pressure, temperature, kinetic energy etc. Researchers try to comprehend diverse physical phenomena by observing features and properties from the interrelationships between the fields. It has been observed that such multi-field or multivariate information could uncover various features that is difficult to observe by using only scalar fields. In topological data analysis we develop various tools and methods for extracting and visualizing important features in such data. Topology-based techniques are very proven to be extremely effective to uncover significant properties which are useful for the domain scientists.

1.1 Motivation

During the past two decades, topological analysis of shapes and data was generally determined by scalar topology, using contour tree, Reeb graph, Morse-Smale complex and their variants. Such techniques have been extended for time-varying scalar field data by characterizing various topology-based similarity measures between two scalar fields. Although work has been done extensively to extract topological features from the scalar field, single fields are still not sufficient to capture all such features [14]. This gives the motivation to extract topological features between multiple scalar fields or simply multi-field. To extract important features in multi-field data, we need to study the similarity between two multi-fields with which we can capture the topological features between these multi-fields and understand the underlying phenomenon existing between them. This study aims to find the similarity between multi-field, by applying different similarity. Comparison of these multi-field data will help in understanding and visualizing the interaction among different scalar field data and their relationship in an efficient manner.

1.2 Background and Related Work

1.2.1 Scalar Topology

When a function uniquely maps points from a two or three dimensional space to real values, then these functions are called scalar functions. The data set obtained from domain scientists are in form of function values measured at discrete points in the domain using procedural methods and requires analytical description of it. The scalar fields are like the temperature, density, pressure or any other physical measure. The domain is triangulated grid and these values are defined on the vertices of the dataset. These triangulated grid are known as *simiplicial complex*.

Let us consider a scalar function $f : M \rightarrow R$ on a d-dimensional manifold M. Then we have the following terminologies defined for this function.

Level Sets. Given a real value $c \in R$, the inverse image $f^{-1}(c)$ of the map f, is called a level set of f at c. A level set can simply be defined as the set of points $p \in M$ where $f(p) = c$. The topological features of a scalar field can be tracked using level sets as shown in Figure FC1.1. A sublevel set is a set of points in M where $f(x) \leq c$ and a superlevel set is a set of point in M where $f(x) \geq c$.

Contours. The connected components of a level set are called contours. The contours of the function f is shown in Figure FC1.1

Morse Theory. In differential topology, Morse theory helps in understanding the topology of a 2-manifold M, by studying differentiable or smooth functions on that manifold [30]. A point $x \in M$ is *critical point* (or *singular point*) if the gradient of f vanishes or $\nabla f(x) = 0$ (obtained by choosing a suitable local coordinates), otherwise x is a regular point. The function value $f(x) = c \in R$ corresponding to a critical point is called a critical value and others are called regular values. Again a level set corresponding to a regular value is called a regular level set, otherwise it is a critical level set. A critical point is *non-degenerate* if its *Hessian* matrix $H_f(x)$ is *non-singular*, otherwise the critical point is *degenerate*. The function $f : M \to R$ is a Morse function, if (i) all its critical points are *non-degenerate* and (ii) have distinct function values. Morse lemma [31] states that near a non-degenerate critical point $x \in M$ it is possible to express the function f as $f(u, v) = f(x) \pm u^2 \pm v^2$ using local coordinates u and v. The number of minus signs in the expression is called the *index* of the critical point. Thus a Morse function on a 2-manifold has three types of critical points - minimum (index 0), saddle (index 1) and maximum (index 2). Morse theory relates the topology of the sublevel set of a Morse function with the local geometry of the critical points. If we consider the change in topology of the sublevel set of a Morse function while function value gradually increases, then Morse theory says while passing a critical point of index i, the topology of the new sublevel set can be obtained by attaching an i-cell with the old sublevel set and there is no topological change in the sublevel set while passing through a regular value.

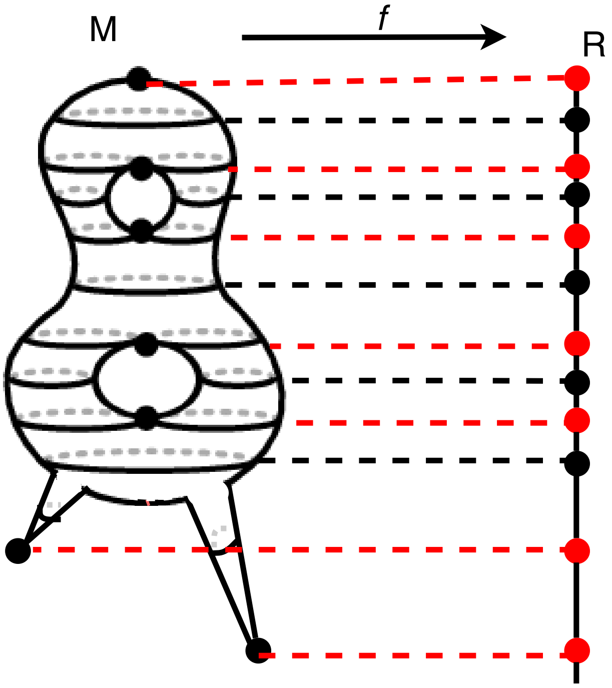


Figure FC1.1: Level sets and corresponding critical points (red) and regular points (black) in range R. The bold lines in manifold M represents the contours.

The connected components of the level sets can be extracted using various topological abstractions. Level set based abstraction is based on gradients and trees/graphs. When the domain is simply connected, i.e. it does not have a hole in it, we represent the abstraction using a tree and if there is a hole in the domain (not simply connected) we represent the domain using a graph.

Reeb Graph. A Reeb graph captures the level set topology corresponding to a scalar field f on a manifold M [16]. Each point of the Reeb graph corresponds to a contour. In particular, the nodes of the Reeb graph corresponds to the contours passing through the critical points of f. The edges connecting the nodes represent the contours which pass through the regular points of f. When the domain of the function is not simply-connected i.e. it does not have a hole in it, then the corresponding Reeb graph contains loops. The Figure FC1.2 shows a simple example of double torus with two legs and the Reeb graph of its height function.

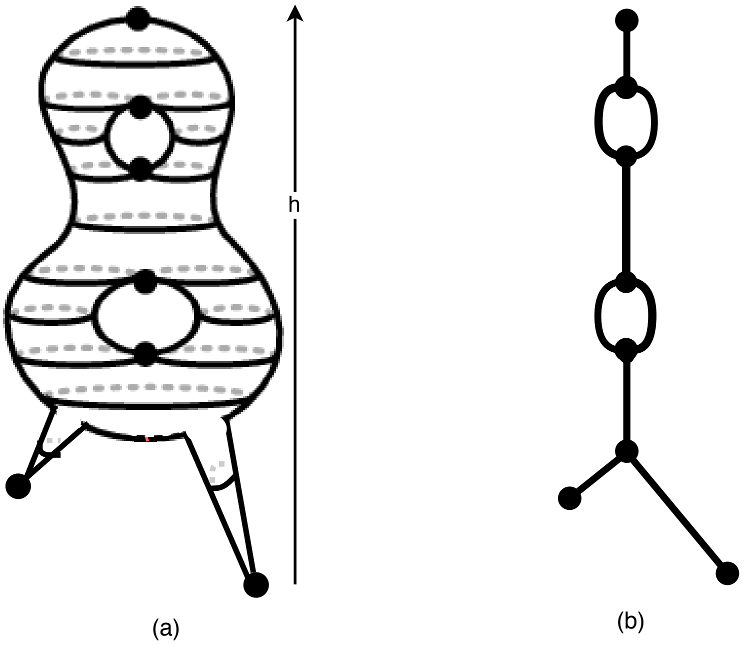
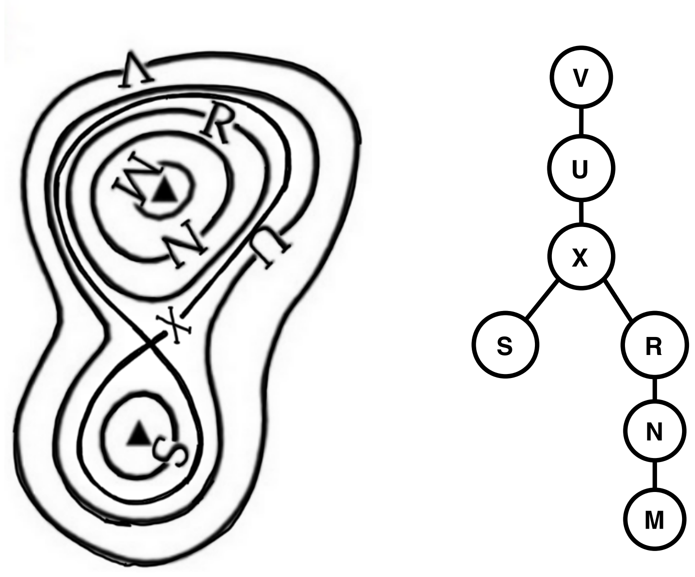


Figure FC1.2: (a) Height field defined on the double torus with two legs and its the level sets, (b) Corresponding Reeb Graph.

Contour Tree. When the domain of the function f is simply-connected, then the Reeb graph of the function has no loops and is called a contour tree [16]. A contour tree is widely used to visualize the topological features of different scalar data in volumetric domain. A contour tree is computed by the union of a join tree and a split tree [11]. A join tree captures the births and deaths of the components of the sublevel sets of scalar field f while changing the scalar values from its minimum to maximum. Similarly a split tree is obtained by computing the join tree of negative of f, i.e. of $-f$. Each point of the domain belongs to a contour at a particular level. All these contours form a contour map. The Figure FC1.3 shows a simple contour map and its corresponding contour tree.



(a) (b)

Figure FC1.3: (a) A contour map (b) Corresponding contour tree.

Multi-Resolution Reeb Graph. The basic idea of the Multiresolution Reeb graph (MRG) is to develop a series of Reeb graphs to capture the topological information of a scalar field (or its underlying domain) at different resolutions [23]. It is obtained by subdividing the data-domain into smaller regions based on their function values (as shown in Figure. FC1.4). A Reeb graph for a finer level is constructed by subdividing each region. For simplicity, this is done using a binary subdivision.

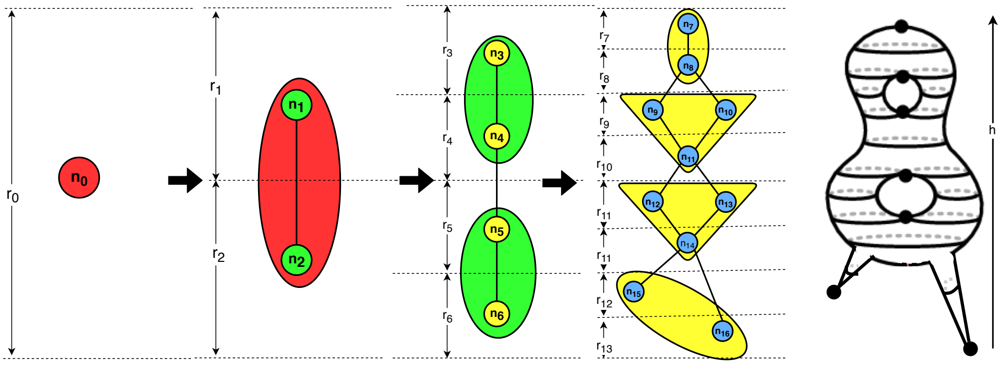


Figure FC1.4: An MRG of the height function h of a standing double torus with legs. The figure shows the MRG with four Reeb graphs at four different resolutions - coarser to finer Reeb graphs are shown from the left to right.

An MRG is a set of graphs that satisfies the following properties: First, a parent-child relationship is maintained between adjacent nodes. Second, by reversing the process of MRG, one can obtain the actual Reeb graph. Third, every Reeb graph of the finer level contains information about its corresponding coarser level. This information is used to construct multiple resolutions in the graph. Figure. FC1.4 shows the construction of MRG over the range of the height function h of the shape. In the Figure. FC1.4, the range of h is divided into intervals at different resolutions. The rightmost is the shape, which is a double torus with legs of different height. Now from left to right, the range of h is a single interval r_0, the resultant Reeb graph obtained will be a single node representing the fully connected object (leftmost). Next the interval r_0 is partitioned into intervals r_1 and r_2, giving rise to a Reeb graph with two nodes n_1 and n_2. According to the connectivity of the object, edges are drawn between n_1 and n_2. Similarly, finer resolution of Reeb graphs are constructed.

1.2.2 Multi-field Topology

A large number of data in scientific simulations and experiments usually consist of more than one scalar field. To visualize and understand these data various tools are used. In this section we describe different terminologies to understand the tools for multi-field topology.

Fiber. Similar to level sets in a scalar field we consider fibers corresponding to a multi-field. Let $f = (f_1, f_2, \ldots, f_n) : R^m \rightarrow R^n$ be a multi-field with each $f_i : R^m \rightarrow R$ being a scalar field ($i = 1, 2, \ldots, n$). Then a fiber of f corresponding to a point $c = (c_1, c_2, \ldots, c_n) \in R^n$ is defined as a set of points $p \in R^m$ such that $f(p) = c$ [13] and is denoted as $f^{-1}(c)$. A fiber $f^{-1}(c)$ can also be expressed as the intersection of the level sets of the component scalar fields, i.e $f^{-1}(c) = f_1^{-1}(c_1) \cap f_2^{-1}(c_2) \cap f_3^{-1}(c_3) \cap \ldots \cap f_n^{-1}(c_n)$. A connected component of a fiber is called a fiber-component or joint contour [13].

Reeb Space. Similar as the Reeb graph for a scalar field, a Reeb space captures the fiber topology of a multifield [13]. Each point on the Reeb space corresponds to a fiber-component or joint contours of a multi-field and vice-versa . For a multi-field $f : R^m \longrightarrow R^n$, when $n \leq m$, the Reeb space consists of n-manifolds joined together in a complicated fashion [18]. Figure FC1.5 shows an example of the Reeb space corresponding to a multi-field consists of two scalar fields $f_1(x, y, z) = x^2 + y^2 - z$ and $f_2 = z$. We compute an approximation of the Reeb space using the joint contour net algorithm [11], as described later.

Jacobi Set and Jacobi Structure. In a multi-field, $f = (f_1, f_2, ..., f_n)$, the set of critical point of one field (say f_i) restricted to the intersection of level sets of rest of the component fields is known as *Jacobi set*. Intuitively, Jacobi set of two generic Morse functions is the set of points where gradients of the individual fields are parallel [15]. The projection of Jacobi set of the multi-field from the domain to the Reeb space is defined as the *Jacobi Structure* of Reeb Space [12]. In Figure FC1.5 the red lines in (a) is the Jacobi set and the red lines in (b) is the corresponding Jacobi structure.

Joint Contour Net. Joint Contour Net (JCN) gives a practical algorithm for approximating a Reeb space [9]. The level sets of the fields are quantized into discrete levels and a connected component of the quantized level set in a mesh is called a quantized contour or contour slab. A JCN is constructed in four phases. First, corresponding to a quantization of each field, all the contour fragments in each cell of the whole mesh is constructed. Second, the contour fragments of the component fields in a cell are intersected to obtain the joint contour fragments. Third, corresponding to each joint contour fragment a node is created, to construct an adjacency graph (dual graph). An

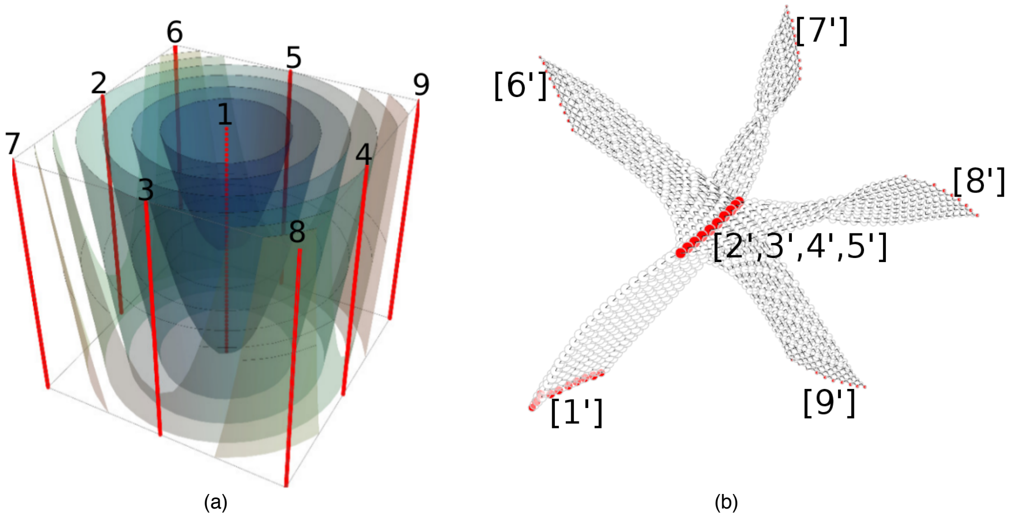


Figure FC1.5: (a) Paraboloid and height field with Jacobi set (red), (b) Reeb Space (JCN) of the corresponding multi-field with Jacobi structure (red).

edge is added between two nodes if the corresponding joint contour fragments are adjacent. Finally, the neighbouring additional nodes with similar isovalues are collapsed to obtain a JCN. Thus, each node in the JCN corresponds to a joint contour slab or quantized fiber-component and an edge represents the adjacency between two quantized fiber-components. Figure FC1.6 (left) shows an example of a JCN for a bivariate field in planar domain.

