Philipp Grohs
Martin Holler
Andreas Weinmann *Editors*

Handbook of Variational Methods for Nonlinear Geometric Data

Springer

Handbook of Variational Methods for Nonlinear Geometric Data

Philipp Grohs • Martin Holler • Andreas Weinmann
Editors

Handbook of Variational Methods for Nonlinear Geometric Data

Springer

Editors

Philipp Grohs
Faculty of Mathematics
University of Vienna
Wien, Austria

Martin Holler
Institute of Mathematics
and Scientific Computing
University of Graz
Graz, Austria

Andreas Weinmann
Department of Mathematics
and Natural Sciences
Hochschule Darmstadt
Darmstadt, Germany

ISBN 978-3-030-31350-0 ISBN 978-3-030-31351-7 (eBook)
https://doi.org/10.1007/978-3-030-31351-7

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG.
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Nonlinear geometry arises in various applications in science and engineering. Examples of nonlinear data spaces are diverse and include for instance nonlinear spaces of matrices, spaces of curves, shapes as well as manifolds of probability measures. Applications where such data spaces appear can be found in biology, medicine, engineering, geography, and computer vision.

Variational methods on the other hand have evolved to be very powerful tools of applied mathematics. They involve techniques from various branches of mathematics such as statistics, modeling, optimization, numerical mathematics and analysis. The vast majority of research on variational methods, however, is focused on methods for data in linear spaces.

Variational methods for nonlinear data are a currently emerging topic. As a consequence, and since such methods involve various branches of mathematics, there exists a plethora of different, recent approaches dealing with various aspects of variational methods for nonlinear geometric data. Research results are rather scattered over different mathematical communities.

The intention of this handbook is to cover different research directions in the context of variational methods for nonlinear geometric data, to bring together researchers from different communities working in the field, and to create a collection of works giving basic introductions, giving overviews, and describing the state of the art in the respective discipline. To this end, we have invited numerous authors, who are leading experts in different disciplines related to variational methods for nonlinear geometric data, to contribute an introductory overview of the state of the art and of novel developments in their field. With this, we hope to both stimulate exchange and collaborations of different research groups and to provide a unique reference work for experts and newcomers in the field.

Our special thanks go to the authors of the chapters for their valuable contributions, which constitute the main part of the book. Also, we thank the referees for their valuable comments. Furthermore, our thanks go to the Springer team. In particular, we want to thank Susan Evans for encouraging us to do this project and for the perfect support in all aspects, and all people from Springer who helped to produce the book.

Wien, Austria	Philipp Grohs
Graz, Austria	Martin Holler
Darmstadt, Germany	Andreas Weinmann
August 2019	

Acknowledgements

Martin Holler acknowledges support by the Austrian Science Fund (FWF) (Grant J 4112). Andreas Weinmann acknowledges support by the DFG Grants WE 5886/3-1 and WE 5886/4-1.

Introduction

This book covers a broad range of research directions connected to variational methods for nonlinear geometric data, the latter being understood as the solution of application-driven problems with nonlinear data via the computation of the minimizer of an energy functional.

The modeling of real-world processes via energy minimization is a classical field in mathematics and is inspired by nature, where stable configurations or the results of certain processes are often those that minimize an energy. Variational methods are well-established tools in mathematics as well as in other scientific disciplines such as computer science, biomedical engineering, or physics. They involve techniques from various branches of mathematics—in particular, analysis, statistics, modeling, optimization, and numerical mathematics—and yield state-of-the-art results in many application-driven problems in science and engineering.

The vast majority of mathematical research in the context of variational methods, however, deals with data and techniques from linear spaces. While this makes sense in view of the fact that the development, analysis, and numerical realization of variational models are challenging already in such a setting, it is important to note the extension of variational methods to incorporate techniques and data from nonlinear spaces is both promising and natural for a broad class of applications.

Indeed, quite often the data of interest, such as diffusion tensors, shapes or fibers of biological objects, functional data, or low-rank matrices, have a natural, nonlinear geometry that needs to be respected when processing or reconstructing such data. In addition, incorporating a nonlinear geometric viewpoint in variational techniques even when dealing with originally linear data, such as in the context of image regularization or classification tasks, can open new perspectives and yield improvements.

