Geometrical Methods for the Analysis of Simulation Bundles

Dissertation
zur
Erlangung des Doktorgrades (Dr. rer. nat.)
der
Mathematisch-Naturwissenschaftlichen Fakultät
der
Rheinischen Friedrich-Wilhelms-Universität Bonn

vorgelegt von
Victor Rodrigo Iza Teran
aus
Quito, Ecuador

Bonn, September 2016

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der
Rheinischen Friedrich-Wilhelms-Universität Bonn

1. Gutachter: Prof. Dr. Jochen Garcke
2. Gutachter: Prof. Dr Martin Rumpf

Tag der Promotion: January 20, 2017
Erscheinungsjahr: 2017

Dedicated to Consuelo, Martin and Yasmin

Summary

Efficiently analyzing large amounts of high dimensional data derived from the simulation of industrial products is a challenge that is confronted in this thesis. For this purpose, simulations are considered as abstract objects and assumed to be living in lower dimensional space. The aim of this thesis is to characterize and analyze these simulations, this is done by examining two different approaches. Firstly, from the perspective of manifold learning using diffusion maps and demonstrating its application and merits; the inherent assumption of manifold learning is that high dimensional data can be considered to be located on a low dimensional abstract manifold. Unfortunately, this can not be verified in practical applications as it would require the existence of several thousand datasets, where in reality only a few hundred are available due to computational costs. To overcome these restrictions, a new way of characterizing the set of simulations is proposed where it is assumed that transformations send simulations to other simulations. Under this assumption, the theoretical framework of shape spaces can be applied wherein a quotient space of a pre-shape space (the space of simulations shapes) modulo a transformation group is used. It is propound to add into this setting, the construction of positive definite operators that are assumed invariant to specific transformations. They are built using only one simulation and as a consequence all other simulations can be projected to the eigen-basis of these operators. A new representation of all simulations is thus obtained based on the projection coefficients in a very much analogous way to the use of the Fourier transformation. The new representation is shown to be significantly reduced, depending on the smoothness of the data.

The new representation of a simulation set allows the introduction of a new simulation data analysis method whose properties will be detailed. The representation is also shown to improve the proper orthogonal decomposition (POD), which is a method that reduces the expensive evaluation of one simulation using a basis obtained from a few simulation snapshots. In the standard POD, the basis is very compact and reduces the size of the problems significantly, nevertheless, it has a strong dependence on the chosen snapshots in evaluating the basis. Our new approach overcomes this limitation and demonstrates this using a numerical example.

Several extensions to this new approach are also included. The first one shows a method to treat conformal transformations and demonstrate it with a given example. Next, an extension for computing paths in the space of simulation shapes is also considered. Paths in shape space are defined as paths along which a shape is transformed by preserving some (attribute) property such as an isometric one. A path with a minimal length is called geodesic, in this context, this thesis shows that the evaluation of geodesics in shape space can be considerably simplified using an invariant basis. An example for planar shape spaces, demonstrating its capabilities will be presented by comparing it with a state of the art approach for geodesic path computation.

Several industrial applications for time dependent datasets from engineering simulations are provided to demonstrate the usefulness of the method and put forward several research directions and possible new applications.

Acknowledgements

I particular I would like to thank Professor Dr. Jochen Garcke for his assistance in the realization of this thesis. His support and scientific advice have been very encouraging. I would also like to thank Professor Dr. Michael Griebel for the support provided allowing me to develop my thesis as a researcher at Fraunhofer SCAI. Also I would like to acknowledge the support of the Bonn International Graduate School and that of Mrs. Karen Bingel

Also my thanks go to all colleagues wo supported me during this time; Bernhard Klassen, Tanja Clees, Arnold Krechel, Kathrin Viertel and Ute Langer, to mention but a few. My special thanks goes to Ivonne Roche-Harth for proof reading of this thesis and for our interesting discussions on the text.

A very special thanks go to my wife Consuelo, my daughter Yasmin and my son Martin. They have always encouraged me to continue working on this thesis. Thank you Consuelo for your patience and your love. I would also like to thank my parents, for giving me tthe love and the opportunity of receiving a sound education. I left my home country several years ago highly motivated to learn mathematics that can be applied in an industrial context. This motivation will, as far as I can see, be maintained for the rest of my live. And I would like to thank the DAAD (Deutsche Akademisches Austausch Dienst). The scholarship to study mathematics in Germany was an incentive to follow a challenging and fascinating adventure. The fruits of this commitment can be discovered in this thesis.

Sankt Augustin, Germany
September 14, 2016

Rodrigo Iza Teran

Table of Contents

Table of Contents **vii**

1 Introduction **1**

2 Simulations **11**

2.1 Simulations as Solutions of Partial Differential Equations 11

2.1.1 Fidelity Models 12

2.1.2 Simulation Bundle and PDE Solution Space 14

2.1.3 Reduced Basis Methods 14

2.2 Simulations in the Product Development 16

2.2.1 Preprocessing 18

2.2.2 Solving 18

2.2.3 Post-processing 19

2.2.4 Simulation Data Management Systems 19

2.2.5 New Generation of Simulation Data Management Systems 20

3 Differential Geometry Notions **21**

3.1 Beyond the Euclidean Space 21

3.2 Differential Geometry 22

3.2.1 Manifolds and the Tangential Space 22

3.2.2 Vector Bundles, Vector Fields 24

3.2.3 Covariant Derivative and Riemannian Metric 26

3.2.4 Geodesics and Laplace-Beltrami Operator on a Manifold 27

3.2.5 Riemannian Curvature and the Second Fundamental Form 29

4 Simulations as a Nonlinear Manifold **31**

4.1 Introduction 31

4.2 Methods of Nonlinear Dimension Reduction 32

4.3 Diffusion Maps 33

4.3.1 Discrete to Continous Formulation: The Laplace-Beltrami Operator 34

4.4 Diffusion Maps for the Analysis of Simulations 37

4.4.1 Stochastic dynamical systems and slow variables in a simulation bundle 38

4.4.2 Slow variable for a simple simulation bundle 39

4.5 Diffusion Maps Methodology for a Simulation Bundle 42

4.6 Methodology 42

5 Quotient Spaces and Group Transformations 45
5.1 Preliminary Definitions . . . 46
5.2 Groups . . . 47
5.2.1 Lie Groups . . . 47
5.3 Quotient Spaces . . . 49
5.3.1 The Riemannian Structure of $\mathcal{M}/G$ . . . 51
5.4 Symmetric Spaces and Representation Theory . . . 53
5.4.1 Notions about Representation Theory . . . 55
5.4.2 Conclusions . . . 56

6 Simulations Space: Group Transformations and Invariant Operators 59
6.1 Simulation Space as Quotient Space over Isometries . . . 60
6.2 Geodesics on the Quotient Space . . . 62
6.3 Group Representations and Invariant Operators . . . 62
6.4 Invariant Operators . . . 63
6.4.1 Simulation Discrete Setting . . . 64
6.4.2 Approximating the Geodesic Distance on a Mesh . . . 65
6.4.3 The Laplace-Beltrami Operator . . . 68
6.4.4 The Laplace-Beltrami Operator on a Mesh . . . 69
6.4.5 The Laplace-Beltrami Operator under Isometric Transformations . . . 70
6.4.6 An Operator for Point Clouds . . . 73
6.4.7 Operator for Planar Shapes . . . 76
6.5 Numerical Evaluation of the Operators . . . 76
6.5.1 Evaluation of the Laplace-Beltrami operator . . . 78
6.5.2 Evaluation of the Operator for the Independent Components . . . 78
6.5.3 Evaluation of the Operator for Planar Shapes . . . 78
6.6 Approximating a Basis in Simulation Space . . . 79
6.7 Spectral Approximation of the Operators . . . 80
6.7.1 Structure of the Eigenfunctions . . . 81
6.7.2 The Discrete Case . . . 81
6.7.3 Discrete Orthogonal Decomposition . . . 82
6.8 Approximation Properties . . . 84

7 New Data Analysis Method 89
7.1 Simulation Data Analysis Method . . . 89
7.2 Analysis of the Building Blocks of the Approach . . . 95
7.3 ODE Bundles and Group Transformations . . . 96
7.4 Comparison with other related Methods. . . . 99
7.5 Operator Invariance Dependence . . . 101

8 Reduced Basis Methods with Invariant Basis 103
8.1 Reduced Basis and Parameter Identifiability . . . 103
8.2 POD Snapshots with an Invariant Basis . . . 104
8.3 Galerkin-POD with Invariant Operators. . . . 109

8.4 POD with an Invariant Basis. . . . 109
8.4.1 Reduced Order Model 110
8.4.2 The Invariant Operator. . . . 111

9 Extensions 117
9.1 Invariance to Conformal Transformations 117
9.1.1 Space of Concentric Spheres 119
9.2 Fast Evaluation of Paths in Shape Space and Simulation Space 121
9.2.1 Analysis on Shape Spaces 121
9.2.2 Shape Space as Quotient Space 122
9.2.3 Geodesics in the Shape Space of Planar Shapes 123
9.2.4 Orbits Paths as a Method for the Evaluation of Paths in Shape Space 124
9.2.5 Orbit Paths in Crash Simulation 129

10 Analysis of Simulation Bundles 133
10.1 Analysis of Simulation Bundles using Diffusion Maps 133
10.1.1 Metal Forming 134
10.2 Applications Analysis Simulation Bundles using Invariance 134
10.2.1 Comparative Analysis of Vibration Response Curves 134
10.2.2 Analysis in Crash Simulation 141
10.2.3 Operator based Approach 143
10.2.4 The Pre-image Problem in Crash-Simulation 148
10.2.5 Time Dependent Analysis of Crash Simulations 152

11 Summary and Open Research 161

Bibliography 163

CHAPTER 1

Introduction

Mathematics for Demanding Applications

Mathematics can be so astonishing and invaluable in confronting the many challenges involved in dealing with demanding applications as they provide an array of perspectives whose resources appear to be inexhaustible. Thus we will endeavor to convince the reader, of how maths can be applied to solve time consuming challenges by demonstrating and presenting one example. The original motivation was brought about by the restrictions involved in the analysis of thousands of simulations. Simulations are an essential tool in the development and improvement of new industrial products and are obtained during the virtual product development process. They are numerical solutions of mathematical models of physical processes such as structural deformation, vibrations or heat exchange. Specialized software tools are used to build extremely detailed models of a product defined on a 3D mesh. These models are then numerically solved using high-performance computer clusters.

A simulation of many product variants (a bundle) that implements specific geometrical changes, material parameters or load cases can thus be completed in a relatively short time. These changes are made by engineers in order to satisfy the functional constraints of the product as well as to fulfill safety regulations. This all contrasts with the time consuming decision process involved in realizing and evaluating those changes, since they are based on engineering judgment of complex 3D simulations; a very detailed mesh of the orders of million of nodes and elements evaluated at several hundred time steps, is a common representation of one simulation. Post-processing tools that actually visualize the model behavior or evaluate meta-data from one simulation are used in such evaluations. The timely development of new products as well as a lack of adequate post-processing tools for the evaluation of many simulations, impose enormous pressure on industries aiming for shorter development phases thus creating a serious bottleneck situation in the product development. The need to find ways of speeding up the analysis capabilities for the industry is, therefore, of the essence. To address these issues we use geometrical approaches which are enhanced by theoretical principles dealing with idealized spaces beyond Euclidean geometry. The principal assumption of data analysis methods, inspired by those principles, is that a high-dimensional dataset has a much lower intrinsic dimensionality and that ways can be found to identify and to represent such low dimensional structures. Use of geometrical approaches has already proved to provide ground breaking methods both for analyzing data in image processing; in medicine with many sound research results in those areas.

One simulation has a dimension which is of the order of the number of nodes times the number of time steps as given by the numerical discretization. That is of the order of several millions. Fortunately, in many cases, an intrinsic dimension is linked to each data set which is much lower; a low dimensional space can then be assumed to contain simulations as objects within this space. Research using this approach for data derived from industrial processes is scarcer than in other scientific disciplines. Our goal is to make use of mathematical principles and deal with ways of characterizing simulation data sets in lower dimensions to develop efficient data analysis methods.

Related Efforts

There are several related research procedures for the analysis of data from engineering simulations. A common approach for the evaluation of several simulations is the so-called sensitivity analysis. In its simplest form, one input variable is varied at a time, scalar output quantities of interest are evaluated for a few simulation results and sensitivity is measured by linear regression. Extensions of the approach are available in several ways, in particular in the form of response surface approaches [60], where the most relevant quantities of interest are trained as fitted functions of the input parameters, based on the results of a few numerical simulations. The underlying assumption is that the information content of all simulations can be concentrated in the scalar quantities of interest. This is also the major drawback of this approach, as it assumes that complex 3D overall behavior of the simulations can be reduced to simple scalar quantities. In contrast to sensitivity analysis we aim at looking at the data in more detail and work with the 3D simulation data. With this focus we group research efforts that deal directly with simulation data adding them to the category of machine learning and other research efforts that are, so far, not directly related to the analysis of simulations but are certainly relevant and should be considered. This includes the analysis of 3D shape geometry, harmonic analysis on graphs and shape space analysis that, in our opinion, play an important role in the methods discussed in this thesis.

Machine Learning

In the area of analysis of simulation bundles, several recent results that use dimensionality reduction are available. The principal assumption is the existence of a low dimensional manifold that represents a simulation bundle and one process that "learns" it from the available data.

One of the first studies analyzing engineering data from full numerical simulations was [Ackermann et al., 2008], where the authors used principal component analysis (PCA) for detecting important parameters from nonlinear finite element simulations. Car crash simulations were analyzed in [Mei and Thole, 2008] using clustering with a local distance measure. Regions of similar deformation behavior were identified and evaluated with this approach. The PCA has also been used in [Thole et al., 2010] for a group of simulations where some parameters were changed; the approach uses the first eigenvectors of the covariance matrix of all given simulations as deformation modes.

Methods based on the PCA are known to be efficient in many cases if the data variability is well reflected by the available data. The eigenvector basis from the PCA reflects those changes in an optimal way and uses it to show individual deformation modes which in turn are used for extrapolating new virtual simulations to show trends without calculating new time consuming simulations. While linear methods, based on the singular value decomposition (SVD), such as the PCA, have proven to be successful for industrial applications, it is nevertheless, known that if nonlinear correlations are present in the data, the use of PCA is not optimal, e.g. [Lee and Verleysen, 2007]. In [Bohn et al., 2013] several methods from machine learning have been used in the analysis of crash simulations. A pre-processing step using either k-means or spectral clustering reduced the original big sized problem to several clusters, which can be analyzed separately using local linear approximation based on tangent space estimation [Zhang et al., 2011], principal manifold learning with sparse grids [Feuersänger and Griebel, 2009], and diffusion maps [Coifman and Lafon, 2006]. In that study, good reconstruction capabilities using the nonlinear principal manifold approach was shown, as well as the detection of principal effects. In addition, a low dimensional representation of the simulations obtained using diffusion maps were shown to correlate with specific input variables.

In [Iza Teran, 2013] the application of diffusion maps in the analysis of engineering data in crash simulation were shown. Examples from vibration analysis and metal forming where also included. Furthermore, it has been shown that from a random variation of several input variables, the most important ones can be identified using clusters in a low dimensional representation.

We also mention recent efforts in machine learning to construct data-driven models for physical systems that incorporate domain knowledge exploiting invariance properties of such systems [Ling et al., 2016] or learn it from training data using deep convolutional neural networks [Farfade et al., 2015]. Those studies have shown the advantages and importance of using invariance properties to design more efficient methods for data analysis or for constructing models for physical systems.

Taking these research studies into account, one can observe that even if good progress has been achieved, there is still a need to develop methods that improve the analysis capabilities especially in the presence of nonlinear relationships in the simulation data.

Related Research

Analysis of 3D Geometry

The analysis of 3D shapes in computer graphics is an important field involving many links to our research study. In the area of 3D shape analysis, objects are represented discretely through a surface mesh embedded in $\mathbb{R}^3$. A number of methods have been developed for pose independent shape classification [Lai et al., 2010], other applications include shape retrieval [Reuter et al., 2006], shape segmentation [Reuter et al., 2009], invariant mesh representation [Lipman et al., 2005], and compression [Ben-chen and Gotsman, 2005], to name but a few. Many of these applications make use of the Laplace-Beltrami operator, a generalization of the Laplace operator for surfaces. There are also several works which use so called spectral descriptors like the heat kernel signature [Sun et al., 2009] and the wave kernel signature [Aubry et al., 2011]. They are based on the calculation of features that requires the calculation of the Laplace-Beltrami operator or the Schrödinger equation for each shape. This has found applications chiefly for pose independent processing, which is a particular requirement in 3D graphics for shape retrieval from archives and shape correspondence in the film industry. Pose changes in these applications are assumed to be isometric, i.e. geodesic distances in the shape are preserved. In this context, a method will not ideally distinguish between shapes with pose changes. By the way, this contrasts with what we would like to achieve, namely to distinguish between two isometric shapes since they are the result of a simulation with different input parameters.

We also mention [Ovsjanikov et al., 2012], where 3D shapes are compared based on the assumption of the existence of a-priori known transformation that morphs one shape into another. While this last approach is linked to ours, it is substantially different since it describes a shape using features which are then implemented for classification or pose independent shape matching.

Harmonic Analysis on Graphs

As will be seen, our approach is based on finding a common orthogonal basis representation for simulations defined on a mesh. In addition a mesh is a particular representation of a graph. Therefore developments in this area are relevant to our work. Wavelets on graphs [Coifman and Maggioni, 2006], [Hammond et al., 2011] and Treelets [Lee et al., 2008] can be used to find orthogonal multi-scale representations on graphs. Treelets, in contrast to wavelets can be used in cases where the functions to be represented are non-smooth. In all of these approaches, one construct from the data specific orthogonal basis that are then effectively used to represent the data in a sparse way with great advantages for classification tasks or further processing. Significant progress has been achieved in this area. Also more recent approaches [Thanou et al., 2014] use dictionaries as frames, they are representations allowing redundancies as opposed to orthogonal ones. The approach can represent datasets defined on graphs very sparsely.

On the theoretical side, important results [Auscher and Hyt 2013] have been achieved in ways of extending harmonic analysis on graphs. In this work, harmonic analysis on spaces of homogeneous type have been analyzed. These spaces are defined as ones where a quasi-metric is defined. Using it, meaningful notions of coarse parts and details of functions defined on a graph, can be studied.

Shape Space Analysis

In statistical shape analysis the idea is to consider geometrical information, in most of the cases as landmark points, and calculate some statistics on it such as finding a mean shape which also means eliminating location, scale and rotational effects. Mathematically speaking if those effects are eliminated, one can define so called homogeneous spaces or quotient spaces where the element of this space are actually equivalent classes which fulfill a specific property (for example being translation invariant). Actually this is the way differential geometry deals with abstract spaces and even extends the concept of differentiability into such spaces. Differential geometry is the study of invariant abstract objects with a mathematical structure. Transformations are studied abstractly using groups which can be given a differential structure (Lie-groups) under certain conditions. Group theory for the analysis of shape goes back to Felix Klein's Erlangen program that proclaims group theory to be the organizing principle of geometry. In simple words the idea is that, if we start with one bigger space, lets say a manifold one, we can take a subset of this space, made of elements such that a specific condition or invariance property is fulfilled. This subset forms a new space, a so called quotient space. The condition is actually expressed through the group action and if this is a Lie-group, then we have again a manifold with a geometrical structure dictated by the group action.

In the area of statistical shape analysis, statistics on shape manifolds, e.g. intrinsic and extrinsic means, are studied. Started by the work of Kendall [Kendall et al., 1999], this is a very active research area, recent work can be found in [Srivastava et al., 2005, Fletcher and Joshi, 2007], as well as in [Huckemann et al., 2010], [Rumpf and Wirth, 2011, Burghard et al., 2013] and references therein. Spaces with different structures, dependent on the chosen group action, can be created in shape space analysis. Specific manifold structures of the quotient space are assumed and are used. For example the so called 2D Kendall's spaces [Kendall et al., 1999] which are generally defined as quotient spaces of landmarks over the orthogonal group. Dealing with Kendall spaces with landmarks in 3D space is more complicated, since the action of the group is not free and the quotient space is no longer a manifold. A fundamental difference to methods applied in manifold learning is that in shape analysis a manifold structure of the shape space is assumed to be known and is thus kept fixed. Additionally, we would like to mention the mathematical work in the area of Riemannian geometries for plane curves and surfaces [Michor and Mumford, 2006, Bauer et al., 2011], those papers, and several others referenced therein, highlight the way to the definition of an abstract mathematical setting for those spaces that can be used for the analysis of the space of simulations.

Characterizing and Analyzing Simulation Datasets

Simulations are numerical solutions of differential equations on a discrete domain. From a mathematical point of view, they are abstract objects assumed living in a space which we would like to characterize. We have a set of simulations for which some parameters have been changed such as material properties, boundary or initial conditions or even the geometry of the domain. The goal, is to understand mathematically the space of simulations as precisely as possible under such parameter variations, using a low dimensional approximation of the original simulations.

The characterization of the space of simulations is done from two different perspectives in this thesis. From the perspective of manifold learning, we use diffusion maps and show their advantages and limitations for applications. Overcoming these limitations we will present a new way of characterizing the space of simulations from the perspective of transformations that send simulations to other simulations complemented by the theoretical framework of shape spaces. An invariant representation of a set of simulations is found that on the one hand allows the derivation of a new data analysis method and on the other hand enables an improvement of a reduced basis method specifically the proper orthogonal decomposition (POD), a method to simplify the numerical solution of partial differential equations.

Learning a Manifold of Simulations

A simulation is obtained as the numerical solution of a partial differential equation on a very detailed geometry, product variants correspond to changes in boundary conditions, material coefficients or local geometrical changes. Having enough datasets, one can learn a space of all those simulations using methods from machine learning. We use diffusion maps in this work and shown that its application already delivers interesting results for applications. The method builds a similarity, positive definite, matrix L of size $m \times m$, where m refers to the number of simulations. In the continuous limit of infinite many data samples and under certain conditions on the function defining the distance between simulations, the matrix operator L converges to a continuous operator on a low dimensional manifold intrinsic to the high dimensional data set. A representation of this space can be obtained using the principal eigenfunctions up to some specific order that can be estimated. In the discrete case one uses the principal eigenvectors to characterize this low dimensional space. Characterizing a simulations space with this method has the limitation however that many datasets are not available, only a few hundred, due to the computational cost. A second constraint of many manifold learning approaches, including diffusion maps is that the reconstruction of datasets unavailable in a training set, is cumbersome. These limitations are confronted under the assumption that simulations in a bundle set are obtained as transformations from a reference simulation in a quotient space setting.

Symmetry and Simulation Shape Space

Addressing the limitations of some machine learning methods, we use the property that simulations are actually numerical solutions of partial differential equations and therefore the principle of symmetry applies, which means that solutions can be considered to be obtained as transformations of a reference configuration. From this point of view a space of solutions can be obtained by defining so called orbits (paths traced when a specific transformation is applied to a reference simulation) along which a specific transformation occurs. The approach is inspired by symmetric principles to solve analytically differential equations. Unfortunately, such approaches cannot be extended in the case of numerical solutions of differential equations, since the underlying complex partial differential equations are not available and thus can only be solved numerically. We will show that under certain conditions the principles can be applied to those cases, that is, we will endeavor to find the way to represent simulations in an orbit space along a transformation group. This requires the use of the theoretical framework of shape spaces, which implies that a space is posed as if created from a so called pre-shape space that contains the set of all shapes modulo a transformation group. Let us mention the space of closed planar curves as an example for illustration. The pre-shape space in this case is the space of all embeddings of the curves in $\mathbb{R}^2$, a shape space in this case can be constructed by taking the quotient with respect to a transformation such as rotations, translations or re-parameterizations of the curves. The quotient space is then made of so called orbits that are the equivalence classes with respect to such transformations. According to this, simulation shape space represents the space of simulations modulo a transformation group. Complementary to this characterization, we will introduce the idea of constructing an orthogonal basis using a positive definite operator that is invariant to the transformation group. The method builds from one arbitrary simulation such an operator as a matrix L_b of size $d \times d$, where d refers to the number of nodes of the discretization. All simulations are thus projected onto this new basis. An equivalent representation of all simulations in the new basis is achieved in this way. Use of this new representation is observed to have several interesting properties. Firstly, the changes in the simulations are concentrated on fewer components of the orthogonal decomposition. This allows for a dimensionality reduction by using the coefficients for those fewer components. Secondly, a separation of effects is enabled. That is, for example, rotations, translations or different deformations modes for crash simulation, can be evaluated independently. Thirdly, the invariant basis is shown to be independent to parameter variations in the simulation so that new simulations can avail of the same basis for its representation for different parameter combinations. A method for analyzing many simulations efficiently in an industrial context can be implemented based on these properties.

Basis of an Invariant Operator and the POD

As mentioned before, the differential equations that are solved in industrial applications are normally not explicitly available. Nevertheless, for several physical processes it is already known which type of mathematical models or their combinations are involved in a solver;

- In structural mechanics, the elasticity equations are involved.
- In crash simulations the plasticity equations.
- For fluid flow simulations, the most general model involves a nonlinear partial differential equation (PDE) namely a Navier-Stokes equation.
- For the transfer of heat we have the convection diffusion equations.
- For vibration analysis often linear eigenvalue problems coming from second order differential equations are involved a so on.

For such cases, the use of a basis from an invariant operator is able to improve the method of reduced basis methods (RBO). Specifically, we concentrated on an extension to the proper orthogonal decomposition (POD). Given a PDE model together with a set of parameters and initial/boundary conditions, the objective is to find a reduced basis representation of a set of simulations. Specifically by using so called snapshots of a set of simulations for specific parameter combinations, the proper orthogonal decomposition (POD) constructs an orthogonal basis with the property that the simulations, have been attributed with, are represented under this new basis using very few basis components. Compared with the size of the simulation or the number of parameters that are varied, a strong dimensionality reduction is achieved. The focus of reduced basis methods of this type is the decomposition in linear subspaces of lower dimension. Projecting the simulations onto this orthogonal basis allows us to obtain an equivalent low dimensional representation of the high dimensional simulations. A simplification of the PDE by means of a Galerkin projection along the reduced basis is obtained. Unfortunately; the basis has a strong dependence on the chosen snapshots so that if the input parameter of the simulation changes, the basis will not be able to represent the new variations and therefore expensive simulations have to be performed to evaluate a new basis. In this thesis we propose to use the basis obtained from an invariant operator into the Galerkin projection, instead of the standard POD basis. It turns out that, the new basis allows for input parameter changes without the need to recompute a new basis.

Additional uses of the New Approach

We propose several other uses of the new concept developed in this thesis; an example of extending the use of invariant operators to other types of transformations has been developed. The specific example shows a way to consider conformal transformations.

The use of an invariant basis is also shown to enable the easy construction of paths in shape space. Starting from an initial and a final shape, a sequence of shapes can be evaluated along a path. The evaluation of minimal paths or geodesics in shape space is a research area where many efforts have been made to speed up its expensive computation. Using an invariant basis we are now able to show examples for planar shapes where such geodesic paths can be computed very easily and rapidly.

Contributions of our Research

The principal contributions of this thesis can be summarized as follows.

- We have shown the successful application of diffusion maps for the analysis of simulation datasets, highlighting capabilities and limitations. To evaluate its use, industrial application examples have been analyzed.

- A main contribution is a new method for the analysis of simulation datasets, based on a new mathematical characterization of such datasets. The approach can be interpreted as a method to estimate group actions on datasets using a link, not explored to our knowledge in this area, between an invariant space and an underlying positive definite operator from which an invariant basis is computed. A kind of dimensionality reduction for all simulations can be obtained along the different components of the orthogonal decomposition. In addition to having a low dimensional representation of a simulation bundle, we have also shown how a separation of effects such as rotations, translations or specific deformations can be obtained. Using such an effect separation, we have been able to construct so called synthetic or virtual simulations that are not obtained from expensive finite element computations. This separation of effects could have very important engineering applications.

- The presented new approach makes use of basis invariant to certain transformations. Using them we have proposed a new way of carrying out the proper orthogonal decomposition (POD) in the context of reduced basis representations. An exploration of the effectiveness of the new approach in the case of a nonlinear convection diffusion equation has been shown. A large range of parameter changes is used along which, the character of the equation is modified from a diffusion dominated to convection dominated one. The problem described is one of the most challenging in the context of reduced basis methods.

- An extension of the invariant basis approach consists on an experimental exploration of the applicability of the new methodology for evaluating geodesics paths in shape space. An example in planar shapes compares the new approach with a state-of-the-art method of geodesics calculation.

- We have proposed and shown how to include conformal invariant transformations, extending the case of isometric ones in a given example.

- A link of the new approach using invariant operators to analytical solution techniques, specifically to a Lie-group method has been demonstrated. An example of the solution of an ordinary differential equation for varying initial conditions is used to demonstrate it.

- In using the new approach as well as diffusion maps, detailed analysis of several industrial datasets has also been achieved.

- Finally, we have highlighted several extensions of the ideas and we have identified open research questions.

Structure of the Thesis

This thesis is structured in the following way. Chapter 2 provides an overview of what simulation bundles are from a mathematical and also from the application point of view. Chapter 3 consists of mathematical definitions and concepts that are used for treating simulations as low dimensional manifold in the context of diffusion maps. This chapter provides also background information that is used further in Chapter 5. Chapter 4 presents an overview of nonlinear manifold learning approaches. The method of diffusion maps is explained including how to use it for the analysis of simulation bundles. Its properties and its limitations for this application are explained . Chapter 5 introduces the abstract spaces we considered for introducing the next chapter; a new approach for the analysis of simulations. Quotient spaces under a group action and its Riemannian structure are considered. Also an overview of how geodesics in a quotient space are calculated is included as well as some notions about representation theory.

Chapter 6 characterizes the space of simulations as quotient space over isometric transformations. We further propose a way to represent transformations group actions by means of invariant operators and its eigenfunctions including an analysis of the discrete setting for two of them. The numerical evaluation of the operators is described and a discussion about the spectral approximation properties is included.

Chapter 7 introduces the new methodology for the analysis of simulations bundles, including its properties.

Chapter 8 propose an extension of reduced basis methods in which the basis is replaced by an invariant basis. An example for a nonlinear 1D partial differential equation (PDE) solved, using a Galerkin approximation for a range of parameters is included for comparison to the proper orthogonal decomposition standard approach.

Chapter 9 describes several extensions of the new method, an extension of the new approach for conformal transformations is included. How to calculate orbit paths and a comparison with geodesic paths in planar shapes is also included.

Chapter 10 describes several industrial applications. First an application of diffusion maps is described in order to illustrate it as a comparison method for the new approach, then several applications of the new method are included for vibration simulation and crash simulation comparing it with diffusion maps if appropriate.

Chapter 11 summarizes this research study highlighting several open problems in different areas and applications scenarios.

CHAPTER 2

Simulations

In this chapter we will describe the type of data we are interested in analyzing as well as how simulations are applied in an industrial environment. An important aspect to mention is that, in practice, we are dealing with a specific mathematical model of a product with boundary and initial conditions and that in the process of designing it, some physical parameters have been changed.

2.1 Simulations as Solutions of Partial Differential Equations

Let us use a specific model for clarifying the setting. Consider the following partial differential equation (PDE) which contains some physical parameters that can be changed,

$$\begin{aligned} -div(q \cdot \nabla u) + u \cdot \nabla u + \frac{\partial u}{\partial t} &= 0 \text{ in } \Omega \\ u &= g_p \text{ on } \Gamma_p \quad p = 1, \ldots, k \\ q\nabla \cdot n &= 0 \text{ on } \Gamma_2, \end{aligned}$$

the parameters are q and the values at the boundary g_p. For this example there are k + 1 parameters with parameter domain $\mathcal{P} \in \mathbb{R}^{k+1}$. The set of parameters can be denoted by $\boldsymbol{\mu} \in \mathcal{P}$. This is a simplified nonlinear model of a quantity u that changes its character from diffusion to convection depending on the value of q and in addition contains parameter dependent boundary conditions. The solution u can be seen as a map $u : \mathcal{P} \to V$ that to any $\boldsymbol{\mu} \in \mathcal{P}$ associates the solution $u(\boldsymbol{\mu})$ belonging to a suitable functional space V.

In a virtual product development environment, this model represents a physical process described by the quantity u. Examples of it are temperature, velocity, strain and so on. Engineers normally do not use the quantity u directly for industrial product decisions, but a derived quantity evaluated on part of the domain Ω_{obs} or on the boundary Γ. An example of such derived quantities are the following integrals,

$$z_1 = \int_{\Omega_{obs}} u(\boldsymbol{\mu}) d\Omega, \qquad z_2 = \int_{\Gamma} u(\boldsymbol{\mu}) d\Gamma,$$

this quantities are actually the ones used for engineering decisions or as a part of an optimization over the parameters or in a control system.

A further analysis goal in the product development is the parameter identification. That is the PDE parameters are varied until the solutions approximate, as accurate as possible, some real measurements from an experimental setup.

We describe the most general setting, the presentation from [Quarteroni et al., 2015] is followed closely here. In general abstract form, a solution of a PDE representing a mathematical model is sought. It can be written as: given $\boldsymbol{\mu} \in \mathcal{P}$, find $u(\mu) \in V$ such that

$$G(u(\boldsymbol{\mu}), \boldsymbol{\mu}) = 0 \quad \in V', \tag{2.1.1}$$

where $G : V \times \mathcal{P} \to V'$ is a parametrized mapping representing the nonlinear PDE.

Taking all solutions for all available parameters, one can talk about the set of solutions,

$$\varphi(\mathcal{P}) = \{u(\boldsymbol{\mu}) \in V : \boldsymbol{\mu} \in \mathcal{P} \subset \mathbb{R}^p\}, \tag{2.1.2}$$

where $\varphi : \mathcal{P} \to V, \ \ \boldsymbol{\mu} \mapsto u(\boldsymbol{\mu})$.

For solving each individual PDE, it is convenient to use the variational formulation. This involves testing the model equations by so called test functions, so that following problem is solved instead

$$g(u(\boldsymbol{\mu}); v; \boldsymbol{\mu}) = 0 \ \ \forall v \in V, \tag{2.1.3}$$

with $g(.;.;\boldsymbol{\mu}) : V \times V \to R$ being defined by $g(w; v; \boldsymbol{\mu}) = \langle G(w; \boldsymbol{\mu}), v\rangle \quad \forall w, v \in V$. It is assumed that the parametrized nonlinear problem is well posed (see [Quarteroni et al., 2015] for details about the conditions) so that we can write the partial Fréchet derivative of $g(z, .; \boldsymbol{\mu})$ with respect to u at z as

$$dg[z](w; v; \boldsymbol{\mu}) = \langle D_u G(z; \boldsymbol{\mu}) w, v\rangle \quad \forall v, w \in V,$$

whereby the mapping G is assumed to be continuously differentiable with partial Fréchet derivatives $D_u G(z, \boldsymbol{\mu}) : V \to V'$ and $D_{\boldsymbol{\mu}} G(z, \boldsymbol{\mu}) : \mathcal{P} \to V'$.

2.1.1 Fidelity Models

To solve numerically the model 2.1.3 involves the use of an approximation on a suitable finite dimensional subspace V_h of V. The so called fidelity model corresponding to 2.1.3 can be written as, find $u_h(\boldsymbol{\mu})$ such that

$$g(u_h(\boldsymbol{\mu}); v_h; \boldsymbol{\mu}) = 0 \quad \forall v_h \in V_h,$$

conditions for the discrete problem to be well posed can also be set (see [Quarteroni et al., 2015] for details). Notice that to solve the nonlinear problem, one has to solve iteratively a linearized problem. This is achieved through the use of Newton iterations, given $\boldsymbol{\mu} \in \mathcal{P}$ and an initial guess $u_h^0(\boldsymbol{\mu}) \in V_h$, for $k = 0, 1, \ldots$ until convergence, solve for $\delta u_h \in V_h$ using

$$dg[u_h^k(\boldsymbol{\mu})](\delta u_h, v_h; \boldsymbol{\mu}) = -g(u_h^k(\boldsymbol{\mu}); v_h; \boldsymbol{\mu}) \quad \forall v_h \in V_h$$

and inserting into $u_h^{k+1} = u_h^k + \delta u_h$.

Introducing a finite dimensional approximation $u_h(\boldsymbol{\mu}) = \sum_{j=1}^{N_h} u_h^j \mathcal{B}^j$ involves the use of a basis $\mathcal{B}^j$ for V_h. This basis is chosen in different ways but normally in industrial problems, the basis is given by linear/nonlinear functions with compact support. A very large number of basis functions, of the order of the number of elements, is required to achieve a given accuracy. As for the iterations in the Newton method, one has to solve correspondingly, very large linear systems at each iteration.

Finite Element Spaces

For industrial problems the finite element method is the de facto high fidelity approximation technique, so we will review here the principal aspects of the method.

Definition 2.1.1. Let Ω be an open set of $\mathbb{R}^d$ and k be a positive integer. A Sobolev space of order k on Ω is the space of all functions f of $L^2(\Omega)$ whose derivatives up to order k belong to $L^2(\Omega)$:

$$H^k(\Omega) = \left\{ f \in L^2(\Omega) : D^{\boldsymbol{\alpha}} f \in L^2(\Omega) \;\; \forall \boldsymbol{\alpha} : |\boldsymbol{\alpha}| \leq k \right\}.$$

Here $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ denote an n-tuple of non-negative integers so that $D^{\boldsymbol{\alpha}} f(\boldsymbol{x}) = \frac{\partial^{|\boldsymbol{\alpha}|} f(\boldsymbol{x})}{\partial x_1^{\alpha_1} \partial x_2^{\alpha_2} \ldots \partial x_n^{\alpha_n}}$, $|\boldsymbol{\alpha}| = \alpha_1 + \alpha_2 + \ldots + \alpha_n$.

In order to define approximations, that depends on a parameter h, a triangulation is defined for the domain Ω. For example in many industrial problems in product design, domains are surfaces embedded in $\mathbb{R}^3$. The triangulation is then a triangular mesh $\mathcal{T}_h$, covering the surface and consisting of non-overlapping triangles K. The diameter of K is defined as h_K and h is the maximum value of $h_K, K \in \mathcal{T}_h$.

A well known strategy to define finite element spaces is to consider continuous functions v_h that are polynomials of degree r on the single triangles, that is to define the space

$$X_h^r = \left\{ v_h \in C^0(\bar{\Omega}) : v_h|_K \in P_r \quad \forall K \in \mathcal{T}_h \right\}, \quad r = 1, 2, \ldots,$$

and is also possible to define $V_h = X_h^r \cap V$. Using element-wise (for each triangle) polynomials makes the calculation of the discretization terms easy. An arbitrary element of V_h is expressed using a basis that has non-empty intersections only with a small number of other functions on the basis, as a consequence a very sparse matrix is obtained.

In order to give an idea about the role of the discretization size of the mesh h_K in the approximation to the original PDE, consider a linear elliptic model, whose variational formulation can be written as: find $u \in V$ such that

$$a(u, v) = f_r(v) \;\; \forall v \in V,$$

$a(u, v)$ is a bilinear form and we use a finite element discretization of this continuous problem defined as: find $u_h \in V_h$ such that

$$a(u_h, v_h) = f_r(v_h) \;\; \forall v_h \in V_h.$$

In this setting the following error estimate can be defined (see [Quarteroni et al., 2015] for details).

Theorem 2.1.2. *Let $u \in V$ be the exact solution of the above variational problem and let $u_h \in V_h$ be its finite element approximation of degree r, where $V_h = X_h^r \cup V$. Then if $u \in V \cap H^{r+1}(\Omega)$, the following a* *priori error estimates hold:*

$$\|u - u_h\|_{H^1(\Omega)} \leq \frac{M}{\alpha} C \left(\sum_{K \in \mathcal{T}} h_K^{2r} |u|^2_{H^{r+1}(K)} \right)^{1/2},$$

$$\|u - u_h\|_{H^1(\Omega)} \leq \frac{M}{\alpha} C h^r |u|_{H^{r+1}(\Omega)},$$

C being a constant independent of h and u.

Note that in order to increase the accuracy, one can refine the size of the mesh or use finite elements of a higher degree. In industrial practice one uses polynomials of lower degree (up to 3) and a very fine mesh. For example an industrial car model contains several million nodes and elements. It is clear that handling many of these simulations for many parameter variations is a formidable task. The need for data reduction methods is of the essence.

2.1.2 Simulation Bundle and PDE Solution Space

A simulation bundle, as we have treated in this thesis, is a set of simulations usually underlying the same geometry and obtained as part of the product development cycle. Input parameters such as material properties or load conditions are changed, generating a simulation for each one of this input changes.

Formally a simulation bundle is considered as a discrete subset of a larger space that is obtained by considering all parameter variations $\boldsymbol{\mu} \in \mathcal{P}$. That is, we assume there is a PDE model which is solved to obtain a simulation, then the so called PDE solution manifold consists of all possible simulations, written as

$$\varphi(\mathcal{P}) = \{u(\boldsymbol{\mu}) \in V : \boldsymbol{\mu} \in \mathcal{P} \subset \mathbb{R}^p\},$$

where $\varphi : \mathcal{P} \to V, \quad \boldsymbol{\mu} \mapsto u(\boldsymbol{\mu})$ is the exact solution map. The corresponding high fidelity discrete solution manifold is

$$\varphi_h(\mathcal{P}) = \{u_h(\boldsymbol{\mu}) \in V_h : \boldsymbol{\mu} \in \mathcal{P}\} \subset V_h,$$

Note that calling it a solution manifold as is usually done in the PDE community is not justified from a mathematical point of view. Nevertheless we will see in section 6.1, that in some cases we can talk about a manifold. The high fidelity discrete solution manifold will be our study object along the following chapters, where we will consider methods to characterize it for data analysis purposes.

The analysis of the effect of parameter changes on the solution of a partial differential equation is cumbersome due to the size of the model, of the order of several millions degrees of freedom. In order to reduce them, so called reduced basis methods have been introduced.

2.1.3 Reduced Basis Methods

In order to be able to handle large simulations under the influence of many parameters, one needs a way to reduce the dimensionality of the data sets. Here one should distinguish two cases: firstly pure data based methods that include linear (PCA) as well as nonlinear dimension reduction methods and secondly the case where in addition to the solutions of the PDE (simulations), one knows or has access to the governing partial differential equations. For this cases, so called reduced basis methods have been developed.

The principle of reduced basis methods [Quarteroni et al., 2015] is based on the construction of suitable projections over a subspace V_N of V_h of reduced dimension $N \ll N_h$. The subspace is generated from a set of snapshots corresponding to solutions of the high fidelity problem for a set of parameters. One special way to do that is the proper orthogonal decomposition (POD) which is another way of referring to the principal component analysis (PCA).

Formally, using the fidelity model,

$$\langle G(u_h(\boldsymbol{\mu});\boldsymbol{\mu}), v_h\rangle = 0 \quad \forall v_h \in V_h, \tag{2.1.4}$$

the reduced basis formulation consist in finding $u_N(\boldsymbol{\mu}) \in V_N$, by enforcing orthogonality of the residual of 2.1.4 to N linearly independent functions of V_h that span a subspace W_N of dimension N, i.e. find $u_N(\boldsymbol{\mu}) \in V_n$ such that

$$\langle G(u_N(\boldsymbol{\mu});\boldsymbol{\mu}), w_N\rangle = 0 \quad \forall w_N \in W_N. \tag{2.1.5}$$

Considering all parameter variations $\boldsymbol{\mu} \in \mathcal{P}$, the general form of the solution set, the so called PDE solution manifold, can be written as

$$\varphi(\mathcal{P}) = \{u(\boldsymbol{\mu}) \in V : \boldsymbol{\mu} \in \mathcal{P} \subset \mathbb{R}^p\},$$

where $\varphi : \mathcal{P} \to V, \quad \boldsymbol{\mu} \mapsto u(\boldsymbol{\mu})$ is the exact solution map. The corresponding high fidelity discrete solution manifold is

$$\varphi_h(\mathcal{P}) = \{u_h(\boldsymbol{\mu}) \in V_h : \boldsymbol{\mu} \in \mathcal{P}\} \subset V_h,$$

the reduced basis approach tries to approximate the elements of φ_h by a linear global approximation under the separable form

$$u_N(x,\boldsymbol{\mu}) = \sum_{j=1}^{N} \alpha^j(\boldsymbol{\mu})\zeta_j(x) \quad \in V_N,$$

while keeping N as small as possible and satisfying

$$\inf_{v\in V_N} \|u_h(\boldsymbol{\mu}) - v\|_V < \epsilon \text{ for all } \boldsymbol{\mu} \in \mathcal{P}.$$

For computations one would like to know the decay rate at which N is achieved. This will depend on the differentiability of the solution map with respect to the parameter, the number of parameters, as well as the nature of the problem.

Finally note that once N has been found, solving 2.1.5 consists in solving a set of N nonlinear equations for every $\boldsymbol{\mu} \in \mathcal{P}$. This is high problem dependent and involves a trade-off between accuracy and computational complexity (since building the nonlinear reduced terms involves computations that depends on the much bigger size N_h). In addition the reduced problem projected using the Galerkin method is not necessarily well posed.

The Proper Orthogonal Decomposition for PDE solutions

The POD (Proper Orthogonal Decomposition) is one of the most applied approaches not only for reducing the dimensionality of the data set, but also in the context of reduced basis approximations. It is a simple scheme that at the same time delivers very good results even in the presence of nonlinearities provided the simulations snapshots adequately sample the parameter space.

Consider a set $P_s = \left\{\boldsymbol{\mu}^1, \boldsymbol{\mu}^2, ..., \boldsymbol{\mu}^{n_s}\right\}$ of n_s parameter samples and the corresponding snapshots

$$\left\{u_h(\boldsymbol{\mu}^1), u_h(\boldsymbol{\mu}^2), ..., u_h(\boldsymbol{\mu}^{n_s})\right\},$$

obtained from the PDE discrete fidelity approximation. The snapshot matrix can be build as $M_s = [\boldsymbol{u}_1|...|\boldsymbol{u}_{n_s}]$, where the vectors $\boldsymbol{u}_i \in \mathbb{R}^{N_h}, 1 \leq i \leq n_s$ are given by $\boldsymbol{u}_i^{(j)} = u_h^{(j)}(\boldsymbol{\mu}^i)$, for $1 \leq i \leq n_s$ and $1 \leq j \leq N_h$.

Given the SVD decomposition of the matrix $M_s = USV^T$, for any $N \leq ns$, the POD basis of dimension N is defined as the set of the first N left singular vectors $\boldsymbol{\zeta}_1, ..., \boldsymbol{\zeta}_N$ of U. Explicitly this basis can be written as

$$\boldsymbol{\zeta}_j = \frac{1}{\sigma_j} M_s \boldsymbol{\psi}_j \quad 1 \leq j \leq N,$$

where σ_j is obtained from $S = diag(\sigma_1, ..., \sigma_r)$ $(\sigma_1 \geq \sigma_2 \geq \sigma_r, r \leq min(N_h, n_s))$ and $\boldsymbol{\psi}_j$ are obtained from $V = [\boldsymbol{\psi}_1|...|\boldsymbol{\psi}_{n_s}]$.

By construction, the POD basis is orthogonal and it minimizes over all possible $N-$dimensional orthonormal basis, the sum of the errors of each snapshot vector $\boldsymbol{u}_i$ and its projection over the orthonormal basis.

In industrial applications the POD has been successfully used and provides adequate solutions provided the variations of a set of simulations is adequately described by the sampling set.

In the following section, we will concentrate on the application setting and what industrial simulations are.

2.2 Simulations in the Product Development

For this thesis we have considered simulations as numerical solutions of differential equations and in particular we deal with such solutions used in the design of industrial products. For the solution of such differential equations, a numerical model is first defined which contains a geometry as a discretization of a domain where the solution is to be found. The model also contains a way to define material parameters as well as boundary and initial conditions used for the solution step.

Simulations are obtained during the virtual product development where mathematical models of physical processes are solved numerically. The models could have a very high level of geometric detail (see figure 2.1 for a car model in crash simulation) that includes information about the materials characterization. This is represented as a function and many times it is the result of a nonlinear regression evaluated using real measurements. A model also contains information about connections elements to join structural components such as welding joints. In addition, the applied loads to the model are characterized as precisely as possible.

Bundles of simulations are generated by solving the PDE many times for specific load cases or changes in material properties, geometry or connections. The way those changes are performed requires engineering knowledge, it is an iterative time consuming process where the performance of the model is improved in several steps. The final goal is to satisfy safety regulations but at the same time fulfilling cost constraints and manufacture-ability.

Model examples are the equations of elasticity, plasticity, fluid dynamics, structural vibrations, heat transport, flow in porous media, radiation transport, chemical transport an so on. They are in general nonlinear, so that they can not be solved analytically except for very simple non application relevant cases. A discretization of the geometrical domain is performed using lamped masses, the finite difference method, the finite element method, finite volume or free mesh methods. To give an example how fine a standard geometric discretization can be, a mesh in car crash simulation is of the order of 3-4 million nodes with similar amount of elements whereas the one for fluid dynamics is of the order of 30 million nodes or more. Computing those models requires the use of a high performance computer cluster that includes thousand of processors. In such clusters one can evaluate hundred variants of a model overnight.

The evaluation of each model requires the use of 3D visualization software where the results of one simulation are animated and/or evaluated according to the deformations, pressure, velocity, acceleration, temperature and so on (see figure 2.1 for an example of the 3D visualization).

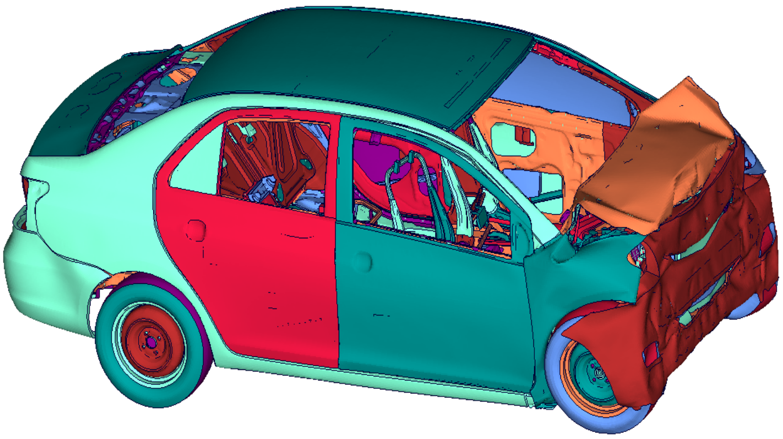


Figure 2.1: Simulation of car crash deformations.

In an industrial context, the process of model generation and evaluation of simulation bundles contains several phases (see figure 2.2). Preprocessing, solving and post-processing. The process is used for different types of so called disciplines in the car industry like frontal crash, lateral crash, or passengers safety. In other industrial areas, different types of physical problems are involved like structural analysis, vibration analysis or computational fluid dynamics including different load case variations.

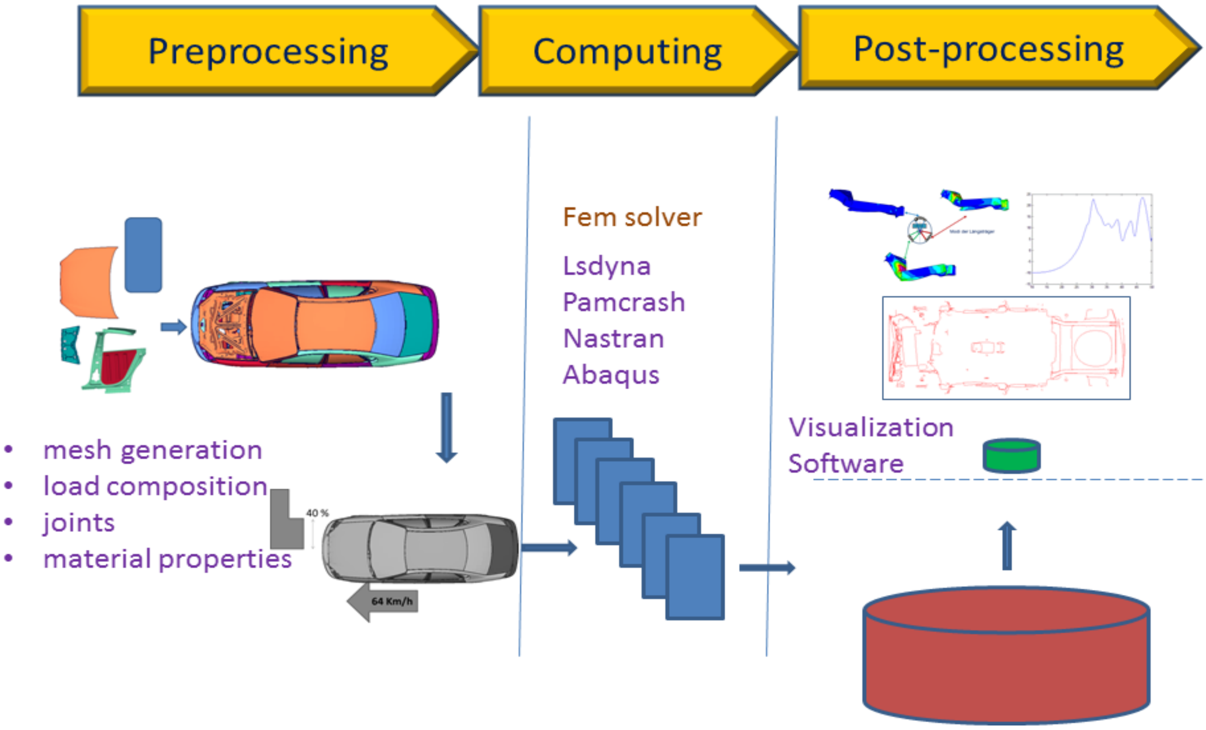


Figure 2.2: Schematic industrial product development process

2.2.1 Preprocessing

This phase starts with the CAD information about the car design, or with some modules defining several components that it can be built up into a complex system. Material or component data has to be specified and assigned as well as the imposed load cases and constraints and this is done through so called preprocessing software that contains meshing routines that helps engineers in the creation of a finite element mesh or routines that automatically configure several components into one calculable system. Also for some models a division into multiple parts is necessary with required material information but also with information about the type of connections between them (as welding joints, kinematic joints and so on). The type of loads and its distribution is also added to the model with the assistance of software tools. This process is time consuming and the final goal is the creation of so called input-decks, the input files for the solvers.

2.2.2 Solving

Once input files for the solvers are available, a batch process can be initiated on a computer cluster where the solver is started. For crash simulation there are different solvers like Pamcrash, Lsdyna and Radioss. For other applications such as structural mechanics there is Ansys or Abacus. For vibration the most popular is MSC-Nastran. For fluid dynamics calculation Star-CD is used as well as OpenFoam. These are just examples, there are also many others commercial software according to the application involved.

2.2.3 Post-processing

Once the solutions are obtained, an evaluation of the solved quantities is performed by the engineer, this evaluation uses specially developed 3D visualization software that is used to show graphically, time dependent deformations, temperature fields, stresses and so on. The level of detail allows for a realistic modeling of specific processes that can also be animated in such post-processing software. In addition to the visualization analysis, engineers make use of predefined special quantities evaluated from simulations that are subject to legal regulations and are a part of a standardization. Examples from such quantities are the HIC index (Head Injury Index) which evaluates the severity of deceleration on the neck of a dummy (a simplified model of a human body) involved in a car crash. Another example is the intrusion that consists on the evaluation of the deformations at critical points in a car structure (like for example in the firewall) for a specific time step that corresponds to the maximal deformation time. Based on such quantities, the actual design process takes place.

Predefined quantities evaluated in the post-processing step are scalar or vector quantities derived from the simulation results, many of them are used as acceptance criteria for products in industry (like the HIC index). Notice that, the use of post-processing quantities implies a reduction of the dimensionality of the problem. The use of those quantities also allows a faster design evaluation since those are scalar or vector quantities but with few components which are meant to replace the time consuming evaluation of detailed geometrical deformations, velocities or accelerations. Nevertheless, how representative those quantities are with respect to the simulations results is actually an open problem.

2.2.4 Simulation Data Management Systems

In car companies the multidisciplinary product development process is assisted by simulation data management systems (SDM) that are in charge of retrieval, versioning and administration of all models, simulations, output quantities and evaluation reports. The product development process demands the creation and evaluation of several thousand models in different productions phases; from concept design up to detail evaluation and optimization. Model evaluation requires extensive engineering judgment and is time intensive. Due to this complexity, car companies use, for all phases mentioned above, specially developed simulation data management systems (SDM) in their product development that allows the control of the type of changes involved in the different development phases, enabling a more rapid model evaluation of best design practices.

In spite of the development of such systems, engineering processes demands more analysis capabilities to achieve economical or safer designs or to being able to reuse previous model construction strategies. An SDM system is not flexible in the sense that once the post-processing quantities are selected at specific time steps at certain fixed locations, the evaluation of those quantities in other positions or at different time steps demand running all the simulations once more, since all results are not saved for further analysis. More flexible SDM systems have to be developed that precisely allows for such, on demand analysis, with reduced computing and storage costs. In the next section we would like to characterize the requirements for such a system to cope with current demands or to enable new types of efficient data analysis capabilities.

2.2.5 New Generation of Simulation Data Management Systems

A flexible and efficient SDM system should have the following features:

- use highly detailed simulation output results as objects for doing database requests or data analysis, instead of using simplified scalar quantities,
- extract simulation outputs or post-processing values at arbitrary time steps without having to run all simulations again,
- predict intermediate time steps not available in the calculated simulations,
- compare simulation results from other designs which do not have the same geometry,
- learn from previous designs and be able to recover such information when needed.

A system with all the above mentioned features do not exists for the industry at the moment. There are nevertheless intensive research that focuses on some of this features. Specially the use of so called "Reduced Models" is today an essential tool in industry. One of the most used methods for finding reduced models is the proper orthogonal decomposition (POD). The method is effective if, the snapshots used to construct the basis cover all the parameter variations adequately. This is actually a difficult task due to the presence of nonlinearities and bifurcations. There are also technological barriers at various levels like data handling, storage and efficient access. Some features such as the prediction of simulations at intermediate steps or learning from previous designs are still a big challenge for the industry.

In this thesis we will approach precisely those challenges using mathematics and propose a new analysis method that contributes to the development of new efficient engineering analysis methods and efficient simulation data managament (SDM) systems.

CHAPTER 3

Differential Geometry Notions

In this thesis, we would like to characterize the space of simulations as an abstract space and be able to define a distance between those simulation objects. Inherent to this characterization is the notion of geometry and we would like to think of simulations as abstract objects living in an abstract space where we can also define a distance between them. In mathematics this can be done using the concept of a manifold (the abstract space) with a metric (to being able to measure distances), further we need also a generalization of differentiability and how to evaluate distances on such spaces. In mathematics differential geometry enables precisely the geometrical characterization of a space of abstract objects. In particular we will consider Riemannian geometry, which studies differentiable manifolds with a Riemannian metric as a distance construct. A generalization of the treatment of spaces beyond the Euclidean space is achieved in this way to general spaces that locally resemble Euclidean space.

The definitions and material in differential geometry given here are standard and can be found in any book on this topic (e.g [do Carmo, 1992, Michor, 2008]).

3.1 Beyond the Euclidean Space

In 1854, Bernhard Riemann gave his professorship lecture in Göttingen with the difficult task of doing this for an audience with no knowledge of his area of study (differential geometry), actually only Gauss was aware of this work. So it was really challenging for him to start talking about the fact that the most familiar concept known as Euclidean geometry, was just one of many possible types of geometries. Actually the concept of geometry itself could be made completely abstract meaning a relationship between abstract objects where one can define a distance to locate them and quantify such relationships through a metric. That approach let us define spaces with different types of objects so that, one can for example, go beyond points in Euclidean space and define points that are for example images, in a space where each point represents an image or perhaps more simply, to use a spherical space where two parallel lines can be traced over the surface, they will cross at some point whereas in the euclidean case, they will never cross each other. Differential geometry studies precisely such settings where abstract objects can be represented in a space.