Multi-Dimensional Reeb Graph. A Multi Dimensional Reeb Graph (MDRG) is a hierarchical decomposition of the Reeb space (or the joint contour net) into set of Reeb graphs in different dimensions [12]. For a bivariate field (f_1, f_2) first we compute the Reeb graph $R(f_1)$ of the field f_1 (in the first dimension). Now each point $p \in R(f_1)$ corresponds to a contour C_p of f_1. We restrict function f_2 (in second dimension) on C_p and define the restricted function $\tilde{f}_2^p = f_2|_{C_p}$. Then for the second dimension, we compute Reeb graphs for each of these restricted functions $\tilde{f}_2^p$. Figure FC1.6 shows

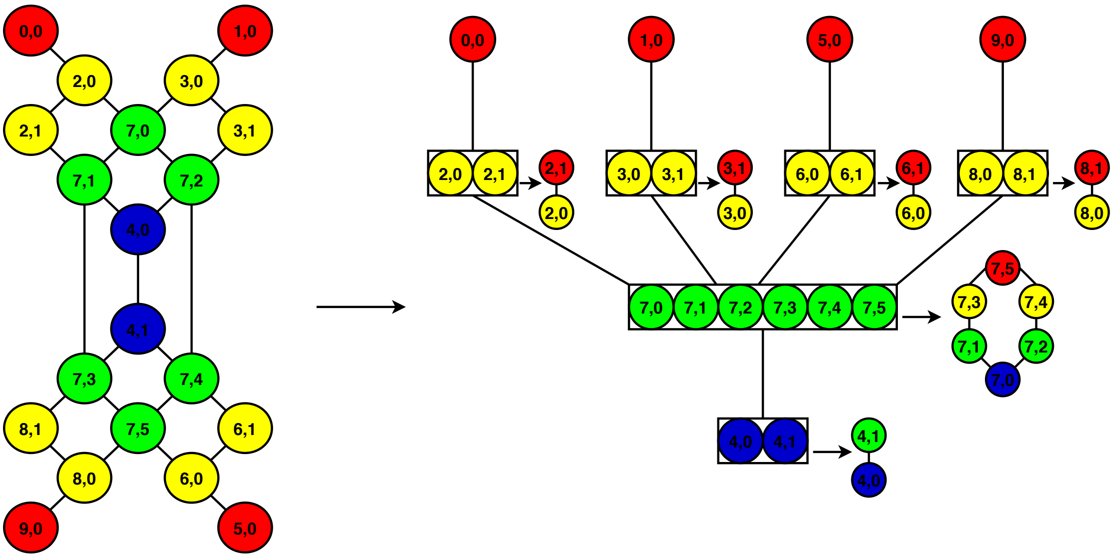


Figure FC1.6: JCN (left) and corresponding MDRG (right) over a bi-variate field data (field 1: Concentric circles and field 2: Parallel lines) over a $5 \times 5 \times 1$ domain.

an example of a quantized Reeb space or JCN and its decomposition into MDRG for a bivariate field.

1.2.3 Similarity/Dissimilarity Calculation

A lot of work has been done to extract topological features from scalar field data. Various distance measures have been proposed between different data structures to extract these features. A distance measure between extremum graphs has been proposed by Narayana et al. [33] to compare scalar fields. A survey by Gao et. al [19] on graph edit distance for application of pattern analysis using different inexact matching algorithms has been studied. Bauer et. al. [6] proposed a functional distortion metric on the Reeb graph and proved its metric properties. An edit distance between merge trees for visualizing features in time varying scalar field data is proposed in [6]. Feature visualization in time-varying single field data [40] is done by Sridharamurthy et al. by proposing an edit distance metric between merge trees. Other related work has been done by proposing a distance metric between merge trees to find similarity between scalar fields [32]. An interleaving distance as a distance metric between merge trees has been proposed by Morozov et al. [32].

Other methods that do not use topology based methods, to visualize and track features in time-varying data have been proposed in the literature. Lee et al. [41] proposed a time-activity curve to visualize features in time-varying data. Earth mover's distance is proposed to solve the problem of aggregate-attribute criteria and volume overlapping. The earth mover's distance is proposed by Jie et. al [25]. The branch-and-bound approach is a global optimization algorithm and the proposed metric can track features efficiently and accurately.

1.2.4 Problem Statements

Generalization of above described data structures and distance metric to time-varying multi-field data is challenging and requires further development. Although other tools such as Reeb space, Jacobi set and Joint Contour Net are already proposed in literature but a comparative study of these methods in time-varying multi-field data requires more development. In this thesis, we deal with two main problems.

First, finding a topological similarity metric that can be used to capture topological changes in time-varying multi-field data. In this work [3], we introduce a topology-aware distance metric between two multi-field based on their fiber-component distributions or histograms in the range space. We also prove the metric properties of the proposed distance measures. We show that the proposed measures capture significant or interesting events in time-varying phenomena, not possible using a study of individual fields. We validate the method by experimenting on a time-varying synthetic data where topological features are known in advance. We show the effectiveness of the method by experimenting on previously studied nuclear-scission data and re-explain how scission events are captured. We also apply our method in capturing important feature in the orbital data of Pt-CO interaction.

Second, to create a data structure that can store topological features of multi-field data. In this work, we introduce a hierarchical data structure called Multi-Resolution Multi Dimension Reeb Graph (MRMDRG) that can hierarchically store Reeb graphs of multiple resolutions and multiple dimensions. We also propose a similarity measure that captures significant features for time-varying data, which is not possible using scalar field data. Later, to validate the method, we experiment with different simulated data where topological features are known in advance. We show the effectiveness of the method by experimenting with previously studied nuclear scission data and re-examine the scission event.

In chapter 2, we will go through the detailed description of the first problem statement along with the proposed method and its application on different datasets. In chapter 3, we will explain the second problem statement in detail, along with the proposed method and its application on various synthetic and real datasets. Finally, we present the concluding remarks in chapter 4 and outline the scope for future study.

CHAPTER 2

TOPOLOGICAL FEATURE SEARCH IN TIME-VARYING MULTI-FIELD DATA

2.1 Chapter Summary

A wide range of data that appear in scientific experiments and simulations are multivariate or multi-field in nature, consisting of multiple scalar fields. Topological feature search of such data aims to reveal important properties useful to the domain scientists. It has been shown in recent works that a single scalar field is insufficient to capture many important topological features in the data, instead one needs to consider topological relationships between multiple scalar fields. In the current chapter, we propose a novel method of finding similarity between two multi-field data by comparing their respective fiber component distributions. Given a time-varying multi-field data, the method computes a metric plot for each pair of histograms at consecutive time stamps to understand the topological changes in the data over time. We validate the method using real and synthetic data. The effectiveness of the proposed method is shown by its ability to capture important topological features that are not always possible to detect using the individual component scalar fields.

2.2 Introduction

Scientists understand different physical phenomena by studying the interrelationships between features in different fields. It has been observed and shown that such multi-field or multivariate data can reveal many important phenomena about an experiment that are impossible to study using a single scalar field data [10, 14]. Development of tools and techniques for extracting and visualizing features in multi-field data is an important topic of research interest [21]. Topology-based methods have been shown to be extremely effective in this context. During the previous two decades, topological analysis of shapes and data was mostly driven by scalar topology, using contour tree, Reeb graph, Morse-Smale complex and their variants [7]. Such techniques have also been extended for time-varying scalar field data by defining different topology-aware similarity measures between two scalar fields [6, 33, 37].

Generalization of the techniques to time-varying multi-field data is challenging and requires further development in both theory and computational methods. More recently, new tools have been proposed for understanding and visualizing multi-field data – Reeb Space [18], Jacobi set [8, 15, 17], Joint Contour Net [9, 14] and Pareto analysis [24]. Extending these methods to time-varying multi-field data requires the development of techniques for comparative analysis and visualization. For example, developing a comparative measure between two Reeb spaces is a challenging open problem. In this thesis chapter, we consider a simpler feature descriptor of a multi-field, namely its fiber-component distribution or histogram. Using this, we make a first step forward towards a topology-aware distance measure between two multi-fields in terms of the distance between their fiber-component distributions. Our contribution in the current chapter is as follows:

- We introduce simple topology-aware distance measures between two multi-fields based on their fiber-component distributions or histograms in the range space. We prove the metric properties of the proposed distance measures.

- We show that the proposed measures capture significant or interesting events in time-varying phenomena, not possible using a study of individual fields. We validate the method by experimenting on a time-varying synthetic data where topological features are known in advance.
- We show effectiveness of our method by experimenting on previously studied nuclear-scission data [14] and re-explain how scission events are captured. We also apply our method in capturing important feature in the orbital data of Pt-CO interaction.

Section 2.3 discusses related works on scalar and multi-field data analysis. Section 2.4 describes different data structures or representations used critical for understanding and visualizing multi-field data. Section 2.5 introduces our proposed topology-aware distance measures and describes important properties of the measure. Section 2.6 discusses the implementation details and Section 2.7 and Section 2.8 describe various results of experiments on synthetic and real data. The experiments are conducted on nuclear scission, fission, and molecular orbital density data of Pt-CO interaction.

2.3 Related Work

Feature extraction in time-varying data is a well studied topic and several approaches have been proposed. We describe a few relevant approaches here.

Various similarity measures between scalar fields have been studied to analyze repeating patterns and similar arrangements in the data. Hilaga et al. studied topological shape matching using a multiresolution Reeb Graph (MRG) [23]. Saikia et al. propose a method for finding repeating topological structure in a scalar data using a data structure called the extended branch decomposition graph (eBDG) [37]. In ther following paper [?] the authors describe a histogram feature descriptor to compare subtrees of merge trees against each other. Narayanan et al. define a distance measure between extremum graphs to compare two scalar fields [33].

Many other comparison measures have been proposed in the literature for finding the distance between graphs or topological data structures. Bauer et al. have proposed a functional distortion metric on Reeb Graph and show its stability properties [6]. A survey on graph edit distance by Gao et al. [19] discusses different inexact graph matching algorithms for the application in pattern analysis. Sridharamurthy et al. propose an edit distance between merge trees for feature visualization in time-varying scalar data [40]. Thomas et al. propose a multiscale symmetry detection technique in scalar fields using contour clustering and studying the similarity between them [42]. In related works, different distance metrics between the merge trees have been proposed to provide a similarity between the corresponding scalar fields [?, 32].

Other techniques that are not based on topological analysis have also been proposed in the literature for tracking and visualizing time-varying features. Ji et al. [25] proposed a global optimization algorithm for time-varying data and resolved the problems of volume overlapping and aggregate-attribute criteria by using the earth mover's distance. A branch-and-bound approach was used for the global cost evaluation. The resultant approach and the metric was able to track features accurately and efficiently. Lee et al. [41] proposed a time activity curve (TAC) to visualize time-varying features.

However, topological feature search in time-varying multi-field data is a comparatively new area of research and only few works can be found in the literature. Duke et al. [14] propose a joint contour net (JCN) based visualization technique for detecting nuclear scission feature in the time-varying multi-field density data. It has been observed that direct visualization of the topological features using JCNs does not scale to large data sizes because the JCN structure can be extremely complicated. In this chapter, our method replaces this JCN visualization technique by a histogram comparison method.

2.4 Background

In this section, we discuss a few tools and techniques from the literature that are required to describe our proposed distance measure.

2.4.1 Histogram and isosurface statistics, continuous scatter plot

A histogram visualizes the distribution of the samples of a scalar field using a bar graph that is constructed by binning the samples in the field range. Histograms provide a measure of importance of isovalues based on the statistics of sample points. Carr et al. [20] show that histograms represent the spatial distribution of scalar fields with a nearest neighbourhood interpolation. Moreover, they show that isosurface statistics, such as the area of isosurfaces [5], betters represent the distribution of a scalar field.

Bivariate histograms represent two fields together. These histograms consist of bins of possibly different shapes such as square, triangle or hexagonal [38]. Square shaped bins of the histogram consist of the count for each pair of values defined on the axes. This count can be used to calculate the variance and bias from the integrated mean square error by using appropriate formulae. The square bins can be stretched to a rectangular shape based on the scale defined on the axes.

The density function corresponding to a collection of continuous input fields is well represented by a continuous scatter plot. Unlike histograms, continuous scatter plots do not depend on the bin sizes. Bachthaler et al [4] describe a mathematical model for generic continuous scatter plots of maps from n-D spatial domain to m-D data domain.

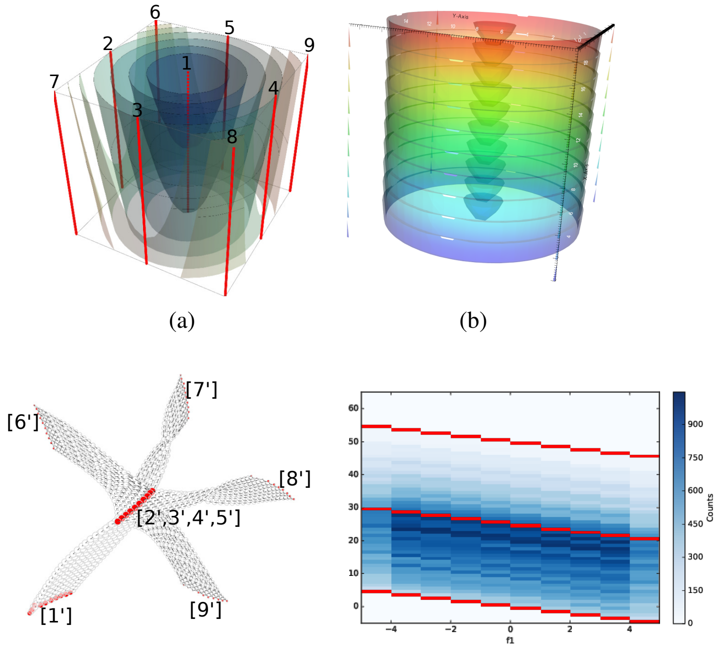
Lehmann et al. [29] describe algorithm for detecting discontinuities in the continuous scatter plots that reveal important topological features in the data.

2.4.2 Multi-field Topology and Jacobi Set

A multi-field on a d-manifold $\mathbb{M}\,(\subseteq \mathbb{R}^d)$ with r component scalar fields $f_i : \mathbb{M} \rightarrow \mathbb{R}$ $(i = 1, \ldots, r)$ is a *map* $\mathbf{f} = (f_1, f_2, \ldots, f_r) : \mathbb{M} \rightarrow \mathbb{R}^r$. In differential topology, $\mathbf{f}$ is considered to be a *smooth map* when all its partial derivatives of any order are continuous. A point $\mathbf{x} \in \mathbb{M}$ is called a *singular point* (or *critical point*) of $\mathbf{f}$ if the rank of its differential map $\mathbf{df_x}$ is strictly less than $\min\{d, r\}$ where $\mathbf{df_x}$ is the $r \times d$ Jacobian matrix whose rows are the gradients of f_1 to f_r at $\mathbf{x}$. And the corresponding value $\mathbf{f}(\mathbf{x}) = \mathbf{c} = (c_1, c_2, \ldots, c_r)$ in $\mathbb{R}^r$ is a *singular value*. Otherwise if the rank of the differential map $\mathbf{df_x}$ is $\min\{d, r\}$ then $\mathbf{x}$ is called a *regular point* and a point $\mathbf{y} \in \mathbb{R}^r$ is a *regular value* if $\mathbf{f}^{-1}(\mathbf{y})$ does not contain a singular point.