Needless to say, extending a well-established field to new application scenarios also requires a nontrivial extension of methodological concepts such as analysis, statistical methods, and optimization techniques.

Now since, as mentioned above, variational methods for linear data already both cover a broad field of applications and incorporate a great variety of mathematical techniques, their extensions towards respecting nonlinear geometries do so even more. This results in a variety of different research directions and application scenarios for variational methods with nonlinear geometric data which are rather spread out over different mathematical communities.

The purpose of this book is to both stimulate exchange between these different communities and to provide a comprehensive reference work for newcomers in the field. To this end, it contains contributions of leading experts for different aspects of variational methods for nonlinear geometric data who provide introductory overviews as well as discussions of recent research results in their fields.

Addressing the different research directions in extending variational methods to respect nonlinear geometry, the book contains different parts. While with this we aim to improve readability by introducing some structure, it is important to mention that the division of the chapters into different parts, and in particular their ordering, is to some extent arbitrary. We emphasize that strong connections between the different parts exist.

In the first part of the book, we deal with different techniques within the area of variational methods to allow processing of data in nonlinear spaces. In this respect, suitable discretizations are necessary.

Chapter 1 addresses this issue by introducing geometric finite elements. Geometric finite elements generalize the idea of finite element methods to maps that take values in Riemannian manifolds. The chapter defines suitable discrete manifold-valued function spaces and deals with the corresponding discretization of smooth variational problems. The resulting discrete problems are solved using methods of manifold optimization. Applications considered are harmonic map problems as well as problems of shell mechanics and micromagnetics.

Another important ingredient for variational methods, in particular, in the context of inverse problems or image processing, is the incorporation of suitable regularization approaches. Classical applications are for instance image denoising and reconstruction, segmentation and labeling tasks as well as joint reconstruction and post-processing. Dealing with nonlinear data spaces, important tasks in this context are the introduction, the analysis as well as the numerical realization of regularization functionals.

Chapter 2 deals with non-smooth variational regularization approaches for nonlinear data. Motivated by their success in the linear context, the chapter considers first- and higher-order derivative/difference-based regularizers as well as wavelet-based sparse regularization approaches and provides an overview of existing approaches. Furthermore, inverse problems in the context of manifold-valued data as well as different techniques for the numerical solution of the corresponding minimization problems are discussed. Applications to denoising, segmentation, and reconstruction from indirect measurements are given.

Chapter 3 also deals with variational regularization pursuing a functional lifting approach for images with values in a manifold. The approach transforms an original variational problem for manifold-valued data into a higher dimensional vector space problem and thereby solves a corresponding relaxed convex problem using the tools of convex optimization in a vector space. Applications are total variation regularization and generalizations as well as segmentation and optical flow estimation.

Often data are not sparse with respect to the canonical basis but with respect to some suitable dictionary such as a wavelet frame or, more generally, a corresponding multiscale transform. For data in linear space, this has led to variational regularization methods such as wavelet sparse regularization which imposes a sparsity prior with respect to a corresponding expansion/transform. Chapter 4 deals with multiscale transforms for geometric data as well as with refinement procedures, in particular subdivision schemes, for geometric data which are needed to define these transforms. In particular, the chapter discusses the differential geometric building blocks and focuses on respecting the nature and symmetries of the data in metric spaces, Riemannian manifolds, and groups. Results on the current state of the art regarding convergence and smoothness are presented.

Besides image processing, variational methods also play an important role in the processing of geometry data. While consistent discrete notions of curvatures and differential operators have been widely studied, the question of the convergence of the resulting minimizers to their smooth counterparts is still open for various geometric functionals. Building on tools from variational analysis, and in particular using the notion of Kuratowski convergence (as opposed to Γ or epi-convergence), Chapter 5 provides a general framework to deal with the convergence of minimizers of discrete geometric functionals to their continuous counterparts. Applications to minimal surfaces and Euler elasticae are provided.