3.2 Differential Geometry

3.2.1 Manifolds and the Tangential Space

To build abstract spaces, first one needs to clarify what a topological space is. Imagine you have a set of objects $\mathcal{M}$ (images, solutions of pdes, musical scores and so on) and you would like to think of them as elements of an abstract space. This can actually be done, if one can define a structure where subsets $\mathcal{T}$ of your set satisfy so called mathematical axioms (a rule or a statement that is accepted as true without proof), namely: a) that the set X itself and the empty set are members of $\mathcal{T}$, (b) that the intersection of any finite number of sets in $\mathcal{T}$ is in $\mathcal{T}$, and (c) that the union of any collection of sets in $\mathcal{T}$ is in $\mathcal{T}$. The sets in $\mathcal{T}$ fulfilling those axioms are called open sets. Certainly to continue describing such an abstract space you need a separability property in order to be able to discern between elements of such space. That is why one introduces a so called Hausdorff property that enables precisely that.

Definition 3.2.1. A topological space is called **Hausdorff** if for any two points $p, q \in \mathcal{M}, p \neq q$, there exists disjoint open sets U_p and U_q such that $p \in U_p$ and $q \in U_q$ with $U_p \cap U_q = \emptyset$.

To describe the open sets one needs a basis B of the topology defined as a collection of open sets in $\mathcal{T}$, such that every open set can be written as a countable union of elements of the basis. The last thing that one needs to characterize an abstract space is to give it, at least locally, a Euclidean geometry. Just think about the surface of the Earth, for us it looks flat (Euclidean) and this is approximately true only locally.

Definition 3.2.2. An ***n*-dimensional topological manifold** is a topological space that fulfills:

- $\mathcal{M}$ is Hausdorff,
- $\mathcal{M}$ has a countable basis of the topology,
- $\mathcal{M}$ is locally homeomorph to $\mathbb{R}^n$.

Defining abstract spaces in this way provides a completely general structure to work with, but you still can define much more. After all, in Euclidean space the concept of derivative is very useful in practice (calculate gradients, velocity, acceleration and so on), but is this possible in an abstract manifold space?, well the answer turns out to be yes!. For doing that, one needs to define differentiability and a way of doing that is by using the concept of local homeomorphisms. They imply the existence of local mappings from the manifold to the Euclidean space and are more generally known as manifold charts.

Definition 3.2.3. Let $\mathcal{M}$ be a topological manifold and let $f : \mathcal{M} \to R$ be continuous. Let $(U, \mathfrak{x}), (V, \mathfrak{y})$ be charts around $p \in \mathcal{M}$, then we can write,

$$f \circ \mathfrak{y}^{-1} = (f \circ \mathfrak{x}^{-1})(\mathfrak{x} \circ \mathfrak{y}^{-1})$$

In order to use a generalization of the concept of differentiability of $\mathcal{M}$, one needs the differentiability of the charts composition $f \circ \mathfrak{y}^{-1}$

Definition 3.2.4. Two **charts** $(\mathfrak{x}, U)$ and $(\mathfrak{y}, V)$ are called C^∞**−compatible** if the chart composition

$$\mathfrak{x} \circ \mathfrak{x}^{-1} : \mathfrak{y}(U \cap V) \to \mathfrak{x}(U \cap V)$$

is a C^∞ **diffeomorphism** (a differentiable continuous mapping where the inverse is also differentiable).

In order to avoid the ambiguity of having a chart dependency by such a definition, one speaks about an atlas of charts with the above property and from that, one can create a maximal atlas as the set containing all such atlases. Thus one can define a differentiable manifold.

Definition 3.2.5. An ***n*-dimensional differentiable manifold** is an n-dimensional topological manifold together with a maximal C^∞ atlas.

Very useful in applications in addition to differentiability is the approximation of a differentiable transformation between manifolds. In Euclidean space one can write formally that, given $f : \mathbb{R}^n \to \mathbb{R}^m$ differentiable, the differential df at x is a linear approximation of f using $df_x = J_x(f)$ with $f(x + v) = f(x) + df_x(v) + \phi(v)$ and $\lim_{v \to 0} \phi(v) \| v \| = 0$ where $J_x(v)$ is the Jacobi matrix from f, the partial derivatives from f are the images of the basis vectors in $\mathbb{R}^n$ under df_x.

To introduce the same concept for an abstract manifold, one can define for each point p on the manifold an abstract vector space $T_p\mathcal{M}$ called the tangential space at p and the linear map $df_p : T_p\mathcal{M} \to T_{f(p)}N$, the differential of f at p. As before in order to fully define a differential for a manifold, one needs to introduce charts on p where such differentiability is defined. This is done through equivalent classes. Two curves $\alpha,\beta : (-\epsilon, \epsilon) \to \mathcal{M}$ with $\alpha(0) = p = \beta(0)$ are called equivalent, $\alpha \sim \beta$ if for one and then for all charts (U, x) on p: $(x \dot{\circ} \alpha)(0) = (x \dot{\circ} \beta)(0)$, then one can define the tangential vectors on M at p.

Definition 3.2.6. The tangential vector on $\mathcal{M}$ at point $p \in \mathcal{M}$ are the equivalent classes with respect to $\sim$ of curves in $\mathcal{M}$ through p. The **tangential space** from $\mathcal{M}$ in p is the set of all such equivalent classes:

$$T_p\mathcal{M} = \{\alpha : (-\epsilon, \epsilon) \to \mathcal{M} \text{ differentiable}, \epsilon > 0, \alpha(0) = p\} \,/\, \sim .$$

For the equivalent classes of a curve, one uses the notation $[\alpha] = \dot{\alpha}(0) = \frac{d}{dt}\big|_{t=0} \alpha(t)$. The disjunctive union of all tangential spaces is called **tangential bundle**: $T\mathcal{M} := \bigcup_{p \in \mathcal{M}} T_p\mathcal{M}$.

Now the differential for the manifold is defined as

Definition 3.2.7. Let $f : \mathcal{M} \to \mathcal{N}$ be a differential map, then the **differential** of f in p is defined as follows

$$df_p : T_p\mathcal{M} \to T_{f(p)}\mathcal{N}, \quad [\alpha] \mapsto [f \circ \alpha].$$

Definition 3.2.8. Let $\mathcal{M}$, $\mathcal{N}$ be differentiable manifolds and $f : \mathcal{M} \to \mathcal{N}$ be a differentiable map between them. The map f is a **submersion** at a point $p \in \mathcal{M}$ if its differential is a surjective map.

Notice that once a differential map can be defined, it is natural to ask how to define the space of differentiable functions on a manifold. In the analogy to defining tangential space, one can define it through equivalent classes.

Definition 3.2.9. Let f, h be differentiable functions defined on an open neighborhood U from p. **Equivalent classes** from functions are defined through,

$$f \sim h \leftrightarrow \exists V \subset U \quad \text{with: } V \text{ open}, \quad f_{|V} = h_{|V}$$

Definition 3.2.10. Germs of differentiable functions in p are equivalent classes with respect to the above definition, so that $C_p^\infty(\mathcal{M})$ = space of germs in p

Definition 3.2.11. A **derivation** on $C_p^\infty(\mathcal{M})$ is a linear mapping $v : C_p^\infty(\mathcal{M}) \to R$ with,

$$v([f][g]) = v([f])g(p) + f(p)v([g]).$$

The set of derivations on $C_p(\mathcal{M})$ is denoted as $D_p(\mathcal{M})$

Definition 3.2.12. Let $v \in T_p\mathcal{M}$ and let $f \in C^\infty(\mathcal{M})$, the **Lie derivative** of f in the direction v is defined as:

$$L_v(f) = df_p(v) = \frac{d}{dt}\bigg|_{t=0} f(\alpha(t)),$$

where $\alpha : (-\epsilon, \epsilon) \to \mathcal{M}$ is a smooth curve in $\mathcal{M}$ with $\alpha(0) = p$ and $\dot{\alpha}(0) = v$.

The tangential bundle is an example of a more general abstract object; called a vector bundle.

3.2.2 Vector Bundles, Vector Fields

Definition 3.2.13. Let E and B be differential manifolds and let $\pi : E \to B$ be a differentiable mapping. A **vector bundle** is a triple (E, π, B) with:

- the mapping π is surjective
- there exists an open covering $\{U_i\}_{i \in I}$ from B and diffeomorphisms

  $$h_i : \pi^{-1}(U_i) \to U_i \times \mathbb{R}^n$$

  with

  - $h_i(\pi^{-1}(x)) = \{x\} \times \mathbb{R}^n$, $x \in U_i$
  - $h_i \circ h_j^{-1} : (U_i \cap U_j) \times \mathbb{R}^n \to (U_i \cap U_j) \times \mathbb{R}^n$

Definition 3.2.14. A differentiable mapping $s : B \to E$ is called **section** of a vector bundle if $\pi \circ s = Id$.

Definition 3.2.15. A **vector field** is a section in a tangential bundle and therefore a differentiable mapping $X : \mathcal{M} \to T\mathcal{M}$ with

$$\pi(X(p)) = p$$

for all $p \in \mathcal{M}$.

As defined, tangential vectors can be identified with derivations on function germs, in an analogous way it is possible to identify vector fields with derivations on the space $C^\infty(\mathcal{M})$ of smooth functions on $\mathcal{M}$. The identification is given by the Lie derivative. For a vector field X one write $L_X(f)(p) := df_p(X_p) = X_p(f)$.

Definition 3.2.16. Let X, Y be vector fields on $\mathcal{M}$, the **Lie bracket** $[X, Y]$ is defined by $L_{[X,Y]} = L_X \circ L_Y - L_Y \circ L_X$. The **Lie derivative** from Y to X is defined as $L_X Y := [X, Y]$

The space of vector fields over a manifold $\mathcal{M}$ together with the Lie bracket is an infinite dimensional Lie-alegebra.

Definition 3.2.17. Let $f : M \to N$ a diffeomorphism, the image of the vector field X under f is a vector field on N defined through

$$(f_* X)_q := df_{f^{-1}(q)}(X_{f^{-1}(q)}).$$

Let X be a vector field on $\mathcal{M}$ and let (U, x) be a chart around p in $\mathcal{M}$. X, restricted to U can be written as:

$$X|_U = \sum_{i=1}^{n} a_i \cdot \frac{\partial}{\partial x_i}$$

for smooth functions $a_i : U \to R$, one search a smooth curves $\gamma : (-\epsilon, \epsilon) \to \mathcal{M}$ with

- $\gamma(0) = p$
- $\dot{\gamma}(t) = X_{\gamma(t)} \quad \forall t \in (-\epsilon, \epsilon)$.

Definition 3.2.18. A curve $\gamma : (-\epsilon, \epsilon) \to \mathcal{M}$ is called **integral curve** of the vector field X through p if

- $\gamma(0) = p$
- $\dot{\gamma}(t) = X_{\gamma(t)}$ for all $t \in (-\epsilon, \epsilon)$.

Theorem 3.2.19. *For all $p \in \mathcal{M}$ there exists an interval I_p around zero and a uniquely defined curve $\gamma_p : I_p \to \mathcal{M}$ with*

$$\gamma_p(0) = p \quad \textit{and} \quad \dot{\gamma}_p(t) = X_{\gamma_p(t)}, \quad \forall t \in I_p$$

Theorem 3.2.20. *For all $p \in \mathcal{M}$ there exists an open neighborhood U from p and an interval I around zero, such that for all $q \in U$ the curve γ_q on I is defined. The mapping*

$$I \times U \to \mathcal{M}, \quad (t, q) \mapsto \gamma_q(t)$$

is differentiable.

Definition 3.2.21. The mapping $(t, q) \mapsto \gamma_q(t)$ is called the **local flow** of the vector field X. The integral curves of X are also called **flow lines** from X. The flow defines for small enough parameter t a local mapping $\varphi_t : U \subset M \to M$ through $\varphi_t(q) := \gamma_q(t)$.

Definition 3.2.22. The mappings φ_t are local diffeomorphisms and it fulfills

$$\varphi_t \circ \varphi_s = \varphi_{t+s}$$

for all parameter t, s for which φ_t, φ_s, φ_{t+s} are defined. The set $\{\varphi_t\}$ is called a 1-parameter group of local diffeomorphisms.

Definition 3.2.23. A **vector field** along a curve $\gamma : I \to \mathcal{M}$ is a smooth mapping $X : I \to TM$ with $X(t) \in T_{\gamma(t)}M$ for all $t \in I$. The space of all vector fields along γ is written as $\mathcal{X}(\gamma)$.

3.2.3 Covariant Derivative and Riemannian Metric

The covariant derivative is a way of specifying a derivative along vector fields of a manifold. Being able to define them is important for defining paths of minimal length on a manifold as we will see.

Definition 3.2.24. A **covariant derivative** or **connection** from Y to X on $\mathcal{M}$ is a mapping,

$$\nabla : \mathcal{X} \times \mathcal{X} \to \mathcal{X}$$

where $Y, X \in \mathcal{X}$ with the following properties:

- $\nabla_X Y$ is $C^\infty(\mathcal{M})$-linear in X
- $\nabla_X Y$ is $\mathbb{R}$-linear in Y
- $\nabla_X(f \cdot Y) = X(f) \cdot Y + f \cdot \nabla_X Y$

Now we would like to describe infinitesimal changes of a vector field along a curve.

Theorem 3.2.25. *Let $\mathcal{M}$ be a manifold with covariant derivative ∇, then there exists a unique mapping*

$$\frac{\nabla}{dt} : \mathcal{X}(\gamma) \to \mathcal{X}(\gamma)$$

with the following properties:

- $\frac{\nabla}{dt}$ *is $\mathbb{R}$-linear*
- $\frac{\nabla}{dt}(f \cdot X) = \frac{df}{dt} \cdot X + f \cdot \frac{\nabla}{dt} X$
- $\frac{\nabla}{dt}(X_\gamma) = \nabla_{\dot{\gamma}(t)} X$

the mapping describe the infinitesimal changes of X along γ.

Definition 3.2.26. The **torsion** of a connection ∇ on $T\mathcal{M}$ is defined as the mapping

$$T : \mathcal{X} \times \mathcal{X}, \quad (X, Y) \mapsto \nabla_X Y - \nabla_Y X - [X, Y].$$

A connection is called torsion free if $T \equiv 0$.

Working with manifolds, turns out to be very useful in practice if a metric can be defined on them.

Definition 3.2.27. Let $\mathrm{Sym}^2 T_p\mathcal{M} := \left\{ b : T_p\mathcal{M} \times T_p\mathcal{M} \to \mathbb{R} \quad | \text{ symmetric bilinear form} \right\}$.

Definition 3.2.28. Let $\mathrm{Sym}^2 T\mathcal{M} := \bigcup_{p \in \mathcal{M}} \mathrm{Sym}^2 T_p\mathcal{M}$ (a vector bundle over $\mathcal{M}$).

Definition 3.2.29. A **Riemannian metric** on M is a smooth section $g \in \Gamma(\mathrm{Sym}^2 T\mathcal{M})$ with g_p positive definitive for all $p \in \mathcal{M}$.

Definition 3.2.30. Let $(\mathcal{M}, g)$ be a Riemannian manifold, a connection ∇ is called **metric** with respect to g if for all vector fields $X,Y,Z \in \mathcal{X}$, the following is fulfilled

$$X(g(Y,Z)) = g(\nabla_X Y, Z) + g(Y, \nabla_X Z).$$

Theorem 3.2.31. *On a Riemannian manifold there exist a uniquely defined connection that is torsion free and metric. This connection is called* ***Levi-Civita****.*

3.2.4 Geodesics and Laplace-Beltrami Operator on a Manifold

Once a covariant derivative along a curve can be defined, a geodesic on a manifold can be defined. Notice that geodesics for a manifold generalizes the use of straight lines on the plane traveled at constant speed.

Definition 3.2.32. A curve $\gamma : I \to \mathcal{M}$ is called a **geodesic** if,

$$\frac{\nabla}{dt}\dot{\gamma} = 0$$

Definition 3.2.33. Let $\gamma : [a,b] \to \mathcal{M}$ be a smooth curve, then the length of γ is defined by

$$L[\gamma] = \int_a^b ||\dot{\gamma}(t)||dt \qquad ||\dot{\gamma}(t)|| = \sqrt{g(\dot{\gamma}(t), \dot{\gamma}(t))}$$

the length of a piecewise smooth curve is the sum of the lengths of the smooth pieces. A Riemannian metric define a distance on a manifold.

Definition 3.2.34. Let $p, q \in \mathcal{M}$, the distance between p and q is defined through

$$d(p,q) := inf\left\{L[\gamma] \mid \gamma : \text{piecewise } C^\infty - \text{curve from } p \text{ to } q\right\}.$$

Geodesics are critical points of the function length L.

Theorem 3.2.35. *Let $\gamma : [a,b] \to \mathcal{M}$ be a curve parametrized by arc length, i.e. $||\dot{\gamma}|| = 1$ and let $\gamma_s(t)$ be a variation of γ i.e a smooth mapping*

$$(-\epsilon, \epsilon) \times [a,b] \to \mathcal{M}, \quad (s,t) \mapsto \gamma_s(t)$$

with $\gamma_0 = \gamma$. Setting $\delta = \frac{\partial}{\partial s}|_{s=0}$, then the following is valid

$$\delta L(\gamma_s) = g(\delta\gamma_s, \dot{\gamma})|_a^b - \int_a^b g(\delta\gamma_s, \frac{\nabla}{dt}\dot{\gamma})dt$$

Lemma 3.2.36. *A curve $\gamma[a,b] \to \mathcal{M}$ parametrized by arc length with $\gamma(a) = p$ and $\gamma(b) = q$ is then a geodesic if and only if, for all variations γ_s with $\gamma_s(a) = p$ and $\gamma_s(b) = q$ $\forall s$, the following is valid,*

$$\delta L(\gamma_s) = 0,$$

that is, if γ a geodesic $\Rightarrow \ddot{\gamma} = 0 \to \delta_L(\gamma_s) = 0$.

Definition 3.2.37. Let $v \in T_p\mathcal{M}$ be a tangent vector to the manifold at p, the **exponential map** is defined as $Exp_p(v) = \gamma_v(1)$, where γ_v is the unique geodesic satisfying $\gamma_v(0) = p$ with initial tangent vector $\dot{\gamma_v}(0) = v$.

Given a Riemannian manifold $(\mathcal{M}, g)$ and a function $f \in C^\infty$ over $\mathcal{M}$, the differential map $d_p\phi : T_p\mathcal{M} \to R$ is linear for all $p \in \mathcal{M}$. There exists a vector field in $T\mathcal{M}$ (the tangential bundle) called the **gradient** of f, $\nabla_g f$ such that,

$$< \nabla_g f(p), X_p >_{g(p)}= d_p\phi(X_p) \quad \forall X_p \in T_p\mathcal{M}.$$

Given a Riemannian manifold $\mathcal{M}$, $X \in \mathcal{X}(\mathcal{M})$ = space of vector fields, $X : \mathcal{M} \to T\mathcal{M}$, C^∞ = space of functions over $\mathcal{M}$, $f : \mathcal{M} \to R$ a **differential form** of grad k on $\mathcal{M}$ is an alternating k-linear mapping over C^∞:

$$w : \mathcal{X}(\mathcal{M}) \times \ldots \mathcal{X}(\mathcal{M}) \to C^\infty(\mathcal{M}).$$

The space of differential forms of grad k is denoted by $\Omega^k(\mathcal{M})$.

Definition 3.2.38. Let X be a vector field on $\mathcal{M}$, the **divergence** of X is defined by

$$div(X) = \sum_i g(\nabla_{e_i}X, e_i),$$

where e_i is a local orthonormal basis from $T_p\mathcal{M}$.

On a manifold we can also talk about operators acting on functions on it. A fundamental one is the **Laplace-Beltrami operator** defined as:

$$\Delta_g : C^\infty \to C^\infty$$

$$\Delta_g = -div_g \cdot \nabla_g.$$

On the manifold $(\mathcal{M}, g)$ we further consider the eigenvalue problem for a given operator as defined by $\Delta_\mathcal{M}\psi = -\lambda\psi$ where λ is an eigenvalue of $\Delta_\mathcal{M}$ and ψ, are the corresponding eigenfunctions. The Laplace-Beltrami operator is positive semi-definite [Rosenberg, 1997] and therefore, all eigenvalues λ_k, $k \geq 0$ are real positive and isolated with finite multiplicity. The set of eigenfunctions $\{\psi_i\}$ of the operator forms an orthogonal base for functions on $\mathcal{M}$, the following decomposition can be written for $f \in C^\infty(\mathcal{M})$,

$$f = \sum_{i=0}^{\infty} \alpha^i\psi_i, \alpha^i =< f, \psi_i > . \tag{3.2.1}$$

For any function f as defined by (3.2.1) one can instead of considering the function itself, equivalently consider the vector $\alpha = [\alpha^1, \alpha^2, \ldots]$ of spectral coefficients obtained by projecting the function along the infinite dimensional eigen-space spanned by the eigenfunctions.

3.2.5 Riemannian Curvature and the Second Fundamental Form

Definition 3.2.39. Let $(\mathcal{M}, g)$ be a Riemannian manifold with Levi-Civita connection, the **Riemannian Curvature** is the mapping

$$R : \mathcal{X}(\mathcal{M}) \times \mathcal{X}(\mathcal{M}) \times \mathcal{X}(\mathcal{M}) \to \mathcal{X}(\mathcal{M}), \quad (X, Y, Z) \mapsto R_{X,Y}Z$$

for the vector fields X, Y, Z defined as

$$R_{X,Y}Z = \nabla_X \nabla_Y Z - \nabla_Y \nabla_X Z - \nabla_{[X,Y]}.$$

Definition 3.2.40. The scalar curvature is defined as

$$scal_g = \sum_{i,j} R(e_i, e_j, e_j, e_i)$$

whereby an orthogonal basis of $T_p\mathcal{M}$ is given by $X = e_n$, $X^\perp = span\{e_1, \ldots, e_{n-1}\}$.

Definition 3.2.41. Let $\mathcal{M} \subset \bar{\mathcal{M}}$ be a Riemannian sub-manifold. The **second fundamental form** is defined by

$$\Pi(X, Y) = (\bar{\nabla}_{\bar{X}} \bar{Y})^\perp,$$

The second fundamental form is symmetric. An important case is the case of hyper-surfaces, in this case choosing a normal vector N, the second fundamental form can be viewed as a $\mathbb{R}-$valued bilinear form

$$A(X, Y) = \left\langle \bar{\nabla}_X Y, N \right\rangle = -\left\langle Y, \bar{\nabla}_X N \right\rangle,$$

for every $p \in \mathcal{M}$ a basis of eigenvectors of the second fundamental form $e_1, \ldots e_n \in T_p\mathcal{M}$ (n is the dimension of $\mathcal{M}$) can be found so that

$$A_p(e_i, e_j) = \lambda_i \delta_{ij}$$

The eigenvalues λ_i are called the **principal curvatures**. Important for us will be the **mean curvature** defined as $tr(A)$ and the **Gauss curvature** defined as $det(A)$.

The mathematical framework introduced here is the one which actually allows us to treat the abstract space of objects, which in our case are simulations. We will discuss in the following chapter a way to obtain a representation of a space starting with a set of simulations. Here one assumes that they are high dimensional objects embedded in a low dimensional space.

CHAPTER 4

Simulations as a Nonlinear Manifold

4.1 Introduction

In this chapter we will describe a mathematical method for the data analysis of a set of numerical solutions of a parametric partial differential equation (PDE). The method is one of the methods applied in the area of manifold learning. It assumes that the data lies on a manifold whose intrinsic dimension is much smaller than the dimension of the fidelity simulations and that the manifold can be recovered approximately. The underlying PDE model itself is not available. Different methods to find such reduced spaces are available, we will present one representative approach in this thesis.

In chapter 2 we presented a description of simulations as numerical solutions of a PDE. In industry several of them are evaluated for different input design parameters as part of the product development process. The set of simulations obtained in this way can be formally written as

$$\varphi_h(\mathcal{P}) = \{u_h(\boldsymbol{\mu}) \in V_h : \boldsymbol{\mu} \in \mathcal{P}\} \subset V_h,$$

obtained by sampling for different parameter combinations in $\mathcal{P}$. In this chapter we will treat the problem of finding a low dimensional representation of the mapping $\varphi_h(\mathcal{P})$ using simulation samples.

We treat simulations $u_h(\boldsymbol{\mu})$ as high dimensional objects in some Euclidean space $\mathbb{R}^d$. An isometric embedding $\varphi_s : \mathcal{M} \to \mathbb{R}^d$ has brought the data to this space from a low dimensional manifold $\mathcal{M}$ ($dim(\mathcal{M}) = M$ with $M \ll d$). Our data objects are the simulations, where the dimensionality is considered to be given as the number of nodes multiplied, by the number of time steps. That gives an order up to several millions.

The objective is to characterize and find the low dimensional object $\mathcal{M}$ in order to represent simulations. We will describe how this can be done in this chapter; after giving an overview of methods of dimension reduction, one method from manifold learning called diffusion maps is chosen. We will explain its principles and its approximation properties in the continuous limit case of many simulations. A justification of the use of this method for simulation data is also included in this chapter. Parts of this chapter has been published in [Iza Teran, 2013].

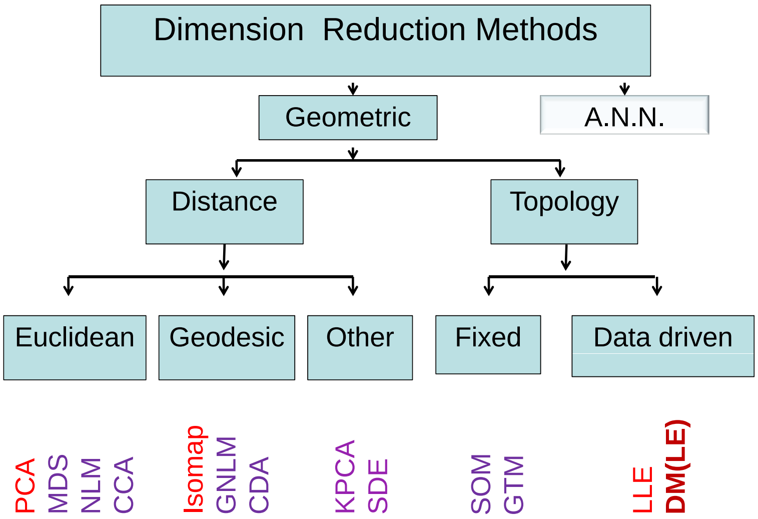


Figure 4.1: Nonlinear Dimension Reduction Methods (based on the classification as given in [Lee and Verleysen, 2007])

4.2 Methods of Nonlinear Dimension Reduction

An overview of some nonlinear dimension reduction methods, that can be useful for the analysis of simulation results, will be given. For details of the different methods see [Lee and Verleysen, 2007].

Nonlinear dimensionality reduction methods assume that the information available from a dataset has implicitly a lower dimension and that this lower dimensional structure can be identified. A natural step in the process of identifying such structures is to embed the data into a lower dimensional space. By using this process some information loss is unavoidable. According to the criteria used to evaluate this loss, it is possible to characterize different methods (see figure 4.1).

We will concentrate on geometry preserving methods that are of different types as can be seen in figure 4.1. Other types of methods such as artificial neural networks (ANN) will not be considered in this study.

Distance preserving methods try to preserve everywhere the distance between the information in the original dataset and in the embedding. If one uses a Euclidean metric, several methods can again be defined, according to how pairwise distance preservation is defined. Traditional principal component analysis (PCA) can be found in this category by its equivalence to multidimensional scaling (MDS).

Other types of methods are called topology preserving methods. Such methods do not aim at preserving pairwise distances globally; a similarity function which is usually a decaying function of the pairwise distances is used instead. As a consequence distance is preserved locally and the focus is more toward preserving relationships such us being near to, similar to; or different from. It means that in the embedding, there is some information loss, but these relationships are preserved in the lower dimensional representation. Inside this classification, one can find methods such us diffusion maps (DM), laplacian eigenmaps (LE) and local linear embeddings (LLE).

Engineers can use both types of methods for extracting low dimensional patterns or structures from several simulation datasets. To show the possible usefulness of such low dimensional representation, let us consider as in illustration a car crash simulation; engineers are interested in designing a car whereby the deformation inside the cabin is as low as possible whereas most of the time design objectives are in concurrence. For example the objective could be to minimize weight and at the same time to maintain structural integrity. In the product development an engineer has to evaluate many designs where several design parameters have been altered; common questions that have to be confronted are: which designs are similar to the optimal one, are they still feasible? which designs are similar for different parameter variations?, can an alternative design be found that lies in between two designs but with fewer costs?. In this thesis; we will treat topology based nonlinear dimension reduction methods to confront such engineering problems and answer those questions. Using dimensionality reduction makes those engineering problems much more manageable.

We would like also to mention that distance preserving methods can and have also been used in this context [Mei and Thole, 2008, Thole et al., 2010]. Which method to use will depend on specific requirements of engineering analysis that at the same time impose requirements on a dimensionality reduction method.

Requirements for the analysis of engineering data:

- the datasets are very big (million from points and hundred of variables for each point),
- the physical problems that are simulated are strongly nonlinear,
- it should be possible to extract some patterns from thousands of such datasets,
- engineers are interested in relative correspondences in the data, as explained in the last section.

The requirements above, justify the use of a nonlinear dimensionality reduction methods for engineering data. There are several topological based methods. From these we will consider diffusion maps.

4.3 Diffusion Maps

The method was introduced in [Lafon, 2004] and [Coifman and Lafon, 2006] and since then, it has evolved into a number of applications and extensions. We summarize here the principles of the method.

Let (X, σ, μ) be a measure space with μ representing the distribution of the points on X, let $k : X \times X \to \mathbb{R}$ be a kernel function with the following properties:

- symmetric and positive i.e $\forall x, y \in X, k(x, y) \geq 0$,
- positive semi-definite, that means that for all real-valued bounded functions f defined on X,

$$\int_X \int_X k(x, y) f(x) f(y) d\mu(x) d\mu(y) \geq 0$$

- $k(x, y) = \eta(||x - y||)$, with η exponentially decaying.

Given a dataset, one can associate a graph $G(V, E)$ with it, where V is the set of nodes corresponding to the observed points of the data set X and E are the edges of the graph with weights calculated from k. Let $d(x) = \int_X k(x, y)d\mu(y)$ be the degree function for the graph. The normalized kernel $p(x, y) = k(x, y)/d(x)$ can be considered as the transition probability of a random walk (a Markov chain) on X, the integral operator $Pf(x) = \int_X p(x, y)f(y)d\mu(y)$ is a random walk operator on the data set. The transition probability for going from x to y in t time steps is given by $p_t(x, y)$, the kernel of the t-th power P^t of P. The idea of the method is that running the Markov chain forward in time, i.e taking powers of P, reveals relevant geometrical structures of X. The Markov chain has a stationary distribution (obtained by taking powers of P until the distribution does not change anymore) given by $\pi(y) = d(y)/\sum_{z\in X} d(z)$. The operator P can be made symmetric using $a(x, y) = k(x, y)/(\sqrt{\pi(x)}\sqrt{\pi(y)})$, this new operator A is compact and has a discrete set of eigenvalues $\{\lambda_l\}_{l\geq 0}$ and eigenfunctions $\{\phi_l\}_{l\geq 0}$ so that $1 = \lambda_0 > |\lambda_1| \geq |\lambda_2| \geq ...$ and $A\phi_l = \lambda_l\phi_l$. The kernel satisfies $a(x, y) = \sum_{l\geq 0} \lambda_l\phi_l(x)\phi_l(y)$ and the first eigenfunction $\phi_0 = \sqrt{\pi}$. The kernel p satisfies $p(x, y) = \sum_{l\geq 0} \lambda_l\psi_l(x)\varphi_l(y)$ with $\psi_l(x) = \phi_l(x)/\sqrt{\pi(x)}$, $\varphi_l(y) = \phi_l(y)\sqrt{\pi(y)}$, $\psi_0(x) = 1$.

The Diffusion Distance

For the powers P^t of P we have $p_t(x, y) = \sum_{l\geq 0} \lambda_l^t\psi_l(x)\phi_l(y)$. A distance between transition probabilities can be naturally be defined as

$$D_t^2(x, z) = ||p_t(x, .) - p_t(z, .)||^2_{L^2(X,d\mu/\pi)}.$$

Using the expression of $p_t(x, y)$ in terms of the above eigenfunctions one gets,

$$D_t^2(x, z) = \sum_{l\geq 0} \lambda_l^{2t}(\psi_l(x) - \psi_l(z))^2.$$

The distance D_t can be computed to a preset accuracy δ with a finite number of terms:

$$M_a(\delta, t) = max\left\{l \in \mathbb{N} \quad \text{such that} \quad |\lambda_l|^t > \delta|\lambda_1|^t\right\},$$

since the first eigenvector is constant, the remaining eigenvectors ψ_i define an M_a-dimensional embedding

$$\Psi_t : x \rightarrow [\lambda_1^t\psi_1(x), \lambda_2^t\psi_2(x), ..., \lambda_{M_a}^t\psi_{M_a}(x)].$$

Depending on the decay of the eigenvalues λ_l, a low dimensional embedding that characterize $\mathcal{M}$ is obtained. Why this is the case can be understood if one treats the continuous limit from the discrete representation evaluated using the high dimensional objects $x, y \in X$. This aspect will be covered in the next section.

4.3.1 Discrete to Continous Formulation: The Laplace-Beltrami Operator

An important flexibility of diffusion maps is how to choose a specific kernel $k_\epsilon(x, y) = exp(-d(x, y)/\epsilon^2)$. A very common choice for $d(x, y)$ is to use $d(x, y) = ||x - y||^2$, the obtained kernel $exp(-||x - y||^2/\epsilon^2)$

is called **Gaussian kernel**. Other options are available such as using $d(x, y) = \sqrt{\|x - y\|^2}$, the obtained kernel $exp(-\sqrt{\|x - y\|^2}/\epsilon^2)$ is called Laplacian kernel. More possibilities include taken $d(x, y)$ as the **earth movers** distance [Rubner et al., 2000], the **graph** distance [Eshera and Fu, 1984] or the **discrete time warping** distance [Petitjean et al., 2014]. For the following we will work, as it is commonly done in the literature, with the Gaussian kernel but we emphasize that this is only one possible choice.

Let $k_\epsilon(x, y) = exp(-\|x - y\|^2/\epsilon)$ with x, y in $\mathbb{R}^d$. The operator P acting on a sub-manifold $\mathcal{M} \subset \mathbb{R}^d$ with uniform probability distribution is

$$P_\epsilon f(x) = \int_{\mathcal{M}} p_\epsilon(x, y) f(y) dy$$

in the limit case $\epsilon \to 0$. The following has been shown in [Belkin and Niyogi, 2008] for the case of uniform density:

$$\Delta_{\mathcal{M}} = \lim_{\epsilon \to 0} \frac{P_\epsilon - I}{\epsilon},$$

where $\Delta_{\mathcal{M}}$ is the Laplace-Beltrami operator on the manifold $\mathcal{M}$. For any t greater than zero, $k_\epsilon(x, y) = exp(-\|x - y\|^2/\epsilon)$, the Neumann heat kernel $e^{-t\Delta_{\mathcal{M}}}$ can be approximated on $L^2(\mathcal{M})$ by $P_\epsilon^{\frac{t}{\epsilon}}$, see [Coifman and Lafon, 2006]

$$\lim_{\epsilon \to 0} P_\epsilon^{\frac{t}{\epsilon}} = e^{-t\Delta_{\mathcal{M}}}$$

For finite data, integrals are approximated by sums. A summary of the discrete expressions and their continuous counterparts is shown in table 4.1.

$K_{i,j} = exp(-\|x_i - x_j\|^2/\epsilon)$	$k_\epsilon(x, y) = exp(-\|x - y\|^2/\epsilon)$
$D_{ii} = \sum_{j=1}^m K_{i,j}$, $i, j = 1, \ldots, m$	$d_\epsilon(x) = \int_X k_\epsilon(x, y) dy$
$D^{-1}K$	$P_\epsilon f(x) = \int_X k_\epsilon(x, y)/d_\epsilon(x) f(y) dy$
$L = D^{-1}K - I$	$\Delta_{\mathcal{M}} = \lim_{\epsilon \to 0} \frac{P_\epsilon - I}{\epsilon}$.

Table 4.1: Discrete expressions and their continuous counterparts.

The eigenfunctions and eigenvalues of the operators P and $\Delta_{\mathcal{M}}$ are related as shown in table 4.2.

Operator	Eigenfunctions	Eigenvalues
$P_\epsilon f(\cdot)$	$\psi_{\epsilon,l}$	$\lambda_{\epsilon,l}$
$\Delta_{\mathcal{M}}$	ψ_l	$-\nu_l^2 = \lim_{\epsilon \to 0} \frac{\lambda_{\epsilon,l} - 1}{\epsilon}$
$\lim_{\epsilon \to 0} P_\epsilon^{\frac{t}{\epsilon}} = e^{-t\Delta_{\mathcal{M}}}$	ψ_l	$e^{-t\nu_l^2} = \lim_{\epsilon \to 0} \lambda_{\epsilon,l}^{t/\epsilon}$,

Table 4.2: Eigenfunctions and eigenvalues relationship for the operators P and $\Delta_{\mathcal{M}}$

Finally the following approximation has been shown in [Singer, 2006a] to be valid for data points that are i.i.d. (ideally identically distributed) sampled from a uniform distribution,

$$\frac{1}{\epsilon} \sum_{j=1}^{m} L_{ij} f(x_j) = \frac{1}{2} \Delta_{\mathcal{M}} f(x_i) + O(\frac{1}{m^{1/2} \epsilon^{1/2 + M/4}}, \epsilon) \tag{4.3.1}$$

where M is the dimension of the manifold $\mathcal{M}$, m is the number of data points and the notation $O(.,.)$ means that there exists positive constants C_1, C_2 (independent of m and ϵ) such that

$$\left| O(\frac{1}{m^{1/2}\epsilon^{1/2+M/4}}, \epsilon^{1/2}) \right| \leq C_1 \frac{1}{m^{1/2}\epsilon^{1/2+M/4}} + C_2\epsilon^{1/2}$$

for large m and small ϵ. The above results are interesting since they tell us that if we have enough data, using the high dimensional objects $x, y \in X$, we can theoretically construct an operator on a low dimensional manifold $\mathcal{M}$. The decay of the eigenvalues will let us decide about the dimensionality of $\mathcal{M}$ and the embedding give us a representation of $\mathcal{M}$. Let us give a simple example in order to understand the principles of the method.

Diffusion Coordinates for a simplified 1D simulation dataset

As described in chapter 2 section 2.1.2, a simulation bundle is a set of PDE solutions obtained by varying a set of input parameters. The PDE is solved usually using a computing intensive finite element solver. Our analysis goal assumes that simulations are located on a low dimensional set that we would like to identify. A parametrization of the simulations along such a low dimensional set allows for an efficient analysis and a method of dimensionality reduction obtains such information based on the simulation data.

For illustration we assume that simulations are given as 1D curves. We observe those curves as high dimensional objects of the size of the number of nodes of the curve. Further we assume that to obtain such curves only one parameter has been varied. We show that a parametrization along this parameter can be found using diffusion maps.

Proposition 4.3.1. *Let S_α be a dataset defined as a 1D curve with N_h nodes. Assume the dataset has been obtained from a numerical solver, where only one parameter α has been changed $m-$times. The dataset containing a set of simulations can be represented as:*

$$x_i = (S_{\alpha_i}(r_1), S_{\alpha_i}(r_2), \ldots, S_{\alpha_i}(r_{N_h})), \quad i = 1, \ldots m. \tag{4.3.2}$$

Assertion: the simulations are located on a 1D curve that depends only on α.

Proof. The simulations are located in $\mathbb{R}^d$ with $d = N_h$. Since the models depend only on the parameter α there exists a mapping $\varphi(q) : \mathbb{R} \to \mathbb{R}^d$ that takes points from R into $\mathbb{R}^d$. Diffusion maps constructs a similarity matrix K with matrix components calculated as $k(x_i, x_j) = e^{-\|x_i - x_j\|^2/\epsilon}$, $i, j = 1, \ldots, m$, where ϵ is a parameter that represents the width of the Gauss function and $x_i \in \mathbb{R}^d$ is given by expression (4.3.2).

The normalized kernel $p(x_i, x_j) = k(x_i, x_j)/d(x_i)$ is obtained using $d(x_i) = \sum_{j=1}^{m} k(x_i, x_j)$. For the normalized kernel the following discrete operator can be constructed,

$$Pf(x_i) = \sum_{j=1}^{m} p(x_i, x_j) f(x_j). \tag{4.3.3}$$

The continuous counterpart of the sum given in equation (4.3.3) is given by an integral operator on the domain of the low dimensional manifold $\mathcal{M}$,

$$Pf(x) = \int_{\mathcal{M}} p(x, y) f(y) d\mu(y), \tag{4.3.4}$$

where the probability density from which the data is sampled is $\mu(y)$. Assuming a uniform probability distribution, an approximation to the Laplace-Beltrami Operator is obtained according to expression (4.3.1). Notice that in this expression, ϵ can not be chosen to be too small, an appropriate value of ϵ has to be found to balance both parts, for details see [Singer, 2006a]. Using (4.3.1), an approximation to the Laplace-Beltrami operator can be obtained. In our specific example, the operator corresponds to a Laplacian in 1D with respect to α, and the eigenvalue problem can be written as,

$$\frac{d^2 f}{d\alpha^2} = \lambda f. \tag{4.3.5}$$

For a non-closed curve with two end points and length α_{max}, the first nontrivial eigenfunction is of the form $cos(\frac{\alpha}{\alpha_{max}})$ [Lafon, 2004]. The eigenvectors of the similarity matrix in (4.3.3) approximate the eigenfunctions of the limit operator (4.3.5) and therefore, one can approximately recover the parameter α that generated the simulation up to a transformation, given by the cosine function. □

This example illustrates the principal idea of the method. We notice that those results assume that the data is sampled from a uniform probability distribution. In practice this assumption is not always true, taking into account those cases, requires modifications to the previous analysis. In addition; the use of different normalizations for the discrete expressions has an influence on the continuous limits, for details see [Nadler et al., 2006, Hein et al., 2007].

Diffusion maps is founded on the above mathematical principles as a method for dimensionality reduction. Its applicability for the intended use in the analysis of simulations can now be explored.

4.4 Diffusion Maps for the Analysis of Simulations

We propose to analyze datasets consisting of simulations using diffusion maps. Each dataset contains several data points which are the individual simulations, altogether they are called a simulation bundle. As highlighted above the method starts with high dimensional data, the dimension being the number of points in one simulation, and ends up with an approximation of a low dimensional manifold $\mathcal{M}$ represented by an embedding on the first eigenvectors of a discrete positive definite operator.

Diffusion maps have been shown to be useful in many applications [Buchman et al., 2011, Neji et al., 2009, Wartak and Bors, 2010, Ferguson et al., 2010, Sonday et al., 2009, Virshup et al., 2012] in particular because of its organizational power allowing a parametrization of datasets and the fact that it allows the identification of so called slow variables, a concept that is very important in the context of simulation data as will be explained in the next section.

There are several aspects that should be taken into consideration in order to apply this method for the analysis of simulation data. Through simple examples, we describe the type of information that can be obtained from the diffusion maps coordinates for this case and how they allow the identification of slow variables in a stochastic dynamical system, even when only a small number of simulations are available.

4.4.1 Stochastic dynamical systems and slow variables in a simulation bundle

Since real data is not uniformly sampled, there is a specific probability distribution μ, mostly unknown, from which the data is sampled. Analyzing the effect of this distribution has been the subject of intensive research [Hein et al., 2007].

As described above in the case of uniform sampling the limit operator is the Laplace-Beltrami operator and for non-uniform sampling the limit operator is a Fokker-Planck type operator [Nadler et al., 2006].

Let us describe this setting, given a transition probability density $p(\mathbf{x}, t|\mathbf{y}, s)$ of finding the system at location $\mathbf{x}$ at time t, given an initial location $\mathbf{y}$ at time s ($t > s$), satisfies the forward Fokker-Planck equation [Coifman et al., 2008]

$$\frac{\partial p}{\partial t} = \nabla \cdot (\nabla p + p \nabla U(\mathbf{x})), \tag{4.4.1}$$

with initial condition

$$\lim_{t \to s^+} p(\mathbf{x}, t|\mathbf{y}, s) = \delta(\mathbf{x} - \mathbf{y}).$$

The backward Fokker-Planck equation in the backward variables $\mathbf{y}$, $s(s < t)$ is

$$\frac{\partial p}{\partial s} = \Delta p - \nabla p \cdot \nabla U(\mathbf{y}), \tag{4.4.2}$$

where $p(\mathbf{x}) = e^{-U(\mathbf{x})} = \mu(\mathbf{x})$.

In [Nadler et al., 2006], a link to this equations to a continuous family of anisotropic diffusion processes, built using a normalized Gaussian kernel have been found. We present the results of [Nadler et al., 2006]. Assume that the data set X consists of a Riemannian manifold with density $p(\mathbf{x}) = e^{-U(\mathbf{x})}$ and let $k_\epsilon(\mathbf{x}, \mathbf{y})$ be a Gaussian kernel, then for any function ϕ on X,

$$\int_X k_\epsilon(\mathbf{x}, \mathbf{y}) \phi(\mathbf{y}) dy = \phi(\mathbf{x}) + \epsilon(\Delta \phi(\mathbf{x}) + q(\mathbf{x}) \phi(\mathbf{x})) + O(\epsilon^{3/2}),$$

where q is a function that depends on the Riemannian geometry of the manifold and its embedding in $\mathbb{R}^n$. Let

$$k_\epsilon^{(\alpha)}(\mathbf{x}, \mathbf{y}) = \frac{k_\epsilon(\mathbf{x}, \mathbf{y})}{p_\epsilon^\alpha(\mathbf{x}) p_\epsilon^\alpha(\mathbf{y})}, \tag{4.4.3}$$

where as shown in [Nadler et al., 2006],

$$p_\epsilon(\mathbf{x}) = p(\mathbf{x}) + \epsilon(\Delta p(\mathbf{x}) + q(\mathbf{x}) p(\mathbf{x})) + O(\epsilon^{3/2})$$

and

$$p_\epsilon^{-\alpha} = p^{-\alpha} \left(1 - \alpha\epsilon \left(\frac{\Delta p}{p} + q\right)\right)(1 + O(\epsilon^{3/2})),$$

Finally the asymptotic expansion of the backward operator is given by

$$T_b^{(\alpha)} \phi = \int_X \frac{k_\epsilon^{(\alpha)}(\mathbf{x}, \mathbf{y})}{d_\epsilon^{(\alpha)}(\mathbf{x})} \phi(\mathbf{y}) p(\mathbf{y}) = \phi(\mathbf{x}) + \epsilon \left(\frac{\Delta(\phi p^{1-\alpha})}{p^{1-\alpha}} - \phi \frac{\Delta(p^{1-\alpha})}{p^{1-\alpha}}\right) \tag{4.4.4}$$

where the normalization factor is given by

$$d_\epsilon^{(\alpha)}(\mathbf{x}) = \int k_\epsilon^{(\alpha)}(\mathbf{x},\mathbf{y})p(\mathbf{y})dy = p_\epsilon^{-\alpha}(\mathbf{x})p^{1-\alpha}\mathbf{x}\left[1+\epsilon\left((1-\alpha)q-\alpha\frac{\Delta p}{p}+\frac{\Delta p^{1-\alpha}}{p^{1-\alpha}(\mathbf{x})}\right)\right]. \quad (4.4.5)$$

The infinitesimal generator of the diffusion is

$$\mathcal{H}_b\phi = \lim_{\epsilon\to 0}\frac{T_b - I}{\epsilon}\phi = \frac{\Delta(\phi p^{1-\alpha})}{p^{1-\alpha}} - \frac{\Delta(p^{1-\alpha})}{p^{1-\alpha}}\phi$$

Inserting the expression $p = e^{-U}$ one obtains

$$\mathcal{H}_b\phi = \Delta\phi - 2(1-\alpha)\nabla\phi\cdot\nabla U$$

For $\alpha = 1/2$ one obtains the backward Fokker-Planck operator 4.4.2. In other words if a normalized weighted graph is built in the way it has been shown, using a simulation dataset from a stochastic process, in the limit a Fokker-Planck (FP) equation is obtained. In general the FP equation describes the time evolution of the probability density function of a stochastic mechanical, physical, chemical or biological system. In the presence of a spectral gap the long term behavior of the system is approximately described using only the first eigenfunctions of this operator. In other words for a stochastic system with many random variables, the first eigenfunctions will describe the behavior of the slow variables (or principal trends), corresponding to the conformational state changes. For example a system can have many variables that oscillate very rapidly around a state as the system progresses in time, only after a long time are there changes to a new state (conformational change). For many applications, the slow variables are precisely the interesting ones. Having enough data sets, we can construct a discrete approximation of this operator and its eigenvectors represent the slow variables of a stochastic process.

Consider a crash simulation where stochastic variability is induced naturally by changes in the input parameters such as material, geometry or even numerical noise in the finite element calculation. In such a process, it is important to study and identify so called conformational changes. In structural dynamics this could correspond to a change in the bending state of a beam. This effect produces a bifurcation in the deformations that propagate to the overall structure. We assume that for this stochastic process, there is a corresponding Fokker-Planck operator from which we can extract the first eigenfunctions. According to the theory, the slow variables of the system will describe the appearance of a bifurcation point in the deformation (buckling mode) and this behavior can be recovered by the eigenvectors of a discrete approximation of this operator. In the next subsection we would like to illustrate these ideas with a specific example.

4.4.2 Slow variable for a simple simulation bundle

In this subsection we will illustrate a specific example that can also be evaluated analytically. This example has been already presented in [Nadler et al., 2006] and [Iza Teran, 2013]. Assume that the probability density μ, normally unknown, is given. We keep a 1D formulation for the purpose of simplicity; this can be extended to 2D thanks to the specially chosen density.

In the context of equation (4.4.2), consider the potential $U(x) = x^2/2\tau$ with the probability density $\mu =$

$e^{-U}/\sqrt{2\pi\tau}$. We assume that a simulation bundle is obtained by sampling from this density (Brownian movement of particles in a potential field [Nadler et al., 2006]). Using this density it is also possible to define the continuous integral random walk operator

$$\tilde{P}_\epsilon\phi(x) = \int_{-\infty}^{\infty} \tilde{p}_\epsilon(x,y)\phi(y)\mu(y)dy, \tag{4.4.6}$$

where the following kernel normalization has been used,

$$\tilde{p}_\epsilon(x,y) = \frac{k_\epsilon(x,y)}{\sqrt{\mu_\epsilon(x)}\sqrt{\mu_\epsilon(y)}} \tag{4.4.7}$$

and $k_\epsilon(x,y) = exp(-\|x-y\|^2/\epsilon)$, $\mu_\epsilon = \int k_\epsilon(x,y)\mu dy$. The eigenvalues of the integral operator (4.4.6) are $\lambda_k = (\tau/(\tau+\epsilon)) < 1$, with corresponding eigenfunctions $\phi_k(x) = r_k(x)exp(-\frac{x^2}{4(\tau+\epsilon)})$, r_k is a polynomial of degree k.

In the limit $\epsilon \to 0$ this integral operator converge to the backward Fokker-Planck operator (4.4.2) and the corresponding eigenfunctions $\psi(x)$ of such an operator can be obtained by normalization of the ones of (4.4.6) with $\sqrt{\mu(y)} = C_a exp(-y^2/(4\tau))$ (see [Nadler et al., 2006] for details).

A extension of this type of potential to two dimensions can be realized in the form $U(x1,x2) = x_1^2/(2\tau_1) + x_2^2/(2\tau_2)$ and choose $\tau 1 \gg \tau_2$ so that x_1 is the slow variable. This specially chosen probability density has a separable structure $\mu(x_1,x_2) = \mu_1(x_1)\cdot\mu_2(x_2)$. The eigenfunctions and eigenvalues are given by

$$\phi_{i,j}(x_1,x_2) = \phi_{1,i}(x_1)\phi_{2,j}(x_2)\ ,\ \lambda_{i,j} = (\tau_1/(\tau_1+\epsilon))^i \cdot (\tau_2/(\tau_2+\epsilon))^j. \tag{4.4.8}$$

Since we set $\tau_1 \gg \tau_2$, ordering the eigenfunctions by decreasing eigenvalue, the first nontrivial eigenfunctions are $\phi_{1,0}, \phi_{2,0}$. Since the order of the polynomial is i and j, it can be seen that

$$\phi_{1,0}(x_1,x_2) = \phi_{1,1}(x_1)\cdot\phi_{2,0}(x_2) = C\cdot x_1\cdot exp\left(-\frac{x_1^2}{4\tau_1}\right)exp\left(-\frac{x_2^2}{4\tau_2}\right). \tag{4.4.9}$$

After conjugation (dividing by $\sqrt{\mu}$ for each variable) it can be seen that this first eigenfunction depends linearly only on x_1

$$\psi_1(x_1,x_2) = C\cdot x_1. \tag{4.4.10}$$

The next task is to verify this analytical result. In order to do that, we will generate several data sets sampled from the given 2D probability distribution $\mu(x_1,x_2)$ (see figure 4.2 using $\tau_1 = 1$, $\tau_2 = 25$). Notice that the specific value of τ is not essential in this analysis, but the fact that $\tau_1 \gg \tau_2$. Using these data sets we build a similarity matrix, normalize it (according to (4.4.7)) and obtain the first diffusion coordinate (first eigenvector), plotting the values of x_1 against the first non-constant eigenvector ψ_1 shows that, for $m = 1000$, a linear dependence is obtained except at the boundaries (figure 4.3 (a)). Next we test the approach after the data sets are reduced drastically for example to only 100 data points, the dependence is still clearly seen even for this very small amount of data sets (figure 4.3 (b)).

Aiming to theoretically support the assumption that even with a very small number of points (number of datasets), the diffusion coordinates can still be useful is in itself difficult. In order to approximate the eigenfunctions of the continuous Laplace-Beltrami operator or the Fokker-Planck operator we need

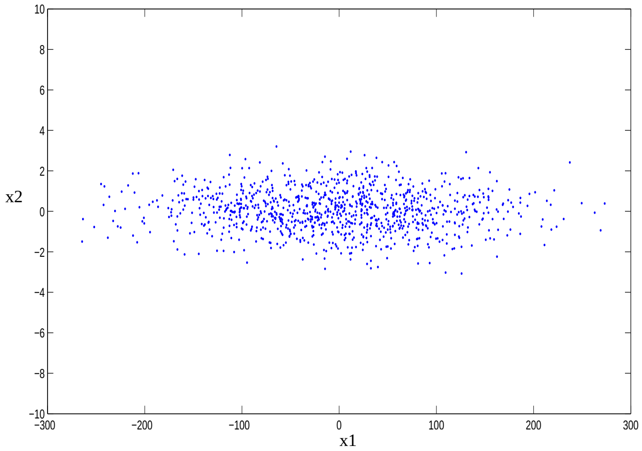


Figure 4.2: Data sampled from a bivariate distribution

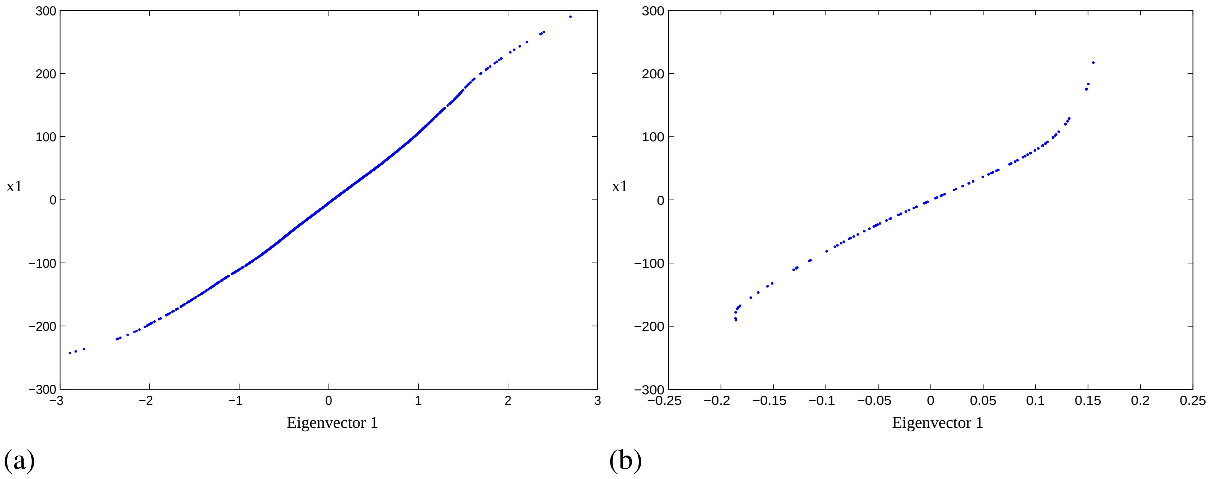


Figure 4.3: Slow variable for data from a bivariate distribution: a) slow variable x_1 vs eigenvector ψ_1 using 1000 data points, (b) slow variable x_1 vs eigenvector ψ_1 using 100 data points

several thousand datasets. In the case of the Gauss kernel with a fixed value of ϵ, it is known (see [von Luxburg et al., 2008]) that the rate of convergence of the eigenvectors is of the order $O(1/\sqrt{m})$. It is also known that eigenvectors obtained from a reduced similarity matrix obtained by sampling from a finer one can approximate the eigenvectors of the original matrix (using the Nyström approximation see [Fowlkes et al., 2004]). A formal analysis of the convergence properties for very few data points is certainly required.

Choosing the value of ϵ is still a very important issue in using eigenvectors for embedding a dataset and implementing a gauss kernel as in our application. Since most of the time only a limited amount of datasets are available, all proposed methods for setting ϵ are not practical. Therefore, an empirical way of setting it has to be used such as for example the median. From a theoretical point of view, which amounts to having enough data, some studies have been done in [Hein and Audibert, 2005].

Based on the theoretical results of this chapter, we propose a method for the analysis of simulation bundles in the next section.

4.5 Diffusion Maps Methodology for a Simulation Bundle

In this section we will provide details of the application of diffusion maps to a simulation dataset in an algorithmic way (see Algorithm 1). The training data set is saved on files in binary proprietary format. So called post-processor software can be used to read this data and extract all or part of it into for example ASCII format. We use the software Animator ® [GNS,] to extract specific components of a car or structure, the components we choose are the ones that are critical for the engineer in the sense of structural behavior under crash or vibration response under excitation.

Notice that in our analysis we use simulation data on a finite element mesh directly or extract some response curves for vibration analysis using again Animator. A finite element mesh contains nodes and elements, we assume that the mesh connectivity is the same and use only the values defined at the nodes on the mesh as data set. If the mesh connectivity is different a reference mesh can be used to map the simulation values to it.

Algorithmus 1 : Diffusion maps for training data

```
Input : $m$ finite element training data sets containing a variable $x \in \mathbb{R}^{N_h}$, where $N_h$ is the
        number of nodes in a finite element mesh or the number of points on a curve
Output : Reduced representation $[\lambda_1^t \psi_1(x), \lambda_2^t \psi_2(x), \dots, \lambda_j^t \psi_j(x) \dots, \lambda_M^t \psi_M(x)] \in \mathbb{R}^M$

foreach $i \in \{1, \dots, m\}$ do
    foreach $j \in \{1, \dots, m\}$ do                    ▷ calculate similarity matrix
        $K(i, j) = exp(-d(x_i, x_j)/\epsilon^2)$
    end
end
$Q = diag(K \cdot \mathbb{1})$
$K^{(\alpha)} = (Q^\alpha)^{-1} K (Q^\alpha)^{-1}$          ▷ calculate kernel as given in 4.4.3
$D = diag(K^{(\alpha)} \cdot \mathbb{1})$        ▷ calculate normalization approximating 4.4.5
$P = D^{-1/2} K D^{1/2}$            ▷ calculate approximation of 4.4.4
$[\Lambda, U] = eig(P, r)$   ▷ calculate first $r$ eigenvalues $\lambda_j$ and eigenvectors $U_j$
$M = max\{j : \lambda_j > 0.1\}, t = 1, \alpha = 1/2$
return $[\lambda_1 U_1 / U_0, \lambda_2 U_2 / U_0, ... \lambda_M U_M / U_0]$         ▷ $M$ dimensional embedding
```

4.6 Methodology

We will describe a general methodology that allows the analysis of finite element simulations-bundles. Whereby four steps are identified:

- **Extraction**: the variables for the analysis are obtained from the simulation. These variables can be of a different type such as scalars, vectors or tensors defined on nodes or elements of a finite element mesh.