The inverse image of the map $\mathbf{f}$ corresponding to a value $\mathbf{c} \in \mathbb{R}^r$, $\mathbf{f}^{-1}(\mathbf{c})$ is called a *fiber* and each connected component of the fiber is called a *fiber-component* [35, 36]. In particular, for a scalar field these are known as the *level set* and the *contour*, respectively. The inverse image of a singular value is called a *singular fiber* and the inverse image of a regular value is called a *regular fiber*. If a fiber-component passes through a singular point, it is called a *singular fiber-component*. Otherwise, it is known as a *regular fiber-component*. Note that a singular fiber may contain a regular fiber-component. Topology of a multi-field data is usually studied based on its fiber-topology [13].

Jacobi set is used to study topological relationship between two or multiple scalar fields. Jacobi set $\mathbb{J}_\mathbf{f}$ of a multi-field $\mathbf{f}$ is the closure of the set of all its singular points, i.e. $\mathbb{J}_\mathbf{f} = cl\,\{\mathbf{x} \in \mathbb{M} : \text{rank } \mathbf{df_x} < \min\{d, r\}\}$. Alternatively, the Jacobi set is the set of critical points of one component field (say f_i) of $\mathbf{f}$ restricted to the intersection of the level sets of the remaining component fields [15]. Intuitively, Jacobi set of two generic Morse functions $f_1, f_2 : \mathbb{M} \to \mathbb{R}$ is the set of points where gradients of the individual fields are parallel, i.e. $\mathbb{J} = \{\mathbf{x} \in \mathbb{M} : \nabla f_1(\mathbf{x}) \times \nabla f_2(\mathbf{x}) = \mathbf{0}\}$. Jacobi set plays a central role in the design of a comparison measure between two or multiple scalar fields [17].



(a)

(b)

(c)

(d)

Figure FC2.1: Figure shows a bivariate synthetic data and corresponding structures to understand its topology. (a) Paraboloid and height field with Jacobi set (red), total 9 connected components of the Jacobi set are numbered as 1 to 9 (b) Singular fiber-components that pass through the Jacobi set points, (c) Reeb Space (JCN) with Jacobi structure (in red). Jacobi structure components that are the projection of the Jacobi set components on the Reeb Space are shown by the corresponding dashed numbers. (d) Histogram with singular values (bins).

2.4.3 Reeb Space and Joint Contour Net

Similar to the Reeb graph of a scalar field, the Reeb space parameterizes the fiber-components of a multi-field and its topology is described by the standard quotient space topology [18]. A Jacobi structure has been defined as a projection of the Jacobi set on the Reeb space, by the quotient map [13]. Figure FC2.1c illustrates a Reeb space with Jacobi structure (in red) corresponding to a bivariate field.

Joint Contour Net (JCN) [9] gives a practical algorithm for approximating a Reeb space. A JCN is built in four stages. The first step of the JCN algorithm constructs all the *contour fragments* in each cell of the entire mesh corresponding to a quantization of each component field. In the second step, the *joint contour fragments* are computed by computing the intersections of these contour fragments for the component fields in a cell. The third step is to construct an adjacency graph (dual graph) of these joint contour fragments where a node in the graph corresponds to a joint contour fragment and there is an edge between two nodes if the corresponding joint contour fragments are adjacent. Finally, the JCN is obtained by collapsing the neighbouring redundant nodes with identical isovalues. Thus, each node in the JCN corresponds to a *joint contour slab* or quantized fiber-component and an edge represents the adjacency between two quantized fiber-components. We use the JCN implementation for computing the quantized fiber-components and its histogram, see Figure FC2.1d.

2.4.4 Histogram Distance Measures

Different measures have been proposed in the literature to study the distance between two histograms [34]. The measures may be classified into two types based on how they are computed – bin-to-bin measures or cross-bin measures. In the former type, bins with the same indices are compared. We list below, a few examples of measures for finding distance between two histograms H and K with bin count h_i and k_i respectively.

Minkowski-form distance:

$$d_{L_r}(H,K) = \left(\sum_i |h_i - k_i|^r \right)^{1/r} \qquad \text{(Eqn 2.1)}$$

Commonly used Minkowski-form distances are d_{L_1}, d_{L_2} and d_{L_∞}. These are often used to compute dissimilarity between two color images.

Histogram intersection:

$$d_{\cap}(H,K) = 1 - \frac{\sum_i \min(h_i,k_i)}{\sum_i k_i} \qquad \text{(Eqn 2.2)}$$

This distance can capture the partial matches when the areas of the two histograms are not equal.

Kullback-Leibler (KL) divergence:

$$d_{KL}(H,K) = \sum_i h_i \log\frac{h_i}{k_i} \qquad \text{(Eqn 2.3)}$$

This is designed from an information-theoretic viewpoint. The measure is non-symmetric and sensitive to histogram binning.

One example of a cross-bin dissimilarity measure is the

Quadratic-form distance:

$$d_A(H,K) = \sqrt{(\mathbf{h}-\mathbf{k})^T\mathbf{A}(\mathbf{h}-\mathbf{k})}, \qquad \text{(Eqn 2.4)}$$

where **h** and **k** are vector representations of H and K, respectively. The matrix $\mathbf{A} = [a_{ij}]$ is the similarity matrix where a_{ij} denote the similarity between the i-th bin of H with the j-th bin of K [34].

2.5 Our Method

Let us consider two continuous multi-fields $\mathbf{f} = (X_1, X_2, \ldots, X_r)$ and $\mathbf{g} = (Y_1, Y_2, \ldots, Y_r)$ over a d-dimensional compact domain $\mathbb{D} \subseteq \mathbb{R}^d$ where each of X_i and Y_i, $(i = 1, 2, \ldots, r)$ are real-valued scalar fields in the domain $\mathbb{D}$. We consider comparing multi-fields **f** and **g** that have almost similar topological features, e.g. multi-fields at two consecutive time steps of a time-varying multi-field data where topological features vary continuously over time. A fiber of the multi-field $\mathbf{f}$ corresponding to a parametric point $\mathbf{c} = (c_1, c_2, \ldots, c_r)$ is the preimage $\mathbf{f}^{-1}(\mathbf{c}) = X_1^{-1}(c_1) \cap X_2^{-1}(c_2) \cap \ldots \cap X_r^{-1}(c_r)$. A connected component of the fiber is called a fiber-component. Fiber-component topology is used to study multi-field topology, similar to the use of contour topology for scalar field studies. The Reeb space is a generalization of the Reeb graph. It captures the fiber-component topology corresponding to a multi-field. However, Reeb space structure is rather complicated and computing an effective distance measure between two Reeb spaces for comparing corresponding multi-fields is an open problem.

In the current work, we consider the change in fiber-component distribution over parametric space to capture the change in topology in two multi-fields with almost similar topological features. We observe that the change in number of fiber-components corresponding to a point on the parametric space implies the change (birth or death) in number of sheets of the Reeb Space. Therefore, to study the topological changes from $\mathbf{f}$ to $\mathbf{g}$ we first consider the fiber-component distributions as the feature-descriptors of the respective multi-fields. Next, we propose few simple distance measures between the fiber-component distributions to capture the difference in terms of topological features.

2.5.1 Fiber-Component Distribution over the Range Space

Let $\mathbf{f} = (X_1, X_2, \ldots, X_r)$ be a continuous multi-field from a d-dimensional compact domain $\mathbb{D} \subseteq \mathbb{R}^d$ to the r-dimensional range space $R_\mathbf{f} = [a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_r, b_r]$, $a_i, b_i \in \mathbb{R}$. Define the function $N : R_\mathbf{f} \rightarrow \mathbb{N}$ as $N(\mathbf{x}) = |\mathbf{f}^{-1}(\mathbf{x})|$ for $\mathbf{x} \in R_\mathbf{f}$, where $|\mathbf{f}^{-1}(\mathbf{x})|$ represents the number of connected components in the fiber $\mathbf{f}^{-1}(\mathbf{x})$. In other words, $N(\mathbf{x})$ maps each point $\mathbf{x}$ of $R_\mathbf{f}$ to the corresponding number of fiber-components of $\mathbf{f}$. We assume that N is a bounded function for multi-fields $\mathbf{f}$ defined over a compact domain $\mathbb{D}$. To compute the total number of fiber-components, we partition the range $R_\mathbf{f}$ into a union of m^r sub-boxes by introducing the partitions of the intervals: $a_i =$

$x_0^{(i)} < x_1^{(i)} < \ldots < x_m^{(i)} = b_i$ for $i = 1, 2, \ldots, r$. Let $\mathbf{x}_{i_1 i_2 \ldots i_r}$ be a point in the sub-box $B_{i_1 i_2 \ldots i_r} = [x_{i_1-1}^{(1)}, x_{i_1}^{(1)}] \times [x_{i_2-1}^{(2)}, x_{i_2}^{(2)}] \times \ldots \times [x_{i_r-1}^{(r)}, x_{i_r}^{(r)}]$ for $i_1, i_2, \ldots, i_r = 1, 2, \ldots, m$ with volume $\Delta V_{i_1 i_2 \ldots i_r}$. Then, $\mathbf{N}$, defined as the sum of number of fiber-components over all points in $R_\mathbf{f}$ is equal to

$$\mathbf{N} = \lim_{\text{all } \Delta V_{i_1 i_2 \ldots i_r} \to 0} \sum_{i_1, i_2, \ldots, i_r = 1}^{m} N(\mathbf{x}_{i_1 i_2 \ldots i_r}) \Delta V_{i_1 i_2 \ldots i_r} = \int_{R_\mathbf{f}} N(\mathbf{x}) d\mathbf{x}. \qquad \text{(Eqn 2.5)}$$

The function N is bounded and hence integrable. Next, we define a density function of the fiber-component distribution as:

$$\mathfrak{p}_\mathbf{f}(\mathbf{x}) = \frac{N(\mathbf{x})}{\mathbf{N}} \text{ for } \mathbf{x} \in R_\mathbf{f}, \qquad \text{(Eqn 2.6)}$$

where

$$\int_{R_\mathbf{f}} \mathfrak{p}_\mathbf{f}(\mathbf{x}) d\mathbf{x} = 1.$$

In practice, to compute the fiber-component distribution over the range space, we first discretize the continuous multi-field $\mathbf{f} = (X_1, X_2, \ldots, X_r)$ in the r-dimensional range space. Let field X_i be discretized (quantized) uniformly at the values $x_1^{(i)} < x_2^{(i)} < \ldots < x_{m_i}^{(i)}$ for $i = 1, 2, \ldots, r$. We denote this discrete range space as $\text{spec}(R_\mathbf{f}) = I_1 \times I_2 \times \ldots \times I_r$, the Cartesian product of $I_i = \{x_1^{(i)}, x_2^{(i)}, \ldots, x_{m_i}^{(i)}\}$ $(i = 1, 2, \ldots, r)$. Then we compute the frequency distribution of the corresponding fiber-components over this discrete range space (spectrum). The probability mass function of the corresponding discrete probability distribution is given by

$$p_\mathbf{f}(\mathbf{x}) = \frac{\tilde{N}_\mathbf{x}}{\tilde{\mathbf{N}}}, \text{ where } \mathbf{x} \in \text{spec}(R_\mathbf{f}). \qquad \text{(Eqn 2.7)}$$

Here, $\tilde{N}_\mathbf{x}$ counts the number of fiber-components at the parametric point $\mathbf{x} = (x_{i_1}^{(1)}, x_{i_2}^{(2)}, \ldots, x_{i_r}^{(r)})$

in spec$(R_{\mathbf{f}})$ (for $i_1 = 1,2,\ldots,m_1; i_2 = 1,2,\ldots,m_2;\ldots; i_r = 1,2,\ldots,m_r$) and $\tilde{\mathbf{N}}$ is the sum of number of fiber-components of $\mathbf{f}$ over all points in the discrete range space spec$(R_{\mathbf{f}})$. Note that $p_{\mathbf{f}}$ defines a probability mass function (p.m.f.) since $p_{\mathbf{f}}(\mathbf{x}) \geq 0$ and

$$\sum_{\mathbf{x}\in\text{spec}(R_{\mathbf{f}})} p_{\mathbf{f}}(\mathbf{x}) = 1.$$

When the quantization level goes to infinity then discrete case converges to the continuous case. Alternatively, one can define p.m.f. using $A_{\mathbf{x}}$ by measuring the size of the quantized fiber-components at the parametric point $\mathbf{x} \in$ spec$(R_{\mathbf{f}})$ and A is the total measure of all the fiber-components over spec$(R_{\mathbf{f}})$. Thus we have

$$p_{\mathbf{f}}(\mathbf{x}) = \frac{A_{\mathbf{x}}}{A}, \text{ where } \mathbf{x} \in \text{spec}(R_{\mathbf{f}}). \qquad \text{(Eqn 2.8)}$$

In the proposed distance measure that we will describe next, we consider the definitions in (Eqn 2.6) and (Eqn 2.7) because they capture the topological changes in the fibers of the multi-field.

2.5.2 Distance between two Fiber-Component Distributions

Let us consider two multi-fields $\mathbf{f}_1 = (X_1, X_2, \ldots, X_r)$ and $\mathbf{f}_2 = (Y_1, Y_2, \ldots, Y_r)$ over the domain $\mathbb{D} \subseteq \mathbb{R}^d$. Let $R_{\mathbf{f}_1}$ and $R_{\mathbf{f}_2}$ be the range spaces of $\mathbf{f}_1$ and $\mathbf{f}_2$, respectively. We note that the range spaces $R_{\mathbf{f}_1}$ and $R_{\mathbf{f}_2}$ may be different but restrict our attention to the case when they are almost equal. To define our distance measures between the fiber-component distributions of $\mathbf{f}_1$ and $\mathbf{f}_2$, first we extend the range spaces $R_{\mathbf{f}_1}$ and $R_{\mathbf{f}_2}$ to an equal range R. We define R as: $R = R_1 \times R_2 \times \ldots \times R_r$ where $R_i = \text{range } X_i \cup \text{range } Y_i$ for $i = 1,2,\ldots,r$. This extended range R is considered as the common domain of fiber-component distributions of both $\mathbf{f}_1$ and $\mathbf{f}_2$. The fiber-component distributions of $\mathbf{f}_1$ on the part $R \setminus R_{\mathbf{f}_1}$, corresponding to which $\mathbf{f}_1$ has no data, is filled with zeros. Similarly fiber-component distributions of $\mathbf{f}_2$ on $R \setminus R_{\mathbf{f}_2}$ is filled with zeros.

For the continuous case: let $\mathfrak{p}_{\mathbf{f}_1}$ and $\mathfrak{p}_{\mathbf{f}_2}$ be the density functions of the fiber-component distributions of $\mathbf{f}_1$ and $\mathbf{f}_2$, respectively, over the extended range R. Let $\mathbf{P}_1$ and $\mathbf{P}_2$ be the corresponding distribution functions. Then we define a point-wise distance measure between $\mathbf{P}_1$ and $\mathbf{P}_2$ as:

$$d_q(\mathbf{P}_1,\mathbf{P}_2) = \left(\int_R |\mathfrak{p}_{\mathbf{f}_1}(\mathbf{x}) - \mathfrak{p}_{\mathbf{f}_2}(\mathbf{x})|^q d\mathbf{x}\right)^{1/q} \qquad \text{(Eqn 2.9)}$$

for any real number $q \geq 1$. In particular for $q = 1$, $q = 2$ or $q = \infty$ we get similar distance measures of practical importance.