The second part of the book deals with approaches that use nonlinear geometry as a tool to address challenging, variational problems. In this context, Chapter 6 gives an introduction to variational methods recently developed in fluid-structure interaction focusing on the dynamics and nonlinear geometry of flexible tubes conveying fluids. Biomedical and industrial applications are in connection with arterial flows as well as in connection with high-speed motion of gas in flexible pipes. Besides the discussion and analysis of variational models, the chapter also considers the numerical realization including appropriate discretizations.

Furthermore, Chapter 7 introduces the geometric concept of roto-translation space, which appears in the representation of object contours from 2D and 3D images in a higher-dimensional space. This lifting technique allows obtaining convex representations of curvature-aware regularization functionals in image processing. Chapter 7 provides the basic concept of this approach, shows some analytical properties and connections of different, existing functionals, and discusses numerical discretization techniques.

Having a different focus, Chapter 8 introduces the framework of assignment flows, which employs in particular methods from information (differential) geometry to model data labeling and related machine learning tasks. Assignment flows provide an adaptive time-variant extension of established discrete graphical models and a basis for the design and better mathematical understanding of hierarchical networks. Chapter 8 provides an overview of recent research and future developments in this field, dealing in particular with the supervised and the unsupervised situation and providing mathematical background on the corresponding dynamical systems as well as geometric integration approaches for their numerical solution.

Last but not least, in the context of large-scale and high-dimensional problems, often low-rank matrix and tensor techniques are used. Chapter 9 deals with this topic and provides numerical methods that explicitly make use of the geometry of rank-constrained matrix and tensor spaces. It illustrates the benefit of these techniques with two topics. One is classical optimization problems such as matrix or tensor completion, where the explicit incorporation of geometry allows to employ Riemannian optimization methods to obtain efficient numerical methods. The other one is ordinary differential equations defined on matrix or tensor spaces, where the authors show how corresponding solutions can be approximated by a dynamical low-rank principle.

In the third part of the book, we consider statistical methods in the context of nonlinear geometric data. There are close links between statistics and variational methods: On the one hand, many variational models are based on statistical considerations. On the other hand, statistical objects in the context of manifolds are themselves often defined as energy minimizers (like most of the statistical objects defined in the following two chapters).

Chapter 10 deals with statistics on manifolds with an emphasis on principal component analysis (PCA). In the linear space context, PCA is a commonly used dimension reduction method, and its generalization to nonlinear data spaces is an important topic for dimension reduction of non-Euclidean data. Starting from the intrinsic mean, the chapter discusses various generalizations to the manifold setting. In particular, Procrustes analysis, principal geodesic analysis, geodesic PCA as well as principal nested spheres, horizontal PCA, barycentric subspace analysis, and backward nested descriptors analysis (BNDA) are considered. The chapter reviews the current state of the art of the corresponding asymptotic statistical theory and discusses open challenges.

Chapter 11 provides a historical review and discussion of intrinsic means in a manifold (which is the best zero-dimensional summary statistics of data). It then proceeds to define the concept of barycentric subspace analysis (BSA). BSA yields flags of “subspaces” generalizing the sequences of nested linear subspaces appearing in the classical PCA. Barycentric subspaces are the locus of weighted means of a finite set of fixed reference points with the varying weights defining the space in the manifold. The corresponding “subspaces” of the BSA are found by minimizing corresponding energies. Here, three variants are discussed generalizing different desired properties from the Euclidean setup. In particular, minimizing the so-called accumulated unexplained variance criterion yields nested spaces while leading to the PCA decomposition in Euclidean spaces when considering a Euclidean setup. The potential of the method is illustrated by applying it to cardiac imaging.

Last but not least, Chapter 12 deals with deep variational inference (VI). After reviewing VI as a fast alternative to Markov Chain Monte Carlo methods, stochastic VI, black-box VI, as well as amortized VI leading to the variational auto-encoder are discussed. The latter involves deep neural networks and graphical models.

Finally, generative flows, the latent space manifold, and the Riemannian geometry of generative models are explored.