- **Preprocessing**: this step is necessary to cope with the problem of the data size (millions of nodes and elements). The preprocessing contains simple sub-sampling, that is to just take consistently only subsets of the data. A further sub-step that is important to mention is the generation of compact signatures. The data can be transformed in different ways in order to obtain a compact representation; examples of such a transformation includes principal component analysis (PCA).

- **Dimension Reduction**: in this step a low dimensional representation is obtained from the dataset that parameterizes the information, allowing the identification of intrinsic parameters. We use diffusion maps as a dimensionality reduction framework. Notice that other dimension reduction methods can be used at this point, without giving a complete justification, we do mention that to our knowledge the connection to stochastic dynamical systems and the identification of slow variables provided by this method is not available using other methods. This is a key point in the analysis of simulation data.

- **Exploration**: this step can be done as long as the simulation variables are organized along the low dimensional embedding space obtained in the dimension reduction step. All simulations can be represented in this low dimensional setting and not only that, since diffusion maps are able to identify intrinsic parameters from the data, we can explore all the dataset in a simplified setting along such parameters. The engineer can obtain an overview of all his designs variants in this way and the effect of the design variations can be explored along the intrinsic parameters.

The proposed method for the analysis of simulations turnouts to be very useful for real industrial cases as has been shown in [Iza Teran, 2013] (see also section 10.1 for an application example), still one can ask the question whether the proposed overall goal of this thesis of using geometrical mathematical principles can be extended even more, specially to address some inherent limitations. The first one relates to the available number of simulations; according to the theory in order to verify the approximation properties, one needs a large number of simulations. This is not achievable in practice, normally at the most several thousand are available and in a more general case only a few hundred are available. The second limitation relates to the reconstruction of high dimensional approximations based on information in the low dimensional embedding. The method does not provide an efficient way of reconstructing such approximations, a great disadvantage, since this is a very important aspect that is very useful in applications.

The mentioned limitations of diffusion maps as a method for the analysis of simulations are critical for applications and we will focus on a new approach to overcome them in the next chapters.

CHAPTER 5

Quotient Spaces and Group Transformations

We have until now treated each simulation as a point in a very high dimensional space with a dimension which is of the order of the number of nodes used for a discretization of the domain. But, a point in this high dimensional space is assumed to be intrinsically low dimensional. The previous chapters were all about how to recover such a low dimensional representation. In the following chapters we will develop a different point of view and consider each simulation as obtained by some transformation from a reference configuration. Although the transformations are not known we will work out this idea from a mathematical point of view and see what can be obtained that can be useful for applications.

Our intended goal is being able to analyze many high dimensional simulations and for that we would like to be able to identify and represent transformations that maps one simulation into another in a parametric form with few parameters. Let us see how far can we get in achieving this. The adequate mathematical setting for describing transformations is to use quotient spaces over manifolds under the action of a transformation group. Specifically, we will consider the case of immersions and embeddings, but in particular embeddings in $\mathbb{R}^3$ and $\mathbb{R}^2$ under the action of a transformation group. Choosing embeddings in $\mathbb{R}^3$ allows the treatment of many interesting cases such as crash simulation as the meshes are, due to computational costs, surface meshes. Embeddings in $\mathbb{R}^2$ covers the case of plane curves obtained as time dependent solutions of ordinary differential equations (ODE). A transformation group action defines an orbit, that is the points that are obtained when a transformation is applied to a manifold. Further the space of all orbits is called orbit space. We will see that one can distinguish between several cases for those quotient spaces depending on the type of group action. A Riemannian structure can, under suitable assumptions, be defined for the quotient space, and geodesics on it can be computed. This characterization from a Riemannian point of view is useful for finding parameterizations of simulations and for moving along the space of orbits, which is done following paths of minimal length on this space, that is geodesics.

A special type of transformations are the ones that preserve the distance on the manifold, they are called isometries of a manifold. For certain type of isometric transformations one can recognize a very useful property for applications, namely that geodesics in the space of isometries can be evaluated following a group action, that means, orbits are under certain conditions geodesics. Additionally, if the group action can be parametrized then simulations can be conveniently parametrized, for at least certain isometric transformations. This is a very important observation, if ways are found to represent those group actions, as otherwise the calculation of geodesics is a very time consuming process that involves the solution of a differential equation in the high dimensional space of the simulations.

Representing group actions for some type of isometries is possible such as in the case of rotations in $\mathbb{R}^n$ where a linear representation can be derived. Nevertheless we would not like to treat each case separately depending on the type of isometry. For that we will use a link between an operator that is invariant to the isometry group and a group representation. A few theoretical results are available here for the rotation and translation group, the Laplace-Beltrami operator is isometric invariant and for the rotation group, the eigenfunctions of it can be used as representations of the group. This link suggests the possibility of using eigenfunctions of an isometric invariant operator to represent simulations. That means, the eigenfunctions can be used as a a new basis for the representation of simulations, considered as isometries that are embeddings in $\mathbb{R}^n$. The projection coefficients to this eigenfunction basis will reflect isometric group actions and any further analysis task can be equivalently done based on the spectral coefficients and what is more important, it opens the possibility to significantly reducing the effort for calculating geodesics since they are defined by the group actions as mentioned above. In chapter 9 we will provide numerical evidence of this assertion.

Relevant material from differential geometry that are presented in this chapter, are standard, for details see [Fletcher, 2004, Onishchik, 1993, Helgason, 1984] and references therein. This chapter uses the concepts of chapter 3 as a prerequisite.

5.1 Preliminary Definitions

In chapter 3 we have introduced the concept of a topological space and based on it, we also defined a differentiable manifold. One of the beauties of mathematics is that one can construct more complex topological spaces by building them up from more simple ones. Subsets of $\mathbb{R}^n$ and also quotient spaces of those spaces allows such constructs.

A simple example of a subspace of $\mathbb{R}^n$ is the circle defined as the set of points

$$S^1 = \left\{(x, y) \in \mathbb{R}^2 : x^2 + y^2 = 1\right\},$$

it is a topological space with the subspace topology of $\mathbb{R}^2$ (U is in S^1 if there is a $V \subset \mathbb{R}^2$ with $U \cap S^1 = V$), the circle is sitting in $\mathbb{R}^2$ but as a topological space it is independent of the ambient space it is sitting on, formally one can speak about an embedding.

Definition 5.1.1. Let X, Y be topological spaces, an **embedding** is a map $\varphi : X \mapsto Y$ such that φ is an homeomorphism (a map that is continuous, invertible, and its inverse is also continuous). $\varphi(X)$ has the subspace topology of Y.

Definition 5.1.2. Let X, Y be topological spaces, an **immersion** is a mapping whose derivative is injective on the tangent space of X at each point.

Spaces embedded or immersed into other spaces can provide a useful framework for constructing topological spaces. Consider for example embeddings of the circle in $\mathbb{R}^n$ like a square in $\mathbb{R}^2$ or the same circle as embedding on the 2-sphere or as a knot (embedding of S^1 into $\mathbb{R}^3$). We notice that in the context of simulations, considered as topological spaces we will encounter embeddings such us surfaces embedded in $\mathbb{R}^3$ as well as immersions such as planar curves immersed in $\mathbb{R}^2$ (self-intersections are allowed). A further way to increase this constructive framework even more are quotient spaces under a group action as we will see in section 5.3. But first, let us give some background information on group theory which is required for making use of quotient spaces.

5.2 Groups

One more mathematical formalism that is needed to be able to construct abstract spaces is group transformations. Actually one can speak of a set of transformations which fulfill some axioms that turn them into a group.

Definition 5.2.1. A **group** is a set G, together with an operation $a \cdot b$ that combines any two elements a, b to form another element denoted $a \cdot b$. A group and the operation satisfy the following axioms, called group axioms.

- closure, for all a, b in G, the result operation is also in G
- associativity, for all a, b, c in G, $(a \cdot b) \cdot c = a \cdot (b \cdot c)$
- identity, there exists an element e in G, such that for every element a in G, $e \cdot a = a \cdot e = a$
- inverse, for each a in G, there exists an element b in G such that $a \cdot b = b \cdot a = e$, with e being the identity element.

Definition 5.2.2. Given a group G and a set $\mathcal{M}$, a **group action** is a map $G \times \mathcal{M} \to \mathcal{M}$ written as $(a, p) \rightarrowtail a \cdot p$ that satisfies

- $e \cdot p = p$ for all p in $\mathcal{M}$
- $a_1 \cdot (a_2 \cdot p) = (a_1 a_2) \cdot p$, for all $a_1, a_2 \in G$ and $p \in \mathcal{M}$

We notice that continuous groups illustrate a set of transformations that can be turned into an abstract manifold; that in turn can even be differentiable. In this case one can talk about Lie groups.

5.2.1 Lie Groups

We will now connect to the definitions from section 3.

Definition 5.2.3. A group G is called **Lie-group** if G is a topological manifold and the mapping $G \times G, (a, b) \mapsto ab^{-1}$ is differentiable.

Further one can also define the tangential space at the identity to the Lie-group manifold. This is called **Lie-algebra**.

Definition 5.2.4. The left, respectively right, **multiplication** on G are defined for each $a \in G$ as:

$$l_a(b) = a \cdot b, \quad r_a(b) = b \cdot a.$$

Definition 5.2.5. A vector field X is called **left invariant** if

$$T_b(l_a)(X(b)) = X(l_a(b) := X(ab)$$

for all elements a and b in G.

Lemma 5.2.6. *If X,Y are left invariant smooth vector fields on G, then their* ***Lie bracket*** $[X, Y]$ ***or commutator*** *is again left invariant.*

Theorem 5.2.7. *Let G be a Lie-group and let $\mathfrak{g}$ be the set of all left-invariants vector fields on G. Then:*

1. *the set $\mathfrak{g}$ is a real vector space that is isomorph to T_eG and dim G = dim $\mathfrak{g}$.*

2. *the set $\mathfrak{g}$ is a Lie-algebra with respect to the Lie bracket of two left-invariants vector fields.*

Definition 5.2.8. The **Lie-algebra $\mathfrak{g}$** of a Lie-group is defined as the space of left-invariants vector fields and accordingly the tangential space on G in e (the unity element in the group), with the commutator of vector fields as Lie-bracket.

Definition 5.2.9. A mapping of Lie groups $\varphi : G_1 \to G_2$ is called a Lie **group homomorphism** if it is a smooth mapping and the following is satisfied:

- $\varphi(e_1) = e_2$, where e_1 and e_2 are the identity elements of G_1, G_2 respectively,

- $\varphi(ab) = \varphi(a)\varphi(b)$ for all $a, b \in G_1$

Definition 5.2.10. The image of a Lie group homomorphism $c : \mathbb{R} \to G$ is called a **one parameter subgroup**. A one parameter subgroup is both a smooth curve and a subgroup of G.

Not any one parameter subgroup is a Lie subgroup of G, but there is a bijective correspondence between the Lie algebra and the one parameter subgroups.

Theorem 5.2.11. *Let $\mathfrak{g}$ be the Lie algebra of a Lie group G. Given any vector $X \in \mathfrak{g}$ there is a unique Lie group homomorphism $c_X : \mathbb{R} \to G$ such that $\dot{c}_X(0) = X$.*

Definition 5.2.12. Let G be a Lie-group with Lie-algebra $\mathfrak{g}$. The mapping,

$$exp : \mathfrak{g} \to G, \quad exp(X) := c_X(1),$$

is called **exponential map** of the Lie-Group G.

Definition 5.2.13. A Riemannian metric $\langle \cdot, \cdot \rangle$ on a Lie group G is said to be a **bi-invariant metric** if it is invariant under both right and left multiplication, that is, $r_a^* \langle \cdot, \cdot \rangle = l_a^* \langle \cdot, \cdot \rangle = \langle \cdot, \cdot \rangle$ for all $a \in G$.

Theorem 5.2.14. *For a Lie group G with bi-invariant metric, the Lie group exponential map $exp(X)$ agrees with the Riemannian exponential map at the identity $Exp_e(X)$, that is for any tangent vector $X \in \mathfrak{g}$*

$$exp(X) = Exp_e(X).$$

This last observation allows us to write the geodesic at a point $a \in G$ as the left multiplication of a geodesic at identity as follows

$$\gamma(t) = a\ exp(tX),$$

with $\gamma(0) = a$ and $\dot{\gamma}(0) = l_a^*(X)$.

5.3 Quotient Spaces

The flexibility of constructing topological spaces can be enhanced through the use of equivalence relations. Remember a equivalence relation is a relation that satisfies transitivity, reflexivity and symmetry properties. The set of equivalence classes can form a new topological space.

Definition 5.3.1. Let $\mathcal{M}$ be a topological space and let $\sim$ be an equivalence relation on $\mathcal{M}$, and let $\pi : \mathcal{M} \mapsto \mathcal{M}/\sim$ be the canonical projection map. The quotient topology on $\mathcal{M}/\sim$ is defined by declaring any subset $U \in \mathcal{M}/\sim$ to be open if its pre-image $\pi^{-1}(U)$ is open in $\mathcal{M}$.

For our purposes we will consider special types of quotient spaces, namely those constructed using a group G acting on $\mathcal{M}$. If a group G acts on a space $\mathcal{M}$ by homeomorphism, then we have the orbit equivalence relation: $x \sim y$ if and only if $x = a \cdot y$ for some $a \in G$.

Definition 5.3.2. Given a group G acting on $\mathcal{M}$, an **orbit** of a point x in $\mathcal{M}$ is the set of elements of $\mathcal{M}$ to which x can be moved by the elements of G. Formally for $x \in \mathcal{M}$, the orbit of x is defined as

$$[x] = \{a \cdot x \in \mathcal{M} \,|\, a \in G\}.$$

The set of all orbits is called **orbit space** and is written as $\mathcal{M}/G$.

In the case that $\mathcal{M}$ consists of a single orbit, $\mathcal{M}$ is called **homogeneous space** and the group action is called **transitive**. The isotropy subgroup of p is defined as $G_p = \{g \cdot p = p\}$, that is G_p is a subgroup of G that leaves p fixed.

Let φ be a distance preserving diffeomorphism (a differentiable mapping where the inverse is also differentiable) $\varphi : \mathcal{M} \to \mathcal{M}'$, between two Riemannian manifolds $(\mathcal{M}, g)$ and $(\mathcal{M}', g')$ with $g = \varphi^* g'$, φ is geodesic distance preserving, it follows that $d_{g'}(\varphi(p), \varphi(q)) = d_g(p, q)$. A global isometry is a map from $\mathcal{M}$ to itself which preserves the distance function d_g, induced from the metric tensor g.

Definition 5.3.3. Vector fields on Riemannian manifolds, whose local flow consists of local isometries are called **Killing vector fields**.

Lemma 5.3.4. *Let X be a Killing vector field and $\gamma : [0, 1] \to \mathcal{M}$ be a geodesic. Then*

$$\frac{d}{dt}\langle X(\gamma), \dot{\gamma}(t)\rangle_{\gamma(t)} = 0.$$

This last result is very interesting since it expresses the existence of a conservation property along a symmetry.

A very special transformation group will prove to be very useful in our context, namely the isometry group. Let G be a group of isometries of a metric space $\mathcal{M}$, then G defines a natural group action

$$\begin{aligned} \Theta : G \times \mathcal{M} &\longrightarrow \mathcal{M} \\ (a, p) &\mapsto a(p) \end{aligned} \tag{5.3.1}$$

Proposition 5.3.5. *([Borzellino, 1992]) Let $\mathcal{M}$ be a C^k Riemannian manifold, then the isometry group $Isom(\mathcal{M})$ of $\mathcal{M}$ is a Lie group and the mapping is of class C^k.*

Definition 5.3.6. An action G on $\mathcal{M}$ is effective if for any $a \neq e$ in G there exists a p in $\mathcal{M}$ such that $a \cdot p \neq p$

Definition 5.3.7. An action G on $\mathcal{M}$ is free if there exists a p in $\mathcal{M}$ with $a \cdot p = p$, then a is the identity.

Definition 5.3.8. An action of G on $\mathcal{M}$ is properly discontinuous if, given any $p \in \mathcal{M}$, there exists an open set U in $\mathcal{M}$ such that the number of elements a of the group for which $g(U) \cap U \neq \emptyset$ is finite.

Proposition 5.3.9. *([Borzellino, 1992]) Every discontinuous group G of isometries of a Manifold $\mathcal{M}$ acts properly discontinuously.*

Dealing with quotient spaces with respect to group actions sometimes leads to manifolds. In other cases one has to be careful since the resulting space could not even be Hausdorff. A useful concept for characterizing some of the resulting spaces is the one of Orbifolds. It is a generalization of the concept of manifold and was introduced by Thurston [Thurston, 2002]. For an orbifold the space locally looks like the quotient of a Euclidean space under the linear action of a finite group. It can also be thought of as a surface with singularities and arise by the analysis of symmetries.

Definition 5.3.10. A (topological) orbifold O is a pair (X_O, U_i), where X_O is a Hausdorff space and U_i is a covering of X_O, that is $X_O = \cup_i U_i$, which is closed under finite intersections. In addition to each element of the open cover U_i there is associated a finite group G_i, an action of G_i on an open subset V_i of $\mathbb{R}^n$ and a homeomorphism $\varphi_i : U_j \approx V_i/G_i$. Whenever $U_i \subset U_j$, there is to be an injective group homomorphism $f_{ij} : G_i \hookrightarrow G_j$ and an embedding $\tilde{\varphi}_{ij} : V_i \hookrightarrow V_j$, which is equivariant with respect to f_{ij} (i.e for each $\gamma \in G_i$ $\tilde{\varphi}_{ij}(\gamma x) = f_{ij}(\gamma)\tilde{\varphi}_{ij}(x)$) such that the following diagram commutes

$$\begin{tikzcd}
V_i \arrow[r, "\tilde{\varphi}_{ij}"] \arrow[d] & V_j \arrow[d] \\
V_i/G_i \arrow[r, "\varphi_{ij} = \tilde{\varphi}_{ij}/G_i"] & V_j/G_i \arrow[d, "f_{ij}"] \\
 & V_j/G_j \\
U_i \arrow[uu, "\varphi_i"] \arrow[r, "\subset"] & U_j \arrow[u, "\varphi_j"']
\end{tikzcd}$$

An immediate application of the above formal definition to quotient spaces is the following

Proposition 5.3.11. *([Borzellino, 1992]) The quotient space of a manifold $\mathcal{M}$ by a group G which acts properly discontinuously and not necessarily free on $\mathcal{M}$ is an orbifold. The orbifold is called a good one if $\mathcal{M}$ is a Riemannian manifold. In this case the underlying space of the orbifold is $\mathcal{M}/G$.*

For the special case of free group actions, the quotient spaces are manifolds:

Theorem 5.3.12. *If G is compact and acts freely on $\mathcal{M}$, there exists a smooth structure on $\mathcal{M}/G$ such that $\pi : \mathcal{M} \to \mathcal{M}/G$ is a principal G−bundle (a bundle where the mapping π is a group G, see definition 3.2.13) and in particular a submersion.*

5.3.1 The Riemannian Structure of $\mathcal{M}/G$

The quotient space of a manifold $\mathcal{M}$ by a group G can, as mentioned above be an orbifold, if the group action is proper and discontinuous. If the action is in addition free, then the quotient is a manifold. This is good news for our analysis but still we have to see how a Riemannian structure for the quotient space in the case of isometric actions is defined. At this point we will need two more results and follow here the approach presented in [Kähler, 2012, Huckemann et al., 2010] where more details can be found.

Theorem 5.3.13 (Hopf-Rinow)**.** *Let $(\mathcal{M}, g)$ be a connected Riemannian manifold. Then the following statements are equivalent:*

- *the closed and bounded subsets of $\mathcal{M}$ are compact.*
- *$\mathcal{M}$ is a complete metric space.*
- *$\mathcal{M}$ is geodesically complete (geodesics are defined for all $t \in \mathbb{R}$, see section 3.2.4).*
- *there exist $p \in \mathcal{M}$ so that Exp_p is defined for all $X_p \in T_p\mathcal{M}$.*

Lemma 5.3.14 (Gauß)**.** *Let $\epsilon > 0$ such that Exp_p on the ball $B_\epsilon(0) \in T_p\mathcal{M}$ is a diffeomorphism on its image and let $r \in (0, \epsilon)$. Then for each vector $v \in T_p\mathcal{M}$ with $\|v\| = 1$, the geodesic $\gamma_v(t) = Exp_p(tv)$ is orthogonal to the image of $Exp_p(\partial B_r(0))$ of the sphere of radius r at the point $\gamma_v(r)$. That is $\dot{\gamma}_v(r)$ is perpendicular to the tangent space $T_{\gamma_v(r)}Exp_p(\partial B_r(0))$.*

Under the conditions of theorem 5.3.12 a Riemannian metric can be defined for the space $\mathcal{M}/G$ as follows.

First, the Hopf-Rinow theorem 5.3.13 asserts that for a complete Riemannian manifold minimizing geodesics can be defined between any two points in the manifold. Under the action of a group by isometries, then

$$\gamma \quad \text{geodesic} \Leftrightarrow a\gamma \quad \text{geodesic for all} \quad a \in G.$$

Under the group action the orbit $[p] = \{ap : a \in G\}$ is a submanifold of M that is diffeomorphic to G/I_p, where I_p is the isotropy group at p defined as $I_p = \{a \in G : ap = p\}$. For a free action all isotropy groups consist of only the unit element.

Now the tangent space $T_p\mathcal{M}$ at p can be decomposed into a vertical subspace and an horizontal orthogonal subspace as,

$$T_p\mathcal{M} = T_p[p] \oplus H_p\mathcal{M}.$$

$T_p[p]$ corresponds to the tangent space of the orbit and $H_p\mathcal{M}$ is defined on $\mathcal{M}$. In this situation one can define curves $t \mapsto \gamma(t)$ based on whether the derivatives at t_0 of them are horizontal or vertical. This means horizontal if $\dot{\gamma}(t_0) \in H_{\gamma(t_0)}\mathcal{M}$ and vertical if $\gamma(\dot{t_0}) \in T_{\gamma(t_0)}[\gamma(t_0)]$.

Still a relationship between the geodesics at $p \in \mathcal{M}$ and the tangential space is needed. In a neighborhood of p defined by means of the exponential map, the Gauss Lemma 5.3.14 states that images under the exponential map of spheres in $T_p\mathcal{M}$ are orthogonal to the geodesics through p.

Now the projection in theorem 5.3.12,

$$\pi : \mathcal{M} \mapsto \mathcal{M}/G := \{[p] : p \in \mathcal{M}\} \tag{5.3.2}$$

is equipped with the quotient space topology and is a submersion. Under this condition one can define a Riemannian metric for $\mathcal{M}/G$. This is possible because $d\pi_p$ induces the isomorphism

$$d\pi_p|_{H_p\mathcal{M}} : H_p\mathcal{M} \to T_{[p]}\mathcal{M}/G.$$

By push forwarding the metric in $\mathcal{M}$ one can define

$$\langle v, w\rangle_{[p]} = \langle (d\pi_p)^{-1}v, (d\pi_p)^{-1}w)\rangle_p.$$

Now what is actually important for applications is the evaluation of distances in the quotient space. The following considerations, allow us to understand the relationship between geodesics in $\mathcal{M}$ with respect to the ones in $\mathcal{M}/G$.

Definition 5.3.15. A curve $\gamma : [a, b] \to M$ is called **horizontal**, if $\dot{\gamma}(t) \in H_{\gamma(t)}\mathcal{M}$ for all $t \in [a, b]$. A **horizontal geodesic** is a geodesic which is also a horizontal curve.

Proposition 5.3.16. *A geodesic, horizontal at one point, is horizontal everywhere. Projections of horizontal geodesics are geodesics in $\mathcal{M}/G$.*

Finally we can define a distance in the quotient space.

Proposition 5.3.17. *Let $\mathcal{M}$ be a complete connected manifold. For any $[p_1], [p_2] \in \mathcal{M}/G$ the following distance definitions coincide*

$$d_{\mathcal{M}/G}([p_1],[p_2]) := inf\,\{L_{\mathcal{M}/G}|\gamma : [0,1] \to \mathcal{M}/G, \gamma(0) = [p_1], \gamma(1) = [p_2]\} \tag{5.3.3}$$

$$\bar{d}_{\mathcal{M}/G}([p_1],[p_2]) := \inf_{a,b\in G} d_{\mathcal{M}}(ap_1, bp_2) = \inf_{a\in G} d_{\mathcal{M}}(p_1, ap_2) \tag{5.3.4}$$

The expression 5.3.4 above implies a minimization problem that has to be solved. The following definition formalizes this situation.

Definition 5.3.18. Let $[p_1], [p_2] \in \mathcal{M}/G$. If there exist $a \in G$ such that

$$\hat{d}_{\mathcal{M}/G}([p_1],[p_2]) = d_{\mathcal{M}}(p_1, ap_2),$$

then p_1 and ap_2 are said to be in **optimal position**.

We notice that until now the quotient space $\mathcal{M}/G$, under the conditions explained above, has a manifold structure. But if the group actions is not free, this is not the case. Nevertheless, as demonstrated in [Huckemann et al., 2010] the expression

$$\hat{d}_{\mathcal{M}/G}([p_1],[p_2]) = \inf_{a\in G} d_{\mathcal{M}}(p_1, ap_2) \tag{5.3.5}$$

can still be used and so called generalized geodesics could be defined instead.

Definition 5.3.19. A curve γ in $\mathcal{M}/G$ is called **generalized geodesic**, if it is the projection of a horizontal geodesic in $\mathcal{M}$.

Up until now the presented mathematical background has an important theoretical context, nevertheless there is still the question of how to evaluate group actions in practice since the transformations sending simulations to simulations are unknown. That is why we would like to introduce some notions about representation theory that provides a constructive way to identify such group actions, as will be seen in the next chapter.

5.4 Symmetric Spaces and Representation Theory

Of special interest for us are symmetric manifolds constructed as compact Lie groups. We will see that being able to define symmetric spaces using a transformation group has interesting properties, but first let us give some necessary definitions.

Definition 5.4.1. Let X be a set and φ be any mapping of X into itself. A point $x \in X$ is called **fixed point** of φ if $\varphi(x) = x$. The mapping is called **involutive** if φ is not the identity mapping, but $\varphi \circ \varphi = id$.

Definition 5.4.2. A (Riemannian) **symmetric space** is a connected Riemannian manifold $\mathcal{M}$ such that at each point $p \in \mathcal{M}$, there is an involute isometry $\varphi_p : \mathcal{M} \to \mathcal{M}$ that has p as an isolated fixed point. The term isolated means that there is a neighborhood U of p such that p is the only point in U that is a fixed point of φ_p.

Theorem 5.4.3. *A Riemannian symmetric space is complete, and if φ_p is an involutive isometry of $\mathcal{M}$, then φ_{p*} is a reflection of the tangent space $T_p\mathcal{M}$, that means $\varphi_{p*}(X) = -X$, and φ_p reverses geodesics through p, that is $\varphi(Exp_p(X)) = Exp_p(X)$ for all $X \in T_p\mathcal{M}$ such that those geodesics exist.*

Definition 5.4.4. A Riemannian manifold $\mathcal{M}$ is called **two-point homogeneous** if for any pairs $p_1, p_2 \in \mathcal{M}, q_1, q_2 \in \mathcal{M}$ satisfying $d(p_1, p_2) = d(q_1, q_2)$, there exists an isometry a of $\mathcal{M}$ such that $a \cdot p_1 = q_1$ and $a \cdot p_2 = q_2$.

Proposition 5.4.5. *Let $\mathcal{M}$ be a Riemannian globally symmetric space of rank one (the rank is the maximum dimension of a subspace of the tangent space (to any point) on which the curvature is identically zero). Then $\mathcal{M}$ is a two-point homogeneous space.*

Definition 5.4.6. Let H be a closed Lie subgroup of the Lie group G. Then the **left coset** of an element $a \in G$ is defined as $aH = \{ah : h \in H\}$. The space of all such cosets is denoted by G/H and is a smooth manifold.

Any given symmetric space $\mathcal{M}$ can always be written as a homogeneous space $\mathcal{M} = G/G_p$, where G is a connected group of isometries of $\mathcal{M}$, and the isotropy subgroup G_p is compact.

An element $a \in G$ induces a smooth mapping $\varphi_a : \mathcal{M} \to \mathcal{M}$ via the group action defined as $\varphi_a(p) = a \cdot p$. The mapping φ_a is a diffeomorphism.

Definition 5.4.7. Given a Lie group action of G on a manifold $\mathcal{M}$, a $G-$invariant Riemannian metric $\langle \cdot, \cdot \rangle$ on $\mathcal{M}$ is a metric such that the mapping φ_a is an isometry for all $a \in G$.

Theorem 5.4.8. *Let G be a Lie group acting transitively on a manifold. If for some point $p \in \mathcal{M}$, the isotropy subgroup G_p is a connected, compact Lie subgroup of G, then $\mathcal{M}$ has a $G-$invariant metric.*

Theorem 5.4.9. *Assume G, $\mathcal{M}$ and p satisfy the conditions of theorem 5.4.8. If $\alpha : G \to G$ is a isomorphism of G onto itself with fixed set G_p, then $\mathcal{M}$ is a symmetric space.*

The converse of this theorem is also true, see [Fletcher, 2004] for details.

Theorem 5.4.10. *A connected Lie group G with bi-invariant metric is a symmetric space.*

Summary. We would like to observe that all this abstract setting is important to be able to establish that first, a Lie Group acting transitively by isometries under certain conditions is a symmetric space and second, that for such an space geodesics are computed through the group action. Since G is a group of isometries acting transitively on $\mathcal{M}$, it is necessary to consider only geodesics starting at a base point p. Geodesics are the image of the action of a one-parameter subgroup of G acting on the base point p. This is summarized in the next theorem.

Theorem 5.4.11. *If $\mathcal{M}$ is a symmetric space with $G-$invariant metric, then a geodesic γ starting at a point $p \in \mathcal{M}$ is of the form*

$$\gamma(t) = exp(tX) \cdot p,$$

where X is a vector in the Lie algebra $\mathfrak{g}$.

5.4.1 Notions about Representation Theory

For symmetric spaces obtained from isometric group actions on a manifold there are very useful results that links the representation of the group and operators that are invariant to the group action.

We follow here closely [Sugiura, 1990] for this section, where details can be consulted.

Definition 5.4.12. A **unitary representation** of a group G is a strongly continuous homomorphism π of G into the group $U(H)$ of unitary operators on a Hilbert space H. A mapping is called an **homomorphism** if it satisfies

$$\pi(ab) = \pi(a)\pi(b) \quad (\forall a, b \in G)$$

and an homomorphism is called **strongly continuous** if the mapping $a \rightarrowtail \pi(a)x$ is a continuous mapping from G to H for all $x \in H$.

Definition 5.4.13. A bounded linear operator $T : H \to H$ on a Hilbert space H is called **unitary** if it satisfies:

- T is surjective,
- T preserves the inner product product i.e $< Tx, Ty >_H = < x, y >$.

Definition 5.4.14. Let (π, H) be a unitary representation of a group G. A closed linear subspace V of H is called **invariant under** π if

$$\pi(a)V \subset V \qquad (\forall a \in G).$$

A unitary representation π is called **irreducible** if $H \neq 0$ and H and 0 are the only invariant subspaces of H. Non irreducible unitary representations are decomposed into irreducible representations.

Definition 5.4.15. Two unitary representations (π_1, H_1) and (π_2, H_2) are equivalent if there exists an isometry A of H_1 onto H_2 satisfying

$$A\pi_1(a) = \pi_2(a)A \;\; (\forall a \in G).$$

Definition 5.4.16. Let (π_1, H_1) and (π_2, H_2) be two unitary representations of G. A linear operator $T : H_1 \to H_2$ satisfying

$$T\pi_1(a) = \pi_2(a)T$$

is called an **intertwining operator** between H_1 and H_2.

Proposition 5.4.17. *Let (π_1, H_1) and (π_2, H_2) be two finite dimensional unitary representations of G and T be an intertwining operator between H_1 and H_2. Then, either $T = 0$ or T is a linear isomorphism.*

Proposition 5.4.18. *Let (π, H) be a unitary representation of G. Then a closed subspace V of H is invariant under π if and only if the orthogonal projection P_V on V commutes with $\pi(a)$ for all $a \in G$. In this case the orthogonal complement $V^{\perp}$ is also invariant under π.*

Definition 5.4.19. Let $\mathbf{B}(H)$ be the algebra of bounded linear operators on a Hilbert space H and M be a subset of $\mathbf{B}(H)$. The **commutant** M' of M is defined by

$$M' = \{L \in \mathbf{B}(H) \mid LU = UL \quad \forall U \in M\}.$$

Theorem 5.4.20 ((Schur's Lemma))**.** *Let (π, H) be a unitary representation of G and $M = \{\pi(a) \mid a \in G\}$. Then, π is irreducible if and only if the commutant M' is equal to the set C_1 of scalar operators.*

Corollary 5.4.21. *Any irreducible unitary representation of a commutative group G is one-dimensional*

Theorem 5.4.22. *Any unitary representation π of a compact group G is a (Hilbert space) direct sum of finite dimensional irreducible unitary representations. In particular any irreducible representation of a compact group is finite dimensional.*

We will now introduce a link between group representations and invariant operators to complete our mathematical setting to be used for analyzing the space of simulations in the next chapter. Let's start with the following results,

Proposition 5.4.23 (Mitsuo Sugiura [Sugiura, 1990])**.** *Let $\mathcal{F}$ be the space of complex valued C^{∞}-functions on $\mathbb{R}^3$. Then a representation T_a of the rotation group* SO(3) *on $\mathcal{F}$ is defined by*

$$(T_a f)(x) = f(xa)$$

for $a \in$ SO(3), $f \in \mathcal{F}$ and $x \in \mathbb{R}^3$. The Laplacian

$$\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$$

commutes with T_a for any $a \in$ SO(3)

$$\Delta \cdot T_a = T_a \cdot \Delta.$$

Theorem 5.4.24 (Mitsuo Sugiura [Sugiura, 1990])**.** *The space H_l of harmonic polynomials of degree l is invariant under the representation T_a. Put $T_a|H_l = D^l_a$. Then $D^l : a \rightarrowtail D^l_a$ is a continous irreducible representation of* SO(3)*.*

5.4.2 Conclusions

Assume a linear operator L exists invariant to a transformation group G. The eigenfunctions of this operator are preserved with respect to this transformation group. In addition one can associate a linear transformation T with elements of the group G say a_1, a_2 in the space of eigenfunctions so that,

$$T(a_1)T(a_2) = T(a_1 a_2).$$

T is called a group representation and it establishes a connection between the eigenfunctions of the linear operator with representations of groups to which the operator is invariant.

According to Sigurdur Helgason [Helgason, 1984], there is a special case where the eigenspaces of the Laplacian on a compact homogeneous space are irreducible representations. Assume we have a compact Riemannian manifold $\mathcal{M}$ on which a compact Lie group G acts isometrically and transitively. The manifold can be thought of as G/K, where K is a closed subgroup of G. According to Helgason: "For G/K symmetric the joint eigenspaces of the G-invariant differential operators on G/K are all irreducible", for details see theorems 4.3 and 3.5 of [Helgason, 1984]. "If G/K is two-point homogeneous, then the G-invariant differential operators on G/K are all polynomial in the operator" (see [Helgason, 1984], page 288).

These results are certainly very interesting since they establish that a group can be "represented" by means of an invariant operator. Indeed, for the rotation group *SO(3)*, it states that one can find a representation in terms of the spherical harmonics functions and those are actually the eigenfunctions of the Laplace-Beltrami operator on S^2 (see [Sugiura, 1990] for details).

The link between an invariant operator and the group representation is of highly theoretical interest, the development of the theory that justifies its uses in applications is open and not within the scope of this thesis. Nevertheless we will use the idea in applications constructing approximations of invariant operators. As we will see in the next chapters we provide numerical evidence that demonstrate that group actions can be represented by projections to the eigenfunctions of isometry invariant operators.

Let us summarize what has been stated along the series of theoretical results presented in this chapter. First a quotient space of a manifold $\mathcal{M}$ with respect to an isometric group G action can be built which can be given a Riemannian structure; this allows to define a metric in the quotient space. Second, under certain conditions, geodesics on the space of isometries are determined through the group actions (geodesics are the image of the action of one-parameter subgroup of the isometry group G). Third using representation theory, a representation of the group can be achieved, also under certain conditions, using eigenspaces of operators that are invariant to the transformation group G. This suggests the possibility of using the projections to the eigenspaces as a representation of the group actions. Notice also that unitary representations and invariant operators are known especially in Quantum Physics. In this context we mention the work in [Heine, 1993], where it is shown that the eigenfunctions of the Hamiltonian are irreducible representations of symmetry groups. This implies or suggests the possibility of also using other types of invariant operators and its eigenfunctions.

In the next chapter we will concentrate on the mathematical characterization of simulation results, considered as being obtained as transformations of a reference simulation, and how to construct invariant operators and the projections of the simulations to the eigenfunctions.

CHAPTER 6

Simulations Space: Group Transformations and Invariant Operators

In the previous chapter we have described the mathematical framework that allows us to analyze abstract spaces making use of the algebraic structure of a transformation group acting on it. In other words one can focus on the transformations to study an abstract space. This idea will be used in our task of characterizing simulations as numerical solutions of partial differential equations.

A remarkable result from the theoretical treatment of differential equations (ordinary and partial) is the use of Lie group symmetry principles, where a continuous transformation is found that transforms solutions into solutions. This is a very powerful theoretical framework that allows to obtain analytical solutions of differential equations. We let ourselves be inspired by Lie group symmetry principles for characterizing the space of simulations.

Lie group symmetry principles are based on finding a way to represent solutions of differential equation as orbits that depend on group parameters. An orbit of solutions curves is a continuous family of solutions parametrized along the group parameters. Methods using Lie group symmetries actually employ such orbits as candidates of new coordinate systems along which the differential equations become easy to integrate. For specific types of differential equations, there are analytical methods to find such coordinates or orbits, for details see [Olver, 1993]. In the context of the analysis of simulation space we only have the simulations as discrete solution of some differential equation, we do not know about the form of the PDE nor are we able to do any transformation on it, nevertheless Lie groups ideas still can be used.

We characterize simulations as members of a quotient space under a transformation group, the question we ask ourselves is whether simulations allow a parametrization along group parameters as done in Lie group symmetry methods for differential equations. Interestingly enough, the answer to this question, turns out to be, under certain conditions, yes.

To address this challenge, we would like to recall the general framework as defined in chapter 2. Finding a PDE solution can be formulated as follows, given a parameter set $\boldsymbol{\mu} \in \mathcal{P} \in \mathbb{R}^p$, find $u(\boldsymbol{\mu}) \in V$ such that

$$G(u(\boldsymbol{\mu}), \boldsymbol{\mu}) = 0 \quad \text{in } V',$$

where $G : V \times \mathcal{P} \to V'$ is a parametrized mapping representing the nonlinear PDE.

The so called fidelity discrete variation formulation corresponding to it can be written as,

$$\langle G(u_h(\boldsymbol{\mu});\boldsymbol{\mu}), v_h \rangle = 0 \quad \forall v_h \in V_h.$$

The PDE solution space is defined as,

$$\varphi(\mathcal{P}) = \{u(\boldsymbol{\mu}) \in V : \boldsymbol{\mu} \in \mathcal{P} \subset \mathbb{R}^p\}$$

where $\varphi : \mathcal{P} \to V, \quad \boldsymbol{\mu} \mapsto u(\boldsymbol{\mu})$ is the exact solution map, and the corresponding high fidelity discrete simulation space is given by,

$$\varphi_h(\mathcal{P}) = \{u_h(\boldsymbol{\mu}) \in V_h : \boldsymbol{\mu} \in \mathcal{P}\} \subset V_h.$$

For characterizing this simulation space, we will consider it as an abstract quotient space made of equivalent classes. Such equivalent classes are defined by isometric group actions, so called orbits, sending simulations to simulations. A representation of the group G is obtained by means of a $G-$invariant operator that can be constructed provided a transformation preserving distance can be defined. For a justification of this approach see section 5.4.1. Once the operator is evaluated, the orbits can be obtained as projections to the eigenbasis of the invariant operator. The group actions are represented at the different components of the orthogonal decomposition. For relevant examples it is shown, that those actions are concentrated on just few components so that a dimensionality reduction can also be achieved in this way.

Treating spaces of surfaces or curves as a quotient space over a group acting by isometries is a procedure used in the context of shape spaces [Michor and Mumford, 2006, Bauer et al., 2011]. In this thesis we assume simulations to be surfaces or curves where an isometric transformation takes place, therefore we can consider them as shapes in the general framework of shape spaces. Parts of this chapter have been presented in [Iza Teran and Garcke, 2016].

6.1 Simulation Space as Quotient Space over Isometries

The theoretical framework of shape spaces assumes that the transformation group G acts by isometries. So we have concentrated on them for the following treatment and left the case of non-isometric transformations as open for further study.

We will characterize the space of time dependent solutions of partial differential equations or ordinary differential equations (PDEs, ODEs). It is obtained by changes on boundary and/or initial conditions on a given fixed domain Ω. Further we assume the following,

- solutions $u(\boldsymbol{\mu})$ are surfaces $\mathcal{S}$ embedded (immersed) in $\mathbb{R}^3$ or are curves embedded (immersed) in $\mathbb{R}^2$
- solutions $u(\boldsymbol{\mu})$ are considered to be isometries, i.e. at each time step the solution is isometric to a reference one.

We will describe the space of solutions of an ODE/PDE as a quotient space. We would like to use the general Riemannian framework of shape spaces [Bauer et al., 2011] for this. That is, we will consider each simulation to be a shape and specify a construction for a quotient space of the space of smooth embeddings (immersions) of $\mathcal{S} \to \mathbb{R}^3, \mathcal{S} \to \mathbb{R}^2$ modulo a transformation group $G(\mathcal{S})$.

The abstract setting of [Michor and Mumford, 2006, Bauer et al., 2011] considers a pre-shape space as the space of embeddings (immersions) of $\mathcal{S}$ into $\mathbb{R}^n$ as follows,

Proposition 6.1.1. *[Bauer et al., 2011] The pre-shape space can be characterized as Frechet manifolds given by*

$$Emb(\mathcal{S}, \mathbb{R}^n) \subset Imm_f(\mathcal{S}, \mathbb{R}^n) \subset Imm(\mathcal{S}, \mathbb{R}^n) \subset C^\infty(\mathcal{S}, \mathbb{R}^n)$$

where dim $\mathcal{S} \leq n$, $Emb(\mathcal{S}, \mathbb{R}^n)$ is the set of embeddings of $\mathcal{S}$ in $\mathbb{R}^n$, $Imm_f(\mathcal{S}, \mathbb{R}^n)$ is the set of free immersions of $\mathcal{S}$ in $\mathbb{R}^n$, $Imm(\mathcal{S}, \mathbb{R}^n)$ is the set of all immersions (free and non-free) and $C^\infty(\mathcal{S}, \mathbb{R}^n)$ is the set of smooth functions from $\mathcal{S}$ to $\mathbb{R}^n$.

Finally the shape space itself is defined as a quotient space,

Proposition 6.1.2. *[Bauer et al., 2011] Let $Imm(\mathcal{S}, \mathbb{R}^n), Emb(\mathcal{S}, \mathbb{R}^n)$ be the space of immersions and embeddings of $\mathcal{S}$ in $\mathbb{R}^n$, respectively. Let $G(\mathcal{S})$ be a transformation group that acts smoothly on $Imm(\mathcal{S}, \mathbb{R}^n)$ and $Emb(\mathcal{S}, \mathbb{R}^n)$ by composition from the right. Also the action of $G(\mathcal{S})$ is by isometries. Then the quotient spaces*

$$Imm(\mathcal{S}, \mathbb{R}^n)/G(\mathcal{S}) \quad Emb(\mathcal{S}, \mathbb{R}^n)/G(\mathcal{S})$$

characterize the space of shapes and it is an infinite dimensional orbifold with a global quotient space topology. For the case of embeddings the space is a manifold.

Shape spaces for simulations are good candidates for the right theoretical framework. It considers continuous group actions which involves the use of an infinite dimensional setting, a highly interesting theoretical topic which has been researched actively in the last years [Kriegl and Michor, 1997], [Michor and Mumford, 2006]. Considering simulations in an infinite dimensional setting is certainly theoretically interesting, but it is beyond the scope of this thesis. For applications, to make things tractable, it is usually assumed that transformations are discrete [Srivastava et al., 2011]. For our case of simulations, they are obtained solving a PDE for a discrete set of parameters and therefore a discrete group action could also be assumed. Under this conditions, the space of simulations can, according to the discussion above, be characterized in a very general form as the quotients,

$$Emb(\mathcal{S}, \mathbb{R}^n)/G(\mathcal{S}) \quad Imm(\mathcal{S}, \mathbb{R}^n)/G(\mathcal{S}). \tag{6.1.1}$$

The specific character of this quotient (manifold or orbifold) will depend on the type of discrete group action. This quotient space will contain orbits of isometries of $\mathcal{S}$. That is, for example, we will have orbits of rotations along specific axis, orbits of translations, orbits of global deformations of different types and orbits of local deformations and so on. How to concretely construct those orbits and then evaluate geodesic paths on such space will be the topic of the next section.

6.2 Geodesics on the Quotient Space

For practical applications we also need a way to define a path along the above defined quotient spaces, let us use for illustration the space $Imm(\mathcal{S}, \mathbb{R}^n)/G(\mathcal{S})$, $n = 2, 3$. In this section we will explain how, starting from a way to determine geodesics on the pre-shape space $Imm(\mathcal{S}, \mathbb{R}^n)$, one can still obtain geodesics in the quotient space. What is actually needed is a way to evaluate a distance between orbits in the quotient space. The procedure requires two steps:

1. define a Riemannian metric in the space $Imm(\mathcal{S}, \mathbb{R}^n)$, this defines the length of infinitesimal deformations of an object, for simplicity think of it as a surface or as a curve suffering a deformation, the metric is denoted by $\langle h, k\rangle_f$ with $f \in Imm(\mathcal{S}, \mathbb{R}^n)$ (the space of immersions) and h, k are vector fields $T_f Imm(\mathcal{S}, \mathbb{R}^n)$, $n = 2, 3$. After this metric has been defined, a computation of geodesics in the space of immersions of $\mathcal{S}$ in $\mathbb{R}^n$ can be realized. This involves a minimization of

$$d_{Imm}(f_1, f_2 \circ a) = \min_{\substack{F:[0,1]\to Imm(\mathcal{S},\mathbb{R}^n)\\ F(0)=f_1, F(1)=f_2}} \left(\int_0^1 (\langle F'(t), F'(t)\rangle)^{1/2} dt) \right),$$

   where f_1, f_2 are elements in $Imm(\mathcal{S}, \mathbb{R}^n)$ and F is a path in $Imm(\mathcal{S}, \mathbb{R}^n)$ indexed by t

2. the metric is chosen invariant under the action of the transformation group and the action is assumed to be free, as a consequence the quotient under this action is a Riemannian submersion. Because of that, a Riemannian metric in the quotient space is obtained. For any $f \in Imm(\mathcal{S}, \mathbb{R}^n)$ those vectors in $T_f Imm(\mathcal{S}, \mathbb{R}^n)$ that are perpendicular to the $G(\mathcal{S})$ orbit through f (with respect to the metric in $Imm(\mathcal{S}, \mathbb{R}^n)$) are called horizontal. As a consequence the geodesic distance between two elements of the quotient space $[f_1], [f_2]$ is given by

$$d_Q([f_1], [f_2]) = \min_{a \in G} (d_{Imm}(f_1, f2 \circ a))$$

that is, finding a geodesic on a quotient space requires the solution of two minimization problems. For shape spaces this approach has been extensively studied. For general references see [Bauer et al., 2014], for the special case of plane curves see [Srivastava et al., 2011, Michor and Mumford, 2006] and for the case of surfaces with genus zero see [Kurtek et al., 2012], [Bauer et al., 2011]. Very important for evaluating this distances is how a parametrization invariant metric is defined.

We observe that for applications using a conceptual framework of representing simulations by group actions, requires a method to represent such actions.

6.3 Group Representations and Invariant Operators

We have stated in section 5.4.1 that finding a linear group representation is possible and not only that, under some conditions as detailed in section 5.4.1, it is even possible to use the eigenfunctions of a group invariant operator as a group representation. A particular example of this link has also been illustrated in that section, the space of harmonic polynomials (eigenfunctions of the Laplace-Beltrami operator on the sphere) of degree l is invariant under a linear group representation and is a continuous irreducible representation of the rotation group *SO(3)*. We would like to focus on this connection between the eigenfunctions of the linear operator with representations of a group to which the operator is invariant. It might be natural to ask the question if such a construction works for other than the rotation group, considering other isometric group actions. Is it possible to say something about the eigenfunctions as unitary representations of the group? Are they irreducible?. These questions are still open, at least to our knowledge and they are very interesting from a theoretical point of view, considering them in our thesis is unfortunately out of scope for the analysis of simulations. Nevertheless the idea of using eigenfunctions of invariant operators is very interesting and very useful for applications. Indeed, as it has been described in section 5.4, for a Lie group acting transitively on a manifold a symmetric space can be defined and geodesics in this space are computed through the group actions! To be precise, they are the image of the action of a one-parameter subgroup of G acting on the base point p, see Theorem 5.4.11. We hope the reader realizes what that means, the calculation of geodesics in the high dimensional space of simulations is very expensive and the statements above suggest a new way to calculate geodesics based on a group representation using eigenfunctions of an invariant operator. Specifically given a simulation bundle assumed to undergo only isometric transformations an operator invariant to those transformations can be constructed, the eigenfunctions can then be used as a basis for all simulations. The projection coefficients to this new basis should, according to the theory, represent isometric group actions and even more for the rotation group and the translation group they are geodesics in the space of simulations! In the course of this thesis, numerical evidence will be provided that shows that the rotation and translation group are represented in the new basis. In the case of plane curves, geodesics in this space are demonstrated to be obtained by using the projection coefficients to the new basis as a representation of the group actions.

Finally, we would like to mention an astonishing link between the orbit paths and geodesics which has a very interesting formal aspect. Vladimir Arnold made the observation (see [Mumford and Michor, 2013] and references therein) that many basic equations in physics including the Euler equation for rigid body motions as well the Euler equation for incompressible fluid dynamics can be seen as geodesic flows on a Riemannian manifold. In addition this manifold is a Lie group manifold with a left invariant metric. Under this condition, a geodesic in the manifold is a one-parameter group orbit. This gives a link from Lie group transformations to geodesics on a manifold. In this thesis we will actually deal with orbits that are the result of group actions!, we hope the reader will find such a link as interesting as we found it from the theoretical as well as from the practical point of view.

6.4 Invariant Operators

An important component of our approach is the use of so called invariant differential operators and their discrete approximations which have a special behavior under isometric transformations, or even more general ones.

Invariant operators, as will be seen, can be constructed by using a distance (metric) with respect to which the invariance is measured. That is, as long as we are able to define a distance that does not change for a certain type of transformation, we can construct such operators using only one (arbitrary) simulation. This implies that the operator built in this form is the same for all simulations. The operator will then depend on the metric used. In practice, each time a different metric is used for the construction of the operator, we obtain a different operator.

Before going into the details of the construction of relevant invariant operators, let us recall that simulations are numerical approximations of ODEs or PDEs and are therefore discrete, so even if the proposed method is described on a continuous setting, for practical applications we will need to resort to and take into account the discrete setting.

6.4.1 Simulation Discrete Setting

Simulations are discrete representations of the solution of a PDE on a domain and we would like to construct discrete invariant operators corresponding to the continuous counterparts. We consider industrial simulations as consisting of finite element numerical solutions on complex domains represented by a mesh. But also, more generally, we assume that PDE solutions are given by values evaluated at specific points over a domain (a point cloud) without necessarily taking into account any specific mesh representation. Nevertheless, using the mesh is handy for the construction of invariant operators as will be seen in the following sections. For the sake of simplicity we will treat surface meshes as approximations of our simulations, in general volume meshes could be treated as well, but surface meshes are typically used in the application domains we will consider

Simulation are defined on meshes K that are approximations of a 2-manifold $\mathcal{S}$ isometrically embedded in $\mathbb{R}^3$. The following definition is one way to quantify how well a mesh K approximates a manifold surface $\mathcal{S}$, see [Belkin et al., 2008] for details.

Definition 6.4.1. Let K be a meshed surface approximating $\mathcal{S}$, where the vertices of K are on $\mathcal{S}$. We say that K is an (ϵ, η) approximation of $\mathcal{S}$ if the following two conditions are fulfilled:

- For a face t in K, the maximum distance between any two points on t is at most $\epsilon\rho$, where ρ is the *reach*, defined as the infimum of the local distance between any point w in $\mathcal{S}$ and the medial axis of the surface $\mathcal{S}$.
- For a face t in K and a vertex $p \in t$, the angle between n_t, the unit outward normal of the plane passing through t, and n_p, the unit outward normal of $\mathcal{S}$ at p, is at most η.

Furthermore, we have restrictions of continuous functions on $\mathcal{S}$ to the mesh K:

Definition 6.4.2. Let $f : \mathcal{S} \to \mathbb{R}$ be a continuous function on $\mathcal{S}$. The function f evaluated at the nodes of a mesh K is called a mesh function $f|_K : K \to \mathbb{R}$.

The approximation conditions above for a mesh to represent a surface can be stated as: first assuring one has a fine enough mesh to represent a shape and secondly that the curvature is also well represented. These conditions are also shown to give approximation conditions under which the graph distance on the mesh d_G approximates well the geodesic distance $d\mathcal{S}$ on the surface. We will present a formal result about it in the following section.

6.4.2 Approximating the Geodesic Distance on a Mesh

Approximation results under which the graph distance d_G approximate the geodesics distance d_S on a manifold are given in [Bernstein et al., 2000], [Garcke, 2016]. The proof uses an intermediate approximation to the geodesic distance called d_P that is made of piecewise sections constructed using points on the manifold S. The proof that $d_G \approx d_S$ is then naturally divided into two parts: $d_S \approx d_P$ and $d_P \approx d_G$. For assuring d_P to be a good approximation to d_S so called sampling condition were introduced in [Bernstein et al., 2000]. For assuring that d_G approximates d_P, an intermediate result is used that relates the geodesic distance and the Euclidean distance on the graph. The combination of both parts is finally used to prove the main result relating those three distances.

We provide in this thesis a variation of the above mentioned result for the case of meshes approximating surfaces in $\mathbb{R}^3$. For this purpose the so called (ϵ, η) approximation condition is used [Belkin and Niyogi, 2008]. This assure that a mesh approximate a surface well. We use this condition to firstly verify that it obeys the sampling condition from the original setting in [Bernstein et al., 2000] and as a consequence d_P approximates d_S arbitrary well provided the (ϵ, η) approximation condition is verified. Secondly an alternative intermediate result to the one provided in [Bernstein et al., 2000], which relates the geodesic distance and the Euclidean distance on the graph is derived. The combination of both parts using the same arguments as for the proof of the main theorem in [Bernstein et al., 2000] completes the proof.

Let us first present some relevant results we need, starting with the sampling condition theorem from [Bernstein et al., 2000]:

Theorem 6.4.3. *Let $\bar{\epsilon}$ and δ be positive, with $4\delta < \bar{\epsilon}$. Suppose:*

1. *The graph $G = (V, E)$ contains all edges $(p, q) \in E$ for which $d_S(p, q) \leq \bar{\epsilon}$.*
2. *For every point m in S there is a data point p_i for which $d_S(m, p_i) \leq \delta$.*

Then for all pairs of data points $p, q \in V$ we have

$$d_S(p, q) \leq d_P(p, q) \leq (1 + 4\delta/\bar{\epsilon})d_S(p, q)$$

In the above theorem $d_G(p, q)$ and $d_P(p, q)$ are distances defined as:

$$d_G(p, q) = \min_P (d_E(p_0, p_1)\| + \cdots + d_E(p_{k-1}, p_k))$$

$$d_P(p, q) = \min_P (d_S(p_0, p_1) + \cdots + d_S(p_{k-1} - p_k))$$

where $P = (p_0 \ldots, p_k)$ varies over all paths along the edges of G connecting $p(= p_0)$ to $q(= p_k)$ and d_E denotes the Euclidean distance.

Also useful is the following proposition from [Bernstein et al., 2000].

Proposition 6.4.4. *We have the inequalities:*

$$\begin{aligned} d_S(p,q) &\leq d_P(p,q) \\ d_G(p,q) &\leq d_P(p,q) \end{aligned}$$

Further estimates are available which bounds the distance between the surface and any point k on the mesh K and the distance between any point in the surface, $p \in S$ and the mesh.

Lemma 6.4.5 ([Belkin et al., 2008]). *If a mesh K (ϵ, η)-approximates S with $\epsilon, \eta < 0.1$ then the following conditions hold:*

1. *For any point $k \in K$, $d_E(k, S) \leq (\epsilon^2 + \eta\epsilon)\rho$.*
2. *For any point $p \in S$, $d_E(p, K) \leq (\epsilon^2 + \eta\epsilon)\rho$.*
3. *$\phi : K \rightarrow S$ is an homeomorphism.*

where $d_E(k, S)$ is the distance from the point k on the mesh to the surface S and $d_E(p, K)$ is the distance from the point p on the surface to the mesh K.

Now we will analyze the (ϵ, η) approximation condition and see whether a set of points satisfying it, also satisfy the sampling condition.

Proposition 6.4.6. *Let the mesh K approximate a surface satisfying the (ϵ, η) approximation conditions. Then points on the mesh also satisfy the sampling condition as given in theorem 6.4.3*

Proof. We will start with item 1 in theorem 6.4.3. Let p, q be points on the surface S which are also points on the mesh K and let G be the graph constructed with those points. According to the (ϵ, η) approximation condition, the maximum Euclidean distance between any two points in a face is at most $\epsilon\rho$. Therefore the graph G constructed using the condition $d_S(p, q) \leq \epsilon\rho$ with $\bar{\epsilon} = \epsilon\rho$, contains all edges for which this condition is fulfilled. This completes the proof for item 1.

Item 2 establishes a closeness condition between every point in S and a point p_i on the mesh. The geodesic distance between a point m in S and a point on the mesh p_i can be bounded using two times the Euclidean distance between those points. This distance in turn can be bounded by the sum of the distance to the mesh from any point on the surface (given by item 2 in Lemma 6.4.5) and the distance between any point on a face and the point p_i which is bounded by the (ϵ, η) approximation condition.

In detail, let us start using the following result from [Niyogi et al., 2008].

$$d_S(m, p_i) \leq 2d_E(m, p_i).$$

The distance $d_E(m, p_i)$ can be bounded by

$$d_E(m, p_i) \leq d_E(m, k) + d_E(k, p_i)$$

where $d_E(m,k)$ is the distance between a point m in the surface and the point k on a face that is nearest to the point m. The expression $d_E(k,p_i)$ corresponds to the distance between the points k and p_i on a mesh face.

Notice that the distance $d_E(m,k)$ is per definition the distance between a point on the surface and the mesh K so that the bound in item 2 from lemma 6.4.5 applies. Further notice that the distance $d_E(k,p_i)$ can be upper bounded using the (ϵ,η) approximation condition, using again that the distance between any two points is at most $\epsilon\rho$. In detail this means:

$$d_S(m,p_i) \leq 2(d_E(m,k) + d_E(k,p_i)),$$

using item 2 of lemma 6.4.5 and the (ϵ,η) approximation, we can write the estimate

$$d_S(m,p_i) \leq 2\rho(\epsilon^2 + \epsilon\eta + \epsilon) = 2\bar{\epsilon}(\epsilon + \eta + 1).$$

Choosing $\delta = 2\rho(\epsilon^2 + \epsilon\eta + \eta)/16 = 2\bar{\epsilon}(\epsilon + \eta + 1)/16$ then

$$4\delta = \frac{\rho(\epsilon^2 + \epsilon\eta + \eta)}{2} = \frac{\bar{\epsilon}(\epsilon + \eta + 1)}{2} < \bar{\epsilon}$$

for ϵ,η small enough. In this way according to theorem 6.4.3 the sampling condition is verified for $\bar{\epsilon},\delta$ and ϵ,η small enough. □

Once the sampling condition is verified, we can assert that d_P is a good approximation of d_S. For assuring that d_G approximates well d_S we first need the following result which is a slight modification of lemma 3.2 in [Belkin et al., 2008].