For the discrete case, let the range space R be discretized (quantized) as $\text{spec}(R) = I_1 \times I_2 \times \ldots \times I_r$ where $I_i = \{x_1^{(i)}, x_2^{(i)}, \ldots, x_{m_i}^{(i)}\}$. Let $\mathbf{P}_1 = \{p_{\mathbf{x}}^{(1)} : \mathbf{x} \in \text{spec}(R)\}$ and $\mathbf{P}_2 = \{p_{\mathbf{x}}^{(2)} : \mathbf{x} \in \text{spec}(R)\}$ be the fiber-component distributions of $\mathbf{f}_1$ and $\mathbf{f}_2$, respectively, over the discrete range space $\text{spec}(R)$. Then we define the point-wise distance measure between the distributions $\mathbf{P}_1$ and $\mathbf{P}_2$ as:

$$d_q(\mathbf{P}_1,\mathbf{P}_2) = \left(\sum_{\mathbf{x}\in\text{spec}(R)} |p_{\mathbf{x}}^{(1)} - p_{\mathbf{x}}^{(2)}|^q\right)^{1/q}. \qquad \text{(Eqn 2.10)}$$

for any real number $q \geq 1$. In particular, for $q = 1$, $q = 2$ and $q = \infty$ we have

$$d_1(\mathbf{P}_1,\mathbf{P}_2) = \sum_{\mathbf{x}\in\text{spec}(R)} |p_{\mathbf{x}}^{(1)} - p_{\mathbf{x}}^{(2)}| \qquad \text{(Eqn 2.11)}$$

$$d_2(\mathbf{P}_1,\mathbf{P}_2) = \left(\sum_{\mathbf{x}\in\text{spec}(R)} |p_{\mathbf{x}}^{(1)} - p_{\mathbf{x}}^{(2)}|^2\right)^{1/2} \qquad \text{(Eqn 2.12)}$$

and

$$d_\infty(\mathbf{P}_1, \mathbf{P}_2) = \sup_{\mathbf{x} \in \text{spec}(R)} |p_{\mathbf{x}}^{(1)} - p_{\mathbf{x}}^{(2)}|. \qquad \text{(Eqn 2.13)}$$

These distance measures are motivated from the observation that the point-wise difference $|\tilde{N}_{\mathbf{x}}^{(1)} - \tilde{N}_{\mathbf{x}}^{(2)}|$ captures the number of changes in fiber-components between two multi-fields at consecutive time steps for $\mathbf{x} \in \text{spec}(R)$. Note that each fiber-component of a multi-field corresponds to exactly one sheet of its Reeb space. So, the difference in number of fiber-components captures the number of possible changes in Reeb space sheets containing the parameter value $\mathbf{x}$. Thus, $|\tilde{N}_{\mathbf{x}}^{(1)} - \tilde{N}_{\mathbf{x}}^{(2)}|$ captures the number of births or deaths of sheets containing the parameter value $\mathbf{x}$ of the corresponding Reeb spaces.

2.5.3 Weighted Distance for the Singular Values

Singular fibers capture the topological changes in the evolution of fibers in a multi-field. The image of a singular fiber in the parametric space is called a singular value. Because of importance of the singular values compare to regular values, we propose a variant to the distance measure that weights the singular values differently,

$$d_q^{\mathbb{S}}(\mathbf{P}_1, \mathbf{P}_2; \omega) = \left[\omega \sum_{\mathbf{x} \in \mathbb{S}} |p_{\mathbf{x}}^{(1)} - p_{\mathbf{x}}^{(2)}|^q + \sum_{\mathbf{x} \notin \mathbb{S}} |p_{\mathbf{x}}^{(1)} - p_{\mathbf{x}}^{(2)}|^q \right]^{1/q}. \qquad \text{(Eqn 2.14)}$$

Here, $\mathbb{S}$ is the set of singular values in the discrete range space $\text{spec}(R)$ and $q \geq 1$. Moreover, $\omega > 1$ is the weight parameter to impose more importance to the singular values than the regular values. We observe from our experiments on different datasets that increasing the weight ω increases the prominence of the events that correspond to topological changes when we plot weighted distances over time. Figure FC2.1d shows a fiber-component histogram with the singular values (in red) corresponding to the bivariate field in Figure FC2.1a.

2.5.4 Metric Space Properties of the Distance Measures

It is important to show that the proposed distance measures between two distributions satisfy the metric space properties for the space $\mathscr{P}_R$ of all possible fiber-component distributions corresponding to different multi-fields with range R. Let us first show that $(\mathscr{P}_R, d_q)$ is a metric space.

1. **Non-negativity.** Note d_q is real-valued, finite and non-negative.
2. **Identity.** We note that for two distributions $\mathbf{P}_1, \mathbf{P}_2 \in \mathscr{P}_R$, $d_q(\mathbf{P}_1, \mathbf{P}_2) = 0$ if and only if $\mathbf{P}_1 = \mathbf{P}_2$, since $\sum\limits_{\mathbf{x} \in \text{spec}(R)} |p_{\mathbf{x}}^{(1)} - p_{\mathbf{x}}^{(2)}|^q = 0$ implies $p_{\mathbf{x}}^{(1)} = p_{\mathbf{x}}^{(2)}$ for all $\mathbf{x} \in \text{spec}(R)$.
3. **Symmetry.** It is straight-forward to show that $d_q(\mathbf{P}_1, \mathbf{P}_2) = d_q(\mathbf{P}_2, \mathbf{P}_1)$. This implies the symmetry property of d_q.
4. **Triangle inequality.** To show the triangle inequality of d_q we consider three fiber-component distributions $\mathbf{P}_1$, $\mathbf{P}_2$ and $\mathbf{P}_3$. Note, for $q = 1$, $|p_{\mathbf{x}}^{(1)} - p_{\mathbf{x}}^{(3)}| \leq |p_{\mathbf{x}}^{(1)} - p_{\mathbf{x}}^{(2)}| + |p_{\mathbf{x}}^{(2)} - p_{\mathbf{x}}^{(3)}|$. For $q \geq 1$, using Minkowski inequality [22] we can show that $d_q(\mathbf{P}_1, \mathbf{P}_3) \leq d_q(\mathbf{P}_1, \mathbf{P}_2) + d_q(\mathbf{P}_2, \mathbf{P}_3)$.

Similar properties can be proved for the other distance measures $d_q^{\mathbb{S}}$, d_1, d_2 and d_∞. However, note the above metric properties hold in the space of fiber-component distributions, not necessarily in the space of actual multi-fields.

2.6 Implementation

We implement the distance measures described in the previous section using Visualization Toolkit (VTK) [27] under the Joint Contour Net [9] implementation framework.

The implementation works for a generic pair for multi-fields but is particularly designed for time-varying multi-fields. We note that the range spaces of two multi-fields at two consecutive time steps are not necessarily the same and may vary slightly. We expand the range of both multi-fields by considering their component wise union and use zero-padding to compute the histogram as described in section 2.5.2. Next, we describe the four main steps of our implementation.

I. **Computing Fiber-Components:** First, we discretize or quantize the common range of the multi-fields into finite numbers of bins. Then corresponding to each bin-value, we compute the quantized fiber-components as described in the JCN algorithm [9]. In other words, compute the contour slabs in each cell for each of the scalar fields and then find intersection of the slabs to get the fragments. Finally an adjacency graph is computed from the fragments to obtain quantized fiber-components. Each quantized fiber-component corresponds to a node of the JCN.

II. **Computing Fiber-Component Histograms:** Next, we compute the r-dimensional fiber-component histogram corresponding to each multi-field on the range space. We use the same binning as used for the quantized fiber-component computation. Each bin in the range is populated with the corresponding fiber-components. We compute the number of fiber-components in each bin for the fiber-component histogram computation. A color map specifying the number of all the nodes is shown in Figure FC2.1d. The color map is chosen over a range of blue values. Light blue shows fewer number of nodes (fiber-components), and as the color darkens the number of nodes (fiber-components) also increases.

III. **Computing Singular Values of multi-fields:** To compute singular values first one needs to compute the singular points or the Jacobi set in the domain of the multi-field and then the corresponding range values of those points are actually the singular values. In the current implementation we first compute the Jacobi structure using a multi-dimensional Reeb graph (MDRG) as described in [12, 13] and then project them in the histogram-bins and call those bins as singular bins. We note that a singular bin of the histogram may contain both singular and regular fiber-components (nodes). In the histogram plot Figure FC2.1d, the red colored bins indicate the singular bins and blue are the regular bins. For the singular bins of the histogram the singular, regular and total nodes (singular and regular together) are stored separately for further computation.

IV. **Computing Distance Metrics between Histograms:** The above three steps are performed for multi-fields at all the time stamps or sites, and the corresponding histograms are stored in different files. A python script is then implemented to compute the corresponding probability density from the histogram. Then the distance metrics between two probability densities at the consecutive time steps are computed as in sections 2.5.2 and 2.5.3. The distance metric $d_q^{\mathbb{S}}$ (as in equation Eqn 2.14) is computed for different values of q and ω. This metric is computed using the singular and regular nodes. Note that if $q = 1$ and $\omega = 1$ the metric $d_q^{\mathbb{S}}$ is same as d_1. To validate the experiment d_1 is calculated using all the nodes (regular and singular nodes together). Along with the measures that we have proposed we even calculated the distance measures for the already defined metrics for histogram comparison as defined in section 2.4.4. The values for these distance metrics are stored and then used to create a comparison line plot. The values were also used to check the metric properties defined in section 2.5.4. We also calculated the simple root mean square distance for bivariate data for experimental comparison.

2.7 Applications

We now describe applications of the proposed comparison driven feature search method to four different datasets, namely (i) a synthetic data consisting of two polynomial functions, (ii) the scission data of plutonium atom, (iii) fission data of Fermium atom and (iv) the DFT data of carbon monoxide and platinum (CO-Pt) molecular bond.

2.7.1 Synthetic Data

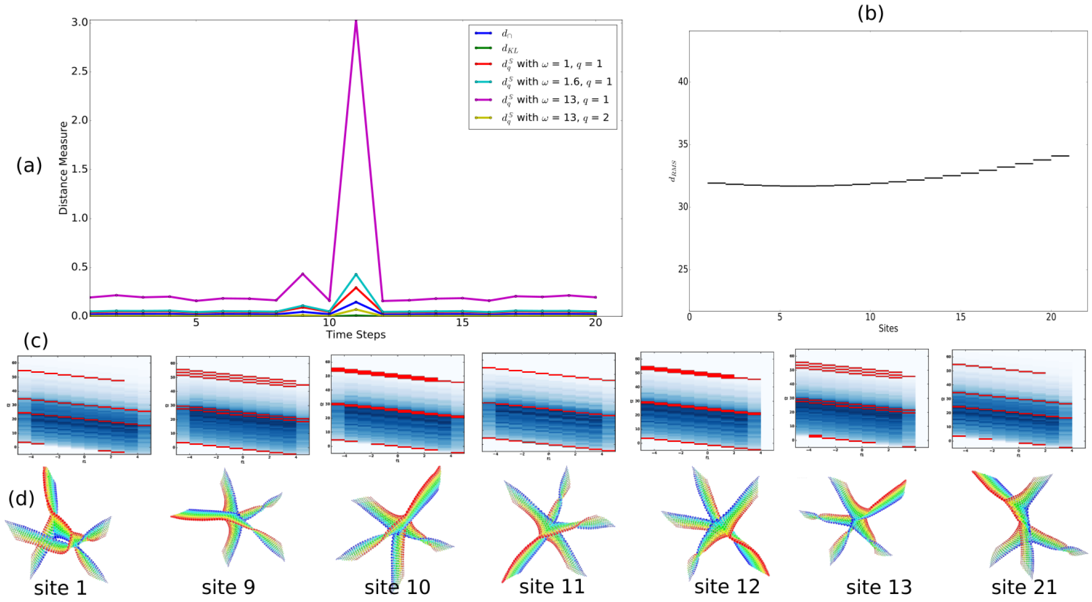


Figure FC2.2: Plots of distance measures between consecutive sites in a series of bivariate (height, paraboloid) fields. (a) Various distance measures show a peak at site 11, indicating a topological change. The proposed metric $d_q^{\mathbb{S}}$ also exhibits a peak, more significant than other distance measures.(b) Root-mean-square plot is not able to capture the topological change. This indicates the need for a topological data structures for multi-field data that captures topological changes. (c) Fiber-component distributions for selected sites. Singular values are highlighted in red. Blue nodes indicate regular nodes and the shades of blue indicate the number of nodes in a particular bin (light indicates low). (d) Corresponding Reeb spaces. The height field is mapped to color (blue is low and red is high).

We generate a synthetic bivariate field whose components are the height field $f_1(x,y,z) = z$ and the paraboloid field $f_2(x,y,z) = x^2 + y^2 - z$. Both fields are defined on an axis-aligned box $[-5.5, 4.5] \times [-5.5, 4.5] \times [-5.5, 4.5]$ and sampled on a grid of size $20 \times 20 \times 20$. Next, we generate a sequence of multi-field data by incrementally translating the domain-box along each of the three axes with small magnitude 0.05, i.e. if (C_x, C_y, C_z) and (c_x, c_y, c_z) are respectively the coordinates of a point on the box before and after the translation, then $C_x = c_x + 0.05$, $C_x = c_y + 0.05$, $C_z = c_z + 0.05$. In total, we create 21 bivariate datasets. To create the consecutive datasets, we begin with the domain $[-5.5, 4.5] \times [-5.5, 4.5] \times [-5.5, 4.5]$ and then apply the above described sequence of translations 21 times until we obtain the domain of the final dataset, namely $[-4.5, 5.5] \times [-4.5, 5.5] \times [-4.5, 5.5]$. The major topological feature is expected in the dataset corresponding to the domain $[-5, 5] \times [-5, 5] \times [-5, 5]$ (which is symmetric about origin) because of degenerate intersections of the fiber-components with the boundary of the box.

Observations and Results

We compute the fiber-component histograms for each dataset in the series and plot the distance between two consecutive datasets, see Figure FC2.2. The distance peaks at site 11 as expected. The red color in the histograms indicates singular nodes and blue color indicates regular nodes. The number of regular nodes in a particular bin is mapped to different shades of blue. Colors in the Reeb space indicate the height field value. Although various distance measures are able to capture the topological change, the peak was not sharp enough. The peak is most prominent using the $d_q^{\mathbb{S}}$ metric and increased weight for singular nodes. Note that all the subsequent experiments are done with $\omega = 13$ in order to keep the consistency in our experiments for all the datasets. If the value of ω is increased better peaks can be obtained and the value is not dependent on the chosen dataset.

Comparison with the Root Mean Squares Metric

To show the usefulness of the proposed metrics, we compute the distance between two multi-fields by directly extending the root mean square metric. The root mean square distance between two multi-fields $\mathbf{f} = (f_1, \ldots, f_r)$ and $\mathbf{g} = (g_1, \ldots, g_r)$ can be generalized as the square root of the mean of the sum of the difference between consecutive component fields:

$$d_{RMS} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\{(f_1(x_i)-g_1(x_i))^2+\cdots+(f_r(x_i)-g_r(x_i))^2\}}.$$

Here m is the number of data points in the domain. Figure FC2.2(b) shows the root mean square distance metric plot. We observe that the rms metric is not capable of capturing the topological change. This further motivates the study of measures such as the one proposed in this thesis chapter for comparing multi-field data.

2.7.2 Plutonium Atom Dataset

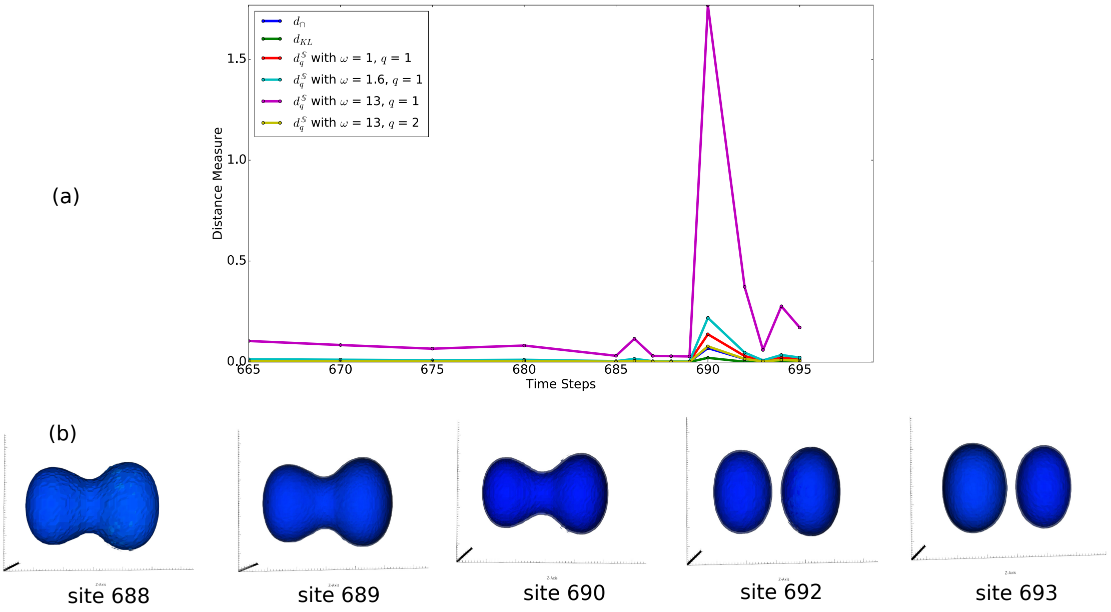


Figure FC2.3: Plots of the distance measures for the scission data for the plutonium atom. (a) Distance measure between fields at consecutive time steps vs. the time step in the range [665 – 699]. The proposed distance measure $d_q^{\mathbb{S}}$ exhibits a prominent peak between time step 690 – 692, which indicates a significant change. (b) Geometry of the plutonium atom at various time steps. The point of scission is between site 690 – 692 and can be seen in the geometry.