In part four of the book, we summarize chapters that deal with geometric methods for the analysis of data from particular nonlinear spaces, such as shapes, curves, and trajectories. Aiming at a statistical analysis of geometric data that is invariant with respect to some particular transformations or reparameterizations, Chapter 13 introduces the framework of square root velocity functions for the representation of scalar functions, curves, and shapes. Building on that, the chapter discusses registration approaches that allow to compute correspondences between different geometric objects. This is then exploited to compute statistics such as principal modes.

On top of that, Chapter 14 extends this framework for a statistical analysis of multi-modality data. There, the authors present a framework for the joint analysis of multi-modality data such as trajectories of functions or tensors. That is, building on representations such as square root velocity functions, they define a metric that jointly registers trajectories and particular data that is defined on such trajectories. With that, they again want to obtain certain invariances and applications to trajectories of functions in the context of functional magnetic resonance imaging, and trajectories of tensors in the context of diffusion tensor imaging are shown.

With similar applications in the background, Chapter 15 deals with geometric methods for topological representations. That is, the authors provide an introduction to tools from topological data analysis to achieve invariant representations and corresponding metrics for diverse data classes such as image, shape, or time series data. They provide several existing approaches, discuss advantages and disadvantages of those, and provide exemplary applications.

Chapter 16 deals with combining machine learning techniques with modeling based on geometric considerations for extracting features from geometric data appearing in computer vision, computer graphics, and image processing. Two scenarios are considered. First, a scheme to discover geometric invariants of planar curves from data in a learning framework is discussed. Here, the invariants are modeled using neural networks. The second scenario considers a reverse setup by imputing principled geometric invariants like geometric moments into standard learning architectures. This enables a significant boost in performance and also provides a geometric insight into the learning process.

Chapter 17 deals with sub-Riemannian methods to study shapes, with a special focus on shape spaces defined as homogeneous spaces under the action of diffeomorphisms. The chapter provides a review of recent developments. It considers sub-Riemannian methods based on control points and their generalization to deformation modules. Furthermore, it considers implicit constraints on geodesic evolution together with the corresponding computational challenges. Numerical examples are provided as illustrations.

The fifth part of the book contains chapters that deal with optimization approaches on manifolds and with particular numerical problems for manifold-valued data. This is broadly relevant for energy minimization-based techniques in general, and in particular also for many of the research directions discussed in the other chapters of the book.

Chapter 18 deals with first order methods for optimization on Riemannian manifolds. In particular, algorithms based on the proximal point method as well as algorithms based on gradient and subgradient methods in the Riemannian setting are defined and analyzed. In particular, an asymptotic and iteration-complexity analysis for the proximal point method on Hadamard manifolds is presented.

Chapter 19 deals with recent advances in stochastic optimization on Riemannian manifolds. Starting from stochastic gradient descend, various methods are discussed. In particular, variance reducing algorithms such as the stochastic recursive gradient algorithm or the stochastic variance reduced gradient algorithm are considered. For the presented schemes, a summary of the presently available convergence results is given. Applications for machine learning problems related to Gaussian mixtures, PCA, and Wasserstein barycenters are also discussed.

In various places in the book, averaging in a manifold appears as a central tool. Chapter 20 considers the particularly interesting problem of averaging in the manifold of positive definite matrices (appearing in medical imaging, radar signal processing, and mechanics) from a numerical optimization point of view. In particular, different averages, all defined in terms of minimization problems, together with their numerical realization are discussed.

In connection with geometric data, differential geometric aspects naturally enter the scene. In particular, differential geometric considerations for efficiently computing the concrete quantities involved in the numerical problems are of central importance (and are often not treated in textbooks on differential geometry due to a different focus). Chapter 21 deals with such aspects focusing on spheres, Graßmannians, and special orthogonal groups. The employed methodology is based on rolling motions of those manifolds considered as rigid bodies, subject to holonomic as well as non-holonomic constraints. As application, interpolation problems on Riemannian manifolds are considered.