Lemma 6.4.7. *Given a positive real number λ such that $\lambda < 1$ and two points p,q on the surface S, let $d_E(p,q) = \|p-q\| < \rho/2\sqrt{6\lambda}$. Then we have that*

$$(1-\lambda)d_S(p,q) \leq d_E(p,q) \leq d_S(p,q)$$

Proof. We follow the steps as given in the proof of lemma 3.2 in [Belkin et al., 2008] and obtain the following inequality

$$d_S(p,q) \leq d_E(p,q) + \frac{d_S(p,q)^3}{6\rho^2}$$

Using the condition $d_E(p,q) = \|p-q\| < \rho/2\sqrt{6\lambda}$ and that $d_S(p,q) \leq 2d_E(p,q)$ [Niyogi et al., 2008], we obtain $d_S(p,q) < \rho\sqrt{6\lambda}$. Now the inequality can be written as

$$(1 - \frac{d_S(p,q)^2}{6\rho^2})d_S(p,q) \leq d_E(p,q),$$

we can upper bound $d_S(p,q)^2$ using $d_S < \rho\sqrt{6\lambda}$ finally obtaining

$$(1-\lambda)d_S(p,q) \leq d_E(p,q)$$

The right hand side of the inequality in the lemma is just only the fact that the line segment between p and q is the shortest arc connecting p and q. Finally we obtain the desired result

$$(1-\lambda)d_S(p,q) \leq d_E(p,q) \leq d_S(p,q)$$

□

Finally we can formulate an analog main theorem to the one in [Bernstein et al., 2000], [Garcke, 2016] as follows

Theorem 6.4.8. *Let $G = [Y, E]$ be a graph defined on a mesh K that (ϵ, η) approximates well the surface S, given as a compact convex sub-manifold of R^3 and let $Y \subset S$. For given two numbers $0 < \lambda_1, \lambda_2, \bar{\delta} = \lambda_2\delta$, the following holds*

$$(1-\lambda_1)d_S(p,q) \leq d_G(p,q) \leq (1+\lambda_2)d_S(p,q)$$

Proof. According to proposition 6.4.3, a mesh K which (ϵ, η) approximates the surface S also satisfies the sampling condition so that theorem 6.4.3 applies and with proposition 6.4.4 we obtain

$$d_G(p,q) \leq d_P(p,q) \leq (1+4\bar{\delta}/\bar{\epsilon}) \leq (1+\lambda_2)d_S(p,q).$$

Now as given in [Bernstein et al., 2000], [Garcke, 2016], choose a path $p_0p_1\cdots p_k$ connecting $p(=p_0)$ to $q(=p_k)$ along graph edges which minimizes the total graph length. For the graph defined on a mesh which (ϵ, η) approximates a surface S, we can use lemma 6.4.7 for each pair p_i, p_{i+1} with $\lambda = \lambda_1$. Then:

$$\begin{aligned} d_S(p, q>) &\leq d_S(p_0,p_1) + \cdots + d_S(p_{k-1},p_k) \\ &\leq (1-\lambda_1)^{-1}d_E(p_0,p_1) + \cdots + (1-\lambda_1)^{-1}d_E(p_{k-1},p_k) \\ &= (1-\lambda_1)^{-1}d_G(p,q). \end{aligned}$$

This proves the left-hand inequality. □

Now we can start to show how to construct an invariant operator that already includes several types of distance preserving transformations, namely the Laplace-Beltrami operator.

6.4.3 The Laplace-Beltrami Operator

Let us start with the Laplace-Beltrami operator and recall some of the concepts already given in chapter 3. The operator is defined as

$$\Delta_g : C^\infty \to C^\infty$$

$$\Delta_g = -div_g \cdot \nabla_g$$

Specifically we consider the operator as defined for a surface $\Delta_{\mathcal{S}} := \Delta_g|_{\mathcal{S}}$. The eigenvalue problem $\Delta_{\mathcal{S}}\psi = -\lambda\psi$ is restricted to the manifold $(\mathcal{S}, g)$, where λ is an eigenvalue of $\Delta_{\mathcal{S}}$ and ψ is the corresponding eigenfunction. Since the operators we consider are positive definite [Rosenberg, 1997], all eigenvalues λ_k, $k \geq 0$ are real, positive, and isolated with finite multiplicity. The set of eigenfunctions

$\{\psi_i\}$ of the operator forms an orthogonal base for functions on $\mathcal{S}$, therefore the following decomposition can be written for $f \in C^\infty(\mathcal{S})$:

$$f = \sum_{i=0}^{\infty} \alpha^i \psi_i, \quad \alpha^i = \langle f, \psi_i \rangle_g \tag{6.4.1}$$

For any function f as defined by (6.4.1) one can instead of considering the function itself, equivalently consider the vector $\alpha = [\alpha^1, \alpha^2, \ldots]$ of spectral coefficients obtained by projecting the function along the infinite dimensional eigenspace spanned by the eigenfunctions. Those coefficients can be used as coordinate functions instead of the original ones as follows:

Proposition 6.4.9. *The difference between two functions $f_1 = \sum_{i=0}^{\infty} \alpha_1^i \psi_i$, $\alpha_1^i = \langle f_1, \psi_i \rangle$ and $f_2 = \sum_{i=0}^{\infty} \alpha_2^i \psi_i$, $\alpha_2^i = \langle f_2, \psi_i \rangle$ using a decomposition with the corresponding eigenfunctions for an operator $\Delta_g(\mathcal{S})$, is given by*

$$\|f_1 - f_2\|^2 = \sum_{i=0}^{\infty} \langle \alpha_1^i - \alpha_2^i, \psi_i \rangle^2 = \|\alpha_1 - \alpha_2\|^2$$

where α_1, α_2 are the spectral coefficients for the orthogonal decomposition.

Proof. to show the above just use the Parseval's identity which establishes that

$$\|f_1 - f_2\|^2 = \|\alpha_1 - \alpha_2\|^2.$$

□

6.4.4 The Laplace-Beltrami Operator on a Mesh

Let $\mathcal{S}$ be a 2−manifold (a surface) isometrically embedded in $\mathbb{R}^3$ and let $\Delta_{\mathcal{S}}$ be the Laplace-Beltrami operator on this manifold. Under the assumption that the mesh K is a (ϵ, η) approximation of $\mathcal{S}$, the following operator can be defined [Belkin et al., 2008].

Definition 6.4.10. Let the number of vertices on t be denoted by $\#t$ and $V(t)$ be the set of vertices of t, then for any $w \in V$, the mesh Laplace operator is defined as

$$L_K^h f(w) = \frac{1}{4\pi h^2} \sum_{t \in K} \frac{Area(t)}{\#t} \sum_{p \in V(t)} e^{-\frac{d_G(p,w)^2}{4h}} (f(p) - f(w)). \tag{6.4.2}$$

In (6.4.2), $d_G(p, w)$ denotes the graph distance on the mesh (shortest path for a triangular mesh) and h is a parameter which corresponds to the size of the local neighborhood at a point. The graph distance on the mesh approximates the geodesic distance on the surface according to theorem 6.4.8.

In the following theorem from [Belkin et al., 2008] one exploits that locally the Euclidean distance is a good approximation of the geodesic distance to give approximation properties for the Laplace-Beltrami operator on a surface by the mesh Laplace operator L_K^h. Note that in [Belkin et al., 2008] the Euclidean distance $d_E(p, w) = \|p - w\|$ is used in the definition of L_K^h instead of $d_G(p, w)$. Due to the (ϵ, η) approximation conditions, the graph distance is a better approximation of the geodesic distance and therefore, it naturally holds for (6.4.2) as well.

Theorem 6.4.11. (Laplace-Beltrami Approximation Theorem [Belkin et al., 2008]) *Let $K_{\epsilon,\eta}$ be a an (ϵ,η)-approximation of $\mathcal{S}$. Put $h(\epsilon,\eta) = \epsilon^{\frac{1}{2.5+\alpha}} + \eta^{\frac{1}{1+\alpha}}$ for an arbitrary positive number $\alpha > 0$. Then for any function $f \in C^2(\mathcal{S})$ it holds*

$$\lim_{\epsilon,\eta\to 0} \sup_{K_{\epsilon,\eta}} \left\| L^{h(\epsilon,\eta)}_{K_{\epsilon,\eta}} f - \Delta_{\mathcal{S}} f|_{K_{\epsilon,\eta}} \right\|_\infty = 0,$$

where the supremum is taken over all (ϵ,η)-approximations of $\mathcal{S}$.

According to this result, for a mesh fine enough which also approximates the curvature of $\mathcal{S}$ well, we expect to get an approximation of the corresponding continuous Laplace-Beltrami operator on this surface.

6.4.5 The Laplace-Beltrami Operator under Isometric Transformations

We will consider in this section surfaces, that are assumed to be smooth connected complete Riemannian 2-manifolds. The geodesic distance induced by the Riemannian metric on the surface $\mathcal{S}$ is denoted by $d_{\mathcal{S}} : \mathcal{S} \times \mathcal{S} \mapsto \mathbb{R}^+$. Also we are given a discrete surface mesh K that in the continuous limit, is assumed to approximate a surface $\mathcal{S}$.

Further let φ^i be a distance preserving diffeomorphism $\varphi^i : \mathcal{S} \to \mathcal{S}^i$ between two Riemannian manifolds $(\mathcal{S}, g)$ and $(\mathcal{S}^i, g^i)$ and let $\bar{K} = \left\{K^i\right\}_{i=1}^m$ be a set of meshes which are assumed to have the same connectivity and approximate a set of φ^i transformed surfaces $\mathcal{S}^i$, $i = 1, \ldots, m$. The transformation is such that geodesic distances are kept the same. In other words the transformation $\varphi^i : \mathcal{S} \to \mathcal{S}^i$, between two surfaces is an isometry. Then $\mathcal{S}, \mathcal{S}^i$ are called isometric and the restriction to the discrete mesh are assumed to preserve geodesic distances as well. In this ideal case the following diagram commutes.

$$\begin{tikzcd}
\mathcal{S} \arrow[r, leftrightarrow, "\varphi^i"'] \arrow[d] & \mathcal{S}^i \arrow[d] \\
K \arrow[r, leftrightarrow, "\varphi^i_{|K}"] & K^i
\end{tikzcd}$$

Here $\varphi^i_{|K}$ is a discrete version of φ^i.

Notice that in a discrete setting it is a-priori not clear as to how to handle geodesic distance preservation. We can nevertheless accept that an error is made by the evaluation of geodesic distances and make the assumption that this error is kept small. A related concept covering this assumptions has already being introduced through the so called $\varepsilon-$ isometry [Bronstein et al., 2006].

Definition 6.4.12. Given two surfaces $\mathcal{S}, \mathcal{S}^i$ in the set of all surfaces, a transformation $\varphi : \mathcal{S} \mapsto \mathcal{S}^i$ is said to have a distortion given by

$$dis\ \varphi \equiv \sup_{s,s' \in \mathcal{S}} |d_{\mathcal{S}}(s, s') - d_{\mathcal{S}^i}(\varphi(s), \varphi(s'))| \tag{6.4.3}$$

If in addition for every $q \in \mathcal{S}^i$ there exists $s \in \mathcal{S}$ such that $d_{\mathcal{S}^i}(q, \varphi(s)) \leq \varepsilon$, φ is called an **ε-isometry** and the spaces $\mathcal{S}, \mathcal{S}^i$ are called ε-isometric.

A discrete point on a mesh is assumed to be sampled on the surface and we approximate the geodesic distance on the surface by using the graph distance on the mesh. Therefore the discrete nature of the mesh introduces an approximation in the calculation of geodesic distances. For the numerical evaluation of the operator we use a Gaussian kernel (see expression (6.4.2)) for all points on a mesh. A discrete version of the Laplace-Beltrami operator is obtained in this way that, as described previously, converges to the continuous counterpart as one refines the mesh. This approximation error is assumed fixed for a certain mesh size. Now consider another mesh subject to an $\varepsilon-$isometric transformation, constructing the operator using the graph distance will inevitably introduce an error with respect to the one on the first mesh, under this setting we assume that the error in the evaluation of the operator due to the discrete approximation has a Gaussian noise character. That is we assume that in the evaluation of the discrete operator, one obtains a noisy representation whose noise is sampled from a Gaussian distribution. Under the above assumptions we can assert the following for an $\varepsilon-$isometric transformation.

Proposition 6.4.13.

1. *Let a deformed surface $\mathcal{S}$ be given and let a geodesic distance preserving mapping φ^i transform $\mathcal{S}$ into a series of deformed surfaces $\mathcal{S}^i$, $i = 1, \ldots, m$. The Laplace-Beltrami operator constructed using the geodesic distance in $\mathcal{S}$ is the same for each $\mathcal{S}^i$.*

2. *Let the set of deformed meshes $\bar{K} = \left\{K^i\right\}_{i=1}^m$ be given as approximations of the surfaces $\mathcal{S}^i$ and also let a set of transformations $\varphi^i_{|K}, i = 1, \ldots, m$ be ε-isometric. Further assume that the ε-isometric transformations produces a Gaussian noise perturbation of the metric. Then the approximation of the Laplace-Beltrami operator L^h_K given in (6.4.2), constructed using graph distances for one mesh K, differs only by a scaling factor with respect to each of the ones constructed using the deformed meshes $i = 1, \ldots, m$ in the set $\bar{K}$.*

Proof. Let φ^i be a diffeomorphism $\varphi^i : \mathcal{S} \to \mathcal{S}^i$ between two Riemannian manifolds $(\mathcal{S}, g)$ and $(\mathcal{S}^i, g^i)$, where $g = \varphi^{i*} g^i$. Further let φ^i be geodesic distance preserving, i.e. $d_{g^i}(\varphi^i(p), \varphi^i(q)) = d_g(p, q)$. Notice that the Laplace-Beltrami operator is invariant to geodesic preserving transformations. For any $\mathcal{S}^i$ that is a deformed surface which is obtained from $\mathcal{S}$ using a geodesic distance preserving mapping φ^i, we can therefore follow that the operator can not distinguish $\mathcal{S}$ and $\mathcal{S}^i$ as different manifolds; we just have different charts. As a consequence, the Laplace-Beltrami operator will be the same for all $\mathcal{S}^i$.

For now, consider a set of deformed meshes $\bar{K} = \left\{K^i\right\}_{i=1}^m$, each K^i approximating a manifold $\mathcal{S}^i$. We assume that, through a specific $\varepsilon-$ isometric transformation $\varphi^i_{|K}$, K^i can be obtained from K. Further K^i approximates a manifold surface $\mathcal{S}^i$ and according to 6.4.8, the graph distance evaluated in K^i approximates the geodesic distance in $\mathcal{S}^i$.

In the discrete setting, for each and every new $\varepsilon-$isometric transformation $\varphi^i_{|K}$ we assume that a perturbation of the operator L^h_K (6.4.2) for K is introduced. According to the assumption in the proposition regarding noise, through the approximate evaluation of the geodesic distances using the graph distances, the transformations introduce Gaussian noise η with covariance matrix σ^2 so that

$$d_G(\varphi(s), \varphi(s')) = d_G(s, s') + \eta(s - s'). \tag{6.4.4}$$

For the numerical calculation of the discrete approximation of the Laplace-Beltrami operator L^h_K given in (6.4.2) on the mesh K a Gaussian kernel weight matrix $\tilde{W} := e^{-d_G(s,s')^2/(4h)}$ has to be calculated.

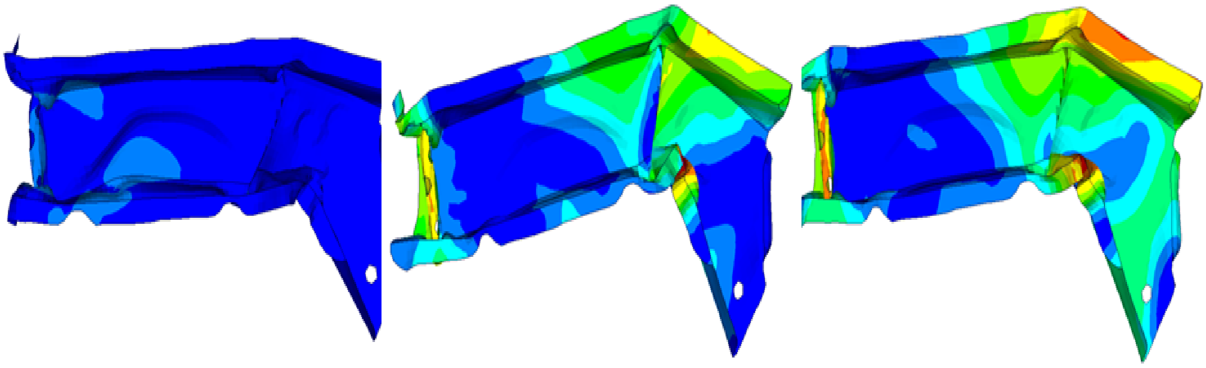


Figure 6.1: Isometric deformations of a car part

For the mesh K^i, a corresponding Gaussian kernel matrix $W := e^{-d_G(\varphi(s),\varphi(s'))^2/(4h)}$ is built instead. The components of these weights matrices can be shown to differ only by a scaling factor [Meyer and Shen, 2014]

$$\tilde{W}_{n,m} = e^{-2(\sigma/\delta)^2} W_{n,m} \tag{6.4.5}$$

where $\delta = 4h$ is the width of the Gaussian kernel. That is, the noise in expression (6.4.4) introduce a perturbation of the weighted kernel matrix of the operator. This argument has already being used for the analysis of the perturbation of the eigenvectors of the graph Laplacian and involves the use of a theoretical result presented in [Karoui, 2010]. The perturbation analysis due to noise as used in [Meyer and Shen, 2014] is analogous to the case treated here, the difference being that in [Meyer and Shen, 2014] the Euclidean distance is used in expression (6.4.4). As for our setting, we assume that the geodesic distance is locally well approximated by the euclidean distance so that the use of expression (6.4.5) can be justified for our case. □

The above result is very interesting, it shows that under some specific assumptions, the approximated operator (6.4.2) can be used for all given $\varepsilon-$isometric transformed meshes. The difference being a scaling factor with respect to the one for mesh K, as long as σ does not vary locally.

Considering simulations which are obtained through the application of an isometric (distance preserving) transformation on a mesh could appear restrictive, but as we have seen small deviations from the isometric case are permissible. In addition it will be seen that even in this setting many applications (see Chapter 9) can be realized. As a motivating example consider deformed surfaces which are obtained from a numerical simulation of a car crash, using different material parameters. The deformations obtained in the example models physical deformations that do not elongate or tear the part. Therefore the car part only deforms but the distances inside the part are preserved, figure 6.1 illustrates these (approximately) isometric deformations.

Notice that, in spite of having evidence that transformations between simulations can be considered as being nearly isometric, there are ways to deal with more general transformations by modifying the distance used in the construction of the operator. We will see this for one example in the next section.

6.4.6 An Operator for Point Clouds

The use of the Laplace-Beltrami operator is already very useful for applications, nevertheless one can certainly think about other types of invariant operators. We will introduce in this section another operator that has been used in the context of nonlinear independent component analysis (NICA) [Singer and Coifman, 2008], and which is valid for a stochastic Itô process. The independent processes are given by

$$dp^{(i)} = \alpha^i(p^{(i)})dt + b^i(p^{(i)})dw^{(i)},\ \ i = 1, \dots N_h,$$

where α^i and b^i are unknown drift and noise coefficients, and $\dot{w}^i$ are independent white noises. In the NICA setting one assumes that one is not able to observe $p = (p^{(1)}, p^{(2)}, \dots, p^{(N_h)})$, but the result of a mapping $\varphi(p)$.

Our application area is the analysis of simulations in engineering, as an example let us take again crash simulations for cars. In industry a finite element model is used that solves the equations of structural mechanics under time dependent deformations. This model is solved for a set of parameters that can vary randomly, this is the case for reliability design or robustness studies. One has some material parameters or some load conditions that vary randomly according to some probability distribution. Whether or not the assumption of an Itô process is valid for this setting is open, but there are approaches in structural dynamics that use such assumption (see [Spanos and Wu, 1994] for the Itô formulation characterizing accumulated structural deformations). In the following we will assume that the operator for the independent components can be used for our application. For its construction, we use the points of the deformed surface meshes from the simulations.

Given N_h points $p^{(1)}, \cdots, p^{(N_h)} \in \mathbb{R}^M$ in the unobservable space $\mathcal{S}^p$, they are mapped to the points $\eta^{(1)}, \cdots, \eta^{(N_h)} \in \mathbb{R}^d$ in another space $\mathcal{S}^\eta$ by a nonlinear transformation φ. This mapping and its inverse can be linearly approximated on a ball $B_\delta(p^{(k)})$ around any given point $p^{(k)}$ by its differential due to the locality of the ball and also assuming that the map is smooth enough. The first order Taylor expansion is then

$$\eta = \varphi(p) = \eta^{(k)} + J_\varphi(p^{(k)})(p - p^{(k)}) + O(\|p - p^{(k)}\|^2), \qquad \text{for } p \in B_\delta(p^{(k)}), \tag{6.4.6}$$

where $J_\varphi(p^{(k)})$ is the Jacobian of φ evaluated at $p^{(k)}$ and also $\eta^{(k)} = \varphi(p^{(k)})$.
This expression allows the evaluation of a first order approximation of the distance

$$\|\eta - \eta^{(k)}\|^2 = \|J_\varphi(p^{(k)})(p - p^{(k)})\|^2 + O(\|p - p^{(k)}\|^3),$$

while for the inverse map φ^{-1} one can write

$$\|p - p^{(k)}\|^2 = \|J_{\varphi^{-1}}(\eta^{(k)})(\eta - \eta^{(k)})\|^2 + O(\|p - p^{(k)}\|^3),$$

see [Singer and Coifman, 2008, Kushnir et al., 2012] for details. According to this approximation, a ball $B_\delta(p^{(k)})$ in $\mathcal{S}^p$ centered at $p^{(k)}$ is therefore mapped to a small ellipsoid in $\mathcal{S}^\eta$ centered around $\eta^{(k)}$. This ellipsoid can be identified with the covariance matrix $C_{k,\delta}$ of its inner points [Singer and Coifman, 2008], i.e.

$$C_{k,\delta} = \mathbb{E}[(\eta - \eta^{(k)})(\eta - \eta^{(k)})^T], \tag{6.4.7}$$

by making use of the linearity of the expectation and using the approximation (6.4.6). It has been proved in [Singer and Coifman, 2008] that

$$J_\varphi(p^{(k)})J_\varphi^T(p^{(k)}) = \frac{d+2}{\delta^2}C_{i,\delta} + O(\delta), \tag{6.4.8}$$

where $\mathcal{S}^p \subset \mathbb{R}^M$, and that a second-order approximation of the local distance $||p - p^{(k)}||^2$ can be written using $C^+_{k,\delta}$, the pseudo-inverse of $C_{k,\delta}$, as

$$||p^{(l)} - p^{(k)}||^2 = \frac{1}{2}\frac{\delta^2}{d+2}(\eta^{(l)} - \eta^{(k)})^T \left[C^+_{k,\delta} + C^+_{l,\delta}\right](\eta^{(l)} - \eta^{(k)}) + O(||p^{(l)} - p^{(k)}||^4) \tag{6.4.9}$$

Using expression (6.4.9), the local distance in the unobservable space can be approximated. We summarize this approximation in the following proposition, see [Singer and Coifman, 2008, Kushnir et al., 2012] for details.

Proposition 6.4.14.

Let $p, w \in \mathcal{S}^p$, and let η, β be their respective mappings to the observable space $\mathcal{S}^\eta$. Then the distance in $\mathcal{S}^p$ (using local coordinates) can be written as:

$$||p - w||^2_{\mathbb{R}^M} = \frac{1}{2}(\eta - \beta)^T \left[(J_\varphi J_\varphi^T)^{-1}(\eta) + (J_\varphi J_\varphi^T)^{-1}(\beta)\right](\eta - \beta) + O(||\eta - \beta||^4_{\mathbb{R}^d}), \tag{6.4.10}$$

where $d \leq M$, and J_ϕ is the Jacobian of the transformation. Using an approximation at the midpoint, the distance in $\mathcal{S}^p$ can be shown to be given by

$$d(p, w) = ||p - w||^2_{\mathbb{R}^M} = (\eta - \beta)^T \left[(J_\varphi J_\varphi^T)^{-1}(\frac{\eta + \beta}{2})\right](\eta - \beta) + O(||\eta - \beta||^4_{\mathbb{R}^d}), \tag{6.4.11}$$

The Jacobian at the midpoint is given by

$$(JJ^T)^{-1}(\frac{\eta + \beta}{2}) = 2[JJ^T(\eta) + JJ^T(\beta)]^{-1} + O(||\eta - \beta||^2)$$

Using the distance in (6.4.11), the matrix operator of size $N_h \times N_h$ can be constructed using the weights

$$A_{p,w} = \exp(-||p - w||^2/\epsilon). \tag{6.4.12}$$

Now the matrix is normalized via

$$W_{r,norm} := D_r^{-1/2} A D_r^{-1/2}, \tag{6.4.13}$$

where D_r is a diagonal matrix with elements

$$D_r^{p,p} := \sum_{p'=1}^{N_h} A_{p,p'}.$$

Further $W_{r,norm}$ is transformed into a row stochastic matrix by

$$W_{r,rs} := D_{rs}^{-1} W_{r,norm} \tag{6.4.14}$$

with D_{rs} the diagonal matrix with elements $D_{rs}^{p,p} = \sum_{p'=1}^{N_h} W_{r,norm}^{p,p'}$. The kernel $W_{r,rs}$ is similar to the symmetric kernel

$$W_s = D_{rs}^{-1} W_{r,norm} D_{rs}^{-1}, \tag{6.4.15}$$

$W_{r,rs}$ with eigenvectors ψ and W_s with eigenvectors ψ_s and share the same eigenvalues. The eigenvectors are related by $\psi_s = D_{rs}^{1/2}\psi$. See [Talmon and Coifman, 2015, Kushnir et al., 2012], for details.

It has been demonstrated in [Singer and Coifman, 2008] that the discrete operator

$$L = D^{-1} W_{r,rs} - I, \tag{6.4.16}$$

with A (6.4.12) evaluated using (6.4.11), converges to a Fokker-Planck operator in the non-observable space $\mathcal{S}^p$. The eigenvectors of the graph Laplacian (6.4.16) approximate also the eigenfunctions in the non-observable space in the limit $N_h \to \infty$, $\epsilon \to 0$

$$L_{\mathcal{S}^p} f = \Delta f - \nabla U \cdot \nabla f, \quad U = -2 \log \mu, \tag{6.4.17}$$

where μ is the density function in $\mathcal{S}^p$. As can be seen from this result, through this construction we are actually recovering an operator in $\mathcal{S}^p$. Additionally, it has been demonstrated that, under some conditions, the top eigenvector corresponding to the top nontrivial eigenvalue of the operator (6.4.16) is actually a function of only the first non-observable variable, the second of the second one and so on, see [Singer, 2006b] for details.

Notice that this construction, not only allows us to obtain the non-observable variables, but it also provides an operator which is invariant to a nonlinear transformation. This constitutes a very important and interesting result for our invariant operator approach.

We propose to use this stochastic setting for the case of simulations as solutions of a PDE model subject to small parameter changes. Given m simulations, they can be assumed to be obtained as a stochastic simulation undergoing a nonlinear transformation from a reference simulation. The transformation is unknown but the realizations for different parameter combinations are known. The data points p are assumed to have been obtained by sampling from an unknown probability distribution μ on the space $\mathcal{S}^p$. To construct the operator of independent components we take the values of the nodes of a surface mesh K, which is defined as before as an approximation of a 2-manifold $\mathcal{S}^p$ isometrically embedded in $\mathbb{R}^3$. A discrete operator for such points can be constructed as an alternative to the approximation of the Laplace-Beltrami operator considered previously.

The operator is valid for one simulation, but for our setting we are considering a simulation bundle where each simulation is assumed to be obtained through a stochastic realization φ^i of a nonlinear transformation φ from a reference simulation, $\varphi^i : \mathcal{S} \to \mathcal{S}^i$. In this setting, a set of m simulations is available to us and our objective is to construct, based only on this information, an operator that is invariant to the nonlinear transformation φ. As explained above, such an operator can be constructed. One starts with one reference simulation and takes all others to form a cloud around each point in the reference one. Then from this cloud, the covariance matrix $C_{k,\delta}$ is used to approximate the term $J_\varphi(p^{(k)}) J_\varphi^T(p^{(k)})$ and its inverse, in order to be able to evaluate expression (6.4.11) locally at each point k of the reference simulation. In this way the discrete operator obtained through expression (6.4.16) is invariant to the given transformation and therefore the same for each and everyone of the given m simulations in the bundle.

Illustrative example

To illustrate the construction of the operator of independent components, we would like to describe a simple example from [Singer and Coifman, 2008]. Let $p = [p_1, p_2]$ with p_1, p_2 being uniformly distributed random variables. We are not able to observe them directly but only after a nonlinear transformations given by $\eta = [\eta_1, \eta_2]$ with $\eta_1 = p_1 + (p_2)^3$ and $\eta_2 = p_2 - (p_1)^3$. See figure 6.2 (a) for a representation of the non-observable variables p and figure 6.2 (b) for the observable η after the nonlinear transformation.

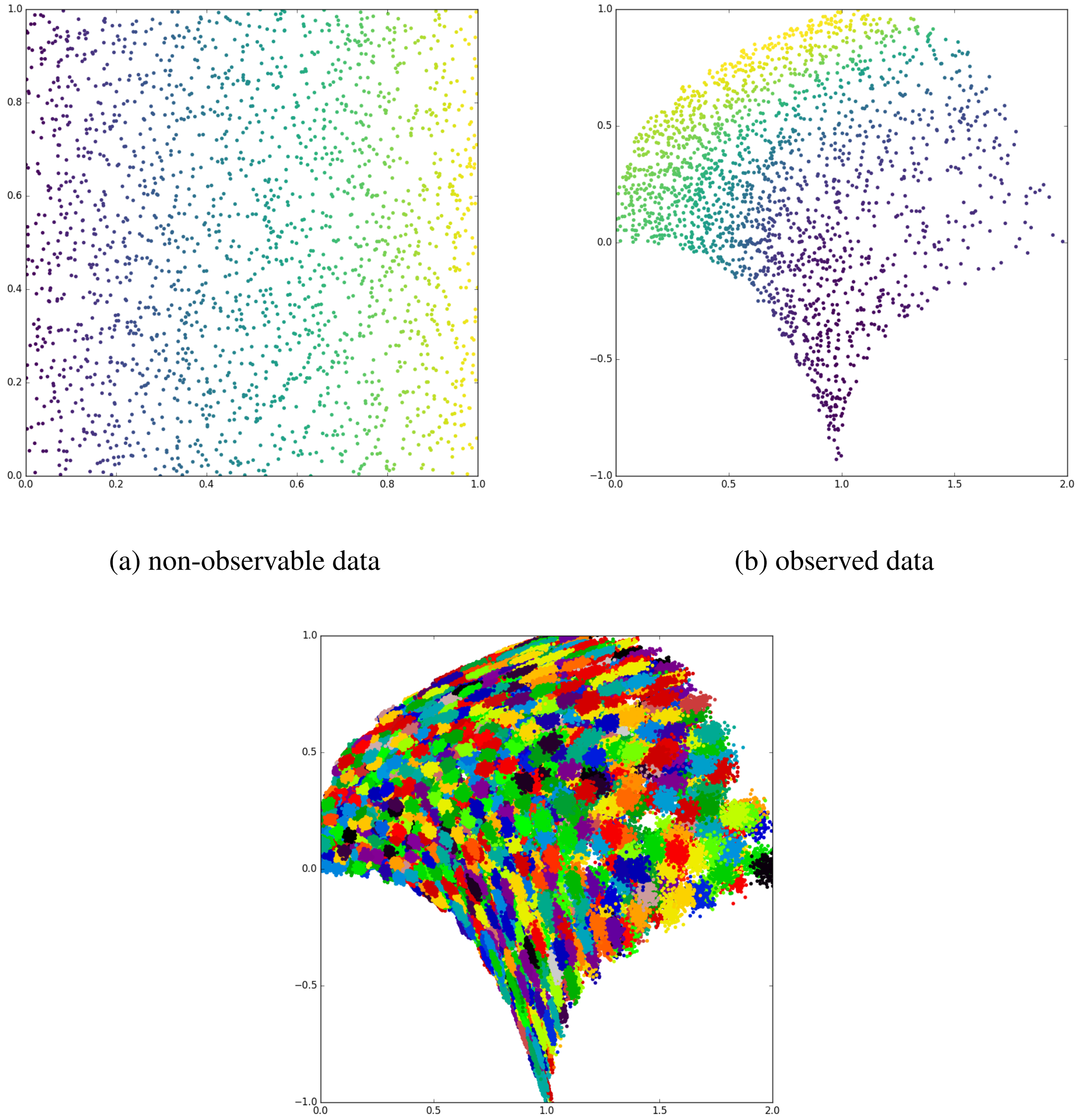
The Jacobian J_φ is not available, but $J_\varphi J_\varphi^T$ and its inverse can be estimated from the data. We use algorithm 3 to evaluate the operator. In the example before, we have $d = M = 2$ and can construct N_h (2×2) sample covariance matrices obtained from a set of m observable simulations in a simulation cloud around each point in $\mathcal{S}^\eta$. The approach assumes that a stochastic simulations can be performed for a small time step, each one starting from a different initial condition. The idea is to create a simulation burst (or cloud) around a starting or nominal configuration, see figure 6.2 (c) for a representation of a simulation burst for our simple example. By using this cloud, one calculates an estimate of the distance between the non-observable variables. Constructing the matrix operator and calculating the eigenvectors of it, one obtains the independent components. In figure 6.2 (a), (b) the color corresponds to the values of the first eigenvector that is, to the first identified independent component for the example. Notice that once the eigenvectors are calculated, they can be used as a common basis for the simulation cloud. All simulations can therefore be projected onto the eigenbasis obtaining as a result the spectral coefficients of all simulations.

6.4.7 Operator for Planar Shapes

Up until now, we have concentrated on treating simulations as surfaces embedded in $\mathbb{R}^3$ or in taking 3D point clouds from a mesh, but extending the approach of constructing an invariant operator for simulations considered as curves embedded in $\mathbb{R}^2$ is straightforward. We now need to consider planar shapes and an invariant operator can be constructed using the distance along the curve, or path length along the curve as a geodesic distance and proceeding as in the previous cases. That is, a matrix operator is constructed using a Gaussian function for all pairwise distances on the curve. Built in this way, the operator will be invariant to changes that preserve this distance and the eigenvector basis of this operator can be used as a basis for the shape space of this type of planar curves.

6.5 Numerical Evaluation of the Operators

In this section we will describe how to evaluate all of these operators numerically.



(a) non-observable data

(b) observed data

(c) simulation burst

Figure 6.2: Illustration of the construction of the operator for independent components

6.5.1 Evaluation of the Laplace-Beltrami operator

The evaluation of the Laplace-Beltrami operator and the corresponding eigenvector basis for a data set bundle is described algorithmically in this section. Expression (6.4.2) is used whereby the graph distance $d(p, w)$ is evaluated using a shortest path algorithm for triangular surface meshes as described in [Mitchell et al., 1987]. The software implementation used in this work is MeshLP [1] that has been modified so that it does not use a matlab interface.

Algorithm 2 describes the general procedure for the evaluation.

Algorithmus 2 : Spectral decomposition of the Laplace-Beltrami operator

```
Input : reference simulation $x^r$: A surface embedded in $\mathbb{R}^3$, given as triangular mesh with $N_h$
        points and $N_f$ faces.
Parameter : $\rho$, $h$, $n_{ev}$
Output : $p$ first eigenvectors of the Laplace-Beltrami operator, invariant to distance preserving
         transformations
foreach $\tilde{k} \in \{1, \ldots, N_f\}$ do                        ▷ estimate areas of each face
    foreach $i \in \{1, 2, 3\}$ do                                 ▷ 1/3 of area assigned to each vertex
        $area[\tilde{k}_i]$ = (area of face $\tilde{k}$)/3            ▷ face $\tilde{k}$ vertices indexed by $\tilde{k}_i$
    end
end
foreach $k \in \{1, \ldots, N_h\}$ do                               ▷ weight matrix with graph distances
    $[ids, dists] = \text{graphdist}(k, x^r, \rho \cdot \sqrt{h})$    ▷ distances on $x^r$ up to $\rho \cdot \sqrt{h}$
    foreach $l, d \in [ids, dists]$ do
        $\mathbf{W}[k, l] = area[k] \cdot area[l] \cdot \exp(-d^2/(4h))/(4\pi h^2)$
    end
end
$\mathbf{D} = diag(\mathbf{W} \cdot \mathbb{1})$
$\mathbf{L} = \mathbf{W} - \mathbf{D}$                                ▷ compute graph Laplacian
solve $[\mathbf{U}, \mathbf{E}] = \text{eig}(\mathbf{L})$              ▷ non-trivial eigenvectors
return first $n_{ev}$ non-trivial eigenvectors $\mathbf{U}$
```

6.5.2 Evaluation of the Operator for the Independent Components

In analogy to the evaluation of the approximation of the Laplace-Beltrami operator, the evaluation of the operator for the nonlinear components is described algorithmically in this section.

Algorithm 3 describes the general procedure for the evaluation.

6.5.3 Evaluation of the Operator for Planar Shapes

Algorithm 4 describes the steps for the evaluation on an operator for curves embedded in $2D$.

[1] http://www.geomtop.org/software/meshlp.html

Algorithmus 3 : Spectral decomposition of the NICA operator

```
Input : 1) reference simulation $x^r$: A surface embedded in $\mathbb{R}^3$, given as $N_h$ points.
        2) data set bundle $\{x^j\}$, $j = 1, \dots, m$: Surfaces embedded in $\mathbb{R}^3$, where each one is
        given by $N_h$ points. The points of the $x^j$ are assumed to form for each point in $x^r$ a
        simulation cloud obtained by a nonlinear transformation of the reference simulation.
Parameter : $\delta$, $\epsilon$, $n_{ev}$
Output : $p$ first eigenvectors of operator, invariant to a nonlinear transformation
foreach $k \in \{1, \dots, N_h\}$ do                                        ▷ local Jacobi matrices at $p^{(k)}$
    empirically estimate local covariance matrix $C_{k,\delta}$ from data set bundle $\{x^j\}$ after (6.4.7)
    JJt[$k$] = pseudoinverse of $C_{k,\delta} \cdot 5/\delta^2$
end
foreach $l \in \{1, \dots, N_h\}$ do
    foreach $k \in \{1, \dots, N_h\}$ do                                    ▷ calculate weight matrix
        estimate $d(k, l)$ after (6.4.11) and using JJt[$k$]
        A[$k, l$] = $\exp(-d(k, l)/\epsilon)$
    end
end
Compute $W_{r,rs}$ using (6.4.13 and 6.4.14)                                   ▷ compute matrix
solve [U, E] = eig($W_{r,rs}$)                                             ▷ non-trivial eigenvectors
return $n_{ev}$ first non-trivial eigenvectors $D_{rs}^{1/2}$U
```

Algorithmus 4 : Calculation of an operator for planar curves

```
Input : A curve $q_0$ given as an immersion in $\mathbb{R}^2$ with $N_h$ points
Output : An operator assumed invariant to transformations between shapes and its
         corresponding eigenvectors as basis
Parameter : $\epsilon$
For the curve $q_0$, evaluate the cumulative path length $Dist(i, j), i = 1, \dots N_h, j = 1, \dots N_h$
along the curve for all points
foreach $j$ in $N_h$ do
    foreach $i$ in $N_h$ do
        Calculate $K_1(i, j) = exp(-Dist(i, j)/(4\epsilon))$
    end
end
Normalize and get symmetric matrix $K$ (see steps 7-9 in Algorithm 1 )
Solve [U, $E$] = $eig(K, p)$
return $p$ first non-trivial eigenvectors U
```

6.6 Approximating a Basis in Simulation Space

The goal of defining operators that are assumed to be valid for all simulations in a bundle is to use the projection on the eigenvectors of these operators to represent all simulations in the bundle. Notice that in this case, the eigenvectors of such operators can be used as a basis for the set of simulations. To explain this, let us return to our setting for analyzing the solution space,

$$\varphi_h(\mathcal{P}) = \{u_h(\boldsymbol{\mu}) \in V_h : \boldsymbol{\mu} \in \mathcal{P}\},$$

obtained by evaluating many numerical solutions $u_h(\mu)$ of a PDE for a given sampling of the parameter space $\mathcal{P}$. As described in section 6.1, our intention is to treat this space as the quotient of the space of surfaces embedded in $\mathbb{R}^3$ or curves embedded in $\mathbb{R}^2$,

$$\mathcal{M} := Emb(\mathcal{S}, \mathbb{R}^n)/G(\mathcal{S}),$$

made of equivalence classes or orbits with respect to a discrete group action G. Now, the use of operators invariant to the transformation group G allows for a very interesting result, namely it opens the possibility of obtaining an equivalent representation of the simulation space. Indeed, under the invariance assumption, the operators are the same for all elements of the space $Emb(\mathcal{S}, \mathbb{R}^n)/G(\mathcal{S})$. As a consequence an orthogonal basis obtained from those operators will also be the same for all the elements of the solution space. In the discrete setting this is achieved using the orthogonal basis of the operators constructed using algorithms 2, 3 and 4 for the Laplace-Beltrami operator, the operator for the independent components, or the operator for planar shapes, respectively. Since the discrete operators and corresponding orthogonal basis are assumed to be approximately invariant to the transformation group G, one can project all simulations to the obtained basis in each case. A way to evaluate what "approximately" means has been explained in section 6.4.5. Note also the analysis of the perturbation due to noise in section 6.4.5 can also be applied for the operator of independent components and the operator of planar shapes since a Gaussian kernel is also used for these cases.

Projecting all simulations to another basis will, in general, not be of any advantage since we need, in the case of a discrete approximation of the operators, as many basis elements as number of nodes N_h on the mesh or points of a curve (for the case of plane shapes) in order to represent each simulation. Nevertheless we will see in section 6.8 that depending on the degree of smoothness of the simulations, the decay of the spectral coefficient is polynomial. We remark though that, since we are dealing with discrete data, we will have to calculate approximations of the operators and this implies that one has to actually justify the use of the eigenvectors as approximations of the corresponding eigenfunctions. This will be clarified in the following section where we will concentrate only on the case of 3D surfaces and 3D point clouds.

6.7 Spectral Approximation of the Operators

We have seen in section 6.4.4, 6.4.6 that under some specific assumptions, starting with a discrete simulation and building a discrete operator, point-wise convergence to a continuous operator can be obtained. Nevertheless a convergence of the eigenspace is not guaranteed by such operator convergence. We would like to clarify, when such convergence can be expected and if it is reasonable to assume that these conditions are fulfilled in the discrete case.

6.7.1 Structure of the Eigenfunctions

The eigenfunctions of the Laplace-Beltrami operator on a surface can be viewed as a generalization of the use of a Fourier basis in Euclidean space for the Laplace operator. Using such eigenfunctions as a basis, a smooth function defined on a surface can be decomposed, so that the first spectral coefficient corresponds to eigenfunctions describing the low frequency part of the function and the rest corresponds to the high frequency parts of it, (see [Grebenkov and Nguyen, 2013, Jakobson et al., 2001]).

For the operator of independent components, the eigenvector corresponding to the first eigenvalue, in increasing order, recovers the first independent component, the second one the second an so on [Singer and Coifman, 2008], see section 6.4.6 for the setting. Additionally, in this stochastic setting, the first eigenfunctions (ordered according to increasing eigenvalues) of the limit Fokker-Planck operator correspond to the slow variables of a stochastic dynamical system and the independent components can be shown to be solutions to Sturm-Liouville differential equations, see again [Singer and Coifman, 2008].

The solutions of those differential equations can be shown to be generators for different type of polynomials, e.g. Jacobi, Laguerre or Hermite types, see [Gomez-Ullate et al., 2008]. In addition, it is also known that special solutions to the Fokker-Planck equation are polynomials that can be used in an orthogonal decomposition for less smooth functions [Tran et al., 2013].

In our setting, an orthogonal decomposition based on the use of eigenfunctions of some specific operator will be able to decompose a function into several components. If the function f is smooth, it is known from spectral approximation theory that the decay of the spectral coefficients will, depending on the degree of smoothness of the function, achieve exponential convergence for $f \in C^{\infty}$, see e.g [Costa, 2004]. If the function is not smooth and has for example some discontinuities, it is also known that the exponential convergence is lost due to the Gibbs phenomena. But at least in regions far away from the discontinuity it can be recovered see e.g. [Gottlieb et al., 1992]. The approach to control the Gibbs phenomena consists in a change of the basis for the decomposition using special polynomials. These again can be shown to be, for specific cases, solutions of specific Fokker-Planck type differential equations, see [Tran et al., 2013]. We notice that in both cases, for smooth and, after changing the basis, for non-smooth functions we do have a few spectral coefficients that are large and correspond to the first eigenfunctions, while the rest are smaller. We will return to this feature, but first let us discuss the discrete setting.

6.7.2 The Discrete Case

In section 6.4, we have reviewed some results that allows us to obtain convergence of a discrete mesh-Laplacian to a continuous operator given some conditions on the mesh or some large sampling limit for a point cloud. Little is known about the convergence of the discrete eigenvectors and eigenvalues to the corresponding eigenfunctions and eigenvalues, this is not obtained from the convergence of the operators. This area is an active field of research, we will mention the following result that follows from [Dey et al., 2010]:

Theorem 6.7.1. *Given a smooth m-manifold $\mathcal{S}$ and a simplicial mesh K which is an (ϵ, η) approximation of it, let $\{\lambda_i\}$ and $\left\{\lambda_i^D\right\}$ denote the set of non-decreasing discrete eigenvalues of the Laplace-Beltrami* *operator $\Delta_{\mathcal{S}}$ and its approximation L_K^h (see 6.4.2) respectively. For any fixed j, we have that*

$$\lim_{h,\epsilon,\eta,\frac{\epsilon}{h^4}\to 0} |\lambda_j - \lambda_j^D| = 0$$

The question of convergence of the eigenfunctions using the same construction as given in [Dey et al., 2010] is to our knowledge still open. We also mention the work in [Reuter et al., 2009], where the Laplace-Beltrami eigenvalue problem is discretized using a finite element formulation and as such can be constructed using convenient convergence properties depending on the mesh size and the degree of the shape functions used for the discretization, although the convergence to the eigenspace has not been treated in this study. Note that in our case we would like to build an operator based on the data, so that we have not the flexibility of choosing a discretization such as in [Reuter et al., 2009].

A different setting to prove convergence is the stochastic one, based on convergence in probability. We have mentioned here the work in [Belkin and Niyogi, 2008] where it has been shown that the eigenstructure of the weighted graph Laplacian converges to the one on the manifold and a recent result that extended such results for the so called connection Laplacian, where the convergence to the Fokker-Planck operator is treated as a special case of the more general setting of the connection Laplacian [Singer and Wu, 2013].

To summarize, based on the mentioned results, the use of a discrete approximation of the eigenfunctions of the Laplace-Beltrami operator and the operator of independent components, can to a certain extend be justified in our approach.

6.7.3 Discrete Orthogonal Decomposition

Once the eigenvectors $\bar{\psi}_i, i = 1, \dots, N_h$ of some discrete operator are given, we can project a given mesh function $f_{|K}$ defined on a mesh K along them and obtain as a result a set of spectral coefficients $\alpha_i, i = 1, \dots, N_h$. The following expression can be written as an analogous expression of the continuous case (6.4.1):

$$f_{|K} = \sum_{i=1}^{N_h} \alpha_i \bar{\psi}_i, \qquad \alpha_i = \langle f_{|K}, \bar{\psi}_i \rangle, \tag{6.7.1}$$

where N_h corresponds to the number of nodes on the mesh K, which can be very large. Using the approximation L_K^h of the Laplace-Beltrami operator a large number of eigenvectors might be necessary to reconstruct the mesh data, depending on its smoothness, which directly affects the degree of decay of the spectral coefficients. The way the corresponding spectral coordinates decay will determine how much information is concentrated in the first coefficients. Note that this is analogously to the Fourier decomposition for 1D signals. In this case exponential decay can be achieved for analytical functions.

Now assume K is a mesh which approximates the reference manifold $\mathcal{S}$. Let $\mathcal{S}^i, i = 1, \dots, m$ be the manifolds obtained by the application of diffeomorphisms φ^i with corresponding approximations given by $\varphi^i_{|K^i}$ for the mesh K^i. Let the function $f_{|K} : K^r \to \mathbb{R}$ be evaluated at the nodes of the mesh and let $f^i_{|K} : K^i \to R$ be the resulting functions after the application of φ^i. The functions f^i, f correspond in our setting to the numerical solutions of a partial differential equation using the finite element method. They are actually discrete and defined on the mesh K^i, K, respectively. In what follows we will use the same notation as the continuous counterpart for the functions on the mesh explicitly mentioning when the context changes. Further, for simplicity we assume that the functions f, f^i are $k-$times differentiable. From now on, we will use ψ for both the eigenfunctions and for the eigenvectors, which case is treated will be clear from the context. Under the assumption of isometric transformations as explained section 6.4.5 the following holds:

Proposition 6.7.2. *Let an orthogonal basis consisting of N_h eigenvectors ψ_j, $j = 1, \ldots, N_h$ be given, which is obtained from the approximation of the Laplace-Beltrami operator (6.4.2). Then all mesh functions f^i can be represented as*

$$f^i = \sum_{j=1}^{N_h} \alpha^i_j \psi_j,$$

where α^i_j are the spectral coefficients obtained by the projection of f^i into the eigenvector basis $\left\{\psi_j\right\}_{j=1}^{N_h}$.

Proof. This is a consequence of the application of expression (6.4.2), under the assumption that the basis is invariant under isometric transformations, see Proposition 6.4.13. □

For the operator (6.4.16), we are in a stochastic setting. Let a set of simulations be represented as functions $f^i : \mathcal{S}^i \rightarrow R$ be obtained as realizations of an independent stochastic Ito processes and obtained as nonlinear transformations from a non-observable reference simulation. This operator approximates a Fokker-Planck operator in the non-observable space and is therefore invariant with respect to the nonlinear transformations (see section 6.4.6 for details). Further, we assume convergence in probability to the eigenfunctions of the continuous operator. The eigenvectors of (6.4.16) approximate the eigenfunctions of the invariant Fokker-Planck operator after [Singer and Coifman, 2008] and can be used as invariant basis for all simulations. Under this assumptions we can write,

Proposition 6.7.3. *Let an orthogonal basis consisting of N_h eigenvectors ψ_j, $j = 1, \ldots, N_h$ be given, obtained from the approximation of the operator as given in expression (6.4.16). Assuming that those eigenvectors approximate the eigenfunctions of a Fokker-Planck operator of an Itô stochastic process as described in section 6.4.6, then all functions f^i can be represented as*

$$f^i = \sum_{j=1}^{N_h} \alpha^i_j \psi_j, \tag{6.7.2}$$

where $\alpha^i_j =< f^i, D^{1/2}_{rs} \psi_j >$ are the spectral coefficients obtained by the projection of f^i into the eigenvector basis $\left\{\psi_j\right\}_{j=1}^{N_h}$ using a weighted scalar product.

Proof. The operator given in (6.4.16), under the assumptions of section 6.4.6, is constructed using a distance in the non-observable space (see Proposition 6.4.14). This makes the operator invariant with respect to the nonlinear transformations of the reference non-observable simulation. Assuming convergence in probability to the eigenfunctions of the continuous operator, the eigenvectors approximate the eigenfunctions of the invariant Fokker-Planck operator and can therefore be used as the invariant basis. □

As explained, for the Laplace-Beltrami operator as well as for the operator of the independent components, they are invariant to specific transformations. As a consequence we could use the eigenvectors as basis and project a set of mesh functions along the same basis. This property, combined with the observation about the possibility of achieving a strong decay behavior of the spectral coefficients, suggest a method for data analysis which potentially reduces the dimensionality of the data along the spectral coefficients.

6.8 Approximation Properties

Note that in this section we only consider the stochastic setting. We will show that under certain conditions, one can expect a strong decay of the spectral coefficients depending on the orthogonal functions used and the smoothness properties of the data.

As before, we will focus on numerical simulations, where we know the approximation properties, since the data stems from a finite element mesh, which discretizes a partial differential equation. The smoothness of the solution, combined with considerations about the spectral approximation estimation based on discrete data, allow us to obtain an estimation that depends on the smoothness of the function and therefore clarifies the observed strong decay of the spectral coefficients, at least for a simplified setting.

We assume that we have obtained an orthogonal decomposition as given in proposition 6.7.3, then the following can be written,

Proposition 6.8.1 (Decay of the Spectral Coefficients)**.** *In the orthogonal expansion given in proposition 6.7.3, for smooth functions $f^i \in C^k, i = 1, \ldots, m$, the spectral coefficients $\alpha^i_l, l = 1, \ldots N_h$ decay as*

$$|\alpha^i_l| \leq \frac{C}{(a_l)^k} \|f^i_{(k)}\|_w,$$

depending on the degree of smoothness (k) of the functions $f^i, i = 1, \ldots, m$.

For the proof of this proposition we will need the following additional results.

Proposition 6.8.2. *Given the eigenvalue problem for the Fokker-Planck operator (expression (6.4.17)), the operator can be separated by the use of an orthogonal transformation and it can be shown that the separated problems are 1D backward Fokker-Planck-like operators obeying specific Neumann boundary conditions. The solution of the eigenvalue problems with such operators corresponds to the solution of Sturm-Liouville problems.*

Proof. This result was shown first in [Singer, 2006b] for linear independent component analysis (ICA). Please note that the operator in expression (6.4.17) is the one on the unobservable manifold $\mathcal{S}^p$ assumed to be a subset of $\mathbb{R}^d$. The observable space is nonlinearly transformed but we are able to approximate the operator in the observable space as detailed in section 6.4.6. Due to this construction, we are left with a Fokker-Planck operator on a linear manifold of dimension d, the linear ICA is posed as given N observations of the $m-$vector X, find the unknown independent random variables $S^1, S^2, \ldots S^d$ and the mixing matrix $A_{m \times d}$ so that $X^{(i)} = AS^{(i)},\ i = 1, 2, \ldots N$. Since the unobservable manifold is planar, the data density is a product of d one-dimensional densities $p(X) = \prod_{j=1}^{d} p_j(S_j)$, where p_j is the density of S_j. The potential U in (6.4.17) is then a sum of one-dimensional potentials $U(X) = \sum_{j=1}^{d} U_j(S_j)$, with $U_j(S_j) = -2logp_j(S_j)$.

The Laplacian operator Δ is invariant to orthogonal transformations so that,

$$\Delta = \sum_{j=1}^{d} \frac{\partial^2}{\partial X_j^2} = \sum_{j=1}^{d} \frac{\partial^2}{\partial S_j^2}.$$

This result shows that by constructing the NICA operator using expression (6.4.16), one approximates the continuous operator

$$\mathcal{L} = \sum_{j=1}^{d} \frac{\partial^2}{\partial S_j^2} - \frac{\partial U_j(S_j)}{\partial S_j}\frac{\partial}{\partial S_j}.$$

This operator is separable and can be written as $\mathcal{L} = \sum_{j=1}^{d} \mathcal{L}_j$, where each $\mathcal{L}_j$ is a one-dimensional Fokker-Planck operator in the interval (a_j, b_j) with Neumann boundary conditions

$$\mathcal{L}_j = \frac{d^2}{dS_j^2} - \frac{dU_j(S_j)}{dS_j}\frac{d}{dS_j}.$$

The eigenvalue problem of $\mathcal{L}_j$ is a Sturm-Liouville problem

$$\frac{d}{dS_j}\left(e^{-U_j}\frac{d\psi_j}{dS_j}\right) + \lambda_j e^{-U_j}\psi_j = 0 \quad S_j \in (a, b). \tag{6.8.1}$$

The operator $\mathcal{L}_j$ has an infinite set of eigenfunctions and eigenvalues. The eigenfunctions of $\mathcal{L}$ are tensor products of the one-dimensional eigenfunctions. □

Proposition 6.8.3. *Let $f \in C^k$, and its spectral coefficients $\hat{f}_l = \langle f, \psi_l \rangle$, where ψ_l, $l = \{0, 1, \dots, \}$ are the eigenfunctions of the Sturm-Liouville eigenvalue problem,*

$$\frac{d}{dx}\left(e^{-U}\frac{d\psi}{dx}\right) + \lambda e^{-U}\psi = 0 \quad x \in (-1, 1). \tag{6.8.2}$$

Then the following holds,

$$|\hat{f}_l| \leq \frac{C}{(\lambda_l)^k} \|f_{(k)}\|_{L^2_w(-1,1)},$$

provided: p is zero at the boundary or the boundary conditions,

$$\alpha_1\psi(-1) + \beta_1\psi'(-1) = 0, \qquad \alpha_1^2 + \beta_1^2 \neq 0 \tag{6.8.3}$$

$$\alpha_2\psi(1) + \beta_2\psi'(1) = 0, \qquad \alpha_2^2 + \beta_2^2 \neq 0 \tag{6.8.4}$$

are satisfied.

Proof. The proposition is a result of spectral approximation theory for Sturm-Liouville problems. The following approach has been shown in [Canuto, 2006]. There are two cases that have to be distinguished, the first one is the regular one, whose approximation properties actually depends on the boundary conditions and the second one is independent of them, but requires that the data density term p decays to zero on one of the boundaries of the domain. We will treat both cases individually.

Let us consider the general form of the Sturm-Liouville problem (6.8.2),

$$-(p\psi')' + q\psi = \lambda w\psi \quad x \in (-1,1). \tag{6.8.5}$$

In this expression p, q, w are given real valued functions such that:

- p is continuously differentiable, strictly positive in (-1,1) and continuous at $x = \pm 1$
- q is continuous, nonnegative and bounded in (-1,1)
- the weight function w is continuous, nonnegative and integrable over $(-1,1)$.

For the problem (6.8.5), boundary conditions for ψ will be defined. Depending on the coefficients p, q and w, specific families of polynomials eigenfunctions (see [Canuto, 2006, Mohammad, 2006]) can be used as basis for a spectral approximation of a general function f. For example under the assumption that the probability distribution from which the data is sampled is Gaussian, then the eigenfunctions are Hermite polynomials.

(a) The "regular" case. The first task is to verify that a kind of convergent "Fourier" decomposition can be found using a conveniently defined inner product. Indeed, it is known (see [Courant and Hilbert, 1953] chapter V section 3) that under the assumptions $\alpha_1\ \beta_1 \leq 0$ and $\alpha_2\ \beta_2 \geq 0$, the eigenvalues of the regular Sturm-Liouville problem form an infinite, unbounded sequence of nonnegative numbers, $0 \leq \lambda_0 < \ldots < \lambda_l < \lambda_{l+1} < \ldots$ and have multiplicity 1. The eigenfunctions ψ_l are mutually orthogonal with respect to the inner product

$$(u,v)_w = \int_{-1}^{1} u(x)v(x)w(x)dx, \tag{6.8.6}$$

that is $(\psi_r, \psi_s)_w = 0$ if $r \neq s$. Additionally the eigensystem $\{\psi_l, \quad l = 0,1,\ldots\}$ is complete in the weighted space $L^2_w(-1,1)$ so that

$$||f - \bar{f}||_{L^2_w(-1,1)} \rightarrow 0 \text{ as } N \rightarrow +\infty,$$

where $\bar{f} = \sum_{l=0}^{N} \hat{f}_l \psi_l, \quad N > 0$ and $||\psi_l||_{L^2_w(-1,1)} = 1$.

Now, an estimation of the convergence rate of the decay of the coefficients of a function $f \in L^2_w(-1,1)$ can be made using (6.8.5), (6.8.6) and integration by parts

$$\hat{f}_l = (f,\psi_l)_w = \frac{1}{\lambda_l}\int_{-1}^{1} f[-(p\psi_l')' + q\psi_l]dx \tag{6.8.7}$$

$$= \frac{1}{\lambda_l}\int_{-1}^{1}[-(pf')' + qf]\psi_l dx - \frac{1}{\lambda_l}[p(\psi_l' f - \psi_l f')]_{-1}^{1} \tag{6.8.8}$$

$$= \frac{1}{\lambda_l}\left(\frac{1}{w}\mathcal{L}_j f, \psi_l\right)_w - \frac{1}{\lambda_l}[p(\psi_l' f - \psi_l f')]_{-1}^{1}. \tag{6.8.9}$$

Using the boundary conditions, the second term vanishes and we obtain the result

$$\hat{f}_l = \frac{1}{\lambda_l}\left(\frac{1}{w}\mathcal{L}_j f, \psi_l\right)_w = \frac{1}{\lambda_l}(f_{(1)}, \psi_l)_w,$$

where $f_{(1)} = \frac{1}{w}\mathcal{L}_j f$ and $\mathcal{L}_j f := -(pf')' + qf$.