Nuclear Density Functional Theory (DFT) is an approach to understand the nuclear fission occurring in a nucleon-nucleon interaction in atomic nuclei. Nuclear fission is a process by which an atom's nucleus splits into two or more fragments. The splitting of the nucleus can be identified as stretching the core, hence it involves some deformation. This deformation can be a crucial indicator of the topology of the atom's nucleus. An important problem in nuclear fission study is the accurate identification of points in a continuous high dimensional manifold where the core is split. The time when the atom breaks into multiple fragments is known as nuclear scission. At this time the topology of the atom changes in terms of the number of components. Physicists typically identify this phenomenon via tedious manual process. Previous works have described a visual approach to identification of scission [14]. However, these methods require the inspection of the geometry of the Reeb space for all time steps. Further, the Reeb space is a complex structure that is difficult to examine. We aim to detect the key time steps that correspond to topological changes by plotting a graph of the distance measure over time.

The dataset consists of nuclear densities of plutonium atom which represents the internal structure of a heavy nucleus. The dataset is a multi-field data consisting of spatial density of proton, the spatial density of neutrons and spatial density of nucleons (protons + neutrons) in the nucleus. These densities, represented as **p**, **n** and **t** are sampled on a $40 \times 40 \times 66$ grid. The dataset available to us is a negative log transformed sample at 14 different time steps, namely [665, 670, 675, 680, 686, 687, 688, 689, 690, 692, 693, 694, 695, 699]. The time step where the nuclear scission occurs is reported in earlier work [14] and confirmed by physicists. We use sufficiently small slab width to capture the topological change. We use the following parameters in our experiments: **p** (slab width 8) and **n** (slab width 2), **p** (slab width 8) and **t** (slab width 2), **n** (slab width 2) and **t** (slab width 2).

Observations and Results

We experiment with all combination of proton, neutrons and nucleon density considering two fields at a time. The plots in Figure FC2.3 show the distance measure for the first combination, **p** (slab width 8) and **n** (slab width 2). We observe a sudden change between time steps 690 and 692. The d_1 distance was typically in the range of 0.0 to 0.02, but at nuclear scission, the measure increases to 0.1. This is due to the change in the number of quantized fiber-components in the range space. After scission, the distance measure dropped down to small values because the number of fiber-components does not change after the split. Figure FC2.3(a) shows a comparison with other bin-to-bin measures that are also able to capture the topology change but the peak is not as prominent. We plot the measure $d_q^{\mathbb{S}}$ for different values of q and weights. As the weight for singular values is increased, the peak becomes more prominent and as q is increased the plot becomes smoother. Figure FC2.3 shows the highest peak in the plot using weight $\omega = 13$ (for singular bins) and $q = 1$.

2.7.3 Fermium Atom Dataset

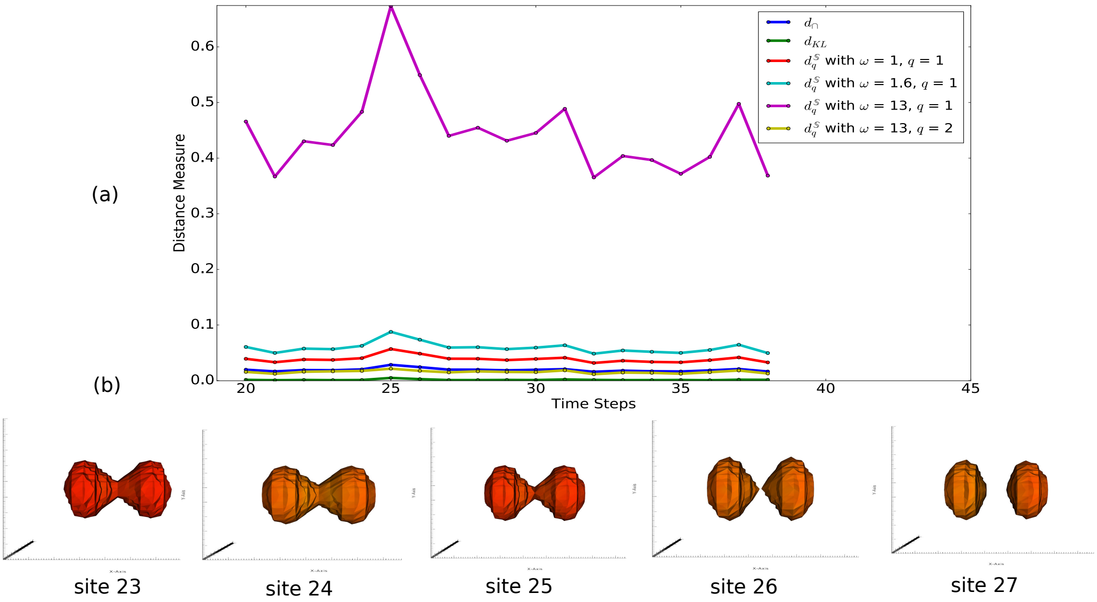


Figure FC2.4: Plots of the distance measures for the scission data for the fermium-256 atom. (a) Distance measure between fields at consecutive time steps vs. the time step in the range $[20, 39]$. The proposed distance measure $d_q^{\mathbb{S}}$ exhibits a prominent peak at time step 26, which indicates a significant change. (b) Geometry of the fermium-256 atom at various time steps. The point of scission is at site 26 and can be seen in the geometry.

We experiment with another scission dataset, namely that of the Fermium-256 atom. In this dataset, our goal is again to find the point where nuclear scission occurs. As described in the literature [14], this dataset consists of three different types of data viz. aEF: asymmetric elongated fission, sCF: symmetric compact fission and sEF: symmetric elongated fission. The dataset that was made available is the sCF data and was sufficient to detect the topological change where the fermium nucleus scission happens symmetrically. The sCF dataset consists of three fields i.e. proton density (**p**), neutron density (**n**) and total density (**t**) defined on a $19 \times 19 \times 19$ sized grid. The field is available at 56 regularly spaced time steps. Time steps 20-55 were chosen for analysis. Choosing the slab width was still an issue, and we end up working with the same slab width as that for Plutonium atom data, namely **p** (slab width 8) and **n** (slab width 2), **p** (slab width 8) and **t** (slab width 2), **n** (slab width 2) and **t** (slab width 2).

Observations and Results

The same set of experiments were done using the fermium-256 atom dataset. Figure FC2.4 shows the plots with proton and neutron density data from time step 20 to 39. We observe a topological change at time step 26. Other bin-to-bin histogram metrics, e.g. the KL divergence and the histogram intersection, exhibit a much smaller peak as compared to the proposed $d_q^{\mathbb{S}}$ distance.

2.7.4 Chemistry Data: Pt-CO Bond

Adsorption of gas molecules on metal surfaces has various applications including heterogeneous catalysis, electrochemistry, corrosion, and molecular electronics [26,39]. Particularly, the adsorption of the CO molecule on platinum surfaces has attracted attention of a wide scientific community, due to its role in the areas of automobile emission, fuel cells and other catalytic processes [2,28]. Therefore, an atomic-level understanding

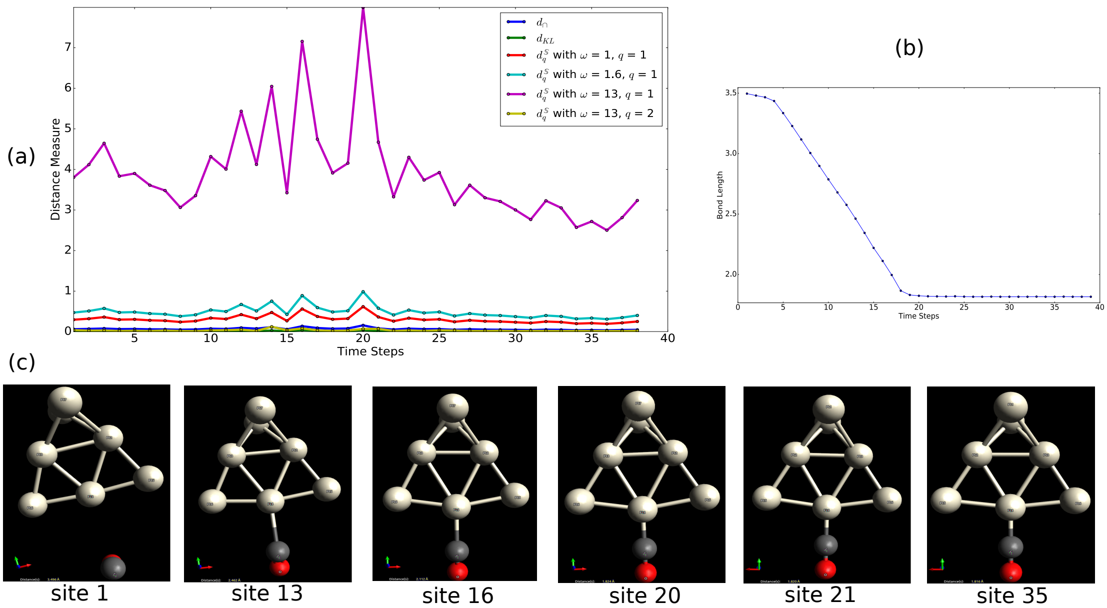


Figure FC2.5: Plots of the distance measures for the orbital density data of Pt-CO bond at different time steps. (a) Distance measure between fields at consecutive time steps vs. the time step in the range $[0, 39]$. The plots are for two field values, HOMO and LUMO and the highest peak is obtained at time stamp 21. The proposed distance measure $d_q^{\mathbb{S}}$ exhibits a prominent peak, which indicates a significant change. (b) Pt-CO Bond length vs time. Bond length stabilizes at time step 21. (c) Geometry of the Pt-CO bond creation at various time steps, visualized using the tool Avogadro. Although the bond is visible at time step 13, the bond length is not stable at this site.

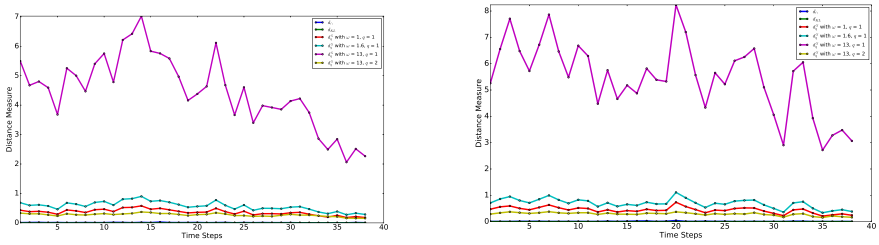
of the CO molecule interacting with the Pt surface is of utmost importance. In this study, we have considered seven Pt atoms representing a platinum surface which interacts with a CO molecule. As the CO molecule approaches towards one of the Pt atoms, the CO bond starts weakening, and Pt-CO bond formation takes place. Quantum mechanical computations were used to generate the electron density distribution corresponding to the highest occupied molecule orbital (HOMO), lowest occupied molecular orbital (LUMO) and HOMO−1. The electron density distribution was computed for varying distance between the carbon atom of the CO molecule and the Pt atom. The Pt-CO bond forms when the distance between the Pt atom and the CO molecule becomes $\sim 1.83A$. This Pt-CO dataset consists of orbital density for orbital numbers 69, 70 and 71. Orbital number 70 corresponds to HOMO, orbital number 71 to LUMO and orbital number 69 to HOMO−1.

Observations and Results

Figure FC2.5 shows different plots for the Pt-CO dataset. At site 21, we get the most stable bond length between Pt and CO molecule. We observe that although the bond is formed at site 13 (as validated by the geometry), the bond-length is not stable. The bond length stabilizes at site 21 and does not change much later. We observe a sharp peak in the plot of the proposed $d_q^{\mathbb{S}}$ distance. This peak corresponds to the formation of the stable bond.

2.8 Single Scalar Field vs. multi-field

We now describe an experiment to demonstrate the importance of studying tools for multi-field data over single scalar field analysis tools. Consider the Pt-CO molecular dataset. Using only orbital 69 (HOMO-1) data the highest peak in the distance measure plot is obtained at site 16 (Figure FC2.6. Distance plots for orbital 70 (HOMO) exhibit the highest peak at site 21. On the other hand, using two fields together, i.e. orbital data 69 and 70, or orbital data 70 and 71, or orbital data 69 and 71, we observe the highest peak is always at site 21. Some topological changes may not be captured using a bivariate data and we may need to consider more than two fields to detect the changes.



(a)

(b)

Figure FC2.6: Distance plot for scalar data for Pt-CO bond detection dataset. (a) Plot for orbital density 69 (HOMO−1). The highest peak is at site 16. (b) Plot for orbital density 70 (HOMO). Significant peak is at site 21.

CHAPTER 3

A TOPOLOGICAL SIMILARITY MEASURE BETWEEN MULTI-FIELDS USING MULTI-RESOLUTION REEB SPACES

3.1 Chapter Summary

Topological similarity measures between scalar-fields has been studied extensively and proven extremely useful in shape matching and time-varying scalar data analysis. However, similar research for computing topological similarity between multi-fields (or multiple scalar fields) is still in its infancy. In the current paper, we propose a topological similarity measure between two multi-fields in volumetric domain based on a similarity measure between their multi-resolution Reeb spaces. Our method generalizes the similarity measure between two multi-resolution Reeb graphs for the the shape matching problem, described by Hilaga et al. [23]. Overall, our method consists of two steps: (i) constructing a multi-resolution Reeb space in different resolutions of the data and (ii) defining a similarity measure between two such multi-resolution Reeb spaces. To satisfy the topological consistency between the points on the respective Reeb spaces, we consider a hierarchical decomposition of each Reeb space into a multi-dimensional Reeb graph. The effectiveness of the proposed similarity measure is shown by applying on different time-varying volumetric real multi-field data.

3.2 Introduction

To understand the physical phenomenon, scientists study the correlation between features in individual fields. In the last two decades, different topological tools and methods to extract topological features from data are mostly developed for single scalar field data such as contour tree, Reeb graphs, morse-smale complex [7] and many more. These techniques are also developed for time-varying data by determining different topological similarity measures between two scalar fields [6, 33, 37].

Multi-field topological features are richer compare to the scalar-field topology. It is observed that a single field data is not able to reveal many important features about an experiment that a multi-field data can reveal [10, 14]. Developing tools that can extract and visualize features for multi-field data is a recent topic of research and is still in its infancy. The generalization of such a technique is a very challenging task in terms of theoretical and computational aspects. Although recently, various tools have been introduced in the literature for explaining and visualizing multi-field data such as Jacobi sets [8, 15, 17], Reeb space [18], and Joint Contour Net [9, 14].

Developing a measure to do the comparison between two data sets is usually very complex and extending these methods to time-varying multi-field data requires much development. For example, visualization of the Joint contour net is very complex. It requires much human effort by digging into the structure to identify any topological feature and extending this to time-varying multi-field is even more difficult. In this paper, we design a data structure that can store topological information in a hierarchical format for each data field and different resolutions and later design a similarity metric to find the similarity between two multi-field data. The main contribution of the paper is described below:

1. We introduce a hierarchical data structure called Multi-Resolution Reeb space that

can hierarchically store Reeb spaces in multiple resolutions.

2. We propose a similarity measure that captures significant topological features for time-varying multi-field data, which is not possible using similarity measures between scalar data.

3. To validate our method, we experiment with different simulated data where topological features are known in advance. We show the effectiveness of our method by experimenting with previously studied nuclear scission data and re-examine the scission event.

Section 3.3 discusses the related works on the scalar field and multi-field topological analysis. Section 3.4 introduces our proposed data structure and our topology-aware similarity metric. Section 3.5 explains the algorithm for creating the data-structure and computing the similarity between the proposed data-structure. Section 3.6 explains the implementation of the proposed algorithm and Section 3.7 describes various applications along with their observations and results.