The sixth part of the book deals with particular, relevant applications of variational methods for geometric data. While partially already covered in the previous chapter, here the focus is on the applications. Chapter 22 deals with applications of variational regularization methods for nonlinear data in medical imaging applications. More precisely, the compounding of ultrasound images, the tracking of nerve fibers, and the segmentation of aortas are considered. In particular, the impact of the regularization techniques for manifold valued data for the final result of processing pipelines is demonstrated.

Chapter 23 considers the Riemannian geometry of facial expression and action recognition. Building on landmark representations, it introduces geometric representations of landmarks in a static and time-series context. Again, a driving factor here is to obtain certain invariances with respect to basic transformations such as a change of the viewpoint or different parameterizations. Building on pseudo geodesics and the concept of closeness for the representation of a set of landmarks in the manifold of positive semidefinite matrices with fixed rank, the chapter considers applications in facial expression and action recognition and provides a comparison of the discussed to previous, existing approaches.

Chapter 24 discusses biomedical applications of geometric functional data analysis. It reviews parameterization-invariant Riemannian metrics and corresponding simplifying square root transforms (see also Chapters 13 and 14). The data space of interest here is probability density functions, amplitude and phase components in functional data, and shapes of curves and surfaces. The chapter provides general recipes for computing the sample mean, covariance, and performing principal component analysis (PCA). The considered applications are the assessment of tumor texture variability, the segmentation and clustering of electrocardiogram signals, the comparison and summarization of planar shapes of tumors as well as the simulation of endometrial tissue shapes, which can be used, for instance, for the validation of various image processing algorithms.

Contents

Part I Processing Geometric Data

Part II Geometry as a Tool

Part III Statistical Methods and Non-linear Geometry

Part IV Shapes Spaces and the Analysis of Geometric Data

Part V Optimization Algorithms and Numerical Methods

Part VI Applications

Contributors

Pierre-Antoine Absil Department of Mathematical Engineering, ICTEAM Institute, Université catholique de Louvain, Louvain-la-Neuve, Belgium

Juan-Carlos Alvarez Paiva University of Lille, CNRS-UMR-8524, Lille, France
Painlevé Laboratory, Villeneuve-d'Ascq, France

Maximilian Baust TU Munich, Chair of Computer Aided Medical Procedures, Garching, Germany

Karthik Bharath School of Mathematical Sciences, University of Nottingham, Nottingham, UK

Ulrich Böttcher Westfälische Wilhelms-Universität Münster, Applied Mathematics Münster, Münster, Germany

Daniel Cremers Technical University Munich, Garching, Germany

Mohamed Daoudi IMT Lille-Douai, University of Lille, CNRS, UMR 9189, CRIStAL, Lille, France

Zhaohua Ding Vanderbilt University, Nashville, TN, USA

Iddo Drori Columbia University, New York, NY, USA

Benjamin Eltzner Felix-Bernstein-Institute for Mathematical Statistics in the Biosciences, University of Göttingen, Göttingen, Germany

Orizon P. Ferreira Universidade Federal de Goiás, Institute of Mathematics Statistics, Goiânia, GO, Brazil

Kyle A. Gallivan Department of Mathematics, Florida State University, Tallahassee, FL, USA,

François Gay-Balmaz CNRS & Ecole Normale Supérieure, Paris, France

Barbara Gris Laboratoire Jacques-Louis Lions, Sorbonne Université, Paris, France

Mengmeng Guo Texas Tech University, Lubbock, TX, USA

Xiaoyang Guo Florida State University, Tallahassee, FL, USA

Hanne Hardering Technische Universität Dresden, Institut für Numerische Mathematik, Dresden, Germany

Martin Holler Institute of Mathematics and Scientific Computing, University of Graz, Graz, Austria

Reshad Hosseini School of ECE, College of Engineering, University of Tehran, Tehran, Iran
School of Computer Science, Institute of Research in Fundamental Sciences (IPM), Tehran, Iran

Wen Huang School of Mathematical Sciences, Xiamen University, Xiamen City, People's Republic of China

Stephan Huckemann Felix