Iterating this result one finally obtains

$$\hat{f}_l = \frac{1}{(\lambda_l)^k}\left(\frac{1}{w}\mathcal{L}_j f_{(k-1)}, \psi_l\right)_w = \frac{1}{(\lambda_l)^k}\left(f_{(k)}, \psi_l\right)_w.$$

(b) The "singular" case occurs when p vanishes for at least one point on the boundary. The solution should, in this case satisfy $p(x)f'(x) \to 0$ as $x \to \pm 1$.

Assume $f \in X$ is square integrable with respect to the weights q and w and that the derivative of f is integrable with respect to the weight p,

$$X = \left\{v \in L^2_w(-1,1) \cap L^2_q(-1,1) \quad | v' \in L^2_p(-1,1)\right\}.$$

Then the following variational formulation of (6.8.5) can be written,

$$\int_{-1}^{1}(pu'v' + quv)dx = \lambda \int_{-1}^{1} uvwdx \text{ for all } v \in X. \tag{6.8.10}$$

as in the regular case, the eigenvalues of (6.8.10) form an unbounded sequence of nonnegative real numbers $0 \leq \lambda_0 \leq \ldots \leq \lambda_k \leq \ldots$; each with finite multiplicity. The system of corresponding eigenfunctions ψ_k is orthogonal and complete in $L^2_w(-1,1)$ (see [Canuto, 2006] chapter 5 for details on the justification of this assertions).

As in the regular case, an approximation of the convergence rate of the decay of the coefficients of a function $f \in L^2_w(-1,1)$ can be estimated using (6.8.10) and integration by parts

$$\begin{aligned}
\hat{f}_l &= \frac{1}{\lambda_l}\int_{-1}^{1}(p\psi_l' f' + q\psi_l f)dx && (6.8.11)\\
&= \frac{1}{\lambda_l}\int_{-1}^{1}[-(pf')' + qf]\psi_l dx + \frac{1}{\lambda_l}[pf'\psi_l]_{-1}^{1} && (6.8.12)\\
&= \frac{1}{\lambda_l}\left(\frac{1}{w}\mathcal{L}_j f, \psi_l\right)_w + \frac{1}{\lambda_l}[pf'\psi_l]_{-1}^{1} && (6.8.13)
\end{aligned}$$

Using the condition at the boundary, the second term vanishes and we obtain the result

$$\hat{f}_l = \frac{1}{\lambda_l}\left(\frac{1}{w}\mathcal{L}_j f, \psi_l\right)_w = \frac{1}{\lambda_l}(f_{(1)}, \psi_l)_w.$$

Iterating this result as before, one finally obtains

$$\hat{f}_l = \frac{1}{(\lambda_l)^k}\left(\frac{1}{w}\mathcal{L}_j f_{(k-1)}, \psi_l\right)_w = \frac{1}{(\lambda_l)^k}\left(f_{(k)}, \psi_l\right)_w.$$

Notice that only in the singular case we can assert that the problem has a convergence decay rate, independent of the boundary conditions, but dependent on the smoothness of the function. □

Proof of Proposition 6.8.1

Proof. For the proof we have combined three results. First the decomposition in expression 6.7.2 is constructed using approximations of the eigenfunctions of the Fokker-Planck operator of an Itô stochastic process. This operator can be decomposed in $1D$ independent Sturm-Liouville problems. Proposition 6.8.2 justify this decomposition. Next using proposition 6.8.3 that establishes that depending on the type of Sturm-Liouville problem and its boundary conditions, an estimation of the decay rate of the spectral coefficients can be found. Finally as shown in [Singer, 2006b], the first eigenfunctions of the operator given in (6.4.17) provides the independent components and they corresponds to the eigenfunctions of the $1D$ independent Sturm-Liouville problems so that for the spectral coefficients $\alpha_l^i, l = 1, \dots N_h$, the decay can be approximated as

$$|\alpha_l^i| \leq \frac{C}{(\lambda_l)^k} \|f_{(k)}^i\|_w,$$

depending on the degree of smoothness k of the functions $f^i, i = 1, \dots, m$.

Continuing in the vein of the main result, notice that proposition 6.8.3 cannot be used directly for our case since only discrete data is available. How to formally apply this continuous result to the discrete case is, to our knowledge still open, nevertheless consider the work done in [Donoho and Johnstone, 1994] where an analysis of the estimation error between a function and its discrete approximation has been completed. An ideal estimator is obtained using a N_t-term approximation using wavelets, the rate of decay is shown to depend on the smoothness of the mother wavelet (for details see [Donoho and Johnstone, 1994]). The combination of these results finally shows the main proposition albeit the above mentioned analysis. □

We note that in the scope of the results above, the operator of independent components can be separated into independent eigenfunctions of 1D Sturm-Liouville problems, they are therefore ideal candidates as a basis for a set of simulations. Also the results above show that at least; for smooth functions, the decay of the first spectral coefficients can be high. For applications this is not necessarily the case. Functions defined at the mesh points are generally not smooth. Nevertheless, we will see in the next chapter that for smooth functions as well as for non-smooth ones, low dimensional information about the data sets can be gained through the use of a single basis and by analyzing the first spectral coefficients.

CHAPTER 7

New Data Analysis Method

Our particular interest lies in the analysis of many simulation results, which are obtained from the numerical solution of partial differential equations with the finite element method. In chapter 6 using the theoretical setting of chapter 5 a way to treat a simulation space as a quotient space with respect to a group G acting by isometries has been proposed. Under certain conditions, geodesics on the space of isometries are determined through the group actions and a representation of the group can be achieved using eigenspaces of operators that are invariant to the transformation group G. This suggests the possibility of using the projections to the eigenspaces as a representation of the group actions. Since simulations are discrete we justified the use of discrete invariant operators in chapter 6 and showed that, under certain conditions, only few spectral coefficients will be large depending on the smoothness of the functions representing the simulations. A method of data analysis for simulations can, using those results, be established. The idea is to make use of the projection coefficients of all simulations to the eigenvector basis from a discrete invariant operator and see how far they can be used for representing isometric group actions. Very interesting results are obtained for such a discrete setting that reflect some of the properties studied in the theoretical part.

7.1 Simulation Data Analysis Method

We propose a method of data analysis for simulations that involves a dimensionality reduction and a separation of effects along several components. A summary of the procedure can be found in algorithm 5.

Algorithmus 5 : Algorithm for Analysis of Simulation Bundles

```
Input : $m$ simulations $f^i \in \mathbb{R}^{N_h}, i = 1, \dots, m$, where $N_h$ is the number of nodes
Parameter : Component index $p$
Output : multiscale analysis of simulations
Take up to $p \leq N_h$ first eigenvectors of either the Laplace-Beltrami
(algorithm 2) the operator of independent components (algorithm 3) or the operator for
plane curves (algorithm 4)
foreach $i$ in $\{1, \dots, m\}$ do
    $\hat{\alpha}^i = \mathbf{U}_p \times f^i \in \mathbb{R}^p$
end
Evaluate the variance of the spectral coefficients $\hat{\alpha}^i$
Take $r$ components with the highest variance $V = \{p_k\}_{k=1,\dots,r}$
Identify group transformations at each component $p_k \in V$ using the spectral coordinates for
component $p_k$
```

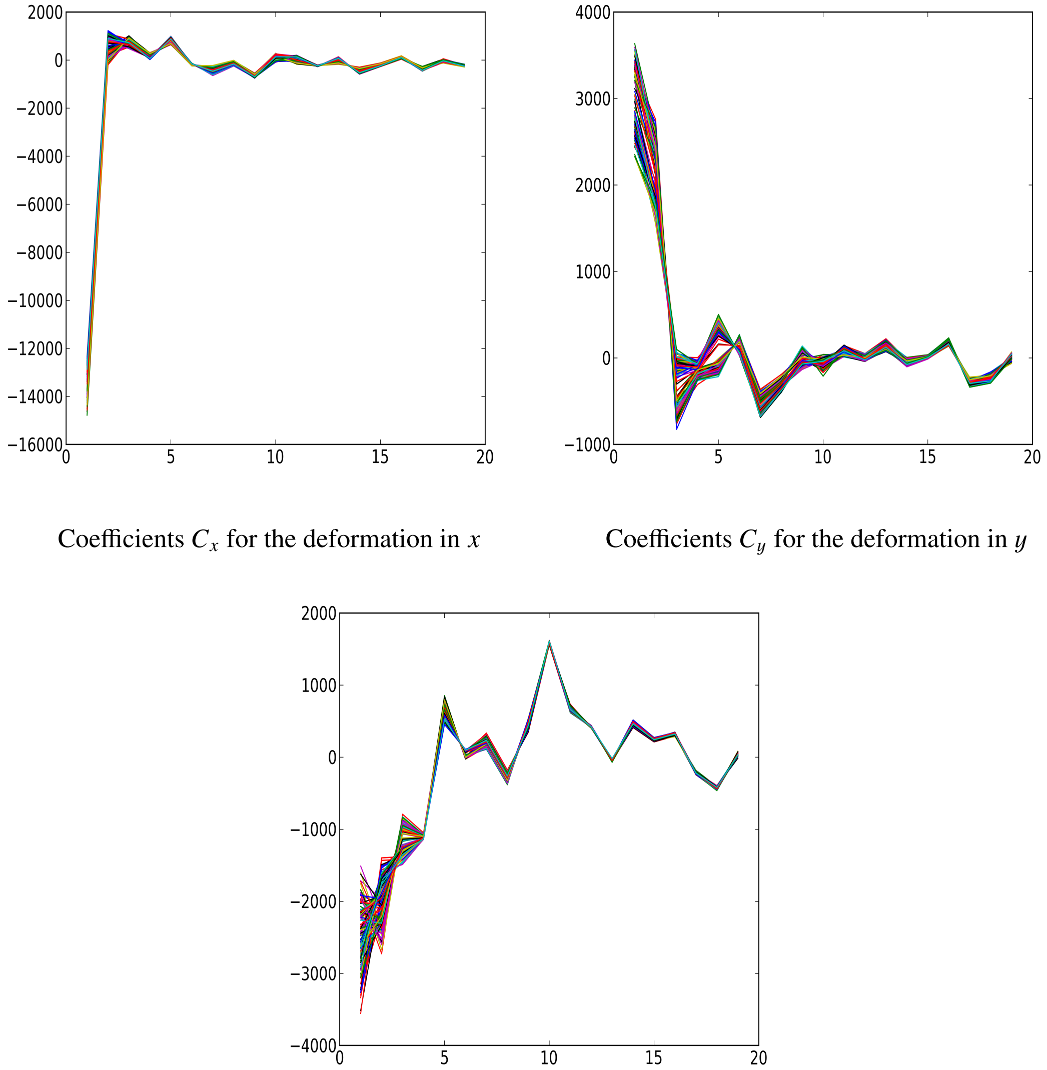
The approach first constructs a positive definite invariant operator using for it an arbitrary simulation. The eigenvectors of this operator are orthogonal and can then be used as a basis for representing all simulations in a dataset. The new representation in this basis are the projection coefficients, their special features we now would like to investigate in detail. Once the projection coefficients are obtained, the method exploits their decay properties. Regarding this, we have applied the approach to many industrial simulations and observed:

- a strong decay of the spectral coefficients,
- variations of the spectral coefficients concentrate on few components of the orthogonal decomposition with decreasing variance,
- using a linear combination of the basis with the spectral coefficients at one specific component of the orthogonal decomposition, translations or rotations for a car part in crash simulation can be obtained.

.

To support these observations let us illustrate them with an example in crash simulation. A total of 116 simulations are evaluated using the finite element solver LSDYNA, the simulations are 3D deformations. The thickness of 9 parts of the car structure are varied randomly up to 30%. A fixed time step is used (for details about the example see section 10.2.2). In Figure 7.1, the first 19 projection coefficients $Cx_k, Cy_k, Cz_k \quad k \in [1, 19]$ for each direction x, y and z using the basis evaluated from the Laplace-Beltrami operator approximation for the undeformed geometry are shown, the first coefficient for $k = 0$ is not used for clarity since it is several order of magnitude bigger then the next ones. It can be observed that the variations are concentrated on a few coefficients for all directions. In order to quantify those variations we evaluate the variance, for the illustrated example they are plotted in Figure 7.2. Thresholding until a certain number of components and deciding about the importance of each component in the orthogonal decomposition can be realized from those variances. For the plots we have also used the coefficients for $k = 0$ to show the variance changes.

The behavior of the variance is a direct consequence of the way the spectral coefficients behave. Proposition 6.8.1 is an attempt to explain this behavior formally. According to it, the decrease of the projection coefficients will be dependent on the degree of smoothness of the function on the mesh. How fast this decay is, depends according to proposition 6.8.1 on the magnitude of the eigenvalues at each component of the orthogonal decomposition. If the eigenvalues decrease fast, so will decay the coefficients.

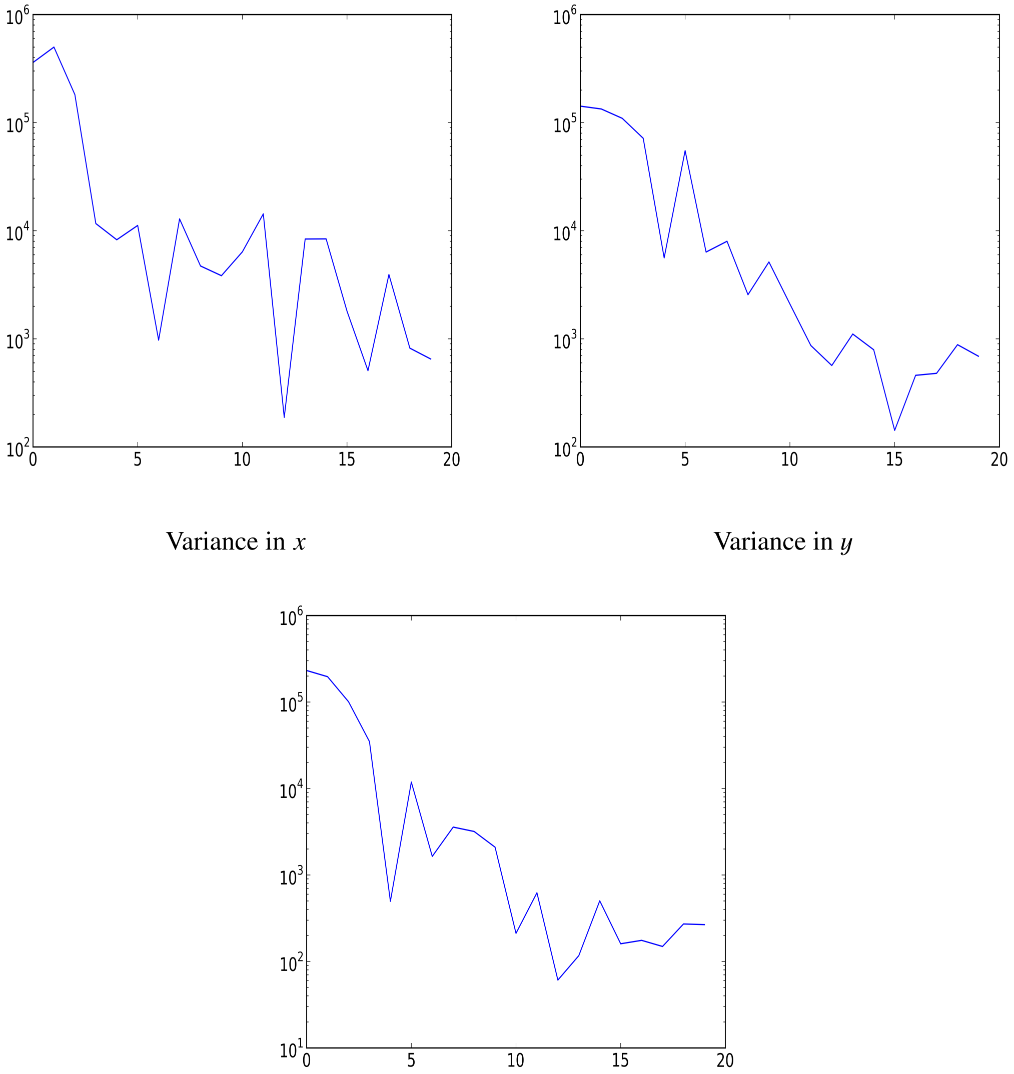


Coefficients C_x for the deformation in x

Coefficients C_y for the deformation in y

Coefficients C_z for the deformation in z

Figure 7.1: First projection coefficients $Cx_k, Cy_k, Cz_k \quad k \in [1, 19]$ of the deformations in the directions x, y and z of 116 crash simulations, the coefficients for $k = 0$ are not included for clarity since they are several orders of magnitude bigger than the next ones



Variance in x

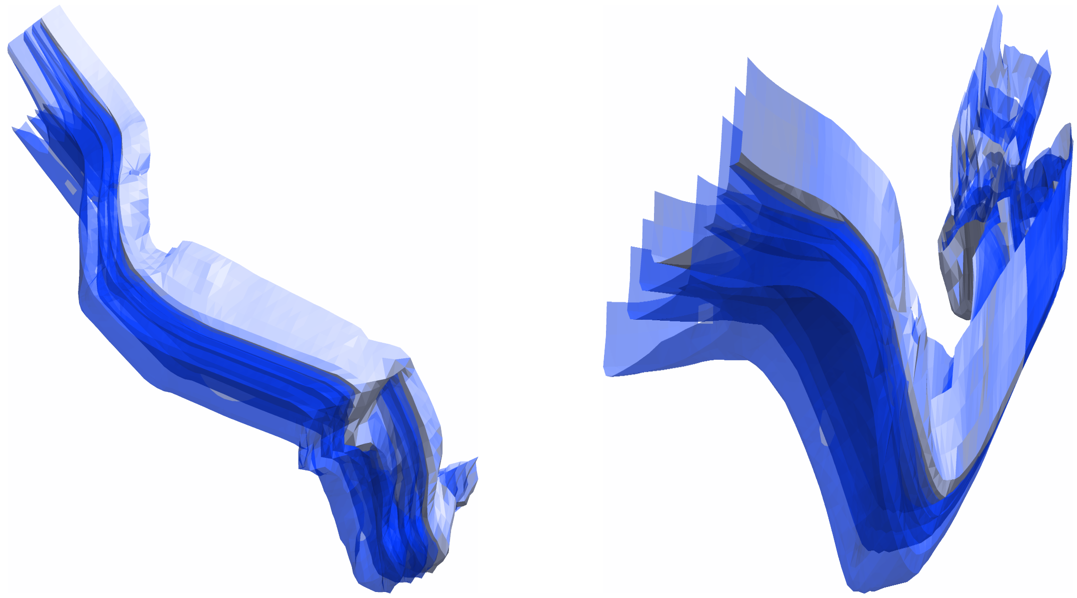
Variance in y

Variance in z

Figure 7.2: Variance of the first projection coefficients $Cx_k, Cy_k, Cz_k \quad k \in [0, 19]$ of the deformations of 116 crash simulations in logarithmic scale

Effect separation along the projection coefficients.

A separation of effects can also be obtained from the evaluation of the spectral coefficients. In order to have a visual representation of them let us use the same example as before. We use the operator of independent components build using algorithm 3. Up until now for the 116 simulations the amount of projection coefficients we obtain are 116 × 3 directions × number of nodes. Taking all spectral coefficients for one simulation for the directions x, y and z and using the common basis, a deformed geometry can be reconstructed as a linear combination of the basis. For this experiment, all coefficients for all components are the same for all reconstructed deformations with exception of the ones for the first component. For it we take the corresponding ones for the first component from each simulation for each direction. Since the basis is the same, this reconstruction is possible and they will reflect the variation of only the first coefficient. Figure 7.3 (a) shows a few geometries reconstructed using this approach. It can clearly be seen that translations transformations are recovered. The same procedure applied for the second component recovers a rotation as seen in figure 7.3 (b). More details about the application of these properties can be found in section 10.2.5.



(a) Translation

(b) Rotation

Figure 7.3: Recovered translations and rotations using the coefficients of the first and second component respectively

The observed behavior of the translations and rotations shows a numerical evidence of the group representation properties of the eigenfunctions of invariance operators. See Theorem 5.4.24 for the case of the rotation group.

Multiscale aspect of the orthogonal decomposition.

A further observation can be made about the orthogonal decomposition obtained using the same basis. We will use one of the crash examples simulations from the example above and project the 3D deformations along the common basis. For each direction x, y and z we obtain a set of coefficients at a fixed time step.

Using the same connectivity of the original mesh, all mesh deformations can be reconstructed using the Laplace-Beltrami basis. For that we use $f^i = \sum_{j=1}^{p} \alpha^i_j \psi_j$ for several values of p, f^i corresponds to the deformation in x, y or z direction respectively.

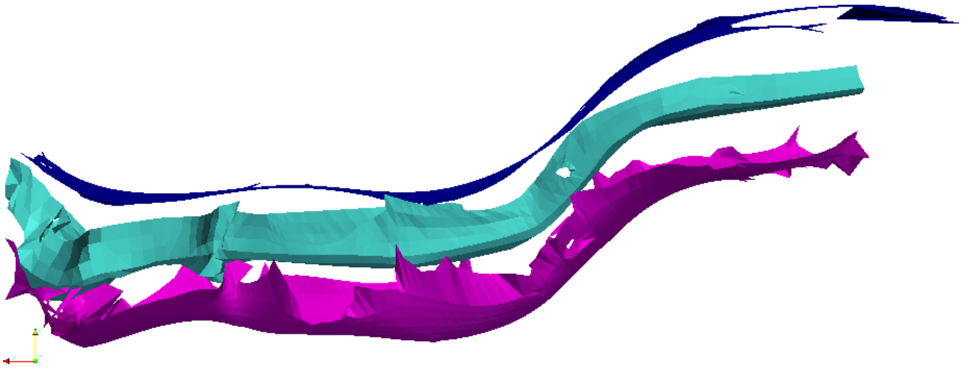


Figure 7.4: Reconstruction using the first p coefficients. $p = 20$ coefficients (upper), $p = 100$ (lower) and original data (middle).

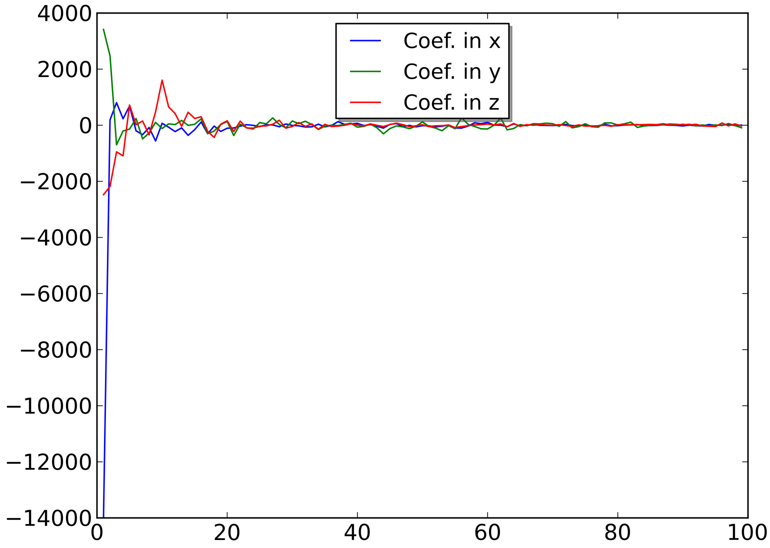


Figure 7.5: Magnitude of spectral coefficients for the deformations in the directions x, y and z at a fixed time step $t = 7$

From figure 7.4, it can be clearly seen that using $p = 20$ only a very coarse approximation of the part can be obtained. Adding more coefficients, e.g. $p = 100$, recovers more details of the part. It can also be clearly seen in figure 7.5 that most of the coefficients are small, with a few bigger ones. This is an essential feature which will be exploited for applications (see chapter 10 for details).

Based on the above observations we can assert that the proposed method enables the analysis of a simulation bundle in a multi-scale fashion. We have observed that in the case of using an approximation of the Laplace-Beltrami operator on a surface, for smooth functions approximating the simulations, coarse variations of a dataset bundle correspond to the spectral coefficients at the first components of the decomposition, details correspond to the higher components. Using the operator for the independent components, the first orthogonal component allows us to identify the variations along the first independent components. This observations has motivated the approach. What precisely coarse is, has many interpretations that have to be analyzed, nevertheless in multiple examples we have evaluated (see Chapter 10 for some of them) one can speak about multi-scale in the sense of coarse variations and fine details for the Laplace-Beltrami operator. Also as seen in section 6.8, under certain conditions the coefficients have a strong decay, therefore the "essential" part of a dataset is concentrated on few coefficients. Notice that also if desired, all coefficients up to some component can be taken as features representing the simulations and that up to dimension three the projection coefficients can be used directly as embedding coordinates for visualization. For dimension larger then three, a further method of dimension reduction can be used to obtain an embedding up to dimension three. For our study we use diffusion maps as a tool for further dimension reduction. See chapter 4 for details about the method and section 10.1 for an application of Diffusion maps to simulation results.

We used here the first coefficients up to component p for each simulation for classification and clustering. For full reconstruction we will take in addition, the coefficients of one reference simulation for components $p + 1$ to N_h for all simulations, where N_h is the number of nodes. Also a reconstruction for analyzing simulations at a particular component p can be achieved by freezing all coefficients with exception of the ones at component p. This will have important applications described in chapter 10.

7.2 Analysis of the Building Blocks of the Approach

We will as before assume that Simulations stems from a 2-manifold $\mathcal{S}$. For the surface $\mathcal{S}$, we will use a mesh representation K approximating $\mathcal{S}$ and represented as a surface in $\mathbb{R}^3$. In the case of solutions of ordinary differential equations (ODE), they are treated as curves embedded in $\mathbb{R}^2$.

The space $Emb(\mathcal{S}, \mathbb{R}^n)$ is the space of embeddings of the surface $\mathcal{S}$ in $\mathbb{R}^3$, or the space of embeddings in $\mathbb{R}^2$ for the case of ODEs respectively. From the elements of this space we will study the ones that are a result of the action of a transformation group G, that is, we will deal with the elements of the quotient $Emb(\mathcal{S}, \mathbb{R}^n)/G(\mathcal{S})$.

We can recognize the following elements in the approach.

- a transformation group G (Lie group), for example $Isom(\mathcal{S})$,
- a metric space $Emb(\mathcal{S}, \mathbb{R}^n)$, where the group action acts on,
- invariant generalized coordinates of $Emb(\mathcal{S}, \mathbb{R}^n)/G(\mathcal{S})$. In our case they are the orthogonal basis from an invariant positive definite operator
- a transformation, which we call the invariantization transform, sending an element of $Emb(\mathcal{S}, \mathbb{R}^n)$ to the invariant space $Emb(\mathcal{S}, \mathbb{R}^n)/G(\mathcal{S})$. In our case this transformation is realized by the projection to the orthogonal basis.

In an approximate way we obtain invariant coordinates due to the eigenvector basis, which is a discrete approximation of the basis of the space $Emb(\mathcal{S}, \mathbb{R}^n)/G(\mathcal{S})$. Since the basis is obtained from an invariant operator, these projection coefficients reflect the group action required to obtain a specific simulation. How group actions are represented in a basis of an invariant operator has a theoretical justification that we have started to explore in section 6.3. We would like to illustrate the aspect of the group representation properties in the following section.

7.3 ODE Bundles and Group Transformations

A set of solutions of a differential equation will be analyzed with the new approach. In particular, we see a connection to Lie-group methods [Olver, 1993] that are well known in the analytical solution of ordinary and partial differential equations. Those methods have evolved under several names, but the principal idea is to use symmetry principles.

We would like to exemplify first the connection of our operator approach to those principles with the following example. Let us consider the system of time dependent ordinary differential equations (ODE),

$$\dot{y} = y^3 + x^2 y - x - y \tag{7.3.1}$$

$$\dot{x} = x^3 + xy^2 - x + y, \tag{7.3.2}$$

the system of odes is to be solved in the integration interval $t \in [-4, 4]$ and we consider a bundle of 11 such solutions solved with different initial conditions given by $y_0 = 0.4 \cdot cos(\theta)$, $x_0 = 0.4 \cdot sin(\theta)$ and $\theta \in [0, 2\pi]$ (see figure 7.6).

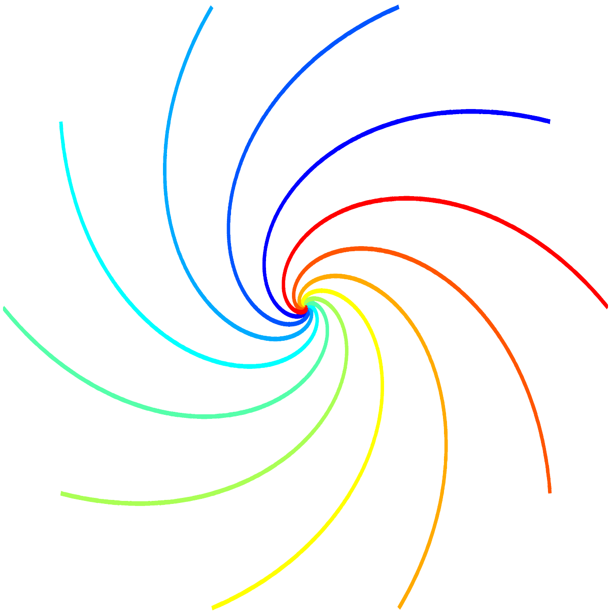


Figure 7.6: ODE bundle made of 11 simulations obtained by solving the equations with different initial conditions

The solution of this ODE using Lie group methods consists in finding a coordinate transformation along which the differential equations are invariant to a transformation group. For this case, the following transformation $x = r \cdot cos(\theta)$, $y = r \cdot sin(\theta)$ transforms this equation into,

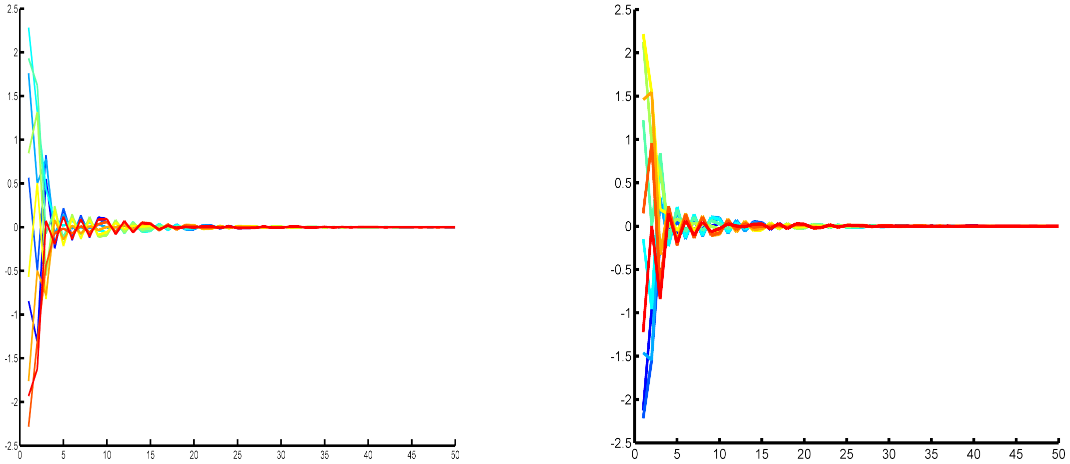
$$\frac{dr}{d\theta} = r(1 - r^2), \tag{7.3.3}$$

which is separable in the new variables. The solutions are invariant under the continuous group transformation $(r, \theta) \mapsto (r, \theta + \gamma)$ where γ is the group parameter and it corresponds to rotations with respect to the origin. Figure 7.6 shows several solutions of this differential equation that have been obtained numerically, i.e. without using the parametrization. As soon as the group parametrization is found, the orbits that represents one specific set of group actions, corresponds to solutions for the ODEs for some specific initial conditions.

For our setting only the numerical solution of the ODEs are available. The solutions of the ODE for different initial conditions are shown in figure 7.6, they are distributed circularly around a center point at the origin, that is, one arbitrary solution and therefore all others can be obtained by a rotation around the origin.

In our method we have constructed an operator using an arbitrary simulation that is assumed invariant to isometric transformations (in this case the rotations). Such an operator can be constructed by considering a parametrization of one arbitrary solution along its length. A distance along such a curve can be defined which is independent on the rotation of the curve and by using this distance a discrete operator can be built with all pairwise distances between all points in the curve. Algorithm 4 in section 6.4.7 describes how this operator can be constructed. From the obtained discrete operator, one can calculate its orthogonal decomposition and use it as basis for the representation of all other solutions.

Figure 7.7 shows 50 projection coefficients to the invariant basis, from a total of 801 obtained for each simulation since the discretization of the solution has $N_h = 801$ points. For the components x and y of the 11 simulations corresponding coefficients are obtained. Observe the strong decay of the coefficients in figure 7.7.

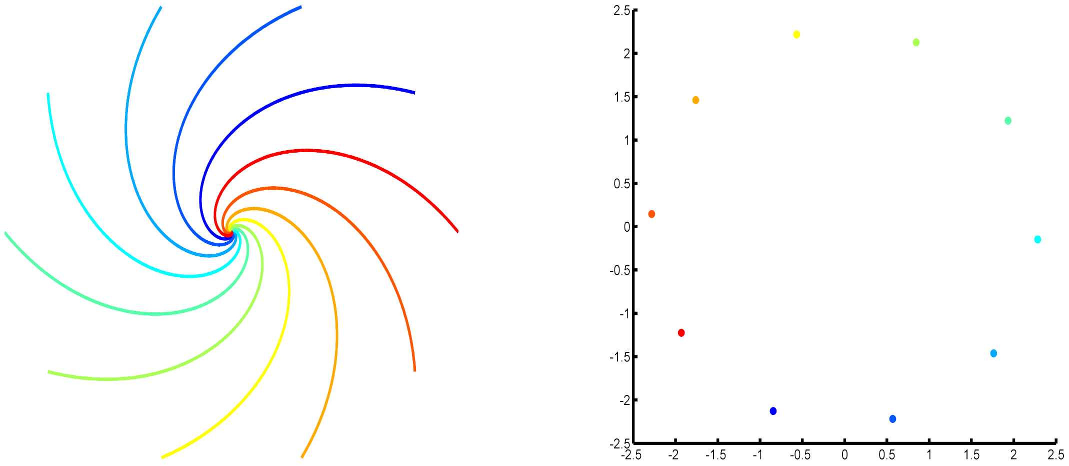


(a)

(b)

Figure 7.7: Projection coefficients for the 11 ODE solutions, (a) in x, (b) in y

According to our proposed approach, the group actions are reflected along the projection coefficients.



(a) (b)

Figure 7.8: 11 ODE solutions (a) and (b) rotation orbit given by coefficients x, y of the first component

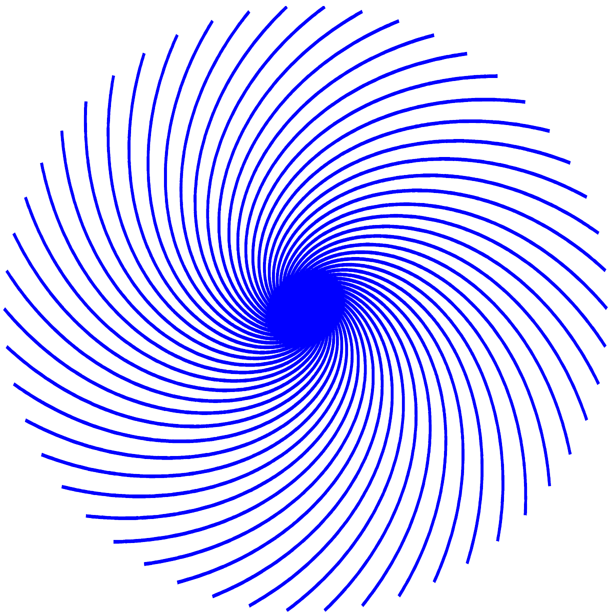


Figure 7.9: 51 ODE solutions interpolated from the rotation orbit

That is, we take the coefficients of the first components, which has the highest absolute values for all the 11 solutions to evaluate the action of the transformation group. There are 11 coefficients for the x direction and 11 for the y direction. The coefficients for the first component represents the orbit of the group action. Plotting all pairs altogether, we obtain figure 7.8 (b). It can be clearly seen that this orbit encode the information of the group action (a rotation) as expected.

In the analytical Lie group method, once the invariant equation is found (equation 7.3.3), one can easily obtain any other solution along the group orbit. In our case we have no equation to evaluate but, we can interpolate intermediate points along the circle orbit to compute any intermediate solution. That is we reconstruct those solutions using the common basis and the interpolated coefficients along the circle as coefficients for the first component. For the other components we get the coefficients using the same approach. The solutions obtained in this way are shown in Figure 7.9.

Notice that the described process of getting those extra "synthetic solutions" justifies our assertion that, using invariant operators a kind of numerical Lie group method is obtained that operates on solutions, without the knowledge of the analytical differential equations.

7.4 Comparison with other related Methods.

We will describe shortly other methods that make use of a convenient basis to represent a dataset. Principal Component Analysis is the most used method for such task but there are other approaches which uses a graph representation of the dataset.

Principal Component Analysis.

Due to its simplicity, the Principal Component Analysis (PCA) is one of the most employed methods for dimensionality reduction [Lee and Wasserman, 2010], we use it as a baseline method. Given a dataset $X = \left\{X^1, X^2, \ldots, X^m\right\}, X^i \in \mathbb{R}^n$, PCA requires the computation of the eigenvalues and eigenvectors of the covariance matrix $X_c X_c^T$. An efficient way of doing it, is via its connection to the SVD decomposition given by $[U S V] = svd(X_c)$, where X_c is a centered representation of X. U is a column matrix of the eigenvectors of $X_c X_c^T$, S is a diagonal matrix whose diagonal elements are the singular values which are the square roots of the eigenvalues of the covariance matrix. V is the complementary base. When the eigenvalues are ordered in a descending order, then the corresponding first eigenvector represents the first main variation of the dataset, the second to the second and so on. Often only a few coefficients are necessary to reconstruct each data so that $X^i \approx \sum_{k=1}^{p} \langle U_k, X^i \rangle U_k$ with $p < m \ll n$. The approach then reduces to the calculation of p eigenvectors.

In the case of numerical simulations in a finite element space, each simulation can be considered as a point in $\mathbb{R}^n, n = 3N_h$, where N_h is the number of vertices, in order to apply the PCA. Comparing the basis for the PCA, Laplace-Beltrami operator and the operator of independent components one observes significant differences. PCA can concentrate the variability of the data in just m coefficients, where m is the number of simulations available. The other operators are bigger, of the order of the number of nodes N_h, but the first coefficients are large in comparison with the rest and that is the reason why we can use them for data analysis. As shown those coefficients can be related to the coarse part of the geometry [Karni and Gotsman, 2000] for the Laplace Beltrami operator and to the first nonlinear components for the operator of independent components [Singer and Coifman, 2008]. For the PCA case, if we attempt to view it as an operator, the first coefficients are related to the variability of the data and not to a smooth part of the geometry nor any nonlinear component. The PCA works well if the data is localized around a linear manifold, in other cases nonlinear dimensionality reduction methods are needed. The operator approach proposed in this thesis a way to obtain the information of this low dimensional structures, we can decompose the data variability in a flexible manner by changing the operator, which is constructed to preserve a quantity or property.

Other Decompositions

For dealing with meshes other decompositions can be used. For example Wavelets for graphs have been used successfully [Coifman and Maggioni, 2006,Hammond et al., 2011] as a multi-scale representation that is able to represent local features on graphs, contrasting with the global character of the PCA decomposition. Treelets [Lee et al., 2008] can be used to find multi-scale representations on graphs for the cases where the data on the graph is non-smooth.

In contrast to these studies, we are interested in finding a multi-scale orthogonal decomposition of not only one dataset but many. We are dealing with a specific type of data sets, namely simulation data, that is numerical solutions of partial differential equations and propose to study the structure of simulation data sets by using its orbits, defined as the trajectories obtained by following the action of a transformation group. The orbits are gained using a basis obtained from an operator invariant to the transformation group. We have observed in our numerical studies that a representation of a specific group can be obtained using an invariant operator. Specifically for the case of isometries using the Laplace-Beltrami operator on a surface mesh and its orthogonal decomposition, we are able to show numerically that orbits can be obtained for the translation group in the first component and for the rotation group in the second component and so on. Other isometric group actions are reflected in other components of the orthogonal decomposition.

The use of wavelets or treelets for the decomposition of data defined on a graph can certainly be a very interesting extension of our work. Nevertheless, in our opinion it is not obvious, in the case of simulations, how to extend the invariance property to such a basis in contrast to the ones obtained from an invariant operator. Notice also, that directly linked to the invariance is the possibility of analyzing simulations in the context of shape spaces. Shape spaces are per definition quotient spaces of a so called pre-shape space with respect to a transformation group acting by isometries.

Using shape spaces in a Riemannian manifold framework allows for the use of a specially defined metric. That is, in such a space a base manifold (in our case the space of embeddings or immersions in $\mathbb{R}^2$ or $\mathbb{R}^3$) is used and the quotient map with respect to a transformation group is assumed to be a Riemmanian submersion. This assumption is commonly used in the area of shape spaces and we would like to think of simulations as a certain kind of shape subject to some transformations so that those concepts can be extended to the treatment of simulations. The use of a Riemannian submersion allows the use of the metric in the space of immersions for the quotient space of simulations allowing an essential simplification.

Finally using invariant operators contains also a fundamental difference to all other approaches mentioned above, namely once the operator is setup, the basis can be used for any instance of a specific transformations even if the data representing them were not available in the training set (see Section 7.3 for an example). The application we have in mind is the analysis of simulation bundles. Those are obtained by the numerical solution of differential equations where some set of parameters has been varied. In this setting we propose to obtain a low dimensional representation of such spaces using the group orbits, that can be represented by the spectral coefficients obtained by projecting the data sets to the common invariant base.

7.5 Operator Invariance Dependence

Taking a closer look at the proposed method, the important step is the construction of the operators. For example pose invariance of a deformation can be achieved using an approximation of the Laplace-Beltrami operator constructed using geodesic distances in the finite element mesh. For a stochastic Itô process one can even obtain invariance to a general nonlinear transformation. After obtaining discrete approximations of these operators, we project the data to the eigenvectors of the operator spectral decomposition and obtain a new representation.

To put the method in a general form, the main idea is to define a metric that is invariant to some transformation group and then using it to construct an invariant operator. The approach is very flexible if we manage to identify the metric and build with it the correct invariant operator corresponding to the way simulations are assumed or observed to behave under change of parameters. Notice that for the isometric case we have used a metric that involves a geodesic distance (approximated by the graph distance). For the case of the independent components the distance in the non-observable space evaluated using data in the transformed space is used to construct the operator. A further example of a way to define a different metric is explained in section 9.1. We also mention the Bi-Laplacian distance as another type of metric that can be used in the context of our method, this distance is pose invariant, noise resilient and insensitive to small topological changes of the mesh [Lipman et al., 2010].

Notice also that other types of operators can be used, a particular related example is the Schrödinger operator [Iglesias and Kimmel, 2012]. A pose invariant signature for the analysis of surface meshes has been successfully implemented in [Aubry et al., 2011]. Notice that an orthogonal decomposition can be obtained from this invariant operator and with it, a similar type of analysis as the one proposed here can be implemented.

CHAPTER 8

Reduced Basis Methods with Invariant Basis

In the context of the analysis of simulations as solutions of partial differential equations, engineers are interested in obtaining approximate surrogate models to avoid the expensive solution phase. These models are constructed based on physical arguments or more generally using so called simulation snapshots from which an orthogonal basis can be gained using principal component analysis, the basis is reduced to a few components. Once the basis is available, for the cases in which the model equations are available, a simplified system of differential equations can be derived using a Galerkin formulation that is later solved for the spectral coefficients. Let us consider the invariance property of our approach using the orthogonal eigenbasis of the corresponding operator in the Galerkin formulation, so that these can be used as an invariant basis. These approaches are studied in the context of reduced basis methods, where as mentioned above, the partial differential equations are supposed to be known. We propose the use of the orthogonal basis of invariant operators in the context of the reduced basis method. Specifically we deal with an extension of one approach into this area, namely the proper orthogonal decomposition (POD).

Firstly, we will analyze the effect of the use of an invariant basis in the case of parameterized partial differential equations. Then we will show by means of an example, a method to utilize simulation snapshots to construct an invariant basis that is able to learn an intrinsic nonlinear parameterization. Finally we will deal with a specific nonlinear PDE and derive a Galerkin formulation using, on the one hand a standard POD for determining a basis; and on the other hand, using an invariant basis. Numerical results obtained with both approaches are then compared.

8.1 Reduced Basis and Parameter Identifiability

As described in section 8, the principle of the reduced basis method (RBM) consists of searching a low dimensional representation in order to approximate the elements of the space of solutions

$$\varphi_N(\mathcal{P}) = \{u_N(\boldsymbol{\mu}) \in V_N : \boldsymbol{\mu} \in \mathcal{P}\} \subset V_N,$$

using a linear global approximation under the separable form

$$u_N(x,\boldsymbol{\mu}) = \sum_{j=1}^{N} \alpha^j(\boldsymbol{\mu})\zeta_j(x), \tag{8.1.1}$$

while keeping N as small as possible and satisfying

$$\inf_{v \in V_N} \|u(\boldsymbol{\mu}) - v\|_V < \epsilon \quad \text{for all} \quad \boldsymbol{\mu} \in \mathcal{P}.$$

In the standard POD, given a set of snapshots one can obtain an orthogonal basis ζ_j using the singular value decomposition. This basis concentrates the variability of the data at few components of the orthogonal decomposition [Quarteroni et al., 2015]. Notice however that the basis obtained in this way depends implicitly on the parameter set $\boldsymbol{\mu}$ since it is obtained from the snapshots evaluated for a specific parameter sampling. In our proposed approach instead, the basis is constructed precisely independent to the transformations that occur as a result of a specific parameter change. This highlights a very interesting observation, the projection coefficients to this basis reflect the parameter variations directly independent of the basis. Let us state this observation a little bit more formally.

Proposition 8.1.1. *Given a simulation dataset $\{u_i\}_{i=1,m} \in \mathbb{R}^{N_t}$ obtained using a sample of input parameters $\boldsymbol{\mu} \in \mathcal{P}$. Assuming all simulations are obtained by a distance preserving transformation or in a stochastic Itô setting by a nonlinear transformation. All simulations can be expressed using an orthogonal basis ψ which is independent of the input parameter variations in $\mathcal{P}$.*

Proof. An operator that is invariant to distance preserving transformations is the Laplace-Beltrami operator, evaluated as described in section 6.4.4. For a stochastic Itô setting an operator invariant to a nonlinear transformation can be constructed as described in section 6.4.6. Evaluating the eigenvectors from each of these operators an orthogonal decomposition of the given simulation bundle can be written as

$$u_i(x,\boldsymbol{\mu}) = \sum_{j=1}^{N_t} \alpha_i^j(\boldsymbol{\mu})\psi_j(x), \tag{8.1.2}$$

the independence of the basis from the parameter $\boldsymbol{\mu}$ is a consequence of the invariance assumption. Further simulations for a specific input parameter combination, not in a training set, can be represented using the invariant basis. □

We will demonstrate the usefulness of this proposition by using it in several forms in the following sections.

8.2 POD Snapshots with an Invariant Basis

In the context of simulations, the variation of several input parameters produces a corresponding solution. Often the solutions depend only on a few parameters, which is a-priori unknown, and in addition, the influence of such parameters on the solution is also unknown. Finding them is very important for engineers when developing new products.

The proposed invariance principles in this thesis are adequate to address both aspects that have been mentioned before. That is, to extract nonlinear parameter dependences from the data and to estimate their influence in the corresponding solutions. The proper orthogonal decomposition (POD) is the standard approach used in industry [Quarteroni et al., 2015]. Simulation snapshots are obtained and with them a basis is calculated representing the variability of the data. We assert that even in the case of being able to sample simulations as desired, the POD has limitations in the presence of nonlinearities and we would like to show them in an example. We will construct a synthetic example that should approximate the behavior of a 3D object under different deformations caused by a nonlinear parameter variation. In applications, each of these deformations will correspond to a finite element solution where some input parameters have been changed.

The example contains synthetic deformations, that is, they are not obtained by a finite element simulation but instead we morph isometrically a reference geometry. The morphing is achieved by modifying several parameters in a nonlinear way. The final objective is to demonstrate how well such parameters are identified using only the morphed deformations as input information using the standard POD and the new approach. In the standard POD, the projection coefficients are used as a reduced representation of a simulation dataset for further processing in a Galerkin solution. In this example we will use a specially chosen set of projection coefficients corresponding to a nonlinear function. Based only on the snapshots, we would like to recover the introduced nonlinear parameter variations using the operator approach and the standard POD method.

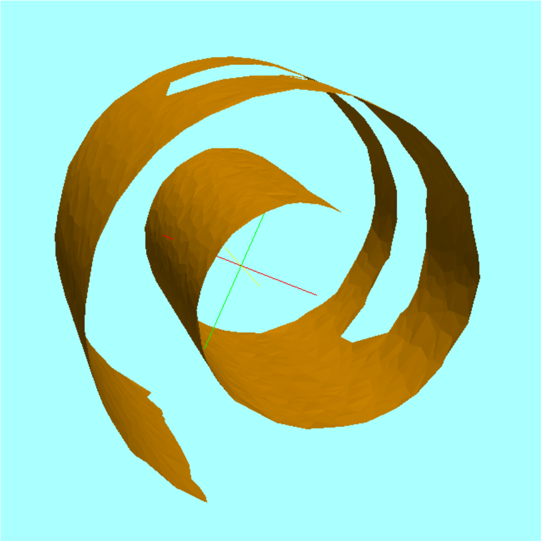


Figure 8.1: original part

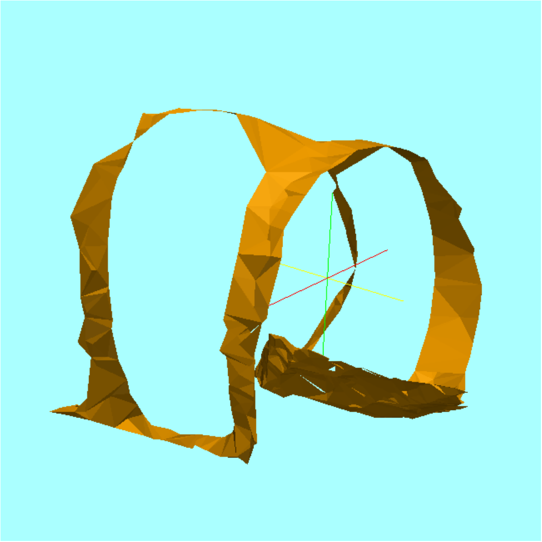


Figure 8.2: deformed part

Consider the 3D object in figure 8.1, it has a total of 1000 nodes. This will be our original geometry that will be deformed to obtain the synthetic simulations. We employ the following procedure:

1. A Laplace operator approximation with a Gauss kernel, using the graph distances between points in the cloud is calculated. The eigenvectors of this operator form an orthogonal basis and we can reconstruct the geometry with them.

2. For the reconstruction we will only use 10 coefficients. A distortion of the geometry is obtained as a consequence of the thresholded reconstruction, see figure 8.2.

3. We define now the following three functions $z_1(x, y)$, $z_2(x, y)$ and $z_3(x, y)$. Let $R = \sqrt{x^2 + y^2} + \epsilon_m$, with $\epsilon_m = 2.2e^{-16}$, then

$$\begin{aligned} z_1 &= \exp(-A_1 \times \sin(R/B_1)/R) \\ z_2 &= A_2 \times \cos(R/B_2)/R \\ z_3 &= A_3 \times \sin(R/B_3)^5/\mathbb{R}^2 \end{aligned}$$

   with $A_1 = 500$, $B_1 = 10$, $A_2 = 1000$, $B_2 = 10$, $A_3 = 10000$, $B_3 = 10$.

4. There are 3 coordinate directions and corresponding spectral coefficients x, y, z for each direction. The values of x, y correspond to a uniform mesh of size 13×13 and those values are taken for the spectral coefficients. The spectral coefficient in the third direction z is replaced by the values of z_1, z_2 and z_3. Therefore, a total of 169 (13×13) spectral coefficients are constructed as follows:

$$C(i, :, :) = [(x(i), y(i), z_1(i)); (x(i), y(i), z_2(i)); (x(i), y(i), z_3(i)); c_4; \ldots; c_{10}], \tag{8.2.1}$$

   with $i = 1, \ldots, 169$. The coefficients $c_4, \ldots, c_{10} \in \mathbb{R}^3$ are kept the same for all synthetic simulations and are equal to the spectral coefficients of one simulation.

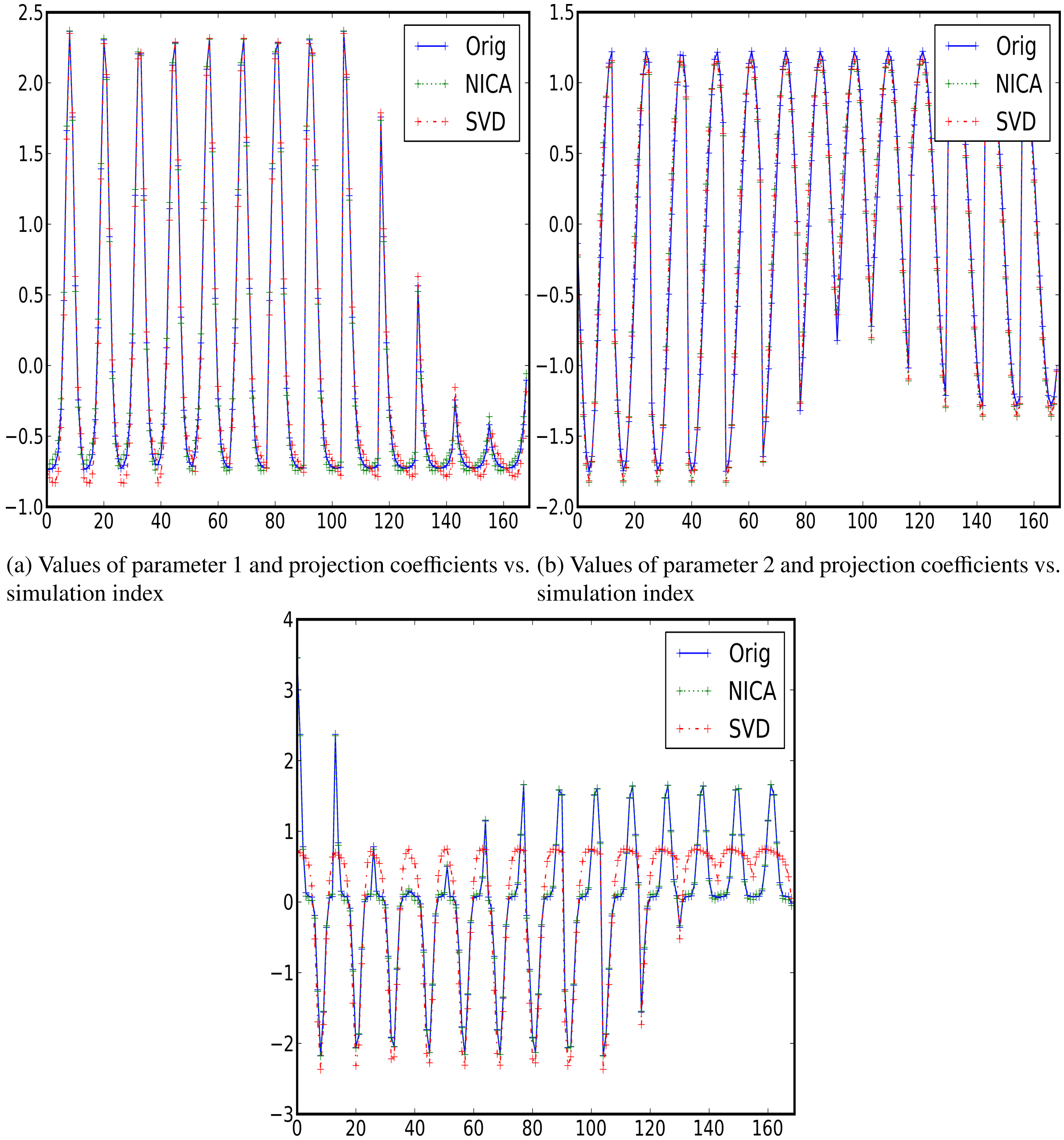
5. On the basis of the set of 169 coefficients, 169 synthetic simulations are obtained using the eigenvectors of the operator from step 1 multiplied by the corresponding coefficients.

We will concentrate on the direction z, which is the direction where changes in the parameter take place and compare the actual variation in this direction with the corresponding ones obtained using the operator approach and the ones obtained with the POD by using the singular value decomposition (SVD). For the POD we took only the component z of the 3D coordinates of the deformations and evaluate the covariance matrix of size 169×1000.

For the operator approach, we will use the one described in section 6.4.6, namely the operator of independent components (NICA). The input data for our algorithm are the 169 3D meshes. The given dataset is projected to the operator basis and the corresponding coefficients 169×3 of size 1000 are evaluated. We have chosen the three largest values of the coefficients for all simulations in the z direction as before getting in total 169 coefficients.

Figure 8.3 shows the corresponding parameters as recovered using the operator of independent components and using the POD, evaluated using the SVD. A normalization is used in order to compare it with the original parameter for each of the three parameters used. The recovered parameters using the operator approach agrees very well with the original parameter variation. Using the POD the agreement is also very good, but not for all parameters as seen in figure 8.3 (c) for the parameter 3.

We have also analyzed the effect of adding 30% uniform noise to our simulations, the corresponding parameters as recovered using the operator approach and using the POD are shown in figure 8.4. The presence of noise affects both approaches negatively, nevertheless the approximation of the parameters for the new approach is overall better in comparison with the one using the standard POD approach.

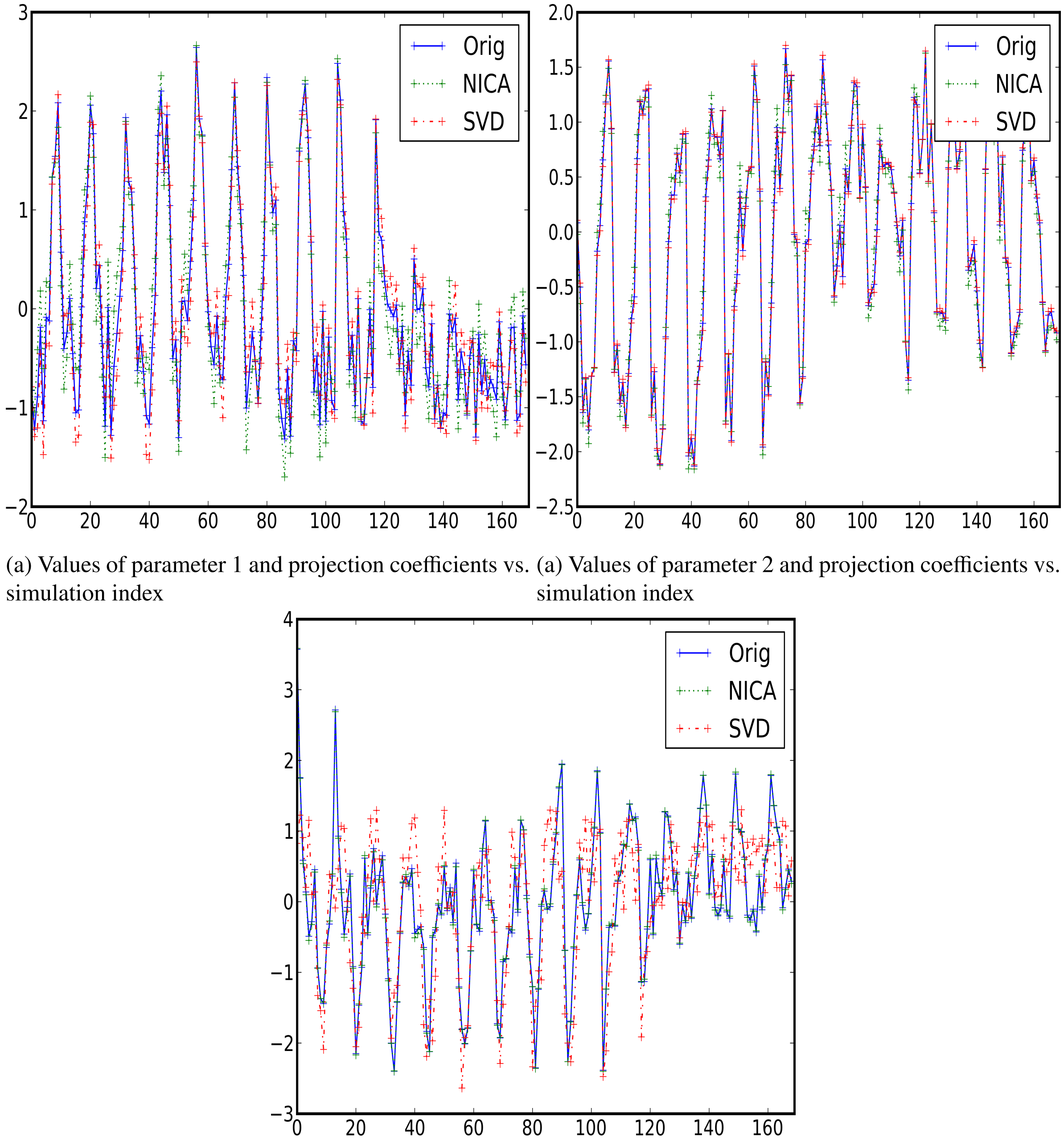


(a) Values of parameter 1 and projection coefficients vs. simulation index

(b) Values of parameter 2 and projection coefficients vs. simulation index

(c) Values of parameter 3 and projection coefficients vs. simulation index

Figure 8.3: Comparison of original and recovered parameters (projection coefficients). A unique orthogonal basis is used and all 169 simulations are projected to this basis. The projection coefficients for the orthogonal components with higher variations from the SVD and from the operator approach (NICA) are used.