3.3 Related Work

Topological similarity and distance measures between scalar field data have been studied extensively. Hilaga et al. [23] proposed a similarity measure between two shapes by computing a Multi-resolution Reeb Graph (MRG) and then applied in topological shape matching. A histogram feature descriptor is proposed by Saikia et al. [**?**] to differentiate between subtrees of the merge tree. Narayanan et al. proposed a distance measure to compare scalar fields using extremum graphs [33]. A survey on graph edit distance and its application of pattern analysis using different inexact matching algorithms is discussed by Gao et al. [19].

Bauer et al. proposed a functional distortion metric for computing distance between two Reeb graphs [6]. A data structure called extended branch decomposition graph (eBDG) [37] is proposed by Saikia et al. using which repeating topological structure in a scalar data is identified. A multiscale symmetry detection technique using contour clustering is proposed in [42] by Thomas et al. Feature visualization in time-varying single field data [40] is done by Sridharamurthy et al. by proposing an edit distance metric between merge trees. Other work has been done to find similarity between scalar fields by proposing a distance metric between merge trees [**?**]. Morozov et al. proposed an interleaving distance as a distance metric between merge trees [32].

There are other techniques that do not use topology based methods to track and visualize time-varying features in the literature. The time-activity curve is proposed by Lee et al. [41] to visualize time-varying features. The problem of volume overlapping and aggregate-attribute criteria is solved using earth mover's distance, proposed by Jie et al. [25]. The proposed branch-and-bound approach is a global optimization algorithm, and the proposed metric can track features efficiently and accurately.

However, only a few investigations have been attempted towards finding a topological measure between two multi-field data. Carr et al. [9] proposed a data-structure known as joint contour net (JCN) that was applied to visualize nuclear scission features in multi-field density data [14]. However, JCN structure is cumbersome for most of the real datasets, and visualization topological features using such a structure is a difficult task. In similar context, Chattopadhyay et al. proposed a hierarchical structure known as Multi-Dimensional Reeb Graph (MDRG) [12] and used to identify critical features, known as Jacobi structure, in the JCN. Recently, Agarwal et al. [3] proposed a distance metric between the fiber-component distributions of multi-fields and shown its application in detecting topological features in time-varying multi-field data.

In the next two sections we describe our method for computing a topological similarity between two multi-fields. First, we propose a new multi-resolution Reeb space data-structure that captures the topology of a multi-field at different resolutions. In the second step, we propose a similarity measure between two such multi-resolution Reeb spaces.

3.4 Multi-Resolution Extension of Reeb Space

In this section we describe a new multi-resolution Reeb space (MRS) that captures the topology of a multi-field data at different resolutions. The Reeb space at a particular resolution is represented by its JCN. The idea is to develop a series of JCNs at different levels of resolution.

3.4.1 Overview

A multi-resolution Reeb space is a hierarchical data structure where each node of Reeb space represents the fiber component of a particular resolution and an edge is added between the nodes if their corresponding joint contour fragments are adjacent to each other. The finer resolution Reeb space is constructed by quantizing the level sets for each component field. For simplicity, the quantization is done in binary fashion. Figure FC3.1 shows an example of a bivariate 2-dimensional data where the first component field is the ring data and second component field is the bar data. In Figure FC3.1 (a) each of the component field is divided into one quantized level set, and hence the whole domain is represented using one node n_0. In Figure FC3.1 (b) both the component field is quantized into two quantized level sets, thereby representing unique values of each contour fragment with nodes n_1, n_2, n_3, n_4 and n_5. In Figure FC3.1 (c) each of the component field is quantized into four level sets, thereby representing each contour fragment with different nodes. These nodes can be used to identify the hierarchy. Nodes n_1, n_2, n_3, n_4, n_5 and n_6 in Figure FC3.1 (e) are united to form node n_0 in the coarset level (Figure FC3.1 (g)). Similarly, nodes n_7, n_9, n_{10}, n_{11} in Figure FC3.1 (i) are united to form node n_1 in corresponding coarser level (Figure FC3.1 (h)). Other nodes in the finer resolution are similarly united to form a node in the corresponding coarser level. The multi-resolution Reeb space should satisfy the following properties:

Property 1: There is parent-child relationship between the nodes of adjacent resolution. In Figure FC3.1, node n_0 is the parent of nodes n_1, n_2, n_3, n_4, n_5 and n_6. Similarly n_1 is the parent of n_7, n_9, n_{10} and n_{11}.

Property 2: The repeating the quantization the multi-resolution Reeb space converges to the original Reeb space. This means, that the finer resolution can approximate the domain more accurately.

Property 3: A Reeb space of a particular resolution implicitly contains all the information of the coarser resolutions. This means, once the Reeb space is constructed for a finer resolution the coarser resolution can be constructed easily by unifying the adjacent nodes based on the values of component fields. In Figure FC3.1, the nodes $\{n_1, n_2, n_3, n_4, n_5, n_6\}$ are unified to n_0 and nodes $\{n_7, n_9, n_{10}, n_{11}\}$ to n_1.

The multi-resolution Reeb space can be constructed using JCN construction satisfying above properties - this is described in the next subsection.

3.4.2 Construction of the Multi-Resolutional Reeb Spaces

The construction of multi-resolution Reeb space is illustrated in Figure FC3.1. In this case, we take a simple example of a 2D bivariate ringBar data for explanation. The construction of multi-resolution Reeb space starts with construction of Reeb space for the desired finest resolution. This Reeb space is constructed by dividing the domain of each component field into K quantized level sets. K determines the number of ranges and the fineness of resolution. In Figure FC3.1 last column, the domain is divided

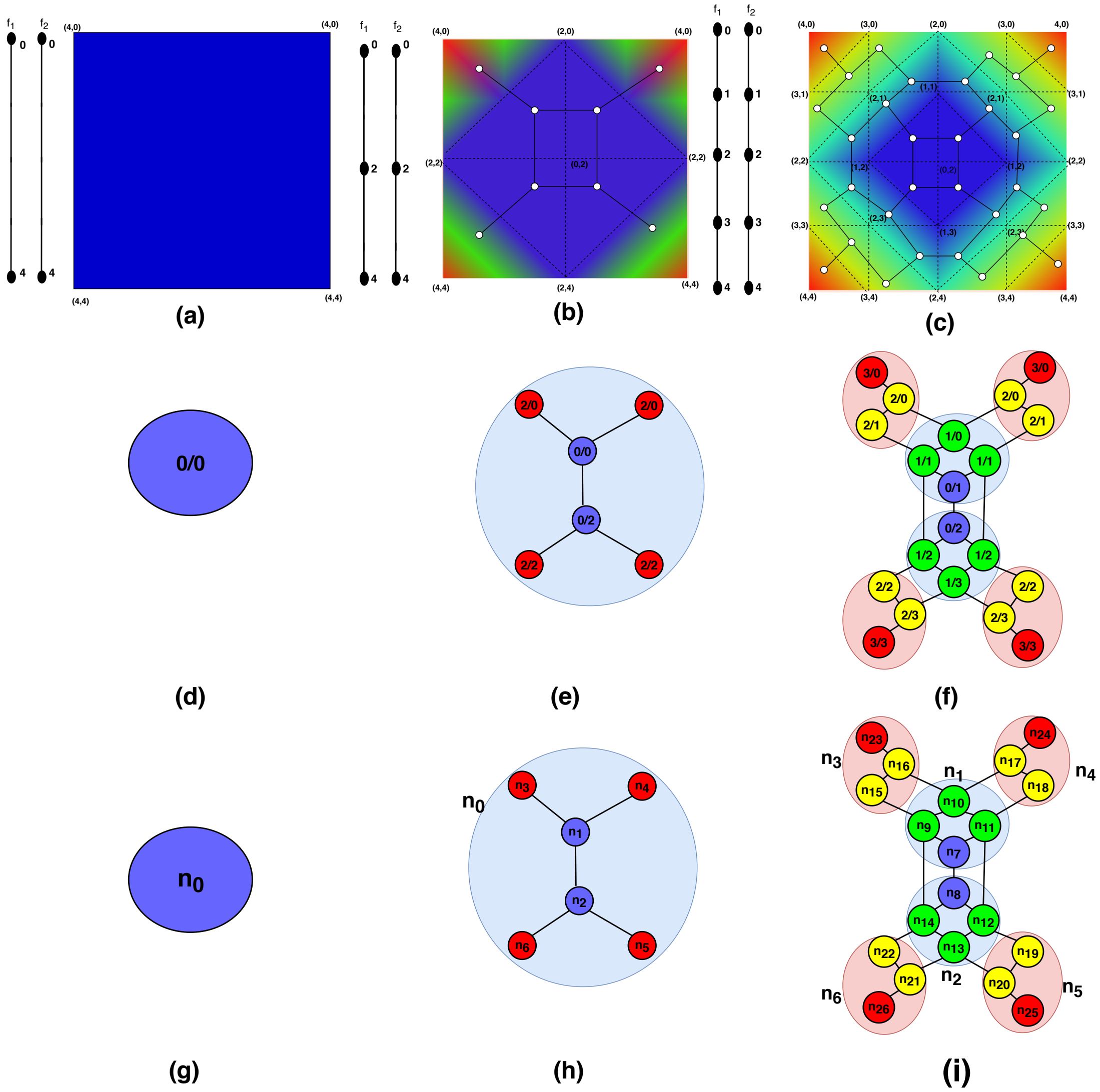


Figure FC3.1: Multi-resolution Reeb Space corresponding to a PL bivariate data: (ring, height). (a) Each component field is quantized into one level, and the corresponding JCN in (d) and (g) consists of only one node representing node value and node name n_0 respectively. (b) Each component field is quantized into two levels and the corresponding JCN is shown is (e) with node value and (h) with node names. (c) each component field is quantized into four levels, and the corresponding JCN is shown in (f) and (i) representing node values and node names repectively. Parent-child relationships between the nodes of JCNs in consecutive resolutions are also shown in (g), (h) and (i), e.g. n_0 is parent of $\{n_1, n_2, n_3, n_4, n_5, n_6\}$.

into 4 ranges for each component field, i.e., $r_{0i} = [0,1)$, $r_{1i} = [1,2)$, $r_{2i} = [2,3)$ and $r_{3i} = [3,4)$, where i determines the component field. Note that, each component field can also be quantized into different ranges. This results in subdivision of domain into triangles in 2D and tetrahedron in 3D. Each of these triangles are known as fragment or simplices. Second, the fragments lying on the boundaries are subdivided so that each fragment belongs to only one range. Third, the fiber components are determined and is represented using a node. Fourth, if two fragments are adjacent to each other and have identical values then the nodes corresponding to the fiber components are united together. And if the fragments are adjacent and the nodes have different value then an edge is added between them.

Next, the multi-resolution Reeb space is constructed from the finest resolution by applying the property 3 defined previously. That is, the multi-resolution Reeb space is constructed by unifying the adjacent nodes while the parent-child relationship is maintained as shown in Figure FC3.1. The edges connected the nodes of same resolution are also calculated. The parent and it corresponding child nodes are also determined at this time by creating a union-find data structure. In this, if a node in the finer resolution Reeb space have the same range value in the coarser resolution and the nodes are adjacent to each other, then these nodes are united to form a single node in coarser resolution Reeb space. In Figure FC3.1 (i) nodes n_{15}, n_{16}, n_{23} have same range value i.e. 2 for first field and 0 for second field in corresponding coarser resolution and are adjacent to each other in the finer resolution Reeb space, therefore, the nodes are united to form a single node n_3. Note that the finest resolution nodes will not have any child nodes and the coarsest resolution nodes will not have their corresponding parent node.

Algorithm 1 coarserResolutionReebSpace

```
**Input:** Finer resolution Reeb space $\mathrm{MRS_f}(V_f, E_f)$
**Output:** Coarser resolution Reeb space $\mathrm{MRS_c}(V_c, E_c)$
**%CREATE COARSER NODES**
Finer resolution nodes $m_0, m_1 \in V_f$
**if** IsSame(range($m_0$),range($m_1$) $\in V_c$ **then**
    **if** (IsAdajcent($m_0, m_1 \in V_f$ ) **then**
        $n_0 \leftarrow$ Union($m_0, m_1$)
    **end if**
**end if**
parent($m_0$) = $n_0$, parent($m_1$) = $n_0$
child($n_0$) = Union$\{m_0, m_1\}$
$V_c$ = Add($n_0$)
**Repeat for other nodes in** $\mathrm{RS_f}$
**%CREATE EDGES BETWEEN COARSER NODES**
Nodes $n_0, n_1 \in V_c$
**if** IsAdjacent(child($n_0$),child($n_1$)) $\in \mathrm{MRS_f}$ **then**
    $e_0$ = AddEdge($n_0, n_1$)
**end if**
$E_c$ = Add($e_0$)
**Repeat for other nodes in** $\mathrm{MRS_c}$
```

Algorithm. The above algorithm is use to create a coarser resolution node of a coarser Reeb space from the nodes of finer resolution Reeb space. Here $\mathrm{MRS_f}$ represents the Reeb space of finer resolution and $\mathrm{MRS_c}$ represents the coarser resolution Reeb space. The nodes m_0 and m_1 belonging to $\mathrm{MRS_f}$ are merged to form a single node n_0 which belongs to $\mathrm{MRS_f}$. The above procedure is repeated for all the nodes of finer resolution. An edge is created between two nodes say n_0, n_1 of coarser resolution if any of the child nodes of n_0, n_1 are adjacent to each other in finer resolution.

3.4.3 Computational Cost for Constructing the Multiresolutional Reeb Spaces

While computing the computational cost of multi-resolution Reeb spaces, we analyse the cost of computing the Reeb space for finest resolution. The computation of finest resolution Reeb space as dicussed in [9] is dependent on number of fragments (simplices) N in the input mesh, the number of quantization levels Q_i of each component field f_i, the number of functions defined r and the number of dimensions d. The process of creating a Reeb space at finest resolution costs $O(rN_e + N_e\alpha(N_e))$, where $N_e = O((2r+d)N_f)$, $N_f = O(kN)$ and α is the inverse Ackermann function. Here, N_f is the number of fragments or the number of nodes in the Reeb space, N_e is the number of adjacent fragments or number of edges in the Reeb space and k is the product of number of Q_i, the quantization level. This results in a polynomial time algorithm which is mostly dependent on k. Since the construction of multi-resolution Reeb space is done from finer to coarser resolution, the other Reeb spaces can be easily constructed by identifying the adjacent fragments and identifying the parent-child relationship which is done in $O(N_f)$ time. Hence constructing the finest resolution Reeb space is predominant in the whole algorithm, therefore, the computational cost of the algorithm is polynomial.

3.5 Algorithm: Computing Similarity Measure

This section describes the second part of our method, i.e. computing a similarity measure between two multi-resolution Reeb Spaces.

3.5.1 Overview

First we give an overview of how the similarity is calculated between two multi-resolution Reeb spaces. Firstly, corresponding to each node (m) of multi-resolution Reeb space, an attribute is computed ($\overline{m}$). Initially, the attribute $\overline{m}$ is calculated for the finest resolution Reeb space using the degree of the node and number of edges in the Reeb space that will be described later. Generally, the attribute can be given by:

$$\overline{m} = \sum_{c} \overline{c} \qquad \text{(Eqn 3.1)}$$

where c is the child nodes of m. The attribute $\overline{m}$ is the sum of attributes of child nodes ($\overline{c}$) of m. Note that, this rule is not applied for finest resolution Reeb space since they do not have any child nodes.

The similarity between two nodes is calculated by finding the similarity between their attributes $\mathrm{sim}(\overline{m},\overline{n})$, where m and n are nodes in two different Reeb space of same resolution. This similarity satisfies following conditions: First, the similarity is maximum when the nodes are matched with itself.

$$0 \leq \mathrm{sim}(\overline{m},\overline{n}) \leq \mathrm{sim}(\overline{m},\overline{m}) \qquad \text{(Eqn 3.2)}$$

Second, the sum of similarities for all the nodes matched with itself is 1.

$$\sum_{m \in \mathrm{MRS}} \mathrm{sim}(\overline{m},\overline{m}) = 1 \qquad \text{(Eqn 3.3)}$$

The similarity between two multi-resolution Reeb spaces MRS_1 and MRS_2 is therefore, defined as the sum of similarities of the attributes of for pair of nodes and is given by following equation:

$$\mathrm{sim}(\mathrm{MRS}_1, \mathrm{MRS}_2) = \frac{1}{R}.\left[\sum_{m \in \mathrm{MRS}_1, n \in \mathrm{MRS}_2} \left(\mathrm{sim}(\overline{m}, \overline{n})\right)^q\right]^{\frac{1}{q}} \qquad \text{(Eqn 3.4)}$$

where R is the number of resolutions and q is any positive real number. These pair of nodes $\{(m_0, n_0), (m_1, n_1), \ldots\}$ are obtained using matching algorithm discussed in section 3.5.3 and hence we call these pairs as matching pairs or simply MPAIRs. The value of similarity for each of these pairs lies between $[0, 1]$ with a value near to 1 indicating, that the multi-resolution Reeb space are more similar. To find the maximum similarity between two multi-resolution Reeb space it is required to match the nodes, such that the topological consistency of the multi-resolution Reeb space is maintained. The rules to maintain the topological consistency are discussed in section 3.5.2. The MPAIRs are calculated from coarse to fine strategy and an overview of algorithm is given below.

Algorithm. The algorithm is an overview of matching algorithm. The nodes in two input multi-resolution Reeb space are inserted in their corresponding NLISTs. Then the matching pairs are created between the nodes inserted in NLISTs. highestSimilarityNode function finds the node in NLIST$_1$ which has highest similarity with itself. This node is known as chosenNode. Then the candidate node is obtained by using checkTopologicalConsistency function which is described in section 3.5.2. This function helps in finding the node that is topologically consistent with chosenNode. If there are more than one candidate node that can match with chosen node then this ambiguity is resolved by finding the node amongst the candidate nodes which has maximum matching with chosen node. The maxMatching function uses the equation defined in equation Eqn 3.7. Then update the MLIST$_1$ and MLIST$_2$ as described in section 3.5.2 and insert the pair of chosenNode and candidateNode as an MPAIR. These MPAIRs are then used to find the final similarity between the multi-resolution Reeb space.

Algorithm 2 MatchingAlgorithm

```
Input: Multi-resolution Reeb space $MRS_1, MRS_2$
Output: $sim(MRS_1, MRS_2)$
%INITIALIZATION
for node $m \in MRS_1$ do
    $NLIST_1 \leftarrow m$
end for
for node $n \in MRS_2$ do
    $NLIST_2 \leftarrow n$
end for
%MPAIR CREATION
for $\forall m \in NLIST_1$ do
    for $\forall n \in NLIST_2$ do
        chosenNode = highestSimilarityNode( m )
        if IsTopologicallyConsistent (chosenNode, $n$) then
            candidateNodes $\leftarrow n$
        end if
    end for
    for $\forall i \in$ candidateNodes do
        candidateNode = maxMatching(chosenNode, i)
    end for
    UpdateMLISTs (chosenNode, candidateNode)
    create MPAIR (chosenNode, candidateNode)
    Remove chosenNode from $NLIST_1$ and candidateNode from $NLIST_2$
    $NLIST_1 \leftarrow$ child(chosenNode)
    $NLIST_2 \leftarrow$ child(candidateNode)
end for
%SIMILARITY CALCULATION
for $\forall MPAIR(m, n)$ do
    $sim(MRS_1, MRS_2)$ += $sim(\overline{m}, \overline{n})$
end for
$sim(MRS_1, MRS_2) = \frac{1}{R}.\ sim(MRS_1, MRS_2)$
return $sim(MRS_1, MRS_2)$
```

3.5.2 Topological Consistency of Multiresolutional Reeb Space

To preserve topological consistency, the nodes in different branches of multiresolution Reeb spaces should not be matched. The following rules are introduced in order to maintain topological consistency. First, two nodes in a particular resolution of two different Reeb spaces can only be matched if they have same range value and their parents are matched. The later condition is not applied on the nodes of coarsest resolution of Reeb spaces, because they do not have any parent. The idea is that two nodes can only be matched if they belong to same part of the domain, which is ensured using the same range value.

Second, the two nodes in a particular resolution of two different Reeb spaces can only be matched if they have same Matching Label List (MLISTs). MLIST is a list of labels propagated to the node. This rule helps in avoiding the chance of matching the nodes in different branches. The MLISTs are maintained for each dimension or for each component field. For example, if we have a bivariate field then each node in the Reeb spcae will have two MLISTs, MLIST_1 for first dimension and MLIST_2 for second dimension. To create MLIST for each node, we extend our Reeb space to multi-dimension Reeb graph (MDRG). If a node in first dimension Reeb graph is matched then an matching label (MLABEL) say X is propagated to all the nodes in a direction of monotonic increase and decrease about the range value of each component field. This way, we can label the branch to which the node belongs. We store the MLABEL a matching list (MLIST) corresponding to each node. Once the nodes in first dimension are matched, the label(s) in the MLIST is propagated to all the nodes in the next dimension Reeb graph. Now similar to the first dimension matching, we match the nodes and propagate a new MLABEL in the direction of monotonic increase and decrease of range value to mark the branch with similar label. Hence the nodes in second dimension will have two MLIST. The label propagation is done from those nodes whoes up-degree and down-degree is one. If the up-degree (label propagation is done in upward direction) is more than one, then it is a case of ambiguity and hence the label not propagated further and vice versa. This case occurs when we reach a critical node, that is, at this node a single component splits or joins, a component is born or a component dies. This is illustrated using an example shown in fig FC3.2.

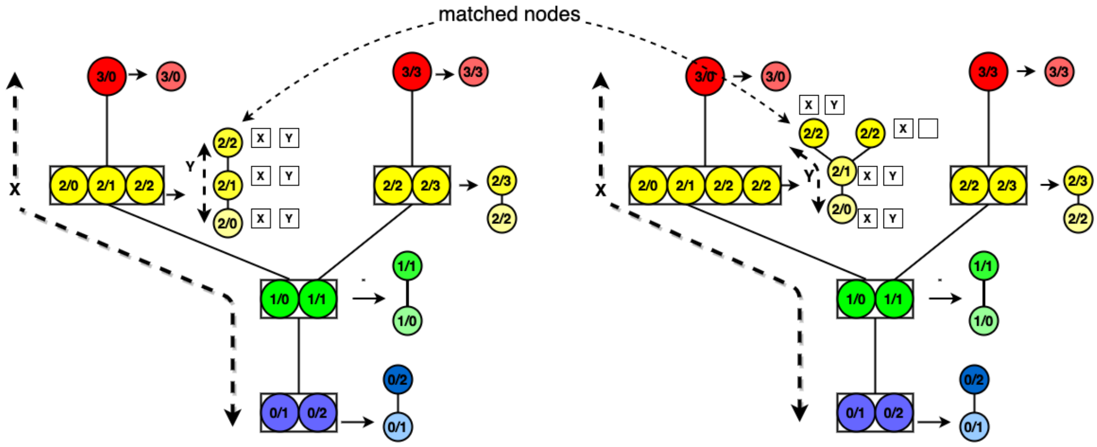


Figure FC3.2: Label propagation is demonstrated for a matching pair in two MDRGs. For a bivariate field, two lists of labels need to be maintained.

Let us consider a bivariate field at two different timestamps, and their corresponding MDRGs as shown in the Figure FC3.2. Suppose the nodes with value 2/2 are matched. We find the node in first dimension Reeb graph and mark the node with an MLABEL X. The label is then propagated in the direction of monotonic increase and monotonic decrease direction of the range value of the first component field. This way the MLIST of nodes in the branch are updated with MLABEL X. If an MLABEL passes a node which has either the up-degree or down-degree greater than one, then the MLABEL is not propagated further. Once all the nodes are assigned with their corresponding MLIST in first dimension Reeb graph or no more matching is possible, the nodes in the second dimension are matched. The MLIST in first dimension is passed to all the nodes in second dimension Reeb graph of the corresponding MDRG. Now each second dimension Reeb graph is treated separately and the above procedure is repeated again.

At the end, each node will consists of two MLISTs. If there are r fields then each node will have r MLISTs. Note that Figure FC3.2 is to demonstrate the label propagation and the MLISTs for all the nodes are not updated. We maintain MLISTs for the nodes which are not yet matched. For the nodes which are already matched, the MLISTs is unnecessary and need not be maintained.

Algorithm 3 IsTopologicallyConsistent(m, n)

```
Input: Nodes $m \in \text{MRS}_1$ and $n \in \text{MRS}_2$
Output: True/False
if range($m$) $\neq$ range($n$) then
    return false
end if
if !(matching(parent($m$), parent($n$))) then
    return false
end if
for each $d$ do
    if MLIST$_d(m) \neq$ MLIST$_d(n)$ then
        return false {d is the dimension of multifield}
    end if
end for
return true
```

3.5.3 Finding the Matching Node Pairs

In this subsection, we define how to find the MPAIRs or the matching node pair. Initially, a list of nodes (NLIST) is created corresponding to each multi-resolution Reeb space that need to be matched. We call these lists as NLIST$_1$ and NLIST$_2$. First, from NLIST$_1$ select a node which has maximum similarity with itself, that is $\text{sim}(\overline{m}, \overline{m})$. This makes us choose a node which affects the final similarity the maximum.

Second, select the node n from different multi-resolution Reeb space by applying the define topological consistency rules. This means, that the node n should have same range value as that of m, the parent node of n should match the parent node of m and the MLISTs of n should be same as that of MLISTs of m. The creation of MLISTs is described before in section 3.5.2. We call this node a candidate node.

Third, if there is no candidate node n that can match with m, then no MPAIR corresponding to m is created, m is removed from NLIST$_1$ and no MLABEL is propagated. The process is then repeated for other nodes in the NLIST$_1$. If there are more than one candidate nodes that can match with m, then we select a node which has maximum matching function value $\text{mat}(\overline{m},\overline{n})$ with m. This ensures that we always get a better match. The matching function is defined below and the equation is given in equation Eqn 3.7.

Finally, an MPAIR corresponding to nodes (m,n) is obtained. Then we remove node m,n from NLIST$_1$ and NLIST$_2$ respectively. An MLABEL corresponding to each of the branches of m and n are propagated and the MLIST is updated accordingly. Note that, once the nodes are matched the MLIST corresponding to these nodes are no longer maintained.

Definition of Matching Function. The matching function $\text{mat}(\overline{m},\overline{n})$ is calculated by using two aspects of the function. First, the loss $\text{loss}(\overline{m},\overline{n})$ representing the final decrease in similarity due to the matching of nodes m and n of an MPAIR. This loss is given by following equation:

$$\text{loss}(\overline{m},\overline{n}) = \frac{1}{2}\{\text{sim}(\overline{m},\overline{m}) + \text{sim}(\overline{n},\overline{n})\} - \text{sim}(\overline{m},\overline{n}) \qquad \text{(Eqn 3.5)}$$

The similarity decreases as the loss increases. Second, we define the matching function by using adjacent nodes as an attribute. The adjacent nodes are taken into account because they define the structure of Reeb space. The attribute of adjacent nodes is defined as:

$$\overline{\mathrm{adj}}(m) = \sum_{a \in \mathrm{adj([s,t),m)}} \overline{a} \qquad \text{(Eqn 3.6)}$$

Hence, the matching function is defined as:

$$\mathrm{mat}(\overline{m}, \overline{n}) = -\mathrm{loss}(\overline{m}, \overline{n}) - \sum_{[s,t)} \mathrm{loss}(\overline{\mathrm{adj}}([s,t), m), \overline{\mathrm{adj}}([s,t), n) \qquad \text{(Eqn 3.7)}$$

3.5.4 Node Attributes and Similarity Functions

The attribute $\overline{m}$ of a node m consists of an attribute $\deg(\mathrm{m})$ and e, where $\deg(\mathrm{m})$ is the degree of the node and e is the number of edges in a particular Reeb space. While, we can use other parameters such as volumetric area $a(m)$, we specifically use $\deg(\mathrm{m})$ and e because the degree of a node can be used to represent the local topology of the domain and dividing the degree with number of edges will result is a degree density. Hence the degree density $D(m)$ is given by:

$$\overline{m} = D(m) = \frac{\deg(\mathrm{m})}{2e} \qquad \text{(Eqn 3.8)}$$

The degree $\deg(\mathrm{m})$ is divided with $2e$ because each edge will be counted twice for each node degree.

As described in section 3.5.1 the parameter $D(m)$ is first calculated for the MPAIRs at the finest resolution Reeb space and then calculated for coarser resolutions.

Hence the final similarity between the nodes of an MPAIR $\{m, n\}$ is defined as:

$$\mathrm{sim}(\overline{m}, \overline{n}) = \min\left(\frac{\deg(\mathrm{m})}{2e_m}, \frac{deg(n)}{2e_n}\right) \qquad \text{(Eqn 3.9)}$$

where e_m is the number of edges in the Reeb space to which the node m belongs, e_n is the number of edges in the Reeb space to which the node n belongs and q is any positive real number. This implies that similarity of a node m with itself is given as follows:

$$\mathrm{sim}(\overline{m},\overline{m}) = \frac{\deg(\mathrm{m})}{2e} \tag{Eqn 3.10}$$

and the summation of all such similarity is equal to 1.

$$\sum_{m\in\mathrm{MRS}} \mathrm{sim}(\overline{m},\overline{m}) = \frac{\sum\deg(\mathrm{m})}{2e} = 1 \tag{Eqn 3.11}$$

3.6 Implementation

We calculate the similarity metric described in the previous section using Visualization Toolkit (VTK) [1] under the Joint Contour Net [9] and Multi-Dimensional Reeb Graph (MDRG) [12] implementation framework. The implementation works for a generic pair for multi-fields but is particularly designed for time-varying multi-fields. The range of two multi-field at two consecutive timestamps may vary slightly, which is fixed during the implementation by finding the maximum and minimum of both the fields and fixing the range for both the data. The range is fixed as the minimum of both the multi-fields and maximum of both multi-field of each component field. The implementation is for the multi-resolution data structure of multiple dimensions, and hence to create coarser resolutions, it is required to fix the number of slabs as dyadic. The slab width is calculated by finding the difference between the maximum and minimum range value of two data and then dividing by the number of slabs which is always a dyadic number. Next, we explain the two main steps required in our implementation.

I. **Computing MRS:** The multi-resolution Reeb space of a particular multi-field is created by using finest resolution JCN. The implementation of JCN is same as described in [9]. Then the number of slabwidth to divide the whole range into quantize ranges, are doubled to obtain the coarser resolution JCNs. The nodes of the coarser resolution JCNs are created by uniting the nodes falling in one coarser range into a single node. The edges are added between these coarser nodes based on their adjacency in its corresponding finer resolution. Corresponding to each node in each of the JCNs, following information is also maintained: its parent node and its corresponding child nodes.

II. **Similarity Calculation:** The similarity between two multi-resolution is calculated by first finding the MPAIRs between them. The MPAIR calculation is done from coarser to finer resolution. First the nodes of MRS_1 and MRS_2 are inserted in NLIST_1 and NLIST_2. Now one of the node is selected from NLIST_1 which has maximum similarity with itself. This similarity is calculated using the formula mentioned in equation Eqn 3.9. The selected node is named as chosen node. Now we select a node from NLIST_2 which is topologically consistent with chosen node. Two nodes are topologically consistent if their range values are same, their parent are also matched and their MLISTs are also same. There can be more than one node in NLIST_2 which are topologically consistent to chosen node. This ambiguity is removed by selecting a node amongst all the selected node which has highest matching with chosen node. The node obtained is named as candidate node. Once the nodes are obtained the MLISTs are updated. For this the corresponding MDRGs are obtained for the JCNs. We use the same implementation as described in [12]. The MLABEL in the MLISTs are propagated as described in section 3.5.2. After this the chosen node and the candidate node forms the MPAIR. Once the MPAIRs for all the nodes of each dimension are constructed, then the similarity for each MPAIR is calculated using the equation Eqn 3.9. The similarity of each MPAIR is summed and then divided by number of resolutions to obtain the similarity between two MRSs.

3.7 Applications

We now depict uses of the proposed MRS and the similarity metric with two different datasets,(i) Fission data of Fermium Atom (ii) Scission data of Plutonium Atom.

An atom splits into two or more parts during the process of nuclear fission. During the process some deformation of the atom nucleus takes place as the core of the atom is stretched. This distortion is important and can be used to identify the topological changes happening inside the atom. Identifying the exact timestamp where the atom has been split in an n-dimension manifold is an important problem as the number of components created after scission changes. The timestamp where the atom splits into two or more fragments is known as the nuclear scission. Previously this problem has been looked, and work has been done [14], but this a very time consuming and tiring task as the physicists have to look into each geometry of the atom for each timestamp. Later work has been done in [3] which again is used to handle the same problem. But in this work, since data is projected from domain to range space, some information might get lost and chances of losing important topological feature increase over the datasets. Our method stores the topological feature in the proposed data-structure and the similarity measure can be use to visualize these features by comparing two data structures for two consecutive timestamps.