(a) Values of parameter 1 and projection coefficients vs. simulation index

(a) Values of parameter 2 and projection coefficients vs. simulation index

(a) Values of parameter 3 and projection coefficients vs. simulation index

Figure 8.4: Comparison of original and recovered parameters (projection coefficients). A unique orthogonal basis is used and all 169 simulations (with noise added) are projected to this basis. The projection coefficients for the orthogonal components with higher variations from the SVD and from the operator approach (NICA) are used.

Summary of Results for the Synthetic Simulation

We can observe that the proposed approach has the following properties:

- it requires as many coefficients as number of nodes in the mesh (several thousand), the POD requires as many as the number of datasets. Nevertheless the approach concentrates most of the data variation in very few spectral coefficients,

- it has the special property of being geometry aware as we obtain eigenvectors of an operator valid for a 3D geometry, in contrast to the POD case,

- from the synthetic data, we were able to approximate very well; the low dimensional independent parameters even in the presence of large amounts of noise in the data,

- the above results imply that the basis obtained from the operator approach has a generalization ability to cope with variations that are not available in the snapshots. Notice that this is a consequence of proposition 8.1.1

The above results are very important for applications using the POD. Obtaining the basis as suggested using an invariant operator, could improve reduced order models of the standard Galerkin-POD method.

8.3 Galerkin-POD with Invariant Operators.

In this section we will show, by means of some examples, firstly the way the new invariant basis is included in a Galerkin approach and secondly we will study awareness to nonlinearities. Special attention is given to whether or not the new basis is able to reflect parameter changes in the simulation not covered by the snapshots.

8.4 POD with an Invariant Basis.

The basis we generate by using an invariant operator has a different character as the one obtained from the snapshots. As mentioned in section 6.8, the number of basis functions necessary to represent a simulation vary according to the smoothness of the data. For smooth functions few basis vectors are necessary to represent the function under the new basis. This invariant basis, once obtained, can be used in a Galerkin approximation to solve a given partial differential equation. It is certainly interesting to investigate how the reduced model behaves since the basis is actually independent to suitable parameter variations.

To demonstrate the behavior of the reduced Galerkin system under the new basis, we use the following nonlinear ODE Burgers equation with appropriate boundary and initial conditions,

$$\frac{\partial}{\partial t}u(t,x) + u(t,x)\frac{\partial}{\partial x}u(t,x) - q\frac{\partial^2}{\partial x^2}u(t,x) = f(t,x), \tag{8.4.1}$$

this is a commonly used model in the context of the POD as a test example, since it has properties similar to the Navier-Stokes equation with a combination of diffusion and convection parametrized by q, which can be taken as the Reynolds number. This example and the standard POD formulation are based on [Jarvis, 2012] where more details can be found. First let us consider the finite element discretization of the above equation which involves the use of an approximation of $u(t,x)$ defined as

$$u_n(t,x) = \sum_{j=0}^{N_h} \alpha_j(t)\mathcal{B}_j(x), \tag{8.4.2}$$

where N_h is the number of nodes and the basis $\mathcal{B}_j(x)$ has local compact support. For this example we choose the following standard hat functions

$$\mathcal{B}_i(x) = \begin{cases} \frac{x-x_{i-1}}{h} & \text{for } x \in [x_{i-1}, x_i] \\ \frac{x_{i+1}-x}{h} & \text{for } x \in [x_i, x_{i+1}] \\ 0 & \text{otherwise,} \end{cases} \tag{8.4.3}$$

Rewriting the nonlinear term as $\frac{1}{2}\frac{\partial[u(t,x)^2]}{\partial x}$, the Galerkin weak formulation is given by,

$$\begin{aligned} \sum_{j=0}^{N_h}\Big(\dot{\alpha}_j(t)\int_0^1 \mathcal{B}_j(x)\mathcal{B}_i(x)dx + \frac{1}{2}\alpha_j(t)^2\int_0^1 \mathcal{B}_j'(x)\mathcal{B}_i(x)dx \\ + q\alpha_j(t)\int_0^1 \mathcal{B}_j'(x)\mathcal{B}_i'(x)dx\Big) = \int_0^1 f(t,x)\mathcal{B}_i(x)dx \end{aligned} \tag{8.4.4}$$

The equation (8.4.4) is a system of N_h+1, including Dirichlet boundary conditions. This can be written as

$$\dot{\alpha}(t) = M^{-1}\left[F(t) - \frac{1}{2}B(\alpha(t))\alpha(t) - qC\alpha(t)\right] \tag{8.4.5}$$

$$\alpha(0) = M^{-1}G \tag{8.4.6}$$

with $M_{ij} = \langle \mathcal{B}_j(x), \mathcal{B}_i(x)\rangle$, $C_{ij} = \langle \mathcal{B}_i', \mathcal{B}_j'\rangle$, $F_i(t) = \langle f(t,x), \mathcal{B}_i(x)\rangle$, $G_i = \langle u_0(x), \mathcal{B}_i(x)\rangle$, $i,j = 0,\dots,N_h$, this equations can then be solved for $\alpha(t)$. The zero Dirichlet boundary conditions are used and for the initial conditions we use

$$u_0 = \begin{cases} 1 & \text{for } x \in [0, 1/4] \\ e^{-(x-0.25)/0.21} & \text{for } x \in (1/4, 1] \end{cases} \tag{8.4.7}$$

8.4.1 Reduced Order Model

The mass matrix M is symmetric and therefore can be decomposed as $M = LL^T$. Then the weighted snapshots matrix is defined as $Y_1 = L^T Y$ where Y is the $N_h \times m$ matrix containing m time snapshots of the finite element calculation. If Y is of rank k then Y_1 is also of rank k

The basis obtained in the proper orthogonal decomposition (POD) minimizes over all possible $m-$dimensional orthonormal basis $W = [w_1|, \dots |w_m] \in \mathbb{R}^{N_h \times m}$, the sum of the squares of the error between each snapshot simulation and its projection onto the subspace spanned by W. This is equivalent [Quarteroni et al., 2015] to finding the eigenvectors of

$$Y_1 Y_1^T \psi = \lambda\psi. \tag{8.4.8}$$

Using the singular value decomposition one can write

$$Y_1 = U\Sigma V^T, \tag{8.4.9}$$

where U is a unitary matrix of dimension $N_h \times N_h$, Σ is diagonal of dimension $N_h \times m$ and V^T is unitary of size $m \times m$. The vectors U_i satisfy

$$max \sum_{j=1}^{m} \left| \left\langle Y_1^j, U_i \right\rangle_{\mathbb{R}^{N_h}} \right|^2 \quad \text{s.t} \quad \|U_i\|_{\mathbb{R}^{N_h}} = 1 \quad \text{and} \quad \langle U_i, U_k \rangle_{\mathbb{R}^{N_h}} = 0 \quad \text{for} \quad k = 1, \ldots, i-1. \tag{8.4.10}$$

If Y has rank k, for $i = 1, \ldots k$, only Σ_{ii} entries are nonzero and one can write $Y_1 = U^k \Sigma^k (V^k)^T$, where k means the first k vectors of the matrix U, V and Σ^k is the $k \times k$ principal sub-matrix of Σ.

Now one can find the spatial POD basis vector by taking $\zeta_i^r(x) = L^{-1} U_i(x)$ for $i = 1, \ldots r$ with $r \leq k$. Using a POD basis, one can replace this basis in the weak form (8.4.4), using the approximation $u^N(t,x) \approx u(t,x)$

$$u^N(t,x) = \sum_{j=1}^{r} \alpha_j^r(t) \zeta_j^r(x), \tag{8.4.11}$$

by doing that, one reduce the order of the ODE system (8.4.4) from $N_h + 1$ to r. For each time step, one solve only for the coefficients α_j^r.

The POD reduced model can then be shown to be given by

$$\dot{\alpha}^r(t) = \left[F^r(t) - \frac{1}{2} (\Phi^r)^T B(\Phi^r \alpha^r(t)) \Phi^r \alpha^r(t) - q C^r \alpha^r(t) \right] \tag{8.4.12}$$

$$\alpha^r(0) = (\Phi^r)^T \alpha(0), \tag{8.4.13}$$

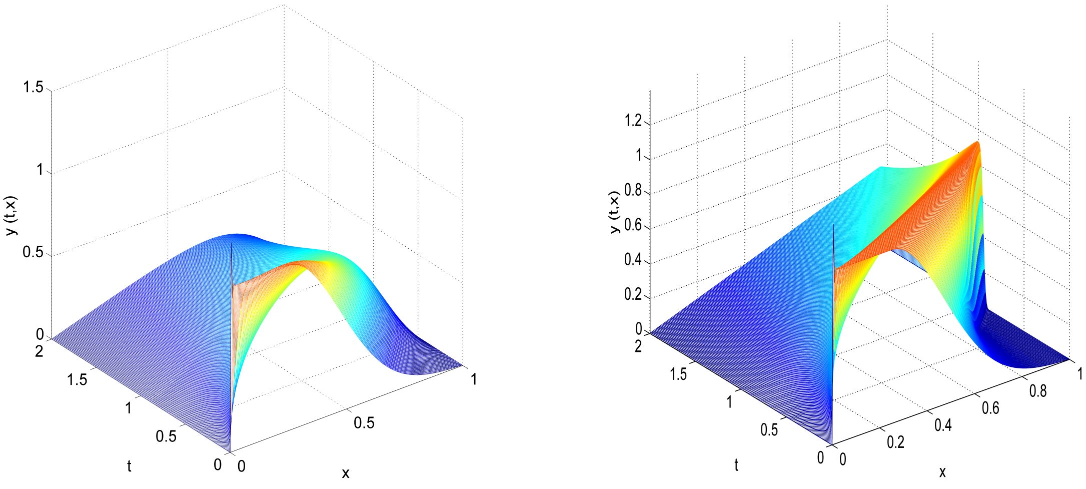
denoting the collection of POD vectors by Φ^r, then $F_i^r(t) = \left\langle f(t,x), \zeta_i^r(x) \right\rangle$ and $C^r = (\Phi^r)^T C \Phi^r$.

Up until now we have presented the traditional POD formulation. Our proposed method uses an invariant basis which is also an orthogonal basis so that a natural extension to the POD consists in replacing the basis with ours to obtain an equivalent expression to (8.4.12). In other words, we have replaced the POD basis $\zeta_i^r(x) = L^{-1} U_i(x)$ with the basis $\psi^i(x)$ obtained from an operator invariant to a transformation that transforms simulations into simulations. We will next describe how we obtained this operator for this specific problem.

8.4.2 The Invariant Operator.

The solutions to the PDE (8.4.1) are curves that, as the time increases, shows an apparent movement along the x axis. Depending on the parameter q, this movement will be dominated by a diffusion or by a pure convection (translation) as can be seen in figure 8.5.

We will now take a closer look at the shapes of the curves, each of them for each and every time step can be assumed to be obtained by transforming a reference curve. That is, a new shaped curve



(a)

(b)

Figure 8.5: Solution curves for the nonlinear PDE (8.4.1) for a) q=20 diffusion dominated and (b) q=400 convection dominated ranges

is obtained from the reference one. In addition the transformation can be assumed as preserving the length along the curve. That is, the transformation is a distance preserving, i.e an isometric one. This observation suggests a way to construct an invariant operator by considering each curve to be a planar shape.

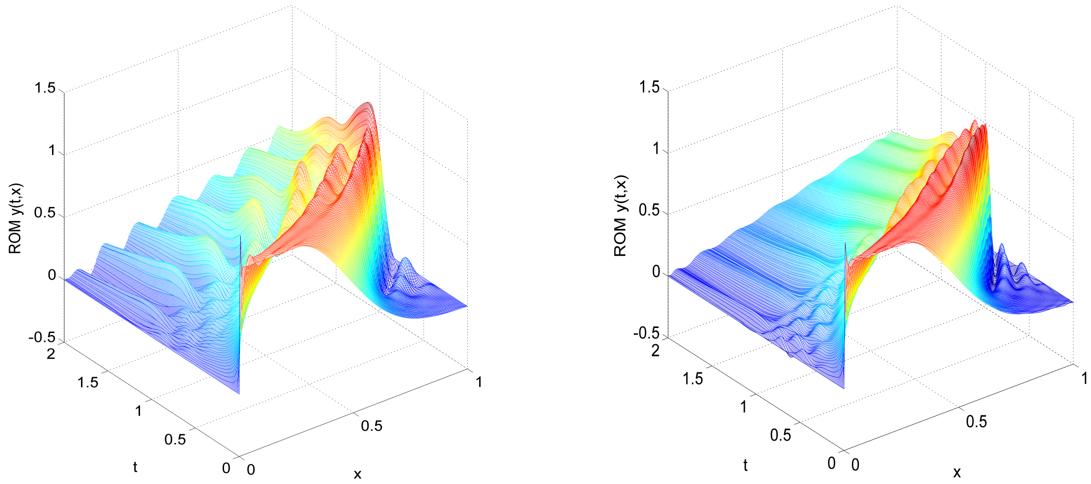
Using as input the solution for one specific time step, lets say for $t = 0$ of the simulation for $q = 20$ we have constructed a distance preserving operator based on Algorithm 4. The eigenvectors of this operator are then the invariant basis that can be used instead of the one in the standard POD approach. Based on this basis we have implemented an extension of the POD method by replacing the new basis in the Galerkin approximation. As before a reduced system of ODEs for the spectral coefficients has to be solved. Specifically we solve (8.4.12) for $q = 20$ using $p = 20$ invariant basis $U^i = \psi^i(x), i = 1, \dots, 20$.

We have noticed that, there are no visual differences between the curves obtained solving the original ODE, the POD approximation or the invariant basis approximation. The result for all of them can be seen in figure 8.5 (a).

Of strong interest in the context of the POD approximation is to obtain reduced models that are valid for a range of parameters of the ODE/PDE. At the same time one would like to avoid the need to obtain a new POD-basis if the parameter range changes, as this implies calculating new simulation snapshots. These requirements have practical interest since the solution of the ODE/PDE in an industrial environment is expensive. The use of a reduced model is also very handy in other processes such as in a control or optimization loops. Calculating new basis in such processes is simply not practicable.

For the reduced solution of (8.4.1) we have solved the Galerkin equation (8.4.12) in two different ways for the same parameter $q = 400$. In the first one we have used the POD basis obtained for $q = 20$ and in the second one we have used our invariant basis obtained from an arbitrary simulation (in this case it is obtained from the curve at $t = 0$).

Figure 8.6 shows the result of these calculation that have to be compared with the solution of the ODE shown in figure 8.5 (b). As can be seen from these results, the solutions for the invariant basis and



(a)

(b)

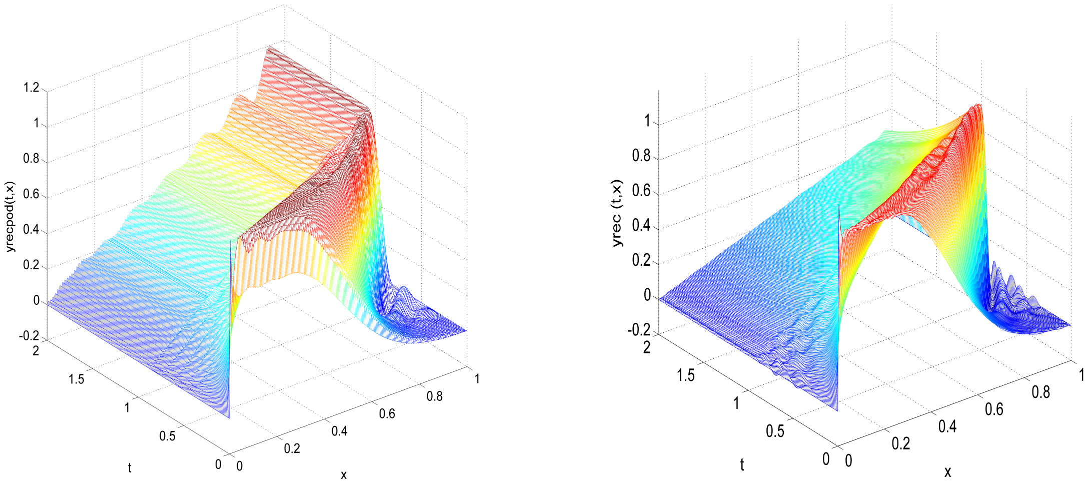
Figure 8.6: Solution curves for the reduced nonlinear PDE 8.4.1 for q=400 using a) 20 POD basis (b) 20 Invariant basis

the ones for the POD, both show deviations from the exact solution. Nevertheless the solution with the invariant basis is closer to the numerical solution with $q = 400$ (figure 8.5 (b)).

In order to be able to assert the reasons for those deviations, we have first focused on the integration of the reduced ODE. In both cases we have used an integration method based on a Matlab solver. There are several implementations of it including the ones that can solve the more difficult stiff problems. We have tested several available implementations ode45, ode23 and ode113 without any noticeable improvement in the final result.

Even if a better ode integrator is not available, we can do the test in an idealized case in which the solver performs in an ideal way. That is, we use the solution for $q = 400$ and project it to the POD-basis as well as to the invariant basis and further reconstruct the solution using 20 basis vectors. Figure 8.7 shows the result of the reconstruction for the POD basis and for the invariant basis, Figure 8.7 (b) shows a significant improvement in the quality of the approximation with respect to Figure 8.6 (b).

The solution of the ODE equation 8.4.4 and the reduced POD system with the new basis can be repeated for several parameters in the interval $q \in [20, 400]$. A projection of all simulations to the POD basis for $q = 20$ and to the invariant basis can then be done with a subsequent reconstruction using 20 components. Doing this evaluation, and taking the mean of the median of the error for all time steps for each value of q, we have obtained the results shown in figure 8.8 where it can be clearly be seen that the error, by using the invariant basis, stays constant for the all range of values of the parameter q (curve labeled as New). This behavior contrasts with the error obtained using the POD basis whose absolute value increases drastically as the value of the parameter q is increased (curve labeled as POD Basis). Recalculating the basis for each parameter maintains the error barely constant as seen in figure 8.8 (curve labeled as Basis recomputed). To have a better overview over the error for the different cases, we repeat the plot using a logarithmic scale in 8.9. This experiment verifies proposition 8.1.1 which establishes that the basis is independent of the input parameters. The solution of the reduced system using the invariant basis (curve labeled POD Basis new) in figure 8.7) shows an improvement with



(a)

(b)

Figure 8.7: Solution curves obtained by reconstruction using 20 basis for q=400 using a) the POD basis (b) the invariant basis

respect to the standard POD but it is not independent of the parameter q. This behavior is assumed to be caused by the integration of the reduced ODE since the basis itself has been shown as capable of representing the simulation curves for the all range of the parameters q (curve labeled New). From this experiment we presume that the results obtained by solving the Galerkin reduced system are affected by the performance of the Matlab integrator. These results suggest that the integrators are not adequate to solve the reduced ODE using the invariant basis.

The POD-Galerkin approximation is very useful as a method for creating a reduced model from a PDE or ODE, the actual dimension of the problem gets reduced in several orders of magnitude. We have shown that modifying the basis in the POD using an invariant basis could produce reduced models that are more general. That is, the basis can still be used, even if PDE/ODE contains parameters that modify the behavior of the solution.

If an invariant basis could be used in the POD approach, then one could actually construct a better POD that overcomes the limitations of the standard one in cases where instabilities are present or when one uses snapshots in a certain parameter range which deviates significantly from the actual parameter variations.

We conclude here with the description of the proposed new methodology for data analysis and the POD extension. Further extensions and industrial applications are described in the next chapters.

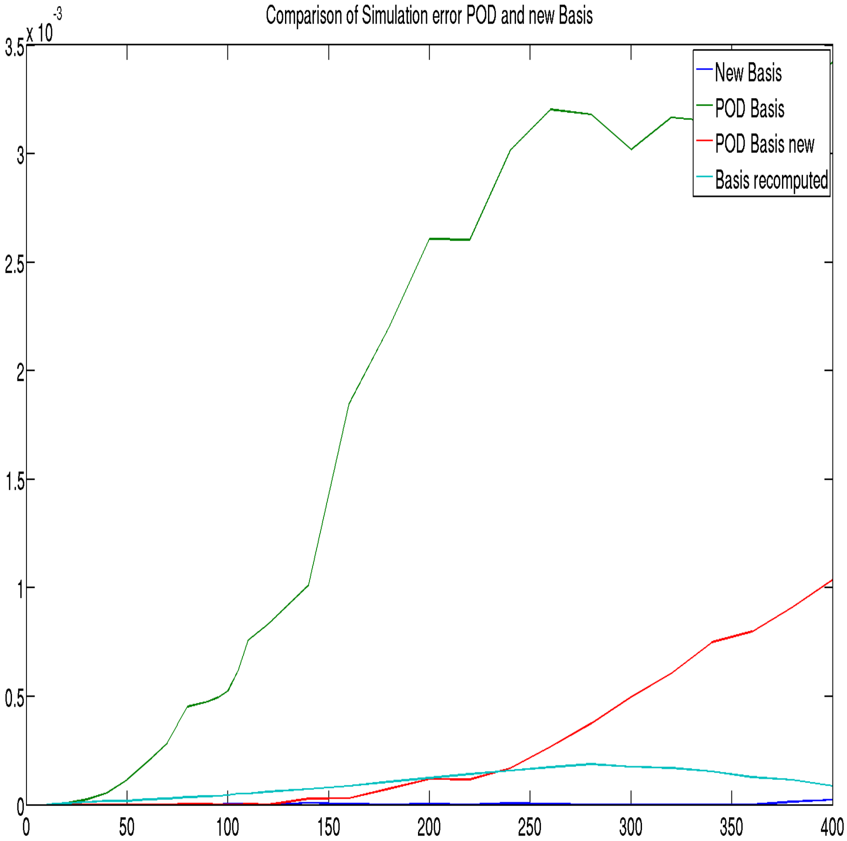


Figure 8.8: Median of the mean of the error with respect to the full Galerkin 8.4.4 for the standard POD reduced solution using a fixed basis for the parameter $q = 20$ (POD Basis), the modified POD solution with the invariant basis (POD Basis new), the reconstruction from the invariant basis (New Basis) and the POD solution where a basis is calculated for each and every parameter (Basis recomputed). A total number of 20 basis functions are taken for the reduced solutions and for the reconstruction

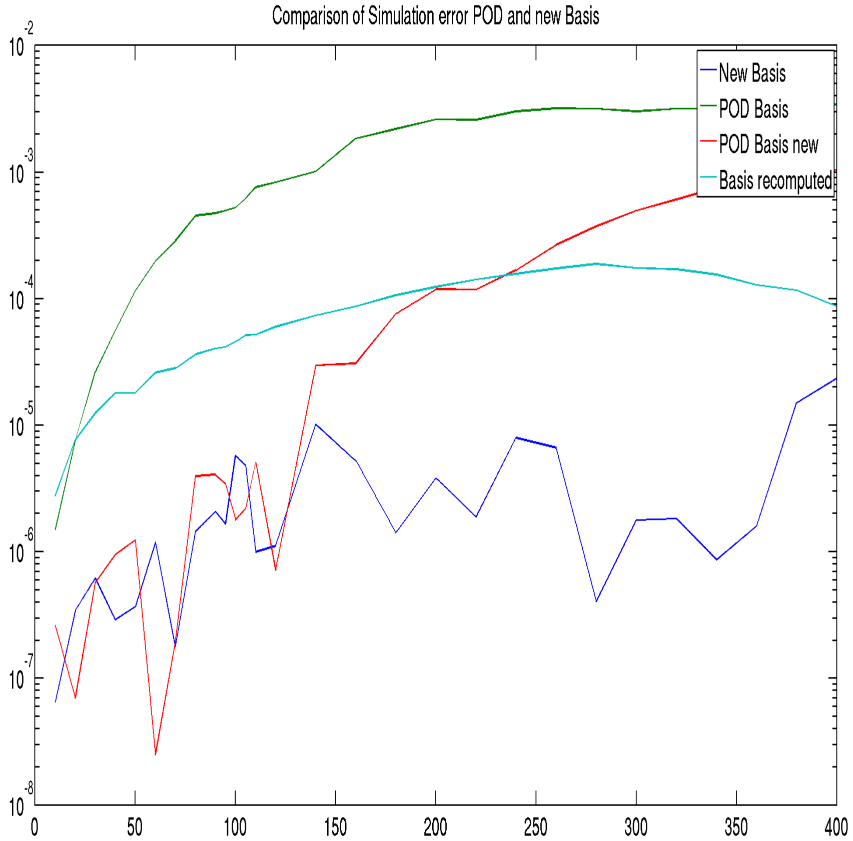


Figure 8.9: Median of the mean of the logarithm of the error with respect to the full Galerkin 8.4.4 for the standard POD reduced solution using a fixed basis for the parameter $q = 20$ (POD Basis), the modified POD solution with the invariant basis (POD Basis new), the reconstruction from the invariant basis (New Basis) and the POD solution where a basis is calculated for each and every parameter (Basis recomputed). A total number of 20 basis functions are taken for the reduced solutions and for the reconstruction

CHAPTER 9

Extensions

Several extensions of the proposed approach described in section 7.1 will be discussed in this chapter. Firstly we will show how to deal with transformations that are not distance preserving, namely conformal transformations and we will provide an example of them. Next we will present a methodology to evaluate paths in shape space. A path in shape space refers to a sequence of shapes from a given initial to a final one. All elements of the sequence are supposed to be contained in the space of the shapes one is dealing with, where paths of minimal length are called geodesics. In chapter 5 we presented an abstract setting that establishes that in the case of isometries, under certain conditions, geodesics are defined as one-parameter group transformations and that group actions can be represented using eigenfunctions of isometric invariant operators. Based on this, we proposed in section 7.1 a method for data analysis that in some way evaluates group actions using the projection coefficients to the eigenvectors of discrete isometric invariant operators. Those projections can be associated with the orbits and they have been shown experimentally to be orbits for the case of rotations and translations. In other words so called orbit paths are shown experimentally to recover approximations of geodesic paths for those cases. We would like to see how far can we get with this approximation for other invariances than rotations and translations, that is are we able to recover geodesic paths in some specific cases? Since for plane shapes, methods for finding geodesic paths are available, an example in the area of plane shapes will be considered to illustrate this approximation and then we will show a method to evaluate orbit paths in the solution space of deformations in crash simulations.

9.1 Invariance to Conformal Transformations

In section 6.4.4 we have proposed an operator that is invariant to isometric (distance preserving) transformations. In section 6.4.6 an operator is proposed for the case of an stochastic Itô process. Here, a dataset contains a small nonlinear variations forming a cloud around a reference configuration. Based on this cloud a local approximation of the Jacobian of the nonlinear transformation can be estimated. As a result, as shown in section 6.4.6 an operator invariant to a nonlinear transformation can be build for this special case. Still one more case can be defined using the same Laplace-Beltrami operator provided a suitable definition of invariance can be found. This is precisely the case for conformal transformations (area preserving). We will use the following definitions for conformal invariance:

Definition 9.1.1. Let $(\mathcal{M}, g)$ be a differentiable connected Riemannian manifold with associated metrics $g, \tilde{g}$. A conformal change is given by $\tilde{g} = e^{2\varphi} g$ where φ is a smooth function on $\mathcal{M}$.

Definition 9.1.2. A metrically defined operator L_g is called conformal invariant if $L_{\tilde{g}} = e^{-2\phi} L_g$

Proposition 9.1.3. *[Ngô, 2012] The Laplace-Beltrami operator on a surface $\mathcal{M}$ is conformally invariant.*

Proof. This has been shown in [Ngô, 2012] and it can be obtained using local coordinates. The Laplace-Beltrami operator in local coordinates can be written as

$$\Delta_{\tilde{g}} = \frac{1}{\sqrt{|\det \tilde{g}|}} \frac{\partial}{\partial x_j} \Big(\sqrt{|\det \tilde{g}|} \tilde{g}^{ij} \frac{\partial}{\partial x^i} \Big).$$

In this expression replace

$$\tilde{g}^{ij} = e^{-2\varphi} g^{ij}$$
$$\sqrt{|\det \tilde{g}|} = e^{n\varphi} \sqrt{|\det g|}.$$

obtaining

$$\begin{aligned}
\Delta_{\tilde{g}} &= \frac{1}{\sqrt{|\det \tilde{g}|}} \frac{\partial}{\partial x_j} \Big(\sqrt{|\det \tilde{g}|} \tilde{g}^{ij} \frac{\partial}{\partial x^i} \Big) \\
&= \frac{1}{e^{n\varphi} \sqrt{|\det g|}} \frac{\partial}{\partial x_j} \Big(e^{n\varphi} \sqrt{|\det g|} e^{-2\varphi} g^{ij} \frac{\partial}{\partial x^i} \Big) \\
&= \frac{1}{e^{n\varphi} \sqrt{|\det g|}} \frac{\partial}{\partial x_j} \Big(e^{(n-2)\varphi} \sqrt{|\det g|} g^{ij} \frac{\partial}{\partial x^i} \Big) \\
&= \frac{1}{e^{2\varphi} \sqrt{|\det g|}} \frac{\partial}{\partial x_j} \Big(\sqrt{|\det g|} g^{ij} \frac{\partial}{\partial x^i} \Big) + \frac{1}{e^{n\varphi} \sqrt{|\det g|}} \frac{\partial}{\partial x_j} (e^{(n-2)\varphi}) \sqrt{|\det g|} g^{ij} \frac{\partial}{\partial x^i} \\
&= e^{-2\varphi} \Delta_g + (n-2) e^{-2\varphi} g^{ij} \frac{\partial \varphi}{\partial x_j} \frac{\partial}{\partial x_i}.
\end{aligned}$$

For a surface $n = 2$ and this implies conformal invariance according to the definition 9.1.2. □

From proposition 9.1.3 one can see that the difference between the operators under a conformal transformation is only the conformal factor. For simulation data sets that undergo only conformal transformations the same operator can be used (albeit the conformal factor). As for our approach presented in section 7, in order to represent a simulation bundle we need a unique basis calculated from an approximation of an operator invariant to a transformation (for example isometric). For this conformal case, under the conditions explained above, this is still possible. We will provide an illustrative example for this case using the space of concentric spheres. Using a unique basis for this case, an approximation of a geodesic path in this space is also achieved using the projections into this basis.

9.1.1 Space of Concentric Spheres

The space of concentric spheres has been treated in [Bauer et al., 2012] where the geodesics under different type of metrics are calculated. The space of concentric spheres is a subspace of the space

$$B_i = Imm(S^2, \mathbb{R}^3)/Diff(S^2)$$

and is therefore a shape space according to [Bauer et al., 2012]. Different type of metrics on the space of immersions are analyzed in [Bauer et al., 2012]. A general form of this metric for surfaces immersed in $\mathbb{R}^3$ is given by

$$G_f(h,k) = \int_{S^2} \Phi(Area(f), \mu)\tilde{g}(h,k)dArea$$

where $\tilde{g}$ is the Euclidean metric on $\mathbb{R}^3$, $h, k \in C^\infty(S^2, \mathbb{R}^3)$ are tangent vectors of the immersion $f : S^2 \to \mathbb{R}^3$ and $\Phi : \mathbb{R}^2 \to \mathbb{R}_{>0}$ is a positive smooth function that depends on the area of the immersion $f(S^2)$ and on the mean curvature μ. Based on this metric, the geodesic equation in the shape space of concentric spheres has been derived in [Bauer et al., 2012] and is given as,

$$r_{tt} = -r_t^2 \frac{1}{\Phi}\left[\frac{1}{r}\Phi + \partial_v \Phi 4r^2\pi + \frac{1}{r^2}(\partial_\mu \Phi)\right]$$

where v and μ corresponds to the area and mean curvature arguments in Φ, that is $\Phi(v, \mu)$. We recall the special case $\Phi = Area^{-1}$, where the equation simplifies to $r_{tt=}0$, an explicit solution can be found in this case

$$r = C_1 \times t + C_2.$$

That is the geodesic equation depends only linearly on the radius for this special case.

Now using our approach, we can take the Laplace-Beltrami basis for the sphere which are given by the spherical harmonics functions given by,

$$\psi_l^m(\theta, \phi) = \begin{cases} N_l^0 P_l(cos\theta) & \text{if } m = 0 \\ \sqrt{2}N_l^m cos(m\phi)P_l^m(cos\theta) & \text{if } m > 0 \\ \sqrt{2}N_l^m sin(-m\phi)P_l^{-m}(cos\theta) & \text{if } m < 0. \end{cases}$$

Notice first that, such a basis is the same for all concentric spheres in the set and that projections of 3d sphere surface points along the spherical harmonics will produce spectral coefficients that only change linearly with respect to the radius of the sphere. A compact representation of a dataset can be obtained using a change of basis to the Laplace-Beltrami basis, i.e. for the concentric spheres a simple linear representation of this shape space is obtained.

Finally we will compare this evaluation with a numerical solution. We have constructed 1000 spheres with different radius, see figure 9.1. For the experiment we have randomly varied the radius of the spheres as shown in figure 9.2 for the values of the radius that have been used for the experiment. The spheres are approximated on a triangular mesh. We have taken one arbitrary mesh and calculated numerically the approximation of the Laplace-Beltrami operator and solve the eigenvalue problem for it as described in Algorithm 2. Finally we have projected all coordinates of all 1000 spheres to the eigenbasis and have obtained the orbit coordinates for all spheres. Figure 9.3 shows a plot of the orbit coordinates corresponding to the spectral coefficients for the second component of the decomposition for all spheres colored according to the value of the radius. The approximation clearly shows a linear relationship as a function of the radius.

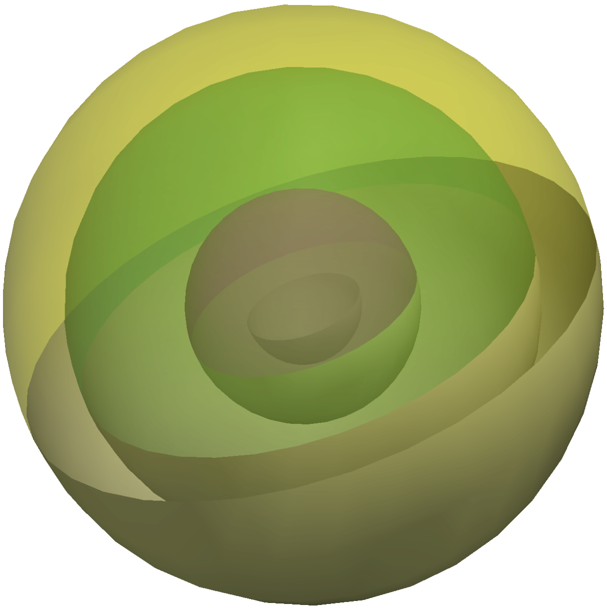


Figure 9.1: concentric spheres

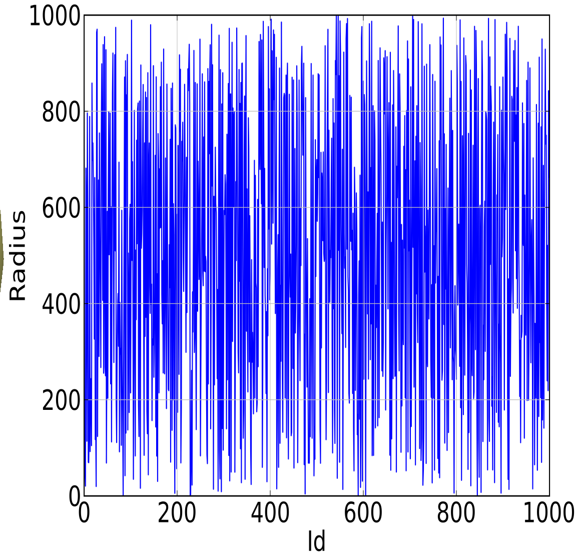


Figure 9.2: set of radius used for all 1000 spheres

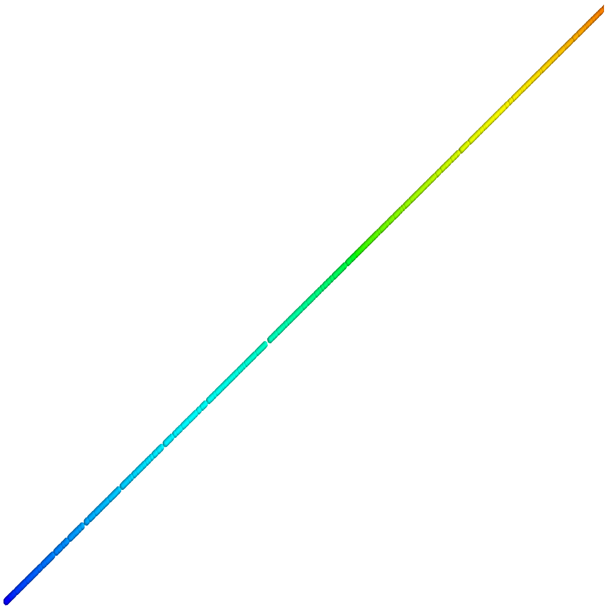


Figure 9.3: simple geodesic for concentric spheres, the color corresponds to the radius of the spheres

9.2 Fast Evaluation of Paths in Shape Space and Simulation Space

In this section we will describe a new method for calculating paths in shape space as a direct extension of the operator approach presented in this thesis. The method uses the eigenbasis of an operator that is invariant to specific transformations. The quotient space of embeddings of a manifold in $\mathbb{R}^n$ modulo a transformation group is considered. Orbits of this space, with respect to a group action, are the spectral coordinates obtained by projecting the given shapes to a unique eigenbasis. The method is applied to the shape space of plane curves and to a simulation bundle of car deformations under crash. It is shown that the proposed methodology can be used for applications exploring shape spaces and also that it has the potential of significantly improving existent methods for geodesic calculations in shape space.

9.2.1 Analysis on Shape Spaces

Shape spaces are infinite dimensional spaces. The analysis of shape spaces has been intensively studied from a theoretical point of view and there are also many applications in several areas such as medicine, biology and archeology among others. For developments in this area see [Bauer et al., 2011, Bauer et al., 2014, Michor, 2008, Srivastava et al., 2011] and the references therein.

A mathematical representation of the space of shapes considers at first a so called pre-shape space that contains those shapes under some constraints such as for curves being an immersion or embedding in $\mathbb{R}^2$, or a set of landmarks representing an object or the set of surfaces as immersions or embeddings in $\mathbb{R}^3$. In a second step, one identifies members of this pre-shape space that belong to the same transformation group or orbit, typically considered transformations are rotations, translations, scaling and re-parameterizations. If the pre-shape space is a Riemannian manifold, then the shape space (set of orbits) inherits, under certain conditions, the Riemannian structure of the pre-shape space, see chapter 5 for details. Shape spaces constructed in this way are quotient spaces of the pre-shape space with respect to a transformation group.

Critical factors that enable the analysis of shape spaces are the representation type (parametrized curve, mesh, parametrized surface, etc) and the definition of a Riemannian metric. The selection of a metric affects also the computing efficiency and analysis interpret-ability.

We propose, based on our results on invariant operators, the use of an equivalent representation of elements of shape space that under certain conditions allows for an efficient calculation of geodesic paths (along which a shape gets transformed into a new target shape) in this space.

The calculation of geodesics has received a lot of attention in the last years, see [Bauer et al., 2014, Srivastava et al., 2011] and references therein. For quotient spaces of shapes, constructing geodesics paths is known to be a very time consuming task. Approaches trying to speed up those calculations have been done recently, in particular see [Srivastava et al., 2011].

Firstly, our contributions consist in describing shapes using a basis representation along group actions or orbits, secondly based on this representation a new method we call, Orbit Path is described that allows the rapid evaluation of geodesics in the orbit space and finally we will show applications of this methodology for the case of plane curves and for simulations of time dependent deformations of cars under crash.

Section 9.2.2 reviews the standard mathematical setting for analyzing the space of plane shapes, following with a description of the invariant operator used for the case of plane curves. Next we will describe the general accepted framework for the construction of geodesic paths for one specific implementation. It follows a description of our proposed methodology and its evaluation and then two types of applications are discussed. Finally we will provide an overview of open research questions.

9.2.2 Shape Space as Quotient Space

We will give an overview of the analysis of plane shape spaces in the square root velocity framework (SRV) as described in [Srivastava et al., 2011], where also full details about the evaluation of geodesic paths can be consulted.

The abstract framework for shape spaces primarily considers a so called pre-shape space. In the case of planar curves this is the space of all immersions $Imm(D, \mathbb{R}^2)$ (notice that this includes embeddings since all embeddings are immersions) of a parametrized curve β in $\mathbb{R}^2$ with domain D. This is a large space that contains immersed closed and open curves. Typically such a pre-shape space already removes translation and scale variability. Other useful transformations are also included, such as the square root velocity representation defined by $q : D \to \mathbb{R}^2$, $q(t) = \dot{\beta} / \sqrt{\|\beta\|}$ (see [Srivastava et al., 2011] for details).

The pre-shape space of closed curves can be defined as

$$C^c = \left\{ q \in L^2(S^1, \mathbb{R}^2) \mid \int_{S^1} \|q(t)\|^2 dt = 1, \int_{S^1} q(t)\|q(t)\|dt = 0 \right\},$$

where the first condition rescales the shape to unit length and the second condition reflects a closure condition (start points are the same as end points) also in general for closed curves one speaks about immersions of the circle S^1 into $\mathbb{R}^2$.

To give a Riemannian structure to those shape spaces, one has to deal with its tangent spaces. For the specific case of closed curves this has been shown to be given by

$$T_q(C^c) = \left\{ v \in L^2(S^1, \mathbb{R}^2) \mid < v, w >= 0, \ \forall w \in N_q(C^c) \right\},$$

where $N_q(C^c)$ is orthogonal to $T_q(C^c)$.

Assuming that a transformation group G acts over C_c by isometries, a quotient space can be build that inherits a Riemannian metric from C^c. The orbit of $q \in C^c$ is given by

$$[q] = \left\{ (q \circ \gamma) \sqrt{\dot{\gamma}} \mid \quad \gamma \in G \right\}.$$

Under these conditions an orbit is associated with the same shape (up to isometries). To compare shapes invariants to a transformation γ a comparison between orbits $[q]$ must be done and for that, a metric in the space of orbits $S = \{[q] \mid q \in C^c\}$ needs to be be defined.

To find such a metric, one starts considering the space $L^2(S^1, \mathbb{R}^2)$, where the space of closed curves C^c is a subset of it. The standard metric in $L^2(S^1, \mathbb{R}^2)$ enables the construction of a Riemannian structure in C^c, using it one can define geodesic paths between elements of C^c. Given a path $\alpha : [0, 1] \to C^c$ such that $\alpha(0) = q_0, \alpha(1) = q_1$, its length is defined by

$$L[\alpha] = \int_0^1 < \frac{d\alpha}{d\tau}, \frac{d\alpha}{d\tau} >^{1/2} d\tau,$$

a minimizing geodesic is obtained by using the infimum over all paths. The length of this geodesic is then given by

$$d_c(q_o, q_1) = \inf_{\{\alpha:[0,1]\to C_c \mid \alpha(0)=q_o,\, \alpha(1)=q_1\}} L[\alpha] \tag{9.2.1}$$

A geodesic in C^c is a critical point of

$$E(\alpha) \equiv \frac{1}{2}\int_0^1 < \frac{d\alpha}{d\tau}, \frac{d\alpha}{d\tau} > d\tau. \tag{9.2.2}$$

A geodesic distance in the quotient space or space of orbits $S = \{[q] \mid q \in C^c\}$ can be shown to correspond to those geodesics in C_c that satify (see section 6.2 for details),

$$d_s([q_o],[q_1]) = \inf_{\gamma\in\Gamma} d_c(q_0, (q_1 \circ \gamma)\sqrt{\dot{\gamma}}). \tag{9.2.3}$$

We will now review the concrete steps for the evaluation of the geodesic distance.

9.2.3 Geodesics in the Shape Space of Planar Shapes

Interesting for applications in the analysis of planar shapes is how to calculate geodesic paths. That is given two plane curves β_0, β_1, how to find a geodesic path between them in the space of closed curves C_c. Also notice that one would consider shapes that differ only by translations, rotations or re-parameterizations, as being the same shape. This involves exactly finding geodesics on the quotient space that have been obtained by taking the quotient with respect to rotations, translations or re-parameterizations.

Evaluating a geodesic path in the quotient space involves two main steps: first a geodesic path is found in the space of immersions $Imm(D, \mathbb{R}^2)$ using the Path-Straightening algorithm in order to solve the minimization problem (9.2.1). Secondly an aligning step is solved over a specific transformation, this involves a second minimization using (9.2.3).

The Path-Straightening method as described in [Srivastava et al., 2011] is used. The idea of the method is to set an arbitrary path in shape space and straighten it in several iterations in order to find a minimizing geodesic. We only list the steps in the algorithm (see algorithm 6). For details of the algorithm we refer to [Srivastava et al., 2011].

Finally to obtain geodesics in the quotient space, the joint optimization task defined in equation (9.2.3) must be solved. This task is specific to the transformation group used, for examples of the approach for rotations and re-parameterizations see [Srivastava et al., 2011].

Up to here we have described the classical approach for the analysis in the space of plane curves. Now we will take our perspective of defining an operator for a set of simulations (or set of planar curves). Is it possible to define a common operator for a set of planar shapes? What type of information with respect to the geodesic paths is recovered in the orbit representation? The next sections try to give answers to

```
Algorithmus 6 : Calculation of geodesic path in $C_c$
    Input : two curves $\beta_0$ and $\beta_1$ in $C^c$
    Output : a geodesic path joining $\beta_0$ and $\beta_1$ given as a series of $m$ intermediate shapes
             $\beta_i, i \dots m$
    Compute the representations $q_o$ and $q_1$ in $C_c$
    Initialize a path $\alpha$ between $q_o$ and $q_1$ in the space of open curves $C^0$
    Project it back into $C^c$
    Compute the velocity vector field $d\alpha/d\tau$ along the path
    Compute the covariant integral of the velocity vector field
    Compute the backward parallel transport
    Compute the full gradient vector field of the energy $E$ given in (9.2.2) with respect to $\alpha$
    Update the path and check if the minimum has been reached otherwise repeat from step 4
```

these questions.

9.2.4 Orbits Paths as a Method for the Evaluation of Paths in Shape Space

As explained in chapter 6, the symmetries of a quotient manifold with respect to a group of isometric transformations can be expressed by means of the eigenfunctions of a positive definitive operator and the corresponding spectral coefficients (or projection coefficients). Transformations are represented along some specific components of an orthogonal decomposition via the spectral coefficients. These coefficients have been shown to provide low dimensional coordinates for simulation data sets. In addition, it has also been shown that paths in such a low dimensional simulation space can be found following such coordinates (see section 7.3 and section 9.1.1). Now we would like to treat planar shapes from this perspective, we will see that this provides interesting results.

We propose a method for evaluating paths in the shape space of plane curves;

- let S be a set of planar shapes,
- let **U** be a set of eigenvectors obtained from a path-length invariant operator constructed using algorithm 4,
- let C be the set of projection coefficients up to order p,

instead of using the shape in its full high dimensional representation, we use "orbits" made of spectral coordinates. For a set of similar shapes, the variation between them can be approximately represented by the changes in the spectral coefficients. The orbits reflects the action of the transformation group on the first few components in an orthogonal decomposition. Taking only the orbits up to a specific component p, a low dimensional representation of a set of shapes is obtained along which (shortest) paths can be calculated.

We show in the next example that surprisingly, orbit paths have the ability to approximate geodesic paths very well, at least for the examples we have shown. To obtain the orbit paths shown, we have taken the spectral coordinates of the initial shape and the ones of the final shape. Between those values of the orbits one can subdivide the orbit ranges in equally spaced sets and reconstruct the obtained shapes at the subdivision points. The procedure is described using algorithm 7

Algorithmus 7 : Calculation of orbit path between source and target shape

```
Input : A source $\mathbf{x_s}$ shape and a target shape $\mathbf{x_t}$, each given as $N_h$ points $\in \mathbb{R}^2$, $m$ the
        number of desired shapes in the orbit path
Output : An orbit path of $P_a$ shapes between source and target shapes, given as points
         $\left\{x_{i,j,k}\right\}_{i=1,2,j=1,...,N_h,k=1,...,m}$ where $\mathbf{x}_{k=1} = \mathbf{x_s}$ and $\mathbf{x}_{k=m} = \mathbf{x_t}$
Calculate an invariant operator and its eigenvectors $\mathbf{U}$ for the source shape $x_s$ using
algorithm 4
$\mathbf{Coef_s} = \mathbf{U} \times x_s$    ▹ spectral coefficients for $x_s$ as matrix of size $(N_h, 2)$

$\mathbf{Coef_t} = \mathbf{U} \times x_t$    ▹ spectral coefficients for $x_t$ as matrix of size $(N_h, 2)$

$\mathbf{Stepx} = [\mathbf{Coef_s} - \mathbf{Coef_t}]_{i,j}/m \quad i = 1, \ldots, N_h, j = 1$    ▹ Subdivision of the orbit
range in $x$

$\mathbf{Stepy} = [\mathbf{Coef_s} - \mathbf{Coef_t}]_{i,j}/m \quad i = 1, \ldots, N_h, j = 2$    ▹ Subdivision of the orbit
range in $y$

foreach $k \in \{1, \ldots, m\}$ do    ▹ reconstruct the orbit path
    $[\mathbf{Coefx}]_k = [\mathbf{Coef_s}]_{j=1} + [\mathbf{Stepx}] \times k$
    $[\mathbf{Coefy}]_k = [\mathbf{Coef_s}]_{j=2} + [\mathbf{Stepy}] \times k$
    $\mathbf{x}_{i=1,k} = [\mathbf{Coefx}]_k \times \mathbf{U}$
    $\mathbf{x}_{i=2,k} = [\mathbf{Coefy}]_k \times \mathbf{U}$
end
}
```

For comparison the geodesic paths in the examples are evaluated using the approach described in [Srivastava et al., 2011] which actually uses projections from the pre-shape space into the quotient space. That is, it computes horizontal geodesics as described in section 9.2.3.

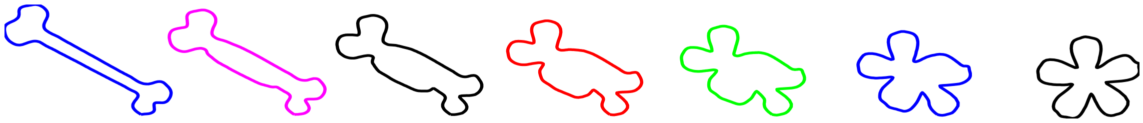


Figure 9.4: orbit path

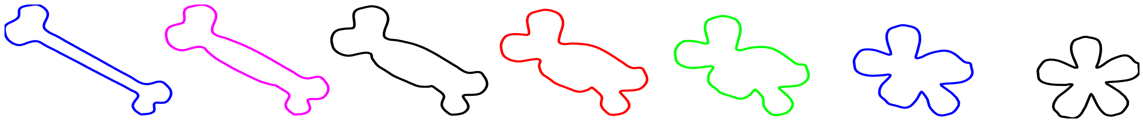


Figure 9.5: geodesic path using elastic metric

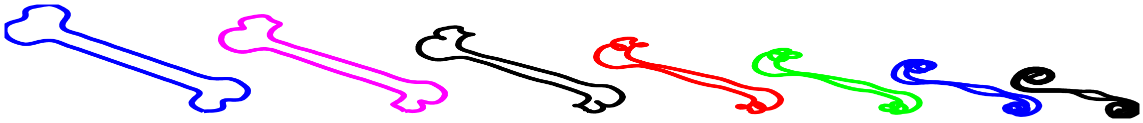


Figure 9.6: orbit path

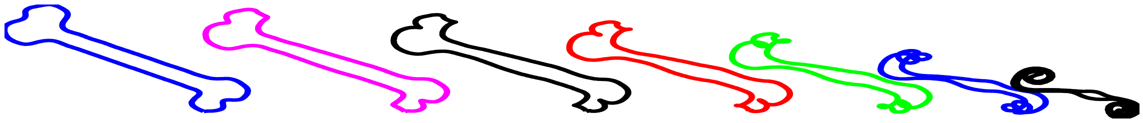


Figure 9.7: geodesic path using elastic metric

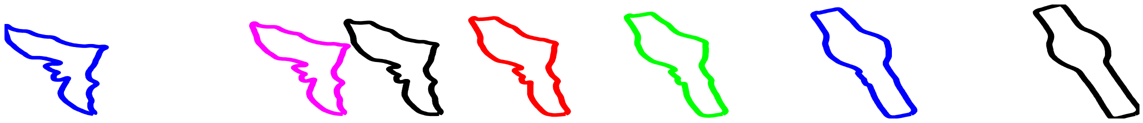


Figure 9.8: orbit path

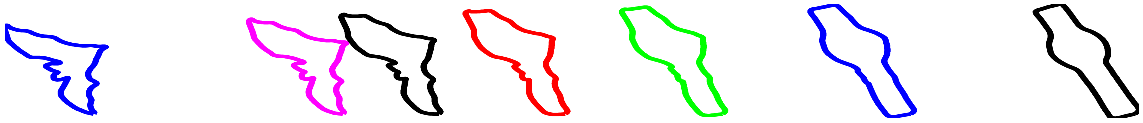


Figure 9.9: geodesic path using elastic metric

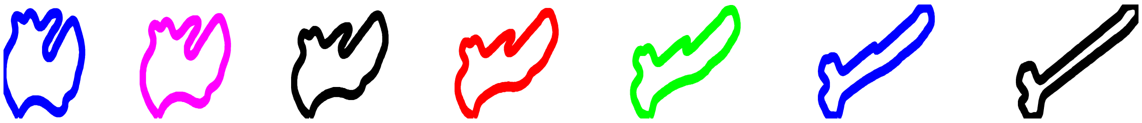


Figure 9.10: orbit path

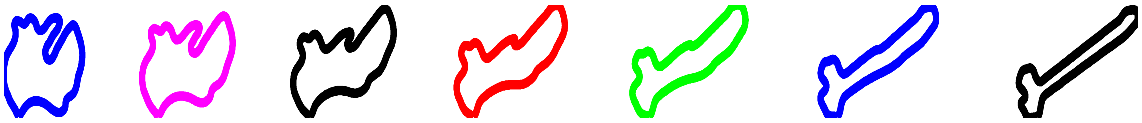


Figure 9.11: geodesic path using elastic metric

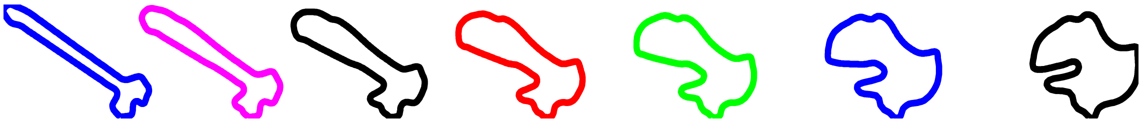


Figure 9.12: orbit path

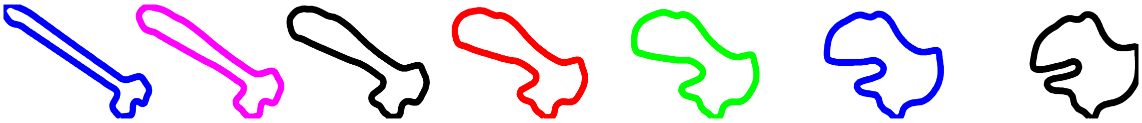


Figure 9.13: geodesic path using elastic metric

The results obtained for the orbit paths in figures 9.4, 9.6, 9.8, 9.10 and 9.12 can be compared with the ones for the geodesic paths obtained in figures 9.5, 9.7, 9.9, 9.11 and 9.13. The results are certainly interesting, at least from two points of view,

- determining shortest paths from given paths in such an orbit space approximately allows you to find geodesic paths. This means that for the examples we have shown, the geodesic path between two shapes is approximately linear for the representation of these plane shapes using an invariant basis.
- following a path in orbit space actually shows geometry preserving transitions between two shapes.

An additional important observation for plane shapes is that, as the examples show, the use of a unique operator such as for example, one based on path length for curves, allows to obtain a basis that can actually be used for arbitrary plane shapes. This aspect needs to be studied further, in order to understand under which conditions a violation of distance preservation is allowed. For now let us give an example where one can see this behavior more clearly.

Using a sequence of hand images where one of the fingers becomes gradually reduced, (see figure 9.14), the procedure in Algorithm 7 is used. We then take the image of the first hand as reference and calculate an operator and the eigenvectors using the path length along this shape. The projection coefficients corresponding to the first image and the final image (where the finger is reduced to its minimum size) are used to recover the complete path in shape space using algorithm 7.

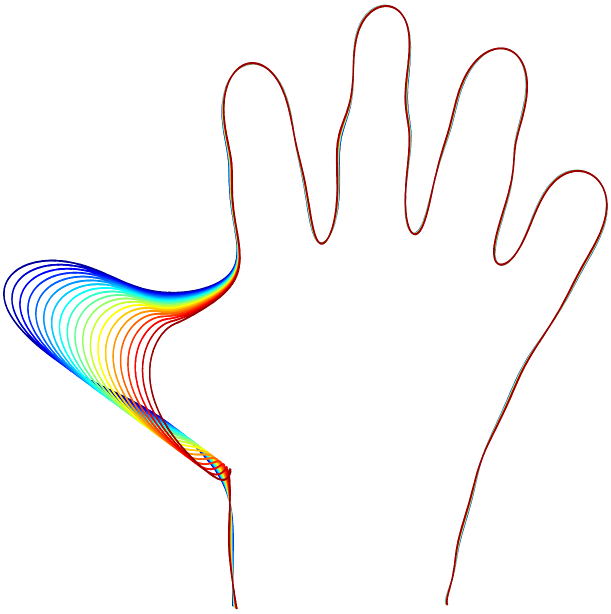


Figure 9.14: Representation of a total of 20 hand shapes with reduction in one of the fingers. These have been obtained by using a linear combination of 50 coefficients of an orthogonal decomposition. The first 5 coefficients corresponds to one specific hand shape, the rest (45) are taken from a reference shape and are kept the same for all 20 hand shapes. Each hand curve has a different color. Here we can observe that the variability in the shape of the hands is reflected already in the first 5 coefficients of the orthogonal decomposition

As before we will use an orbit path as an approximation of the geodesic path. Figure 9.15 shows the approximated geodesic path between the initial hand shape and the final one (with the maximal finger reduction). The reconstructed path provides a geometry preserving transformation of the hand. For this case the decay of the spectral coefficients starts at around the component fifth. We have used the first 5 components for representing the variability of the shapes. For shape reconstruction we have used instead 50 (the last 45 coefficients are kept constant equal to the ones on the initial hand). The total number of points for representing each of the original test hands were 704 points.

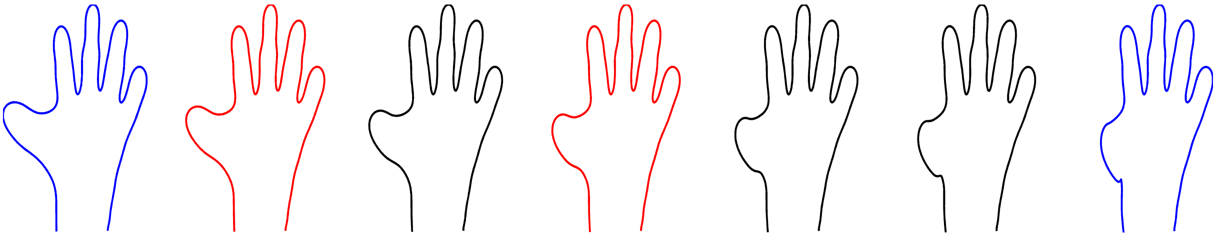


Figure 9.15: Recovered path in orbit space

We would like to mention another observation about the type of operator used. The hand shape changes are very localized and even if they are not locally distance preserving, the rest of the hand shape is preserved so one might just ask whether the use of an operator based on Euclidean distance

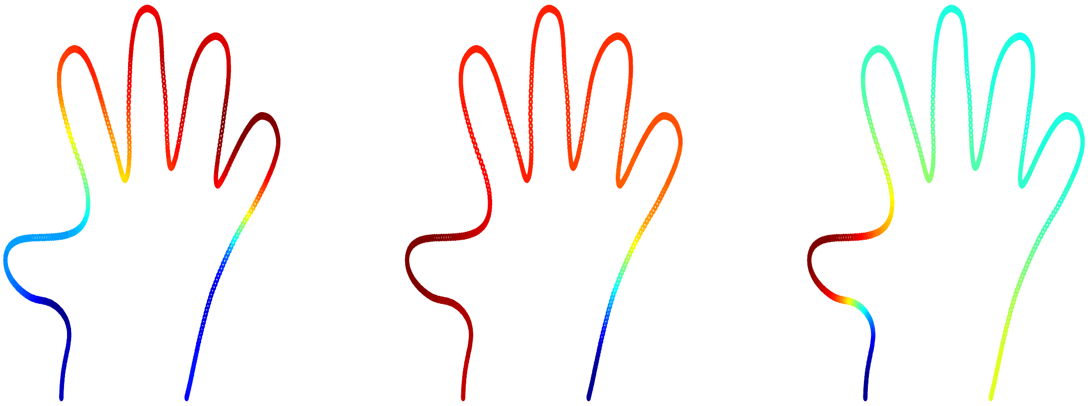


Figure 9.16: First eigenvectors of the Laplacian using Euclidean distances. The colors correspond to the values of the second, third and fourth eigenvectors respectively starting from the left.

could also be used in this case. That is, we can build the graph Laplacian as in Algorithm 4, but instead of using the path length, we just use the Euclidean distance to form the matrix K. Since there are no pose changes for most of the hand shape, we expect that the eigenvectors of this Laplacian are a good basis for all hand geometries. This can be verified by looking at the first eigenvectors for this operator (see figure 9.16).

The eigenvectors follow the form of the fingers, so that one can use such a basis for this case and obtain the same orbit path.

Summarizing, we can say that in plane space, there are several operators that can be constructed and the basis obtained from them can be used. The spectral coefficients corresponding to the first components are adequate to calculate a geodesic path between two shapes, for the examples used. In the next section we will show whether the orbit path approach can be applied for the more complicated shapes obtained from crash simulation.

9.2.5 Orbit Paths in Crash Simulation

The previous section has detailed a way to evaluate orbits in plane shape space. Simulations can actually be considered as a special type of different shapes obtained as a result of the solutions of ordinary or partial differential equations. The calculation of what we call an orbit path, can also be used in the more difficult case of simulations.

We would like to apply it to an example of a strong deformation of a beam under compression. The beam model is solved using LSDYNA and 17 time steps are calculated. The initial geometry (without deformation) has been used for calculating the discrete approximation of the Laplace-Beltrami operator as described in algorithm 2. The initial deformation and the deformations corresponding to the last time step, can be projected along the eigen-basis. Figure 9.17 shows fewer time steps of the simulation obtained from LSDYNA.

The orbit path approach can also be used for this case. Algorithm 7 can be used replacing the invariant operator by the discrete Laplace-Beltrami operator and using one more set of spectral coefficients for the direction z. We assume that no intermediate shapes for the time steps are available between the original

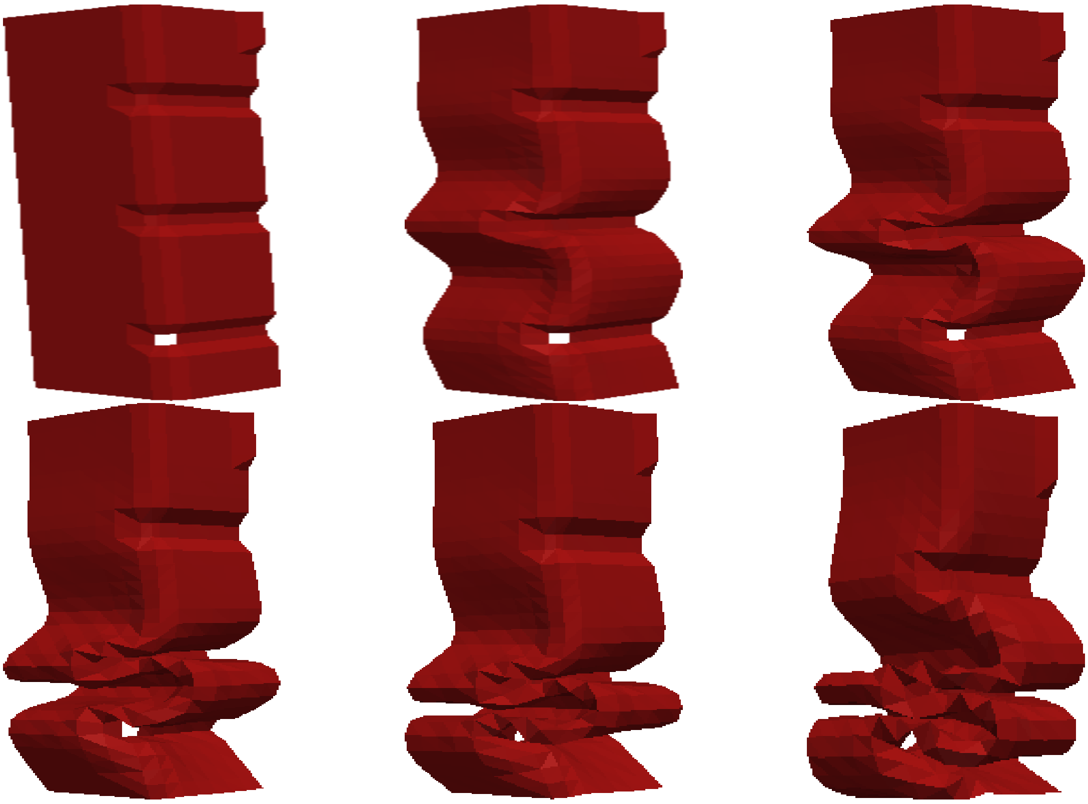


Figure 9.17: Beam deformation obtained from an LSDYNA crash simulation.

and final deformed shape and so try to recover the deformation trajectory between them using the orbit coordinates (the spectral coefficients) from the initial and final deformation states. We will evaluate in this case a linear orbit path that can be re-sampled at an arbitrary number of points in shape space. See figure 9.18 for the reconstructed orbit path obtained in this way.

The deformations in figure 9.18 and figure 9.17 are the same for the last and end states following the order of the rows and from left to right. The state 2 shows approximately the same deformation. Starting with state 3, the differences are much greater as shown in the folding of the structure.

Even though there are differences in the corresponding deformations compared to the LSDYNA calculations, we emphasize that they were obtained without using any intermediate LSDYNA calculation. We have linearly interpolated in the orbit space and for that we have used only the initial and the end deformations. An improvement of this approximation can be achieved if intermediate simulations are available. Then one can evaluate an approximation of a geodesic distance using intermediate orbit points.

As it can be seen, the orbit path approach has the potential to enable the interpolation of deformations at time steps not available in the original simulation. There is a keen interest in gaining some more information for engineering applications on a deformation process where only the initial and final stages are available. That is, one is able to measure only those stages and engineers would like to know, what the intermediate deformed shapes look like to achieved this final stage. We have shown that a first approximation can indeed be obtained using the orbit path estimation. A few samples around final and

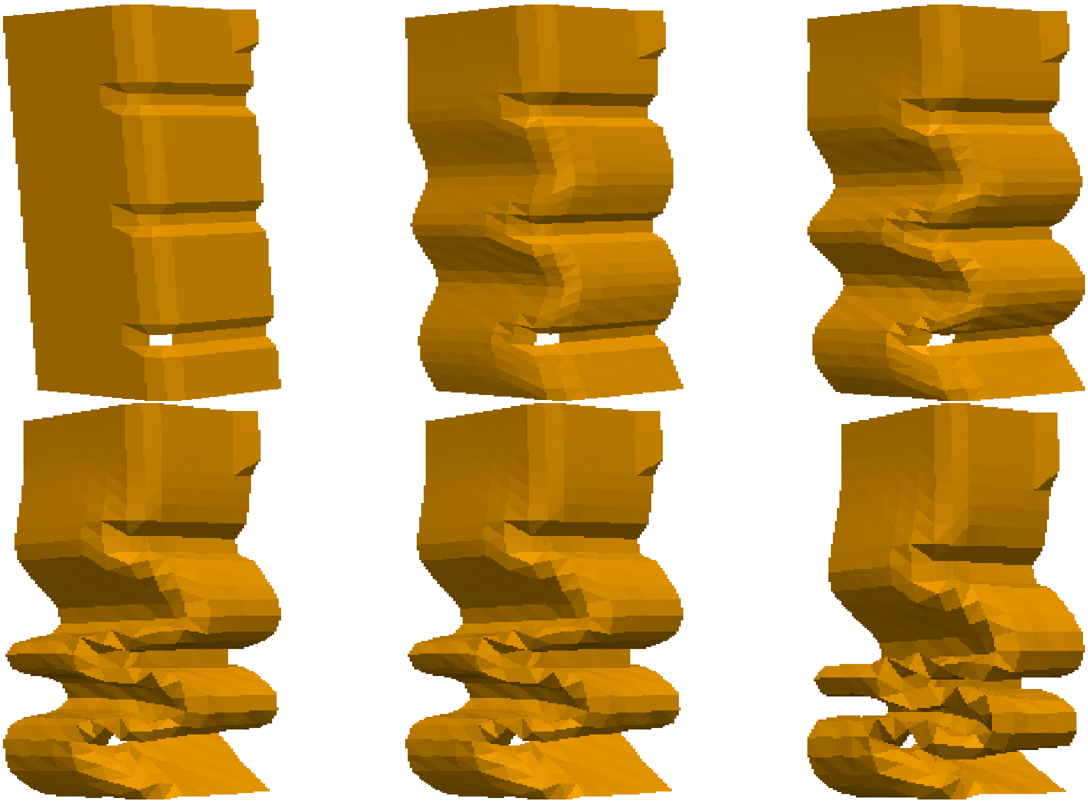


Figure 9.18: Beam deformation obtained from evaluation of the orbit path in the range between the initial and the final deformations.

initial shapes will certainly improve a lot the approximation compared to a LSDYNA simulation.

There are other approaches that could be used for finding a geodesic path in shape space which unfortunately initially requires the solution of the geodesic equation. This involves the solution of a global optimization whose size is of the order of the number of nodes (see [Kilian et al., 2007]). We also notice that even this task can be greatly simplified using our approach, that is, one solves the geodesic equation but uses instead orbit coordinates. Because of the spectral decay of the orbits for the first coefficients, a lower dimensional problem have to be solved instead.

CHAPTER 10

Analysis of Simulation Bundles

In this chapter we will present a series of industrial applications as examples for the type of engineering analysis that can be done using the methods developed in this thesis.

We will first describe the application of diffusion maps for metal forming, as a way to illustrate how this methodology of dimension reduction can be used for the analysis of simulations. For details about this and other applications of diffusion maps see [Iza Teran, 2013]. Next we will concentrate on response curves from vibration analysis (NVH). We will compare the results given in [Iza Teran, 2013] with the ones obtained using the new proposed invariant operator approach. Finally we will concentrate on an example in crash simulation taken from [Iza Teran, 2013] and we show how to extend the type of analysis that can done, on the basis of the new approach using the same data sets. A part of these examples have been presented in [Iza Teran and Garcke, 2016].

10.1 Analysis of Simulation Bundles using Diffusion Maps

The application of the methodology as described in section 4.6 is presented here for an application in metal forming. In this approach a similarity matrix is constructed using all available simulations and from it a set of eigenvalues and eigenvectors are obtained that approximate a low dimensional representation of all simulations. See section 4.3 for some details about the method. In this case in addition to the trivial eigenvalue by 1, only the first few eigenvalues are significant, correspondingly we use the second and third eigenvectors (ψ_1, ψ_2) as diffusion coordinates. As explained in section 4.4.1 simulation data can be considered as a high dimensional dynamical system. Using diffusion maps one can extract the slow variables of the system in such a way that the first non-trivial eigenvector corresponds to the first slow variable, the second to the next slow variable and so on. There could be more variables, nevertheless by projecting to a 2D embedding, we are at least able to see the first most dominant ones. Another aspect is the point of view of clustering and classification, the second eigenvector is very informative (spectral clustering). Organizing the simulations according to the order given by the second eigenvector shows a way to parameterize or cluster all the data. The clustering obtained in this way is already very useful in practice, since the engineer can automatically see which of the variables have a larger impact on his design.

10.1.1 Metal Forming

We will consider material parameter changes that produce variations in the plastic strain of a plate (a quarter disc). In the metal forming simulation, this is an interesting output variable that evaluates the permanent (inelastic) distortion of metals under applied stresses that strain the material beyond its elastic limit. The ability of metals to flow in a plastic manner without fracture is the fundamental basis for all metal forming processes.

The material law considers the effect of elastic-plastic hardening with a Swift-Law $\sigma = K \cdot (\varepsilon_0 + \varepsilon)^n$ as well as anisotropies, defined by several parameters r_0, r_{45}, r_{90}, r_m and Δr.

In order to obtain the simulation data for the analysis, the parameters K and r_0 were varied randomly up to 3 times the nominal value generating input data for a total of 80 simulations, generated using LSDYNA ® [LSTC, 2004]. This and other applications of diffusion maps are described in [Iza Teran, 2013].

From each simulation the plastic strain (extraction step) is extracted, so that a set of vectors $m = 80$ of dimension $N_h = 1780$ is obtained. The general procedure for nonlinear dimension reduction with diffusion maps described in algorithm 1 is used, building a similarity matrix with weights $w_{ij} = exp\left[-(\frac{\|x_i - x_j\|}{\epsilon})\right]$. The value of ϵ is chosen equal to the standard deviation of the data.

The result can be seen in figure 10.1, each point in the plot corresponds to one simulation, in addition the value of the corresponding parameter K is displayed as a color for each point, corresponding to each simulation.

In figure 10.1 by analyzing the information from this 2D embedding (exploration step), one can observe that the simulations have been categorized according to the intensity of the plastic strain. This can be verified for example by observing the corresponding strain distributions for a number of simulations on the left compared with the ones on the right of the 2D Embedding. Also observing the simulations toward the middle ($x = 0.05$), one can recognize points where the strain distributions show a more local distribution on the disc, whereas the ones on the right branch of points, shows a plastic strain distributed around the border of the disc.

An important additional observation in this 2D plot is that the input parameter (in this case the stiffness factor K) follows a very clear pattern as a function of the changes on the plastic strain for all simulations.

10.2 Applications Analysis Simulation Bundles using Invariance

In this section, the new approach presented in this thesis will be compared with diffusion maps. Several industrial examples are used to highlight the differences and usefulness of the new method.

10.2.1 Comparative Analysis of Vibration Response Curves

Standard Diffusion Maps

The first example is concerned with the analysis of noise, vibration & harshness. This example was first presented in [Iza Teran, 2013]. A car is considered as a structure made of several components (the doors, roof parts, console, lateral stiffeners, support structure, etc.). We will use an industrial example

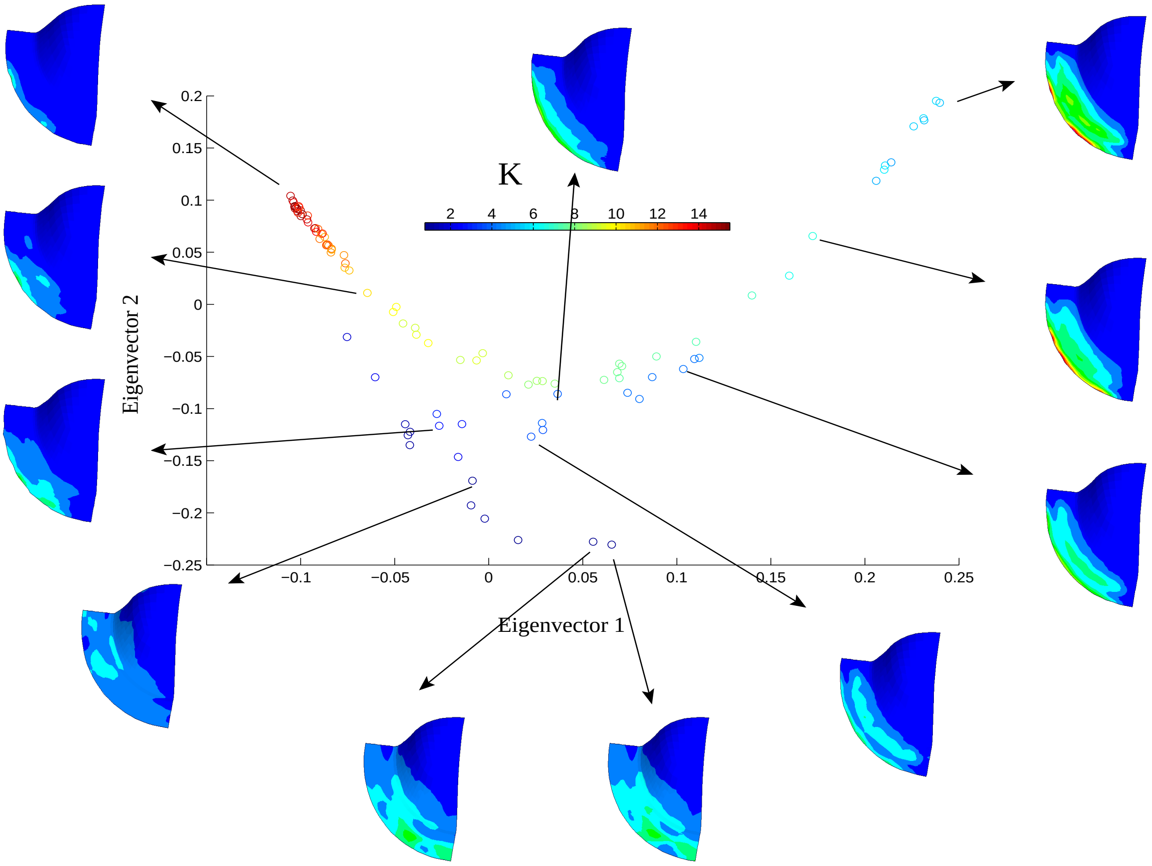
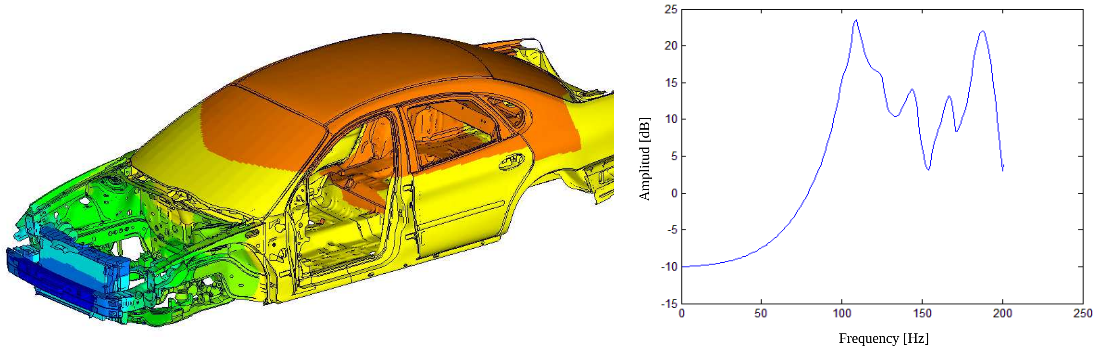


Figure 10.1: Metal forming embedding in 2D, the color insert represents the value of the parameter K

that contains around 1000 components and each of them is represented using a very fine mesh structure with different types of elements (shells, volumes, beams). For this discretized structure a dynamic simulation is performed that calculates the behavior under simulated road operating conditions. For the engineers it is important to simulate the response of the structure to specific excitations. This response was calculated with the dynamic simulation software NASTRAN ® .

For this study several model variations have been performed changing some input parameters such as thickness, material or geometry. The decision about an ideal configuration is normally a conflicting objective, subject to several design restrictions. For example a structural component should be, from the point of view of the dynamics, stiff but at the same time this implies an increase in weight.

The response of the structure is analyzed at several points of interest in the structure. Figure 10.2 (a) shows a typical vibration mode (amplified for visualization), the response (see figure 10.2 (b)) to a specific excitation at points of interest (for example at the console front) shows the engineer how well a design fulfills the dynamical requirements for comfort and vibration, and can be evaluated from these vibration modes.



a) Vibration mode (scaled to show deformations)

b) Response curve at the console

Figure 10.2: NVH-Analysis, vibration mode and response curve

In the EU project SIMDAT (IST-2004-511438), the AUDI-AG has completed a series of model variations, with changes in the thickness of some of the components, changes in the mass of the cooler or the introduction of small structural changes such as small stiffeners at the doors. A total of 39 models were simulated using NASTRAN ® software, each one of them representing a specific change.

The variable of interest here is the response of the structure at specific points. A set of 39 vectors of length ($N_h = 500$) is extracted from the NASTRAN ® simulations. The general procedure for nonlinear dimension reduction using diffusion maps (see Algorithm 1) is used. For this example we would also like to show the effect of using two different type of distances for building a similarity matrix using the so called kernel weight $w_{ij} = exp\left[-d(x_i, x_j)/\epsilon^2\right]$ in algorithm 1. The first kernel weight using the distance $d(x_i, x_j) = \sqrt{||x_i - x_j||^2}$ is called Laplacian kernel and the second kernel weight using the distance $d(x_i, x_j) = ||x_i - x_j||^2$ is called Gaussian kernel. For both cases, the second and third eigenvectors have been chosen for the embedding purposes. The result can be seen in figure 10.3 and figure 10.4, where in addition to the response curves, the change in some of the car models is superposed graphically. Each response curve has been given a number and this number is displayed in figure 10.3 and figure 10.4.

In [Iza Teran, 2013] only a Laplacian kernel was used, important observations can be done based on the results in figures 10.3, 10.4. Each point represents one response curve in the plot and one can visually verify that there is a specific shape of the response curve that can be associated to a specific car model parameter change for the mass, thickness, stiffness and for the introduction of ribs at the doors. The cluster of results for the response curve that are associated with changes in thickness is very dense. The response curve for this cluster is correspondingly very similar. It can be stated that modifications in the car design based on the thickness variations that have been done, does not have a strong effect on the shape of the response curve. The introduction of changes in the cooler mass or on the cooler stiffness has a strong influence on the shape of the response curve as well as changes in the mass distributions. The introduction of doors-stiffeners changes has also a significant influence. It can be stated that in general both types of kernels produce embeddings that separates specific outliers with one observable exception for the out-lier 12 (marked as red star in figure 10.3, 10.4). As it is shown in figure 10.4 the simulation 12 is not identified as an out-lier for the Gauss kernel whereas in figure 10.3 this simulation

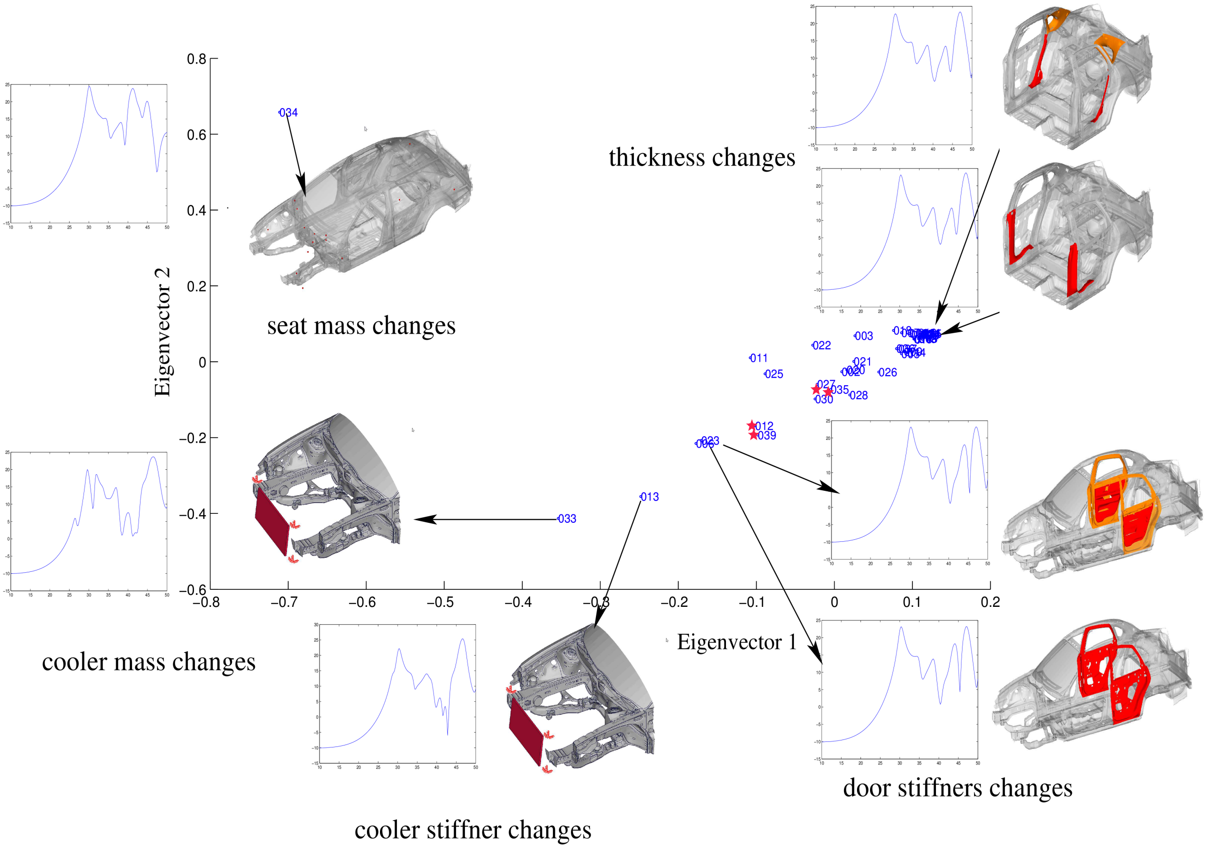


Figure 10.3: 2D embedding with diffusion maps using a Laplacian kernel showing a parameterization of vibration response curves obtained from 39 NVH simulations. Some geometrical or material parameter changes that were made in the input model are shown together with the corresponding simulation response curves

is clearly separated using the Laplacian kernel.

In the next section we will compare diffusion maps with the operator approach, for that we will use only the result with the Laplacian kernel. A systematic comparison of both kernels with respect to the operator approach will be the focus of a future publication.

The Operator Approach

For the invariant operator approach that has been proposed in this thesis, one needs to define an adequate operator that can be considered as invariant to the changes present in a simulation. For this case we observe that the response curves are 2D plane curves and therefore one can use the operator approach as described in Algorithm 4. Using the eigenvector basis obtained in this way, all the 39 curves can be projected into this basis. In Figure 10.5, the first 30 coefficients for the 39 NVH response curves are plotted, obtained after projection to a common basis.

We have observed that the first coefficients are responsible for the largest variations and the further one gets to the higher ones, then the more details are recovered from the curves.

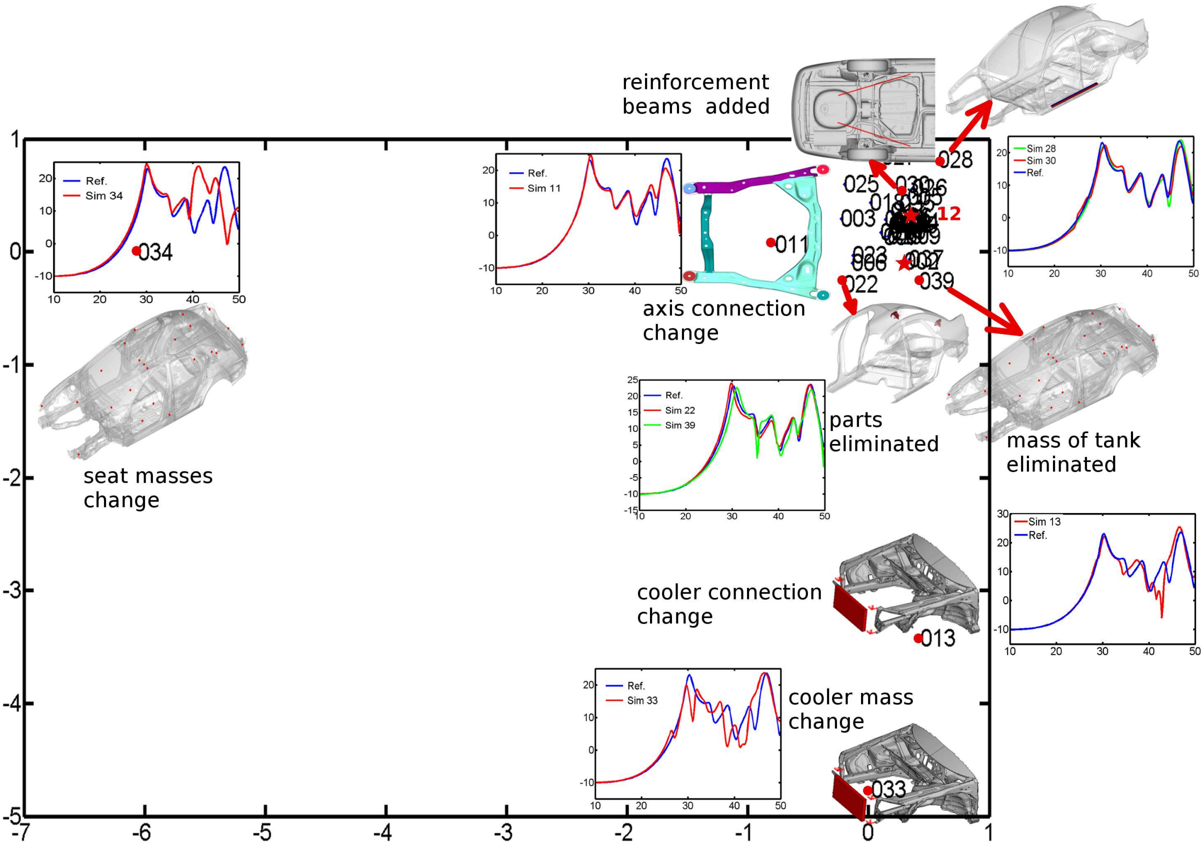


Figure 10.4: 2D embedding using a Gauss kernel showing a parameterization of vibration response curves obtained from 39 NVH simulations. Some geometrical or material parameter changes that were made in the input model are shown together with the corresponding simulation response curves

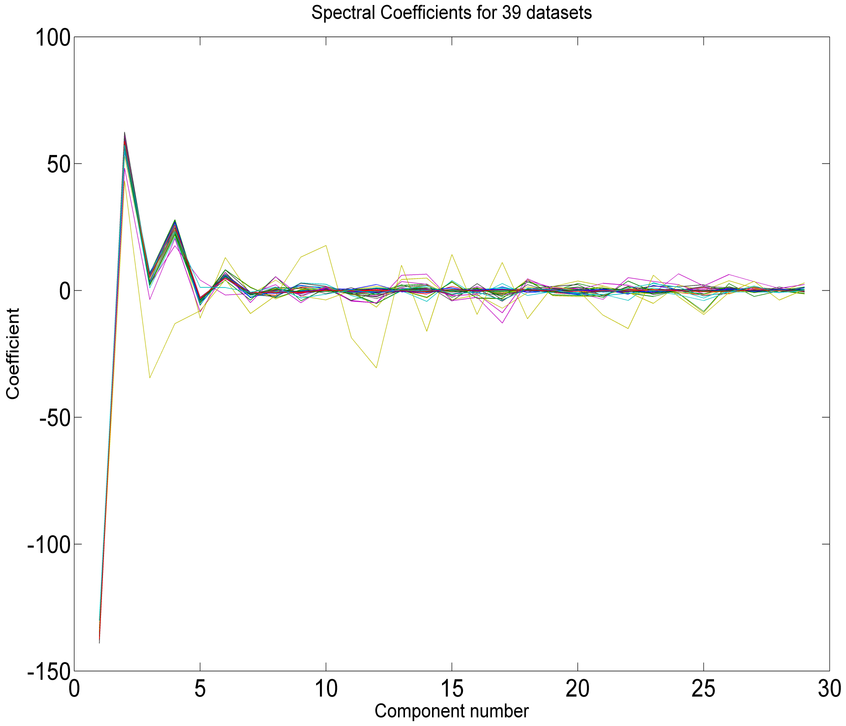


Figure 10.5: Spectral coefficients for 39 NVH response curves

To compare the operator approach with diffusion maps, we have chosen two scales with the largest variations from the first components. In this example we have chosen the third and fourth components. Notice that the plot of the spectral coefficients in Figure 10.5 suggests also to consider other components. In this context we mention that still a method has to be developed to decide automatically about the embedding dimensions. The variance of the spectral coefficients as mentioned in section 7.1 can be used for that purpose. Nevertheless the proposed method has a multi-scale behavior since at each component of the orthogonal decomposition, different variations of the response curves can be obtained. That is, the comparison between simulations turns out to be far more profuse in the new approach as it will be seen. The 2D-embedding shown in Figure 10.6 for the new approach is different to the one obtained with diffusion maps. We would like to mention specific differences with two examples in more detail.

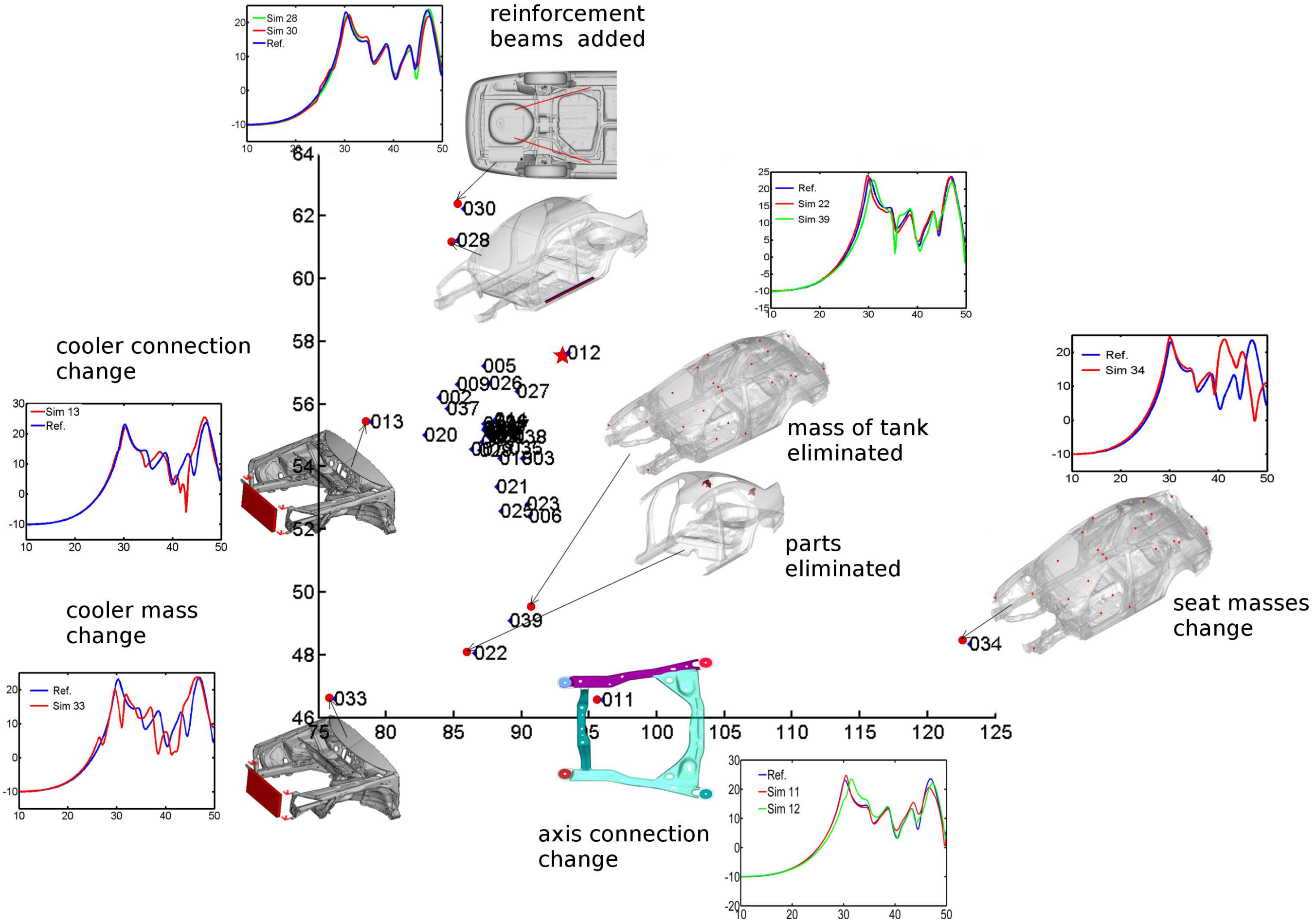


Figure 10.6: Embedding with coefficients for the third and fourth components, of an orthogonal decomposition using the new approach. Also shown are the response curves and the corresponding variation of the input data for the simulations for a few outliers

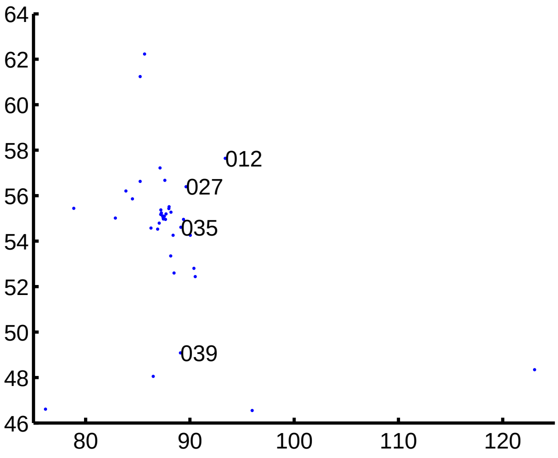


Figure 10.7: Embedding with locations of points corresponding to simulations 12, 39, 35 and 27

Figure 10.7 shows the points corresponding to simulations 12, 39, 35 and 27. As can be seen in figure 10.3 (see red stars), using diffusion maps for calculating a low dimensional embedding, those points are shown as very near. The new approach instead shows that they are different. As can be seen in 10.8 the curves corresponding to simulations 12 and 39 do differ at specific locations. Similarly for simulations 35 and 27, they are shown in diffusion maps as very near, but as Figure 10.9 shows, they differ locally at several positions.

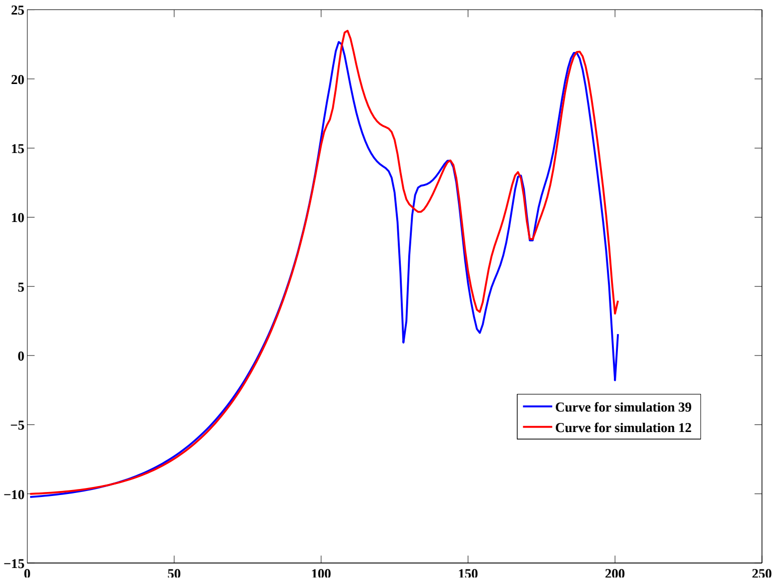


Figure 10.8: Curves for simulations 12 and 39

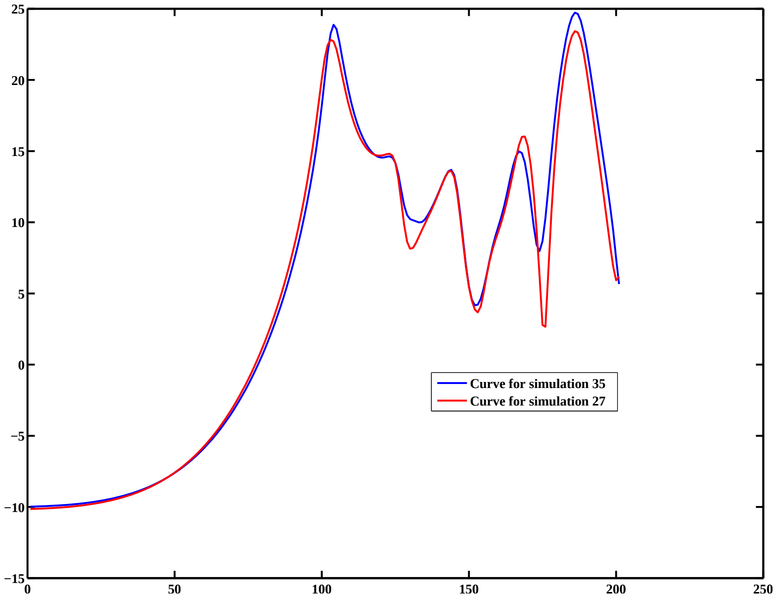


Figure 10.9: Curves for simulations 35 and 27

The example shows that the new approach can improve the clustering obtained from the curves. Specially, at least for this example, a much richer capability to differentiate between response curves is observable.

10.2.2 Analysis in Crash Simulation

Nowadays, finite element simulations in industry are used to study the physical behavior of products. We consider the example of car parts under deformations due to a crash. The development of new car models demands the creation of thousands of simulations where some material parameters or the geometry are varied. The analysis of the combined effect of those changes in the crash behavior is an open problem and designs are nowadays made, based on engineering judgment in a time consuming trial and error process.

We illustrate in this section the application of our proposed data analysis method to the case of crash simulations comparing it with diffusion maps where possible. Consider a frontal crash simulation of a Chevrolet C2500 pick-up truck, a model with around 60,000 nodes from the National Crash Analysis Center[1]. We use 126 simulations[2] of a vehicle frontal crash (see Figure 10.10) where 9 part thicknesses are varied randomly by up to ±30%. The variation in the thickness of these 9 parts results in different deformations of the original structure, see also [Bohn et al., 2013].

[1] http://www.ncac.gwu.edu/

[2] Computed with LS-DYNA http://www.lstc.com/products/ls-dyna

The Model and Standard Diffusion Maps

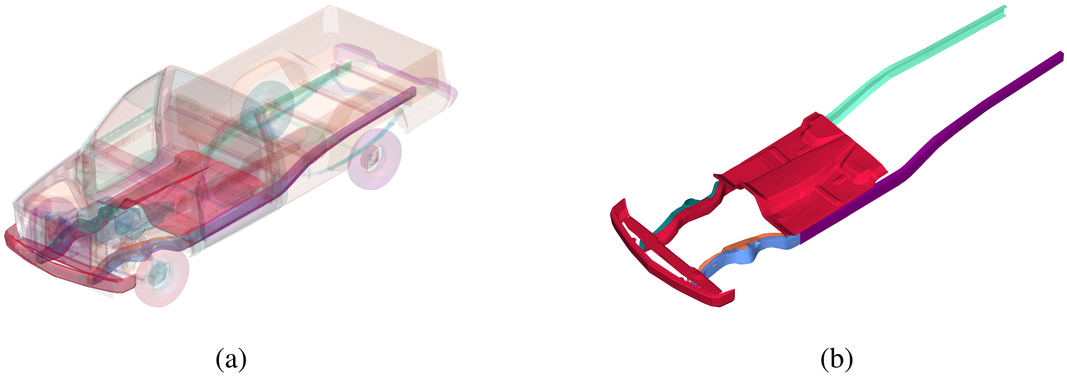

(a)
(b)

Figure 10.10: Truck under frontal crash (a) and parts that have thickness changes (b)

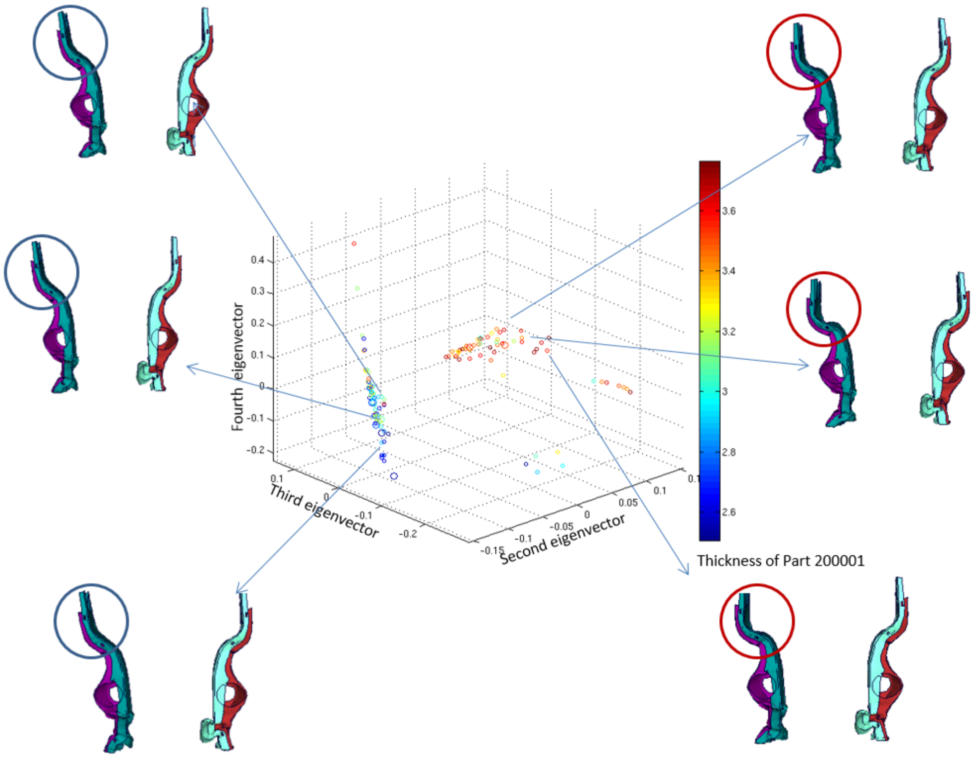


Figure 10.11: Embedding using the second, third and fourth eigenvectors. Each point corresponds to a crash deformation shape for the two structural beams shown in the inset. The color of the points in the plot corresponds to the thickness of one of the parts

In [Bohn et al., 2013] this data set was used for nonlinear dimensionality reduction. There, the norm of the difference of the deformations of two structural beams between the two time steps 6 and 7 was used as feature vectors. A total of $m = 116$ simulations were used with $f_k^i = \sqrt{\|x_k^i - y_k^i\|}$, where $x_k^i, y_k^i \in \mathbb{R}^3, k = 1, \ldots, N_h = 1714$, and x_k^i, y_k^i denote the position of grid point k of simulation i at the time step 6 and 7, respectively. An $m \times m$ matrix was obtained and the second and third and fourth eigenvectors where used as a lower-dimensional representation of all simulations using diffusion maps, see section 4.6 and [Bohn et al., 2013, Iza Teran, 2013] for details.

The distribution of the values for the deformation in each branch differs, a closer look at the frontal part of the car shows two types of bending (buckling mode) of a structural beam (see inserted deformed parts of figure 10.11). The color coding in the 3D plot corresponds to the value of the thickness of one of the structural parts and can, as it is seen, be used as indicator of the buckling behavior.

Now we would like to use the operator approach for analyzing this dataset. As for the clustering shown in figure 10.11, the same branching behavior is observed using orbit coordinates so that we concentrate in other type of results that are only possible with the new approach.

10.2.3 Operator based Approach

Based on the same data set, we have constructed an approximation of the Laplace-Beltrami operator using geodesic distances for one structural part. Algorithm 2 is used for the mesh at time $t = 0$. The eigenvectors of the operator can then be used to evaluate the scalar product $\alpha^i_j = \langle f^i, \psi_j \rangle$ to obtain the spectral coefficients for any mesh function.

The new approach enables several types of analysis in addition to the ones provided by diffusion maps. First, we will investigate the decomposition into coarse parts and details of a function on a mesh stemming from crash simulations.

Decompositions of Deformation and Mesh Associated Variables

The deformations are now the considered mesh functions f^i, one for each coordinate x, y and z, i.e. 3 per simulation at time step $t = 7$.

Using the same connectivity of the original mesh, all mesh deformations can be reconstructed using the Laplace-Beltrami basis for the x, y, and z directions. For that we use $f^i = \sum_{j=1}^{p} \alpha^i_j \psi_j$ for several values of p.

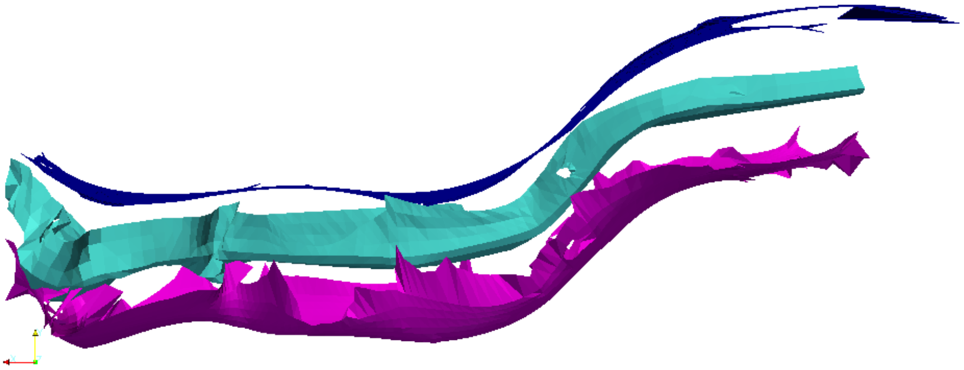


Figure 10.12: Reconstruction using the first p coefficients. $p = 20$ coefficients (upper), $p = 100$ (lower) and original data (middle).

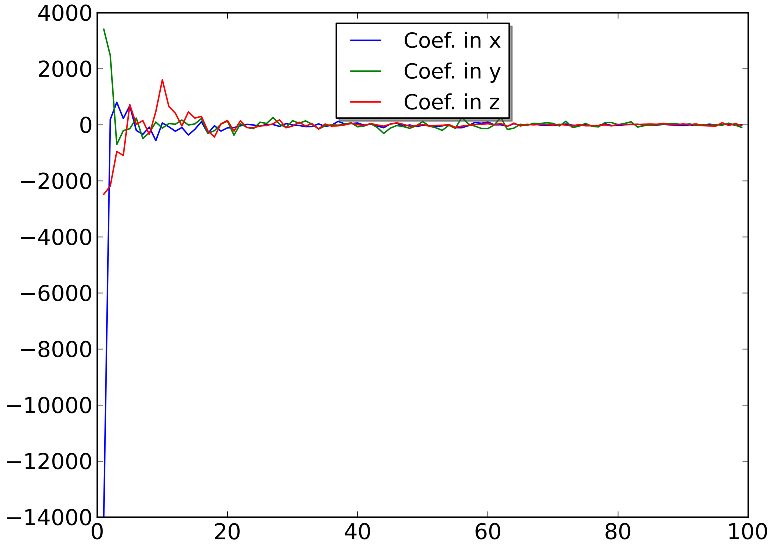


Figure 10.13: Magnitude of spectral coefficients for the deformations in the directions x, y and z at a fixed time step $t = 7$

From figure 10.12, it can be clearly seen that using $p = 20$ only a very coarse approximation of the part can be reconstructed. Adding more coefficients, e.g. $p = 100$, recovers more details of the part. It can also be clearly seen in figure 10.13 that most of the coefficients are small, with a few bigger ones. This is an essential feature which will be explored for a classification task later in this section.

So far, we have been able to show that the deformed geometry of a car part can be represented using an orthogonal basis of the eigenvectors of an operator. In addition, any function $f \in C^k(M)$ on the mesh can be represented by a linear combination of the eigenvectors. This implies that also, other variables associated to each node can be represented using the same orthogonal basis for example: nodal strains, temperatures, velocities, and so on. To demonstrate it, we will use as nodal variable the norm of the difference of the deformations between two time steps (6 and 7) of the car crash simulation. We use the same basis obtained from the Laplace-Beltrami operator evaluated from the geometry at time $t = 0$. The norm of the difference between the deformations at time steps 6 and 7 is a mesh function with values at the nodes of the mesh. This quantity is projected to the basis and a reconstruction of it using different values of p can be seen in figure 10.15. As it can be observed in figure 10.15, the reconstructed quantity shows almost no difference with respect to the original one (see figure 10.14). The same color scale is used for figure 10.14 and figure 10.15.

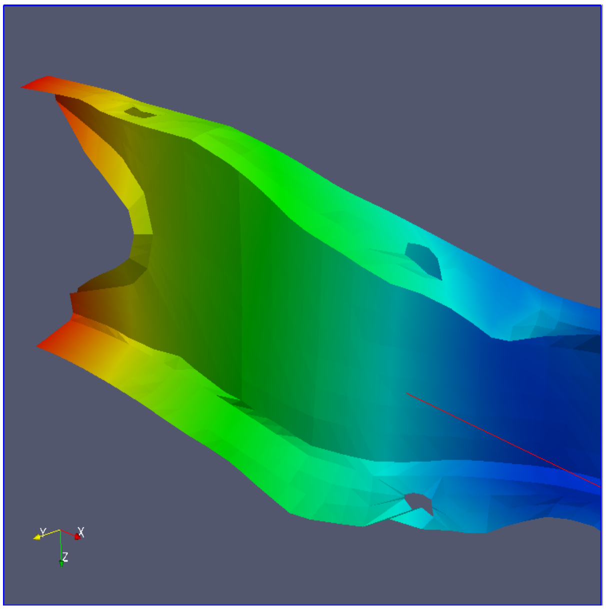
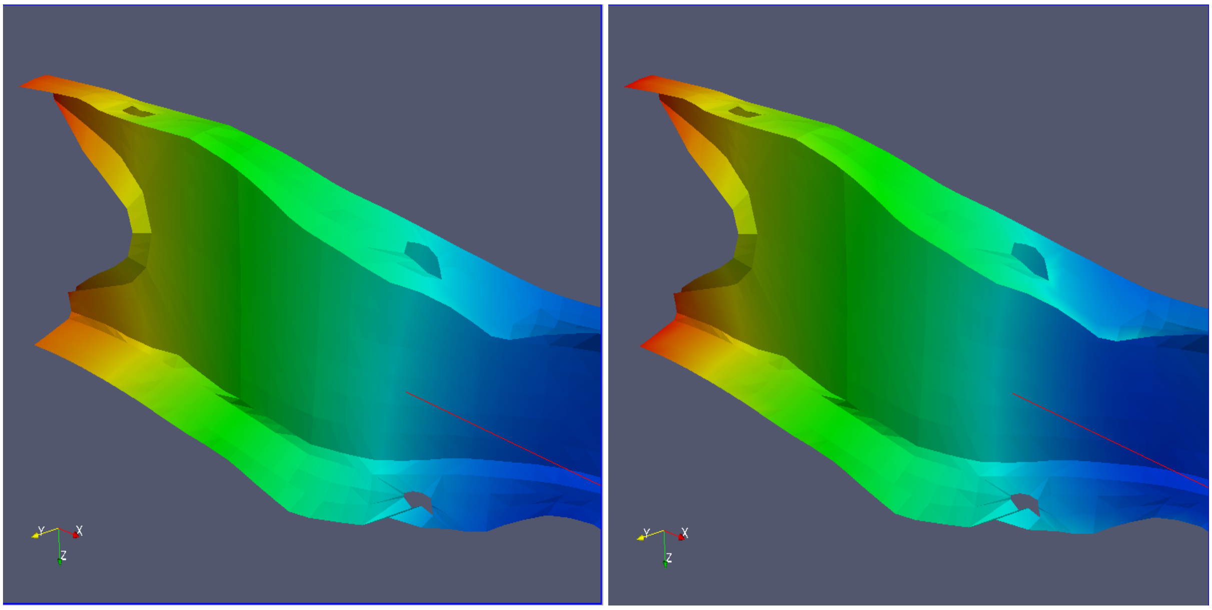


Figure 10.14: Difference of the deformations between time steps 6 and 7 in a crash simulation. The color on the shape represents the magnitude of the norm of the difference between the deformations at the specified time steps



(a) reconstruction with $p = 20$

(b) reconstruction with $p = 200$

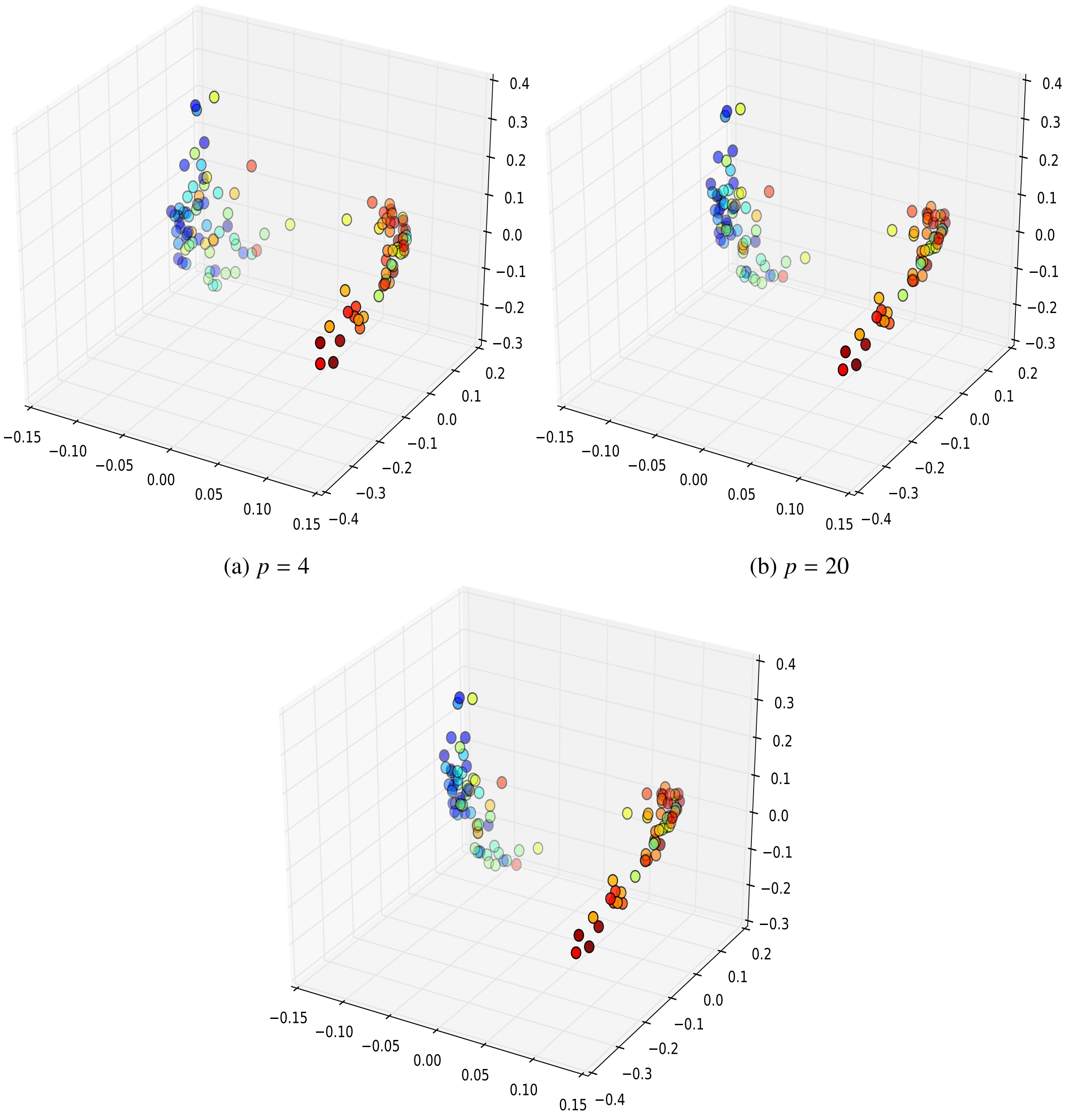
Figure 10.15: Reconstruction of the deformations between two time steps using p coefficients, the color on the shape represents the magnitude of the norm of the difference between the deformations at time steps 6 and 7

Data Analysis of Deformations in Crash Simulations

We will also investigate the use of the spectral coefficients as input to the nonlinear dimensionality reduction procedure. We will project as before all m feature vectors f^i corresponding to the norm of the deformations difference along the eigenvectors of the approximation of the Laplace-Beltrami operator. The result is a set of m coefficients, each of size N_h, where N_h is the number of nodes of the deformed beam.