3.7.1 Fermium Atom Data

The dataset of Fermium-256 atom consists of the nuclear densities that represent the heavy nucleus of the internal structure of an atom. The dataset consists of two spatial densities, namely, proton density (p) and neutron density (n). We aim to find the time stamp where the nuclear scission occurs. We experiment with sCF data [14] in which the fermium atom undergoes symmetric compact fission. The dataset consists of 40 timestamp (0-39) with a dimension of 19 × 19 × 19. We expect the major topological change at site 26, as this at this time step the nucleus splits into two atoms.

Observation and Results. We experiment with a number of resolutions for the fermium atom data. The range of values for spatial density of proton (**p**) is site 1 is 0.0 to 126.0 and the range of values for spatial density of neutron (**n**) is 0.0 to 139.0. This range may vary slightly for each time-step. The plots in Figure FC3.3 shows the similarity measure plot for the combination of, **p** (number of slabs 64) and **n** (number of slabs 64). We have chosen the slabs based in the experiments done in [3] and [14]. We observe a sudden change at time step 26 in Figure FC3.3 (c). We experiment with different values of q to check the results. Figure FC3.3 shows the plot for three different levels of resolutions and value of q is 1. We observe that as we the number of resolutions increases the depression in the plot is observed at site 26.

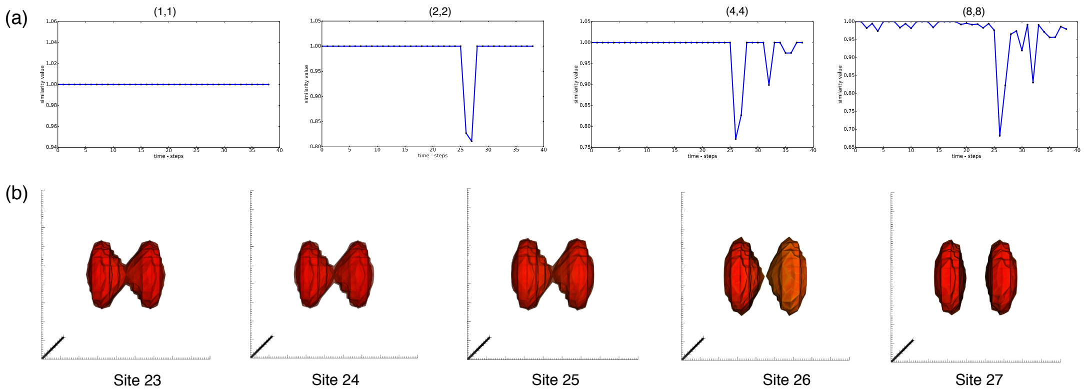


Figure FC3.3: Plots for similarity measure for Fermium atom data. **First row:** Left to Right are the similarity measure vs. the time step range [0-39] with number of resolutions as 1, 2, 3 and 4 respectively and q as 1. The proposed similarity measure exhibits a prominent dip at site 26, which indicates a significant change. **Bottom row:** Geometry of Fermium atom nucleus at different time-steps. The nucleus split at site 26 and can be seen in the geometry.

3.7.2 Plutonium Atom Data

The dataset consists of nuclear scission data of plutonium atom. The dataset is a multi-field data with two spatial densities i.e. proton density (**p**) and neutron density (**n**). The densities are sampled on a $40 \times 40 \times 60$ grid. The dataset if a negative log transformed sample at 14 different time steps, namely [665, 670, 675, 680, 686, 687, 688, 689, 690, 692, 693, 694, 695, 699]. The nuclear scission occurred between time step 690-692 and is reported in [14] and confirmed by the physicists.

Observation and Results. We experiment with number of resolutions for the plutonium atom data. The range of spatial density of proton (**p**) is 7.0 to 216.0 and the range of spatial density of neutron (**n**) is 6.0 to 214.0. These ranges may vary slightly for every time-step. The plots in Figure FC3.4 shows the similarity measure for the combination of, **p** (number of slabs 32) and **n** (number of slabs 128). These slabs are chosen carefully based on the experiments done in [3] and [14]. We experiment with different values of q and the result shown in Figure FC3.4(a) is with q value as 2000. We observe a sudden change between time-stamps 690-692 which shows that a sudden change has occurred at this time-stamp.

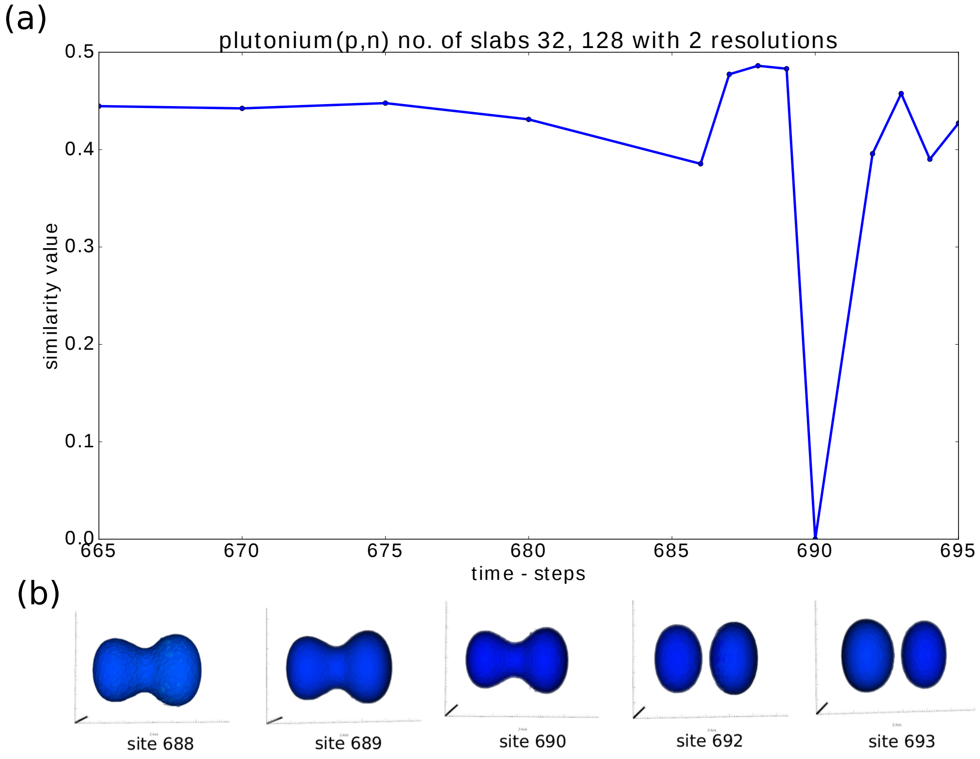


Figure FC3.4: Plots for similarity measure for Plutonium atom data. (a) is the similarity measure vs. the time step range [665-699] with number of resolutions as 2 and q as 2000. The proposed similarity measure exhibits a prominent dip between site 690-692 as the number of resolutions increases, which indicates a significant change. (b) Geometry of Plutonium atom nucleus at different time-steps. The nucleus split between site 690-692 and can be seen in the geometry.

CHAPTER 4

CONCLUSION AND FUTURE WORK

In our work we proposed two different approaches to find topological features in time-varying multi-field data. In the first method, we proposed a fiber component distribution as a feature descriptor for multivariate data. Then a novel method to extract topological features between time-varying multi-field data has been proposed, by introducing a distance measure between fiber-component distributions. The effectiveness of this method is shown by applying it on different synthetic and real data. The proposed distance measure can be used to find important timesteps and intervals and the features at such time stamps [3]. Next we proposed a data structure that can store topological information at different resolution for multi-field data. A similarity measure that can be used to extract topological feature has also been introduced that can make comparative analysis between the proposed data structure. The effectiveness of this method has again been tested on different real datasets. The proposed data structure can store topological features and the measure is able to extract the feature from time-varying multi-field data.

Despite the success of the two approaches, we still think there is a room for future improvement. The first approach captures essential changes in the range space, but it captures unimportant changes as well. These false changes are the main drawbacks of this method. For example, in the plot for the Pt-Co data, we observe additional peaks. To overcome this problem in future, we can find other distance measures between the

Reeb Spaces [3]. The Reeb space could be studied in a subsequent step for detailed analysis. The distance measure can also be computed for sub-domains, thereby allowing for finer-grained analysis. In the second approach, extensive use of Reeb space and its generalized algorithm JCN is made, for the construction of the data structure. The future work includes designing a data structure directly from the domain and not relying on Reeb space or JCN. Although the similarity measure between the data structure can capture the topological features, we would still like to explore more similarity measures to capture these features.

Bibliography

[1] `Visualization Toolkit`. http://www.vtk.org/, 2013.

[2] H. A. Gasteiger, N. Markoviċ, and P. Ross. H2 and co electrooxidation on well-characterized pt, ru, and pt-ru. 2. rotating disk electrode studies of co/h2 mixtures at 62 .degree.c. *The Journal of Physical Chemistry*, 99, 11 1995,. doi: 10.1021/j100045a042

[3] T. Agarwal, A. Chattopadhyay, and V. Natarajan. Topological feature search in time-varying multifield data. *arXiv preprint arXiv:1911.00687*, 2019.

[4] S. Bachthaler and D. Weiskopf. Continuous scatterplots. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1428–1435, Nov 2008. doi: 10.1109/TVCG.2008.119

[5] C. L. Bajaj, V. Pascucci, and D. R. Schikore. The contour spectrum. In *Proceedings of the 8th Conference on Visualization '97*, VIS '97, pp. 167–ff. IEEE Computer Society Press, Los Alamitos, CA, USA, 1997.

[6] U. Bauer, X. Ge, and Y. Wang. Measuring distance between reeb graphs. In *Proceedings of the thirtieth annual symposium on Computational geometry*, pp. 464–473, 2014.

[7] S. Biasotti, L. D. Floriani, B. Falcidieno, P. Frosini, D. Giorgi, C. Landi, L. Papaleo, and M. Spagnuolo. Describing shapes by geometrical-topological properties of real functions. *ACM Comput. Surv.*, 40:12:1–12:87, 2008.

[8] P.-T. Bremer, E. M. Bringa, M. Duchaineau, D. Laney, A. Mascarenhas, and V. Pascucci. Topological Feature Extraction and Tracking. *Journal of Physics: Conference Series*, 78:012007, 2007.

[9] H. Carr and D. Duke. Joint contour nets. *IEEE Transactions on Visualization and Computer Graphics*, 20(8):1100–1113, Aug 2014. doi: 10.1109/TVCG.2013.269

[10] 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, 2015. doi: 10.1111/cgf.12636

[11] H. Carr, J. Snoeyink, and U. Axen. Computing contour trees in all dimensions. *Computational Geometry*, 24(2):75–94, 2003.

[12] A. Chattopadhyay, H. Carr, D. Duke, and Z. Geng. Extracting Jacobi Structures in Reeb Spaces. In N. Elmqvist, M. Hlawitschka, and J. Kennedy, eds., *EuroVis - Short Papers*, pp. 1–4. The Eurographics Association, 2014. doi: 10.2312/eurovisshort.20141156

[13] A. Chattopadhyay, H. Carr, D. Duke, Z. Geng, and O. Saeki. Multivariate topology simplification. *Computational Geometry: Theory and Application*, 58:1–24, 2016.

[14] D. Duke, H. Carr, A. Knoll, N. Schunck, H. A. Nam, and A. Staszczak. Visualizing nuclear scission through a multifield extension of topological analysis. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2033–2040, Dec 2012. doi: 10.1109/TVCG.2012.287

[15] H. Edelsbrunner and J. Harer. Jacobi Sets of Multiple Morse Functions. *In Foundations of Computational Matematics, Minneapolis, 2002*, pp. 37–57, 2004. Cambridge Univ. Press, 2004.

[16] H. Edelsbrunner and J. Harer. *Computational topology: an introduction*. American Mathematical Soc., 2010.

[17] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. Local and Global Comparison of Continuous Functions. In *Proceedings of the conference on Visualization*, pp. 275–280, 2004.

[18] H. Edelsbrunner, J. Harer, and A. K. Patel. Reeb Spaces of Piecewise Linear Mappings. In *SoCG*, pp. 242–250, 2008.

[19] X. Gao, B. Xiao, D. Tao, and X. Li. A survey of graph edit distance. *Pattern Analysis and applications*, 13(1):113–129, 2010.

[20] C. Hamish, B. Duffy, and B. Denby. On histograms and isosurface statistics. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1259–1266, Sep. 2006. doi: 10.1109/TVCG.2006.168

[21] C. Hansen, M. Chen, C. Johnson, A. Kaufman, and H. Hagen, eds. *Scientific Visualization*. Mathematics and Visualization. Springer-Verlag London, London, 2014.

[22] G. H. Hardy, J. E. Littlewood, and G. Pólya. *Inequalities*. Cambridge: Cambridge University Press. ISBN 0-521-35880-9, second ed., 1952.

[23] M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii. Topology matching for fully automatic similarity estimation of 3d shapes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pp. 203–212, 2001.

[24] L. Huettenberger, C. Heine, H. Carr, G. Scheuermann, and C. Garth. Towards Multifield Scalar Topology Based on Pareto Optimality. *Computer Graphics Forum*, 32(3.3):341–350, 2013.

[25] G. Ji and H.-W. Shen. Feature tracking using earth mover's distance and global optimization. In *Pacific graphics*, vol. 2, 2006.

[26] I. Kendrick, D. Kumari, A. Yakaboski, N. Dimakis, and E. Smotkin. Elucidating the ionomer-electrified metal interface. *Journal of the American Chemical Society*, 132, 11 2010. doi: 10.1021/ja1081487

[27] Kitware, Inc. *The Visualization Toolkit User's Guide*, January 2003.

[28] G. Kresse, A. Gil, and P. Sautet. Significance of single-electron energies for the description of co on pt(111). *Phys. Rev. B*, 68, 08 2003. doi: 10.1103/PhysRevB.68.073401

[29] D. J. Lehmann and H. Theisel. Discontinuities in continuous scatter plots. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1291–1300, Nov 2010. doi: 10.1109/TVCG.2010.146

[30] Y. Matsumoto. *An introduction to Morse theory*, vol. 208. American Mathematical Soc., 2002.

[31] J. W. Milnor, M. Spivak, and R. Wells. *Morse theory*, vol. 1. Princeton university press Princeton, 1969.

[32] D. Morozov, K. Beketayev, and G. Weber. Interleaving distance between merge trees. *Discrete and Computational Geometry*, 49(22-45):52, 2013.

[33] V. Narayanan, D. M. Thomas, and V. Natarajan. Distance between extremum graphs. In *2015 IEEE Pacific Visualization Symposium (PacificVis)*, pp. 263–270, April 2015. doi: 10.1109/PACIFICVIS.2015.7156386

[34] Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99–121, 2000.

[35] O. Saeki. *Topology of Singular Fibers of Differentiable Maps*. Springer, 2004.

[36] O. Saeki, S. Takahashi, D. Sakurai, H.-Y. Wu, K. Kikuchi, H. Carr, D. Duke, and T. Yamamoto. *Visualizing Multivariate Data Using Singularity Theory*, vol. 1 of

Mathematics for Industry, chap. The Impact of Applications on Mathematics, pp. 51–65. Springer Japan, 2014.

[37] H. Saikia, H.-P. Seidel, and T. Weinkauf. Extended branch decomposition graphs: Structural comparison of scalar data. In *Computer Graphics Forum*, vol. 33, pp. 41–50. Wiley Online Library, 2014.

[38] D. W. Scott. A note on choice of bivariate histogram bin shape. *Journal of Official Statistics*, 4(1):47, 1988.

[39] G. A. Somorjai and Y. Li. *Introduction to Surface Chemistry and Catalysis*. John Wiley & Sons, 2010.

[40] R. Sridharamurthy, T. Bin Masood, A. Kamakshidasan, and V. Natarajan. Edit distance between merge trees. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2018. doi: 10.1109/TVCG.2018.2873612

[41] Teng-Yok Lee and Han-Wei Shen. Visualizing time-varying features with tac-based distance fields. In *2009 IEEE Pacific Visualization Symposium*, pp. 1–8, 2009. doi: 10.1109/PACIFICVIS.2009.4906831

[42] D. M. Thomas and V. Natarajan. Multiscale symmetry detection in scalar fields by clustering contours. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):151–165, 2014. doi: 10.1109/TVCG.2014.2346332