The coefficients decay very fast, i.e. similar to figure 10.13, the energy of the signal is concentrated in very few coefficients and one can equivalently use these coefficients instead of the feature vectors f^i (our raw data). Due to the decay, we just would like to keep the most significant ones and compare the embedding with diffusion maps using the first $p = 4$ and $p = 20$ coefficients with the embedding obtained using the raw data.

Figure 10.16 shows a comparison of these embeddings, where each point corresponds to the embedding of one feature vector f^i. The color of the points is given by the value of one of the varied part thicknesses. It shows that this thickness has a high correlation with the resulting clustering of the deformation results and therefore graphically verifies the differences in the classification due to the eigenvectors. It is interesting to see that with only m vector of coefficients of size $p = 4$, the structure of the embedding is almost the same and can be used for classification instead of using the m raw feature vectors information of size $N_h = 1714$. This has a number of implications, but the important one is that the embedding is actually completely dominated by the first few components in the orthogonal decomposition, which correspond to the coarse variations.



(a) $p = 4$

(b) $p = 20$

(c) raw data

Figure 10.16: Comparison of diffusion maps embeddings for the car crash data using different number of projection coefficients to the Laplace-Beltrami basis as feature vectors. The color of the points corresponds to the value of the thickness of one part which is varied for each simulation

10.2.4 The Pre-image Problem in Crash-Simulation

The use of dimensionality reduction in applications is characterised through the interplay between the so called restriction and extension phases. This has already being studied in several applications such as in image processing and earth structure classification, see [Thorstensen et al., 2011, Kushnir et al., 2012]. We will elaborate on this for a particular application in crash simulation.

Given a training set of data in a high dimensional space $\mathbb{R}^{N_h}$, dimensionality reduction methods extract a low dimensional representation in $\mathbb{R}^p, p \ll N_h$ in such a way that each of the training data can be identified with an element in this new space, this is sometimes called the restriction phase. Such a mapping can be done in several ways and generally one tries to preserve some property of the high dimensional space such as the geometric distance or the topology (see [Lee and Verleysen, 2007] for a detailed description). The structure of the embedding should enable you to find the corresponding position in the lower dimensional space for any new test data point in the higher dimensional space.

Ideally, after obtaining this parameterization, one would like to obtain a high dimensional approximation that corresponds to a specific point in the low dimensional representation not in the training set. This is very important for applications in image segmentation and denoising. Given some embedding coordinates, this so called extension phase now enables the reconstruction of the corresponding high dimensional data set in $\mathbb{R}^{N_h}$. This is an ill posed problem since the same embedding coordinates could correspond to different high dimensional extensions. How to find them is known as the pre-image problem and has been extensively studied and is still an open problem, see e.g. [Kushnir et al., 2012].

The use of the proposed operator approach introduces a transformation with a fundamental simplification, the pre-image now gets parameterized with very few coefficients that are able to reconstruct a coarse version of the analysis quantity, e.g. in the example from section 10.2.3 with as few as four parameters.

We will now investigate the application of the method to the pre-image problem in crash simulation, where it has another very interesting property, namely it corresponds to obtaining an approximation of a numerical simulation without using the finite element software. Note that crash simulation is a highly nonlinear process and trying to interpolate between existing simulations based on the original input parameter space will in general not produce satisfactory results. One simulation of a real model in the car industry takes several hours using up to hundreds of computing nodes, the advantage is clear if one would be able to get an approximate behavior without investing that huge amount of computing resources. An approach in this direction was presented in [Bohn et al., 2013].

We will use again the data set from section 10.2.3, a frontal crash simulation of a truck with around 60,000 nodes and 17 time steps. A total of 126 simulations were obtained by varying randomly the thickness of 9 structural components, 116 data sets are used for the training phase and 10 data sets are used as testing data. As in section 10.2.3 the analysis variable is the norm of the difference between the deformations at time step 6 and 7 (the feature vector).

We have 116 feature vectors which are functions defined at the nodes of a mesh. Just to evaluate how different the feature vectors might be, let us take the squared nodal distance difference between two arbitrary selected feature vectors. Figure 10.17 shows two examples of such differences. As expected the squared differences could be large for arbitrary selected datasets.

We use the same procedure as done in section 10.2.3 where instead of using the feature vectors, we use the projection coefficients to the eigenvectors of the Laplace-Beltrami operator. As shown in section 10.2.3, using the coefficients until component $p = 4$ where enough for classification using diffusion maps. The training data is accordingly projected to the basis of eigenvectors and we use the coefficients until component $p = 4$ as feature vector for each simulation as before.

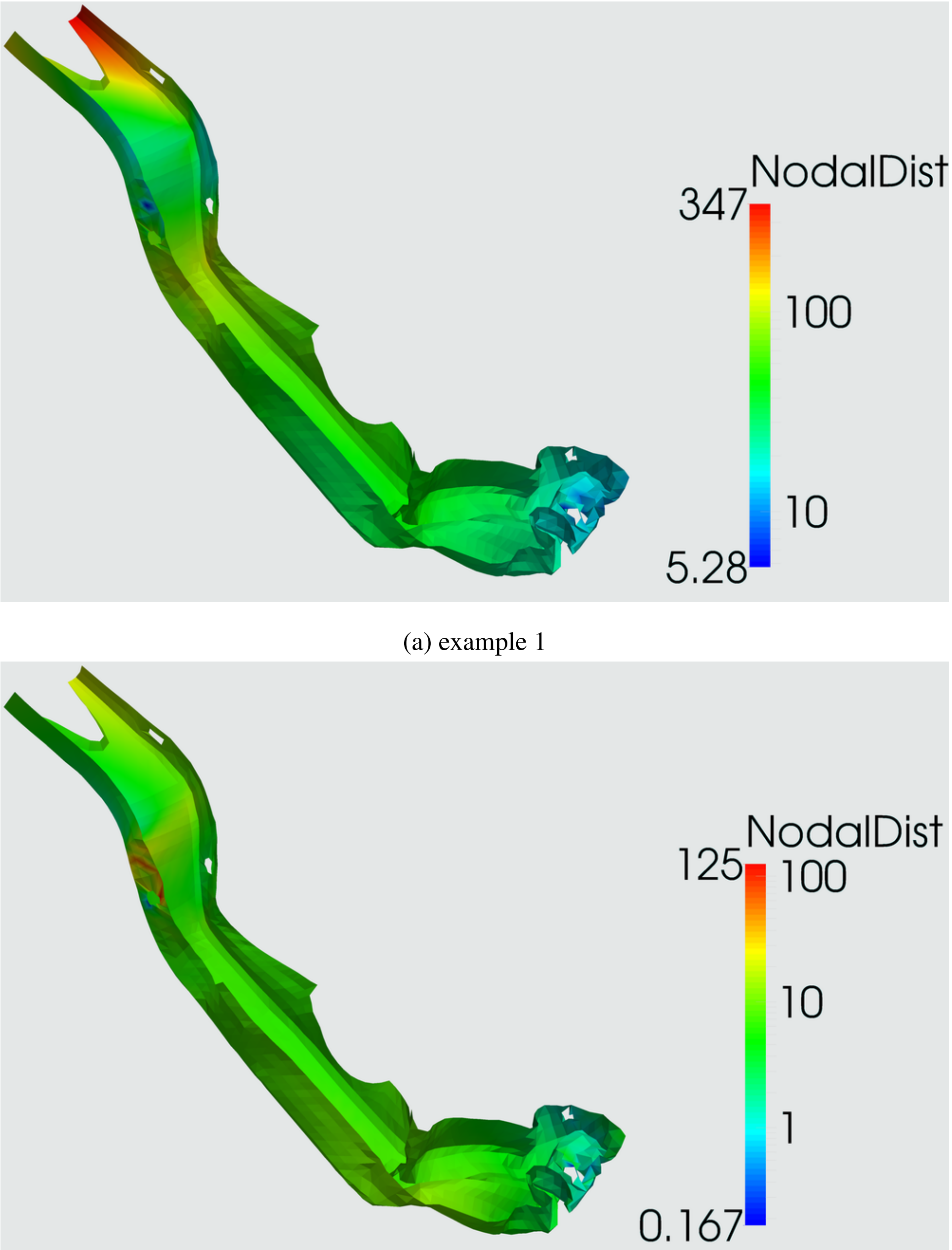
Our goal is to measure the quality of the low dimensional embedding, found using the projection coefficients until component $p = 4$ as a substitute of the embedding using diffusion maps. The idea is, based on the coordinates of the training data as vectors in $\mathbb{R}^4$, to pick a training simulation which is near to a test simulation. Each chosen training data set can be considered as an estimate of a corresponding test simulation.

We will proceed as follows. First the ten simulations from the test set are also projected to the eigenvector basis. Next we use the low dimensional coordinates of the test set and find the nearest neighbor, for each test simulation (now represented as a vector in $\mathbb{R}^4$), to the coordinates of the training simulations (also in $\mathbb{R}^4$).

There are ten simulations in the training set which are near to a corresponding test simulation. As a way of quantifying how close those training simulations are to the test simulations, the mean of the squared differences between the test data set and the corresponding nearest neighbor from the training set for all ten test simulations is shown in figure 10.18.

For other nonlinear dimensionality reduction procedures, e.g. diffusion maps, we have observed in the course of our research that using the nearest neighbor in the low dimensional coordinates produces on average a worse “estimation” result.

We can see that using the embedding coordinates from the training set, an estimation of a simulation result for several test simulations is possible. How to predict so called “virtual” numerical simulations using a combination of the eigenvectors will be discussed in section 10.2.5.



(a) example 1

(b) example 2

Figure 10.17: a), b): We consider deformations between two consecutive time steps. Shown in each picture is the differences of the deformations between two arbitrarily chosen simulations from the training data

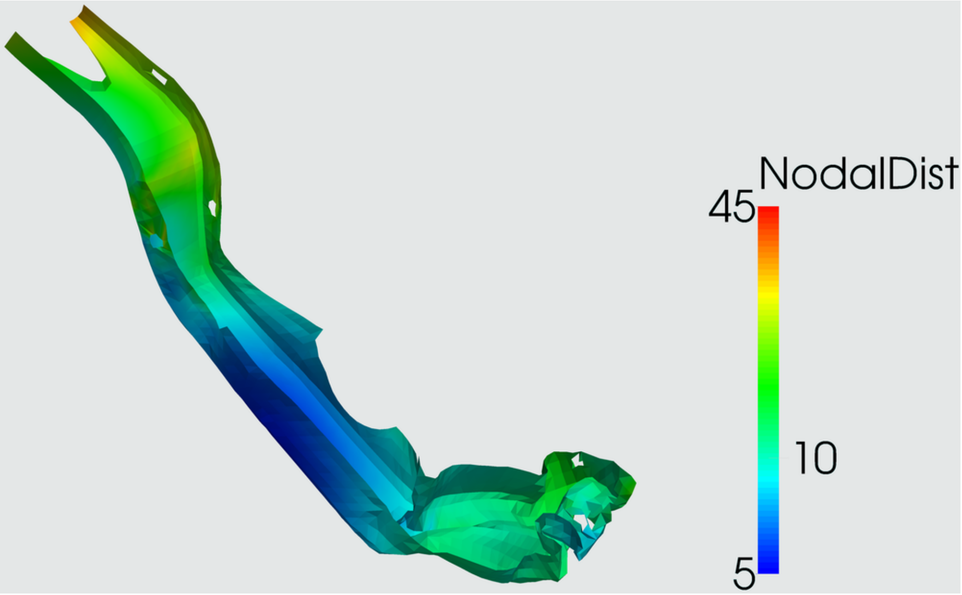


Figure 10.18: Squared mean error between ten test simulations and corresponding estimations from the training data, found using nearest neighbors evaluated on the projection coefficients for each simulation up to component $p = 4$.

10.2.5 Time Dependent Analysis of Crash Simulations

The properties of the introduced approach allow also the efficient analysis of time dependent information, this is due to the use of a common basis for all simulations and time steps. Car crash simulations are time dependent, in very few milliseconds the structure of a car can deform greatly. Furthermore, an unstable behavior can originate from small variations in the material properties, initial load conditions or numerical conditioning. This phenomenon is called buckling and is a serious problem for the robust design of car components. Relevant for an engineer is not only the identification of principal bifurcation modes, but also the study of the time of origin of the unstable behavior.

Our approach now allows a time dependent analysis of such unstable deformation characteristics.

We will use the same Chevrolet truck example as before, but to visualize the time dependent behavior we employ more time steps and therefore now use 167 simulations and 141 time steps, where again 9 plate thicknesses are varied.

We proceed along the lines of section 6.4 using the Laplace-Beltrami operator in algorithm 2 for the mesh at time $t = 0$. The eigenvectors of the operator are taken as basis for all 167 simulations × 141 time steps. The projection coefficients represent all simulations over all time steps in this basis. There are $N_h = 1741$ coefficients, nevertheless we observe that the variations for all simulation and time steps is concentrated around the first ones.

We would like to study the type of information that can be gained from the coefficients of the first component ($p = 1$) of the orthogonal decomposition. For this first component, we have chosen the projection coefficients for the deformations in the direction x, y and z. Figure 10.19 shows a graphical representation of those coefficients. Each point represents a simulation at a specific time step, the color of the points corresponds to the time step.

There are several observations that can be made from the obtained configuration in figure 10.19. We can see that these reduced coordinates of the simulations give an organization of the data in time. A bifurcation clearly starts about half-way during the crash simulation, while one can see how the positions of the simulations for the last time steps appear mixed with those of previous ones. This corresponds with the rebound effect in a crash, where the car bounces back from the obstacle after the inertia of the movement is absorbed. Other embedding methods can be applied to the same data, but dealing with all time steps simultaneously is very limited. The usual approach is to use the PCA for time dependent analysis as well, so as a first example let us calculate it using all simulations from one time step, where the bifurcation is clearly present. The spectral coefficients obtained by projecting the deformations for some selected time steps along these principal components becomes the low dimensional structure shown in figure 10.20 (a), this approach does not produce adequate results since the variability over all time steps is not taken into account. Computing a PCA using all simulations and time steps improves the results, but only to some degree, see figure 10.20 (b). Although a time behavior is now visible and a small separation is recognizable near the end, the clear separation and different results due to the bifurcation cannot be recognized in the embedding, in particular the time around which the bifurcation initiates is not visible as is the case in figure 10.19. Other nonlinear dimensionality reduction methods could in principle be used for dealing with this data set. But then, either the embedding method has to be computed as many times as time steps are available, where the switching of the eigenvectors makes re-

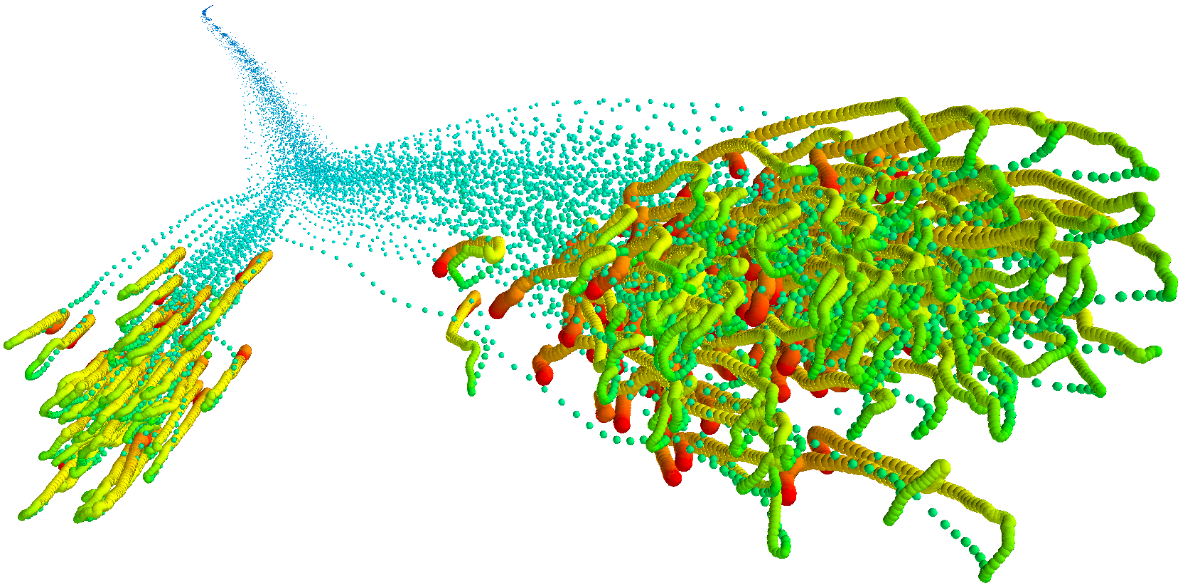
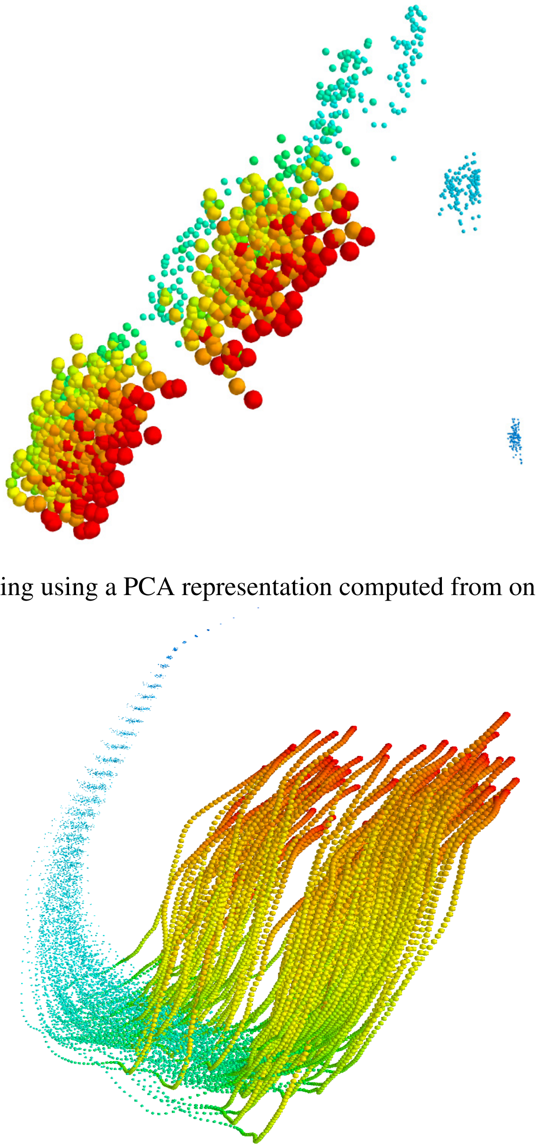


Figure 10.19: Reduced 3D representation of 23547 (167 simulations × 141 time steps) time dependent simulation results, obtained by the spectral decomposition of the Laplace-Beltrami operator. The coordinates of each point, which indicates a simulation at a specific time, are the first spectral coefficients for each direction x, y, and z of the deformations of a car part, they are colored according to the corresponding time step of the simulation.

covering a time dependent low dimensional structure very cumbersome, if at all possible. Alternatively, one could attempt to embed other time steps into the coordinates obtained from one time step using the Nyström method, but this is as limited as the PCA example above. Using all simulations and all time steps is not feasible since one has to deal with large full matrices, and cannot go to a formulation in the size of the mesh, or relevant parts of it, as is possible for PCA or our approach.

As can be seen in this example, projecting all simulations to the same basis from the Laplace-Beltrami operator enables a new type of data analysis approach. The projection coefficients to the basis are used for analysis. For this example, the coefficients of the first component of the orthogonal decomposition, already delivers a lot of information about the variability encountered in a simulation bundle where a set of parameters have been changed.



(a) Embedding using a PCA representation computed from one time step.

(b) Embedding using a PCA representation computed from all time steps.

Figure 10.20: Reduced 3D representation of simulations over several time steps, obtained with PCA. The points indicate a simulation at a specific time and are colored according to the corresponding time step of the simulation.

Reconstruction of Simulations

Using the same data as for the example above, we have also observed an interesting behavior by the reconstruction making use of the fast decay of the first spectral coefficients. We have selected one arbitrary simulation at a specific time step (time step 11 from 152) as a reference from which we save all spectral coefficients. For all other simulations we keep only the first 100 coefficients for all time steps and append to each one of them, the coefficients of the reference simulation starting from 101 to the end, that is to the number of nodes of the simulation.

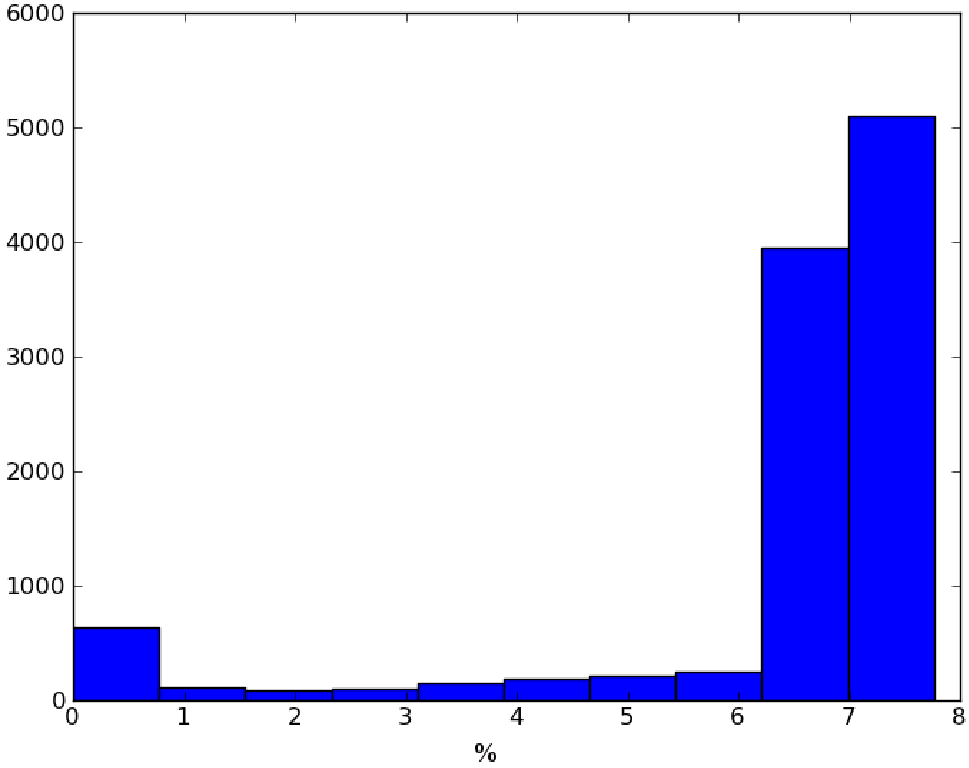


Figure 10.21: Histogram of the error in percentage for the reconstruction of 76 simulations × 142 time steps (= 10792) using the first 100 spectral coefficients. The coeffcients starting from 101 to the end (number of nodes of the simulation) are kept constant (for all simulations and for all time steps) and correspond to the coefficients of one reference simulation for time step 11

We use 142 time steps for a total of 76 simulations and reconstruct, using the appended coefficients, a total of 10792 = (142 × 76) simulations. A relative maximum difference of 7.7% is obtained. The maximal displacement in [mm] is 4553 and the minimum is 3352, and we have subtracted the mean value to compensate for rigid body motions. This result shows numerically that the principal variations of the dataset are concentrated on the first coefficients. Figure 10.21 shows the distribution of the error with respect to the corresponding simulation for all the dataset.

A further capability of the use of single basis from an operator has been studied. The experiment suggest that a further application of the proposed approach could be in the area of data compression. Nevertheless much more research have to be done to verify it.

Finally we would like to present a further application of the proposed approach which goes into the area of creating synthetic simulations. We will see that this is possible, based on the information on the variability of the projection coefficients for the different components of the orthogonal decomposition.

Crash Modes

The deformations in a crash process are a complex mixture of different effects such as translations, rotations, local and global bending and torsion. How the different modes interact, in particular in relation to the changes in the model is of great importance for engineering analysis.

In another context, comparing simulations with experiments is a difficult task due to the variability of manufacturing parts, the material modeling and due to experimental conditions. Analyzing how different deformation modes differ from the real experiment and morphing a mesh corresponding to it is a mayor need.

Our approach turns out to be useful in addressing the above mentioned challenges. We have used the same Chevrolet truck example as before but we studied ways of identifying and decomposing different crash effects along independent modes.

We mentioned in section 6.3 a very interesting theoretical link between the eigenfunctions of an operator invariant to a transformation group and the representation of the group. The idea is that in some cases, a linear representation of a transformation group can be found. We will show in an experiment that in the discrete case, using the eigenvector basis a representation of group actions along different components of the orthogonal decomposition of the operator of independent components can be achieved.

We will fix a specific time step (7 from 17) and evaluate for this shape, the operator of independent components. A total of 115 simulations are used and it is assumed that at each point of a reference simulation, a cloud of 115 points is formed that allows us to evaluate a local Jacobian. See algorithm 3 for the construction of this operator. Our aim is to be able to identify the independent modes of a complex mixture of them in a crash process. That this is possible can be seen in the following results. We use the first $N_t = 4$ components of the orthogonal decomposition. A reconstruction of a simulation reflecting only the transformations along some component p can be done by fixing all the coefficients of the orthogonal decomposition with exception of the p one. The value of the coefficient for component p corresponds to the one of the specific simulation and time step, in this way we can reconstruct 115 simulations reflecting the changes only for the component p.

A few reconstructed simulations for the first component, that is for $p = 1$ is shown in figure 10.22. It can be seen that it corresponds to a translation.

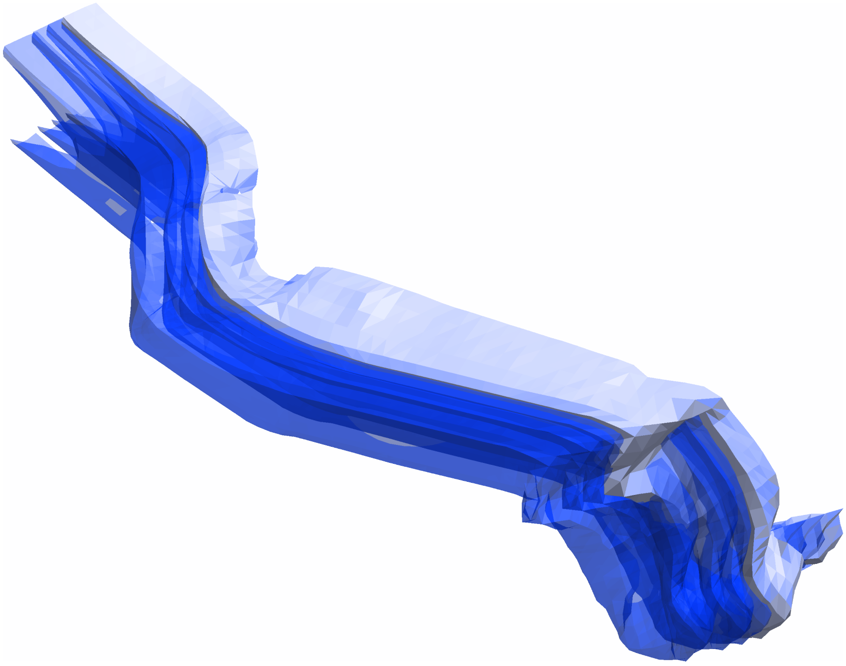


Figure 10.22: Translation mode identified using component $p = 1$ of an orthogonal decomposition of the operator of independent components

Some few reconstructed simulations obtained for the next component, that is for $p = 2$ are shown in figure 10.23. It can be seen that it corresponds to a rotation.

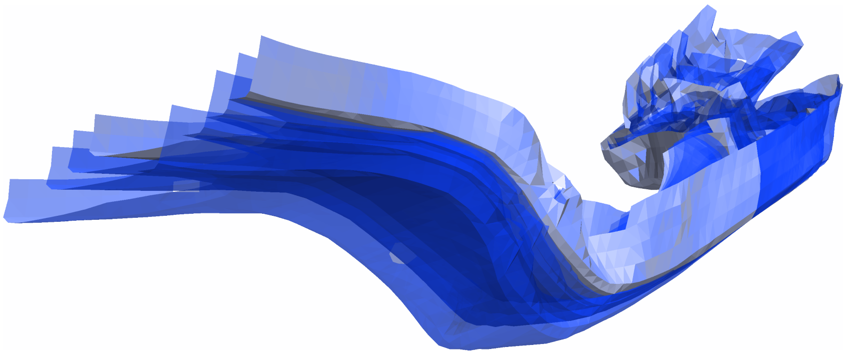


Figure 10.23: Rotation mode identified using component $p = 2$ of an orthogonal decomposition of the operator of independent components

The next reconstructed simulations are obtained for the third component in the orthogonal decomposition, that is for $p = 3$. Figure 10.24 shows that it corresponds to a deformation. The next reconstructed simulations are obtained for the fourth component, that is for $p = 4$. Figure 10.25 shows that it corresponds to a local deformation.

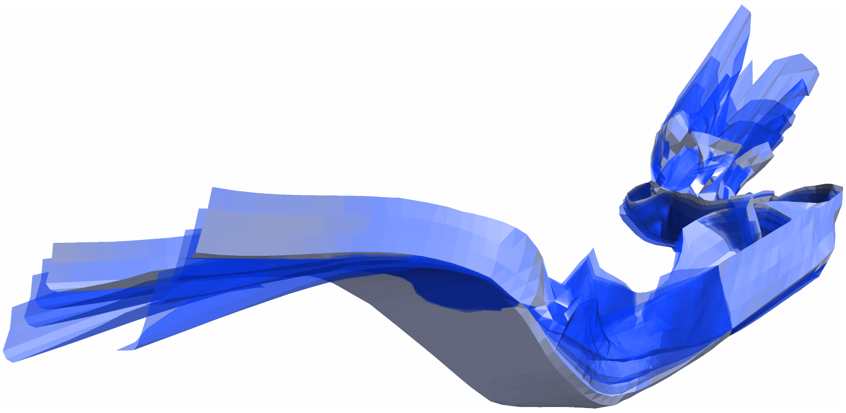


Figure 10.24: Deformation mode identified using component $p = 3$ of an orthogonal decomposition of the operator of independent components

Notice that the extraction of these independent modes constitute a numerical example of the relationship between group actions and its representation through eigenvectors of an invariant operator as discussed in section 6.3.

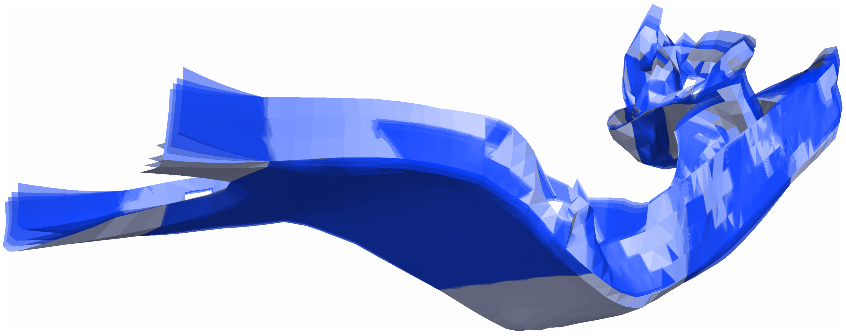


Figure 10.25: Deformation mode identified using component $p = 4$ of an orthogonal decomposition of the operator of independent components

Still we would like to mention one more possibility to increase the number of modes that can be obtained if a certain amount of simulations are available. We recall the process of obtaining orbit paths introduced in section 9.2.4. The process described there, makes use of algorithm 7 to obtain a deformation path between two given shapes. Given only two deformation stages, one can obtain an arbitrary amount of intermediate deformations between those two. We have chosen two arbitrary deformations and calculate an orbit path between them. This is illustrated in figure 10.26.

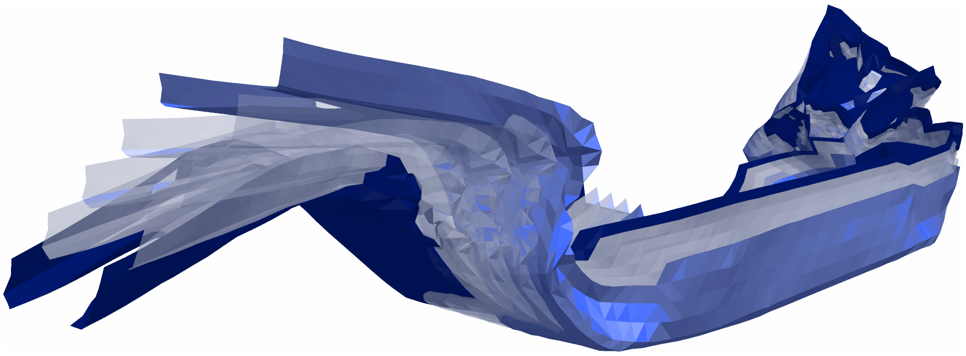


Figure 10.26: Deformation orbit path evaluated using an orthogonal decomposition of the operator of independent components

We would like to emphasize the flexibility inherent into this process. A large amount of different deformation modes can be obtained in this way by just changing the initial and final deformations. We can expect a large number of applications of this procedure in an engineering context.

CHAPTER 11

Summary and Open Research

Now at the end of this thesis we hope the reader agrees fully or partially with us on the statement: "Mathematics can be so astonishing and invaluable in confronting challenges in applications, providing perspectives and resources that appear to be inexhaustible". In any case, even if you disagree with that, we hope we were able to provide the reader with interesting topics and perspectives in an area which, in our opinion, still requires further research.

We can summarize the developments in this thesis from different aspects;

From a mathematical point of view we have demonstrated a method to characterize simulation results as elements of an abstract simulation space. Conditions under which this space is a manifold have also been explained. Orbits of some group actions as elements of this space have been represented in a lower dimensional space, enabling easier analysis of such simulation bundles. In addition a new method for dimension reduction has been proposed that is able to cope with nonlinearities. The method has been justified mathematically using approximation theory for spectral orthogonal decomposition.

Even if the objective of the thesis was to originally deal with simulation bundles, an extension for the analysis of plane shapes has been realized. The use of an invariant basis for the analyzed examples suggests a significant reduction in the evaluation costs for calculating geodesic paths for the space of curves embedded in $\mathbb{R}^2$. This topic has to be investigated further.

An extension of the proper orthogonal decomposition (POD) method that is more general than the standard approach has been presented. For a nonlinear PDE example we have been able to show that, due to the invariance property, a Galerkin approximation can still be used if the parameters of the differential equations change. The basis does not need to be recalculated as in the standard POD approach.

The proposed approach is shown to provide a convenient representation for simulation results of finite element simulations with the possibility of creating approximate solutions without carrying out an expensive simulation. In addition, a reduced representation of an orbit space is shown to potentially reduce the evaluation cost for calculating geodesic paths for simulations.

Several industrial applications are able to show the usefulness of the new approach as a data analysis tool. We expect that many areas, not only in the area of simulation in engineering, could benefit from it.

We see a need for further research also in several of these aspects.

Though we have been able to show that approximative geodesic paths can be obtained using the new approach, it still remains to be demonstrated formally and numerically as to how well they can be approximated.

Where the differential equations are available we have shown that an extension of the proper orthogonal decomposition (POD) can be realized that is independent, under certain conditions, of the parameter variations of the equations for which the invariance assumption is valid. Nevertheless the solution of the nonlinear ODE example underwent problems in the integration of the reduced POD system, so that one had to analyze as to why different integration schemes failed. Once this is known an approach that overcomes such limitations can be devised.

Solving the geodesic equation for 3D shapes is very time consuming. Putting this differential equation in terms of orbit coordinates could potentially reduce the solution times due to the lower dimensionality of those coordinates.

Our approach has the flexibility to incorporate other types of transformation invariance through the operator. A direct extension to the approach is the use of the affine invariant Laplace-Beltrami operator as described in [Raviv et al., 2011]. Its use could significantly increase the flexibility in representing variations in the solutions of finite element simulations of physical processes. Also other types of invariant operators can be used as the Schrödinger operator [Iglesias and Kimmel, 2012] as well as other types of invariant metrics such as the Bi-Laplacian distance [Lipman et al., 2010]. It is expected that using them, the analysis capabilities of the proposed method can be significantly enhanced.

A known open problem in the numerical solution of parametrized partial differential equations by spectral methods is how to find the best basis for solving a PDE. In this thesis we have identified a relationship between special functions (solutions of the Sturm-Liouville problem) and invariant operators, we conjecture that the way to choose a basis for a spectral method or for a reduced basis method should be based on the types of invariance that are expected in the solutions. How to formalize and prove this conjecture is still open.

Further research is needed to explore the link between group representations and invariant operators. In particular a theoretical treatment about group representation in the cases of invariant operators on meshes is needed.

The basis of the Laplace-Beltrami operator is known to have a slow decay rate, particularly for non-smooth functions. Much better alternatives have been developed in the last years using wavelets on graphs or treelets. A better basis that compactly represents a given mesh function can be obtained using those approaches. Use of such basis for the analysis of simulations can provide a natural extension to our approach if ways can be defined to consider it as invariant to some type of transformations.

Bibliography

[Ackermann et al., 2008] Ackermann, S., Gaul, L., Hanss, M., and Hambrecht, T. (2008). Principal component analysis for detection of globally important input parameters in nonlinear finite element analysis. *IAM Institute of Applied and Experimental Mechanics, Stuttgart*.

[Aubry et al., 2011] Aubry, M., Schlickewei, U., and Cremers, D. (2011). The wave kernel signature: A quantum mechanical approach to shape analysis. In *IEEE International Conference on Computer Vision Workshops, ICCV 2011 Workshops, Barcelona, Spain, November 6-13, 2011*, pages 1626–1633.

[Auscher and Hyt 2013] Auscher, P. and Hyt T. (2013). Orthonormal bases of regular wavelets in spaces of homogeneous type. *Applied and Computational Harmonic Analysis*, 34(2):266 – 296.

[Bauer et al., 2014] Bauer, M., Bruveris, M., and Michor, P. W. (2014). Overview of the geometries of shape spaces and diffeomorphism groups. *J. Math. Imaging Vision*, 50(1-2):60–97.

[Bauer et al., 2011] Bauer, M., Harms, P., and Michor, P. W. (2011). Sobolev metrics on shape space of surfaces. *Journal of Geometric Mechanics*, 3(4).

[Bauer et al., 2012] Bauer, M., Harms, P., and Michor, P. W. (2012). Curvature weighted metrics on shape space of hypersurfaces in n-space. *Differential Geometry and its Applications*, 30(1).

[Belkin and Niyogi, 2008] Belkin, M. and Niyogi, P. (2008). Convergence of Laplacian eigenmaps. *preprint, short version NIPS 2008*.

[Belkin et al., 2008] Belkin, M., Sun, J., and Wang, Y. (2008). Discrete Laplace operator on meshed surfaces. In *Proceedings of the Twenty-fourth Annual Symposium on Computational Geometry*, SCG '08, pages 278–287. ACM.

[Ben-chen and Gotsman, 2005] Ben-chen, M. and Gotsman, C. (2005). On the optimality of spectral compression of mesh data. *ACM Trans. Graph*, 24:60–80.

[Bernstein et al., 2000] Bernstein, M., Silva, V. D., Langford, J. C., and Tenenbaum, J. B. (2000). Graph approximations to geodesics on embedded manifolds.

[Bohn et al., 2013] Bohn, B., Garcke, J., Iza Teran, R., Paprotny, A., Peherstorfer, B., Schepsmeier, U., and Thole, C.-A. (2013). Analysis of car crash simulation data with nonlinear machine learning methods. In *Proceedings of the ICCS 2013*, volume 18 of *Procedia Computer Science*, pages 621–630. Elsevier.

[Borzellino, 1992] Borzellino, J. E. (1992). *Riemannian Geometry of Orbifolds*. PhD dissertation, University of California, Los Angeles, Department of Mathematics.

[Bronstein et al., 2006] Bronstein, A. M., Bronstein, M. M., and Kimmel, R. (2006). Efficient computation of isometry-invariant distances between surfaces. 28(5):1812–1836.

[Buchman et al., 2011] Buchman, S. M., Lee, A. B., and Schafer, C. M. (2011). High-dimensional density estimation via SCA: An example in the modelling of hurricane tracks. *Statistical Methodology*, 8(1):18 – 30.

[Burghard et al., 2013] Burghard, O., Berner, A., Michael, W., Mitra, N., Seidel, H.-P., and Klein, R. (2013). Compact part-based shape spaces for dense correspondences. Technical report, Bonn University.

[Canuto, 2006] Canuto, C. (2006). *Spectral Methods : Fundamentals in Single Domains*. Scientific computation. Springer, Berlin.

[Coifman et al., 2008] Coifman, R., Kevrekidis, I. G., Lafon, S., Maggioni, M., and Nadler, B. (2008). Diffusion maps, reduction coordinates, and low dimensional representation of stochastic systems. *Multiscale Modeling and Simulation*, 7(2):842–864.

[Coifman and Lafon, 2006] Coifman, R. and Lafon, S. (2006). Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30.

[Coifman and Maggioni, 2006] Coifman, R. R. and Maggioni, M. (2006). Diffusion wavelets. *Applied and Computational Harmonic Analysis*, 21(1):53 – 94. Special Issue: Diffusion Maps and Wavelets.

[Costa, 2004] Costa, B. (2004). Spectral methods for partial differential equations. *CUBO A Mathematical Journal*, 6(4):1 – 32.

[Courant and Hilbert, 1953] Courant, R. and Hilbert, D. (1953). *Methods of Mathematical Physics*, volume 1. Interscience Publishers, Inc, New York.

[Dey et al., 2010] Dey, T. K., Ranjan, P., and Wang, Y. (2010). Convergence, stability, and discrete approximation of Laplace spectra. In *Proceedings of the Twenty-first Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '10, pages 650–663, Philadelphia, PA, USA. Society for Industrial and Applied Mathematics.

[do Carmo, 1992] do Carmo, M. P. (1992). *Riemannian Geometry*. Birkhäuser.

[Donoho and Johnstone, 1994] Donoho, D. L. and Johnstone, I. M. (1994). Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81:425–455.

[Eshera and Fu, 1984] Eshera, M. A. and Fu, K. S. (1984). A graph distance measure for image analysis. *IEEE Transactions on Systems, Man, and Cybernetics*, 14(3):398–408.

[Farfade et al., 2015] Farfade, S. S., Saberian, M. J., and Li, L.-J. (2015). Multi-view face detection using deep convolutional neural networks. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, ICMR '15, pages 643–650, New York, NY, USA. ACM.

[Ferguson et al., 2010] Ferguson, A. L., Zhang, S., Dikiy, I., Panagiotopoulos, A. Z., Debenedetti, P. G., and Link, A. J. (2010). An experimental and computational investigation of spontaneous Lasso formation in Microcin j25. *Biophysical Journal*, 99(9):3056 – 3065.

[Feuersänger and Griebel, 2009] Feuersänger, C. and Griebel, M. (2009). Principal manifold learning by sparse grids. *Computing*, 85(4):267–299.

[Fletcher and Joshi, 2007] Fletcher, P. T. and Joshi, S. (2007). Riemannian geometry for the statistical analysis of diffusion tensor data. *Signal Processing*, 87:250–262.

[Fletcher, 2004] Fletcher, T. (2004). *Statistical Variability in Nonlinear Spaces: Application to Shape Analysis and DT-MRI*. PhD dissertation, University of North Caroline at Chapel Hill, Department of Computer Science.

[Fowlkes et al., 2004] Fowlkes, C., Belongie, S., Chung, F., and Malik, J. (2004). Spectral grouping using the nyström method. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26:214–225.

[Garcke, 2016] Garcke, J. (2016). Lecture Notes in Scientific Computing II: Kernel Methods and Nonlinear Dimensionality Reduction, Universität Bonn.

[GNS,] GNS. Gns mbh. http://www.gns-mbh.com/.

[Gomez-Ullate et al., 2008] Gomez-Ullate, D., Kamran, N., and Milson, R. (2008). An extended class of orthogonal polynomials defined by a sturm-liouville problem.

[Gottlieb et al., 1992] Gottlieb, D., Shu, C.-W., Solomonoff, A., and Vandeven, H. (1992). On the gibbs phenomenon i: recovering exponential accuracy from the fourier partial sum of a nonperiodic analytic function. *Journal of Computational and Applied Mathematics*, 43(1–2):81 – 98.

[Grebenkov and Nguyen, 2013] Grebenkov, D. and Nguyen, B. (2013). Geometrical structure of Laplacian eigenfunctions. *SIAM Review*, 55(4):601–667.

[Hammond et al., 2011] Hammond, D. K., Vandergheynst, P., and Gribonval, R. (2011). Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 30(2):129 – 150.

[Hein and Audibert, 2005] Hein, M. and Audibert, J.-Y. (2005). Intrinsic dimensionality estimation of sub-manifolds in rd. In *Proceedings of the 22nd international conference on Machine learning*, ICML '05, pages 289–296, New York, NY, USA. ACM.

[Hein et al., 2007] Hein, M., Audibert, J.-Y., Luxburg, U. V., and Dasgupta, S. (2007). Graph Laplacians and their convergence on random neighborhood graphs. *Journal of Machine Learning Research*, 8:1325–1370.

[Heine, 1993] Heine, V. (1993). *Group Theory in Quantum Mechanics: An Introduction to Its Present Usage*. Courier Corporation.

[Helgason, 1984] Helgason, S. (1984). *Groups and geometric analysis : integral geometry, invariant differential operators, and spherical functions*. Pure and applied mathematics. Academic press.

[Huckemann et al., 2010] Huckemann, S., Hotz, T., and Munk, A. (2010). Intrinsic shape analysis: Geodesic PCA for riemannian manifolds modulo isometric lie group actions. *Statistica Sinica*, 20:1–100.

[Iglesias and Kimmel, 2012] Iglesias, J. A. and Kimmel, R. (2012). *Schrödinger diffusion for shape analysis with texture*, pages 123–132. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Iza Teran, 2013] Iza Teran, R. (2013). Enabling the analysis of finite element simulation-bundles. *International Journal for Uncertainty Quantification*, pages 95–110.

[Iza Teran and Garcke, 2016] Iza Teran, R. and Garcke, J. (2016). Operator based multiscale analysis of simulation bundles. *Submitted*.

[Jakobson et al., 2001] Jakobson, D., Nadirashvili, N., and Toth, J. (2001). Geometric properties of eigenfunctions. *Russian Math. Surveys*, 56:67–88.

[Jarvis, 2012] Jarvis, C. (2012). Reduced order model study of Burgers' equation using proper orthogonal decomposition. Master's thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

[Kähler, 2012] Kähler, M. (2012). Shape spaces and shape modelling. Master's thesis, Durham University, Durham.

[Karni and Gotsman, 2000] Karni, Z. and Gotsman, C. (2000). Spectral compression of mesh geometry. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, pages 279–286. ACM Press/Addison-Wesley Publishing Co.

[Karoui, 2010] Karoui, N. E. (2010). On information plus noise kernel random matrices. *Annals of Statistics*, 38:3191–3216.

[Kendall et al., 1999] Kendall, D. G., Barden, D., Carne, T. K., and Le, H. (1999). *Shape and Shape Theory*. Wiley.

[Kilian et al., 2007] Kilian, M., Mitra, N. J., and Pottmann, H. (2007). Geometric modeling in shape space. *ACM Trans. Graph.*, 26(3).

[Kriegl and Michor, 1997] Kriegl, A. and Michor, P. W. (1997). *The Convenient Setting of Global Analysis*, volume 53 of *Mathematical Surveys and Monographs*. American Mathematical Society.

[Kurtek et al., 2012] Kurtek, S., Klassen, E., Gore, J. C., Ding, Z., and Srivastava, A. (2012). Elastic geodesic paths in shape space of parameterized surfaces. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(9):1717–1730.

[Kushnir et al., 2012] Kushnir, D., Haddad, A., and Coifman, R. (2012). Anisotropic diffusion on submanifolds with application to earth structure classification. *Applied and Computational Harmonic Analysis*, 32(2):280–294.

[Lafon, 2004] Lafon, S. (2004). *Diffusion Maps*. PhD dissertation, Yale University, Department of Mathematics.

[Lai et al., 2010] Lai, R., Shi, Y., Scheibel, K., Fears, S., Woods, R., Toga, A., and Chan, T. (2010). Metric-induced optimal embedding for intrinsic 3D shape analysis. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2871–2878.

[Lee et al., 2008] Lee, A. B., Nadler, B., and Wasserman, L. (2008). Treelets—an adaptive multi-scale basis for sparse unordered data. *Ann. Appl. Stat.*, 2(2):435–471.

[Lee and Wasserman, 2010] Lee, A. B. and Wasserman, L. (2010). Spectral connectivity analysis. *Journal of the American Statistical Association*, 105(491):1241–1255.

[Lee and Verleysen, 2007] Lee, J. and Verleysen, M. (2007). *Nonlinear Dimensionality Reduction*. Springer Publishing Company, Incorporated.

[Ling et al., 2016] Ling, J., Jones, R. E., and Templeton, J. A. (2016). Machine learning strategies for systems with invariance properties. *Journal of Computational Physics*, 318(C).

[Lipman et al., 2010] Lipman, Y., Rustamov, R. M., and Funkhouser, T. A. (2010). Biharmonic distance. *ACM Trans. Graph.*, 29(3):27:1–27:11.

[Lipman et al., 2005] Lipman, Y., Sorkine, O., Levin, D., and Cohen-Or, D. (2005). Linear rotation-invariant coordinates for meshes. *ACM Trans. Graph.*, 24(3):479–487.

[LSTC, 2004] LSTC (2004). *LS-DYNA Keyword User's Manual*. Livermore Software Technology.

[Mei and Thole, 2008] Mei, L. and Thole, C.-A. (2008). Data analysis for parallel car-crash simulation results and model optimization. *Sim. Modelling Practice and Theory*, 16(3):329–337.

[Meyer and Shen, 2014] Meyer, F. G. and Shen, X. (2014). Perturbation of the eigenvectors of the graph Laplacian: Application to image denoising. *Applied and Computational Harmonic Analysis*, 36(2):326 – 334.

[Michor, 2008] Michor, P. (2008). *Topics in Differential Geometry*. American Mathematical Society.

[Michor and Mumford, 2006] Michor, P. W. and Mumford, D. (2006). Riemannian geometries on spaces of plane curves. *J. Eur. Math. Soc.*, 8:1–48.

[Mitchell et al., 1987] Mitchell, J. S. B., Mount, D. M., and Papadimitriou, C. H. (1987). The discrete geodesic problem. *SIAM J. Comput.*, 16(4):647–668.

[Mohammad, 2006] Mohammad, M.-J. (2006). *Special Functions: A Symmetric Generalization of Sturm-Liouville Problems and its Consequences*. Dissertation, University of Kassel.

[Mumford and Michor, 2013] Mumford, D. and Michor, P. W. (2013). On Euler's equation and EPDiff. *Journal of Geometric Mechanics*, 5(3):319–344.

[Nadler et al., 2006] Nadler, B., Lafon, S., Coifman, R. R., and Kevrekidis, I. G. (2006). Diffusion maps, spectral clustering and the reaction coordinates of dynamical systems. *Applied and Computational Harmonic Analysis: Special issue on Diffusion Maps and Wavelets*, 21(1):113–127.

[Neji et al., 2009] Neji, R., Langs, G., Deux, J.-F., Maatouk, M., Rahmouni, A., Bassez, G., Fleury, G., and Paragios, N. (2009). Unsupervised classification of skeletal fibers using diffusion maps. In *Proceedings of the Sixth IEEE international conference on Symposium on Biomedical Imaging: From Nano to Macro*, ISBI'09, Piscataway, NJ, USA. IEEE Press.

[Ngô, 2012] Ngô, Q. A. (2012). *The Einstein-Scalar Field Lichnerowicz Equations on Compact Riemannian Manifolds*. PhD dissertation, University of Singapore, Department of Mathematics.

[Niyogi et al., 2008] Niyogi, P., Smale, S., and Weinberger, S. (2008). Finding the homology of submanifolds with high confidence from random samples. *Discrete and Computational Geometry*, 39(1-3):419–441.

[Olver, 1993] Olver, P. J. (1993). *Applications of Lie Groups to Differential Equations*, volume 107 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 2nd edition.

[Onishchik, 1993] Onishchik, A. (1993). *Lie Groups and Lie Algebras I*. Encyclopedia of Mathematical Sciences. Springer-Verlag.

[Ovsjanikov et al., 2012] Ovsjanikov, M., Ben-Chen, M., Solomon, J., Butscher, A., and Guibas, L. (2012). Functional maps: A flexible representation of maps between shapes. *ACM Trans. Graph.*, 31(4):30:1–30:11.

[Petitjean et al., 2014] Petitjean, F., Forestier, G., Webb, G. I., Nicholson, A. E., Chen, Y., and Keogh, E. J. (2014). Dynamic time warping averaging of time series allows faster and more accurate classification. In *2014 IEEE International Conference on Data Mining, ICDM 2014, Shenzhen, China, December 14-17, 2014*, pages 470–479.

[Quarteroni et al., 2015] Quarteroni, A., Manzoni, A., and Negri, F. (2015). *Reduced Basis Methods for Partial Differential Equations: An Introduction*. UNITEXT. Springer International Publishing.

[Raviv et al., 2011] Raviv, D., Bronstein, A. M., Bronstein, M., Kimmel, R., and Sochen, N. (2011). Affine-invariant diffusion geometry for the analysis of deformable 3D shapes. In *Proc. Computer Vision and Pattern Recognition (CVPR)*.

[Reuter et al., 2009] Reuter, M., Biasotti, S., Giorgi, D., Patanè, G., and Spagnuolo, M. (2009). Technical section: Discrete Laplace-Beltrami operators for shape analysis and segmentation. *Comput. Graph.*, 33(3):381–390.

[Reuter et al., 2006] Reuter, M., Wolter, F., and Peinecke, N. (2006). Laplace-Beltrami spectra as 'Shape-DNA' of surfaces and solids. *Computer-Aided Design*, 38(4):342–366.

[Rosenberg, 1997] Rosenberg, S. (1997). *The Laplacian on a Riemannian Manifold*. Cambridge University Press.

[Rubner et al., 2000] Rubner, Y., Tomasi, C., and Guibas, L. J. (2000). The Earth Mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40:2000.

[Rumpf and Wirth, 2011] Rumpf, M. and Wirth, B. (2011). An elasticity-based covariance analysis of shapes. *International Journal of Computer Vision*, 92(3):281–295.

[Singer, 2006a] Singer, A. (2006a). From graph to manifold Laplacian: The convergence rate. *Applied and Computational Harmonic Analysis*, 21(1):128–134.

[Singer, 2006b] Singer, A. (2006b). Spectral independent component analysis. *Applied and Computational Harmonic Analysis*, 21(1):135–144.

[Singer and Coifman, 2008] Singer, A. and Coifman, R. R. (2008). Non-linear independent component analysis with diffusion maps. *Applied and Computational Harmonic Analysis*, 25(2):226 – 239.

[Singer and Wu, 2013] Singer, A. and Wu, H.-t. (2013). Spectral convergence of the connection Laplacian from random samples. *ArXiv e-prints*.

[Sonday et al., 2009] Sonday, B. E., Haataja, M., and Kevrekidis, I. G. (2009). Coarse-graining the dynamics of a driven interface in the presence of mobile impurities: Effective description via diffusion maps.

[Spanos and Wu, 1994] Spanos, P. and Wu, Y. (1994). *Probabilistic Structural Mechanics: Advances in Structural Reliability Methods*. Springer My Copy UK.

[Srivastava et al., 2005] Srivastava, A., Joshi, S., Mio, W., and Liu, X. (2005). Statistical shape anlaysis: Clustering, learning and testing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27:590–602.

[Srivastava et al., 2011] Srivastava, A., Klassen, E., Joshi, S. H., and Jermyn, I. H. (2011). Shape analysis of elastic curves in euclidean spaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(7):1415–1428.

[Sugiura, 1990] Sugiura, M. (1990). *Unitary Representations and Harmonic Analysis: An Introduction*. Developments in Animal and Veterinary Sciences. North-Holland.

[Sun et al., 2009] Sun, J., Ovsjanikov, M., and Guibas, L. (2009). A concise and provably informative multi-scale signature based on heat diffusion. In *Proceedings of the Symposium on Geometry Processing*, SGP '09, pages 1383–1392, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association.

[Talmon and Coifman, 2015] Talmon, R. and Coifman, R. R. (2015). Intrinsic modeling of stochastic dynamical systems using empirical geometry. *Applied and Computational Harmonic Analysis*, 39(1):138 – 160.

[Thanou et al., 2014] Thanou, D., D. I Shuman, and Frossard, P. (2014). Learning parametric dictionaries for signals on graphs. *IEEE Transactions on Signal Processing*, 62(15):3849–3862.

[Thole et al., 2010] Thole, C. A., Nikitina, L., Nikitin, I., and Clees, T. (2010). Advanced mode analysis for crash simulation results. In *Proc. 9th LS-DYNA Forum*.

[Thorstensen et al., 2011] Thorstensen, N., Ityngier, P., Ségonne, F., and Keriven, R. (2011). Diffusion maps as a framework for shape modeling. *Comput. Vis. Image Underst.*, 115(4):520–530.

[Thurston, 2002] Thurston, W. P. (2002). *The Geometry and Topology of 3-Manifolds (Electronic Edition)*. http://www.msri.org/publications/books/gt3m/.

[Tran et al., 2013] Tran, T. D., Hofrichter, J., and Jost, J. (2013). An introduction to the mathematical structure of the wright-fisher model of population genetics. *Theory in Biosciences*, 132(2):73–82.

[Virshup et al., 2012] Virshup, A. M., Jiahao, C., and Martinez, T. J. (2012). Nonlinear dimensionality reduction for nonadiabatic dynamics: The influence of conical intersection topography on population transfer rates. *Journal of Chemical Physics*, 137.

[von Luxburg et al., 2008] von Luxburg, U., Belkin, M., and Bousquet, O. (2008). Consistency of spectral clustering. *Annals of Statistics*, 36(2):555–586.

[Wartak and Bors, 2010] Wartak, S. and Bors, A. G. (2010). Optical flow estimation using diffusion distances. In *Proceedings of the 2010 20th International Conference on Pattern Recognition*, ICPR '10, pages 189–192.

[Zhang et al., 2011] Zhang, P., Qiao, H., and Zhang, B. (2011). An improved local tangent space alignment method for manifold learning. *Pattern Recognition Letters*, 32(2):181–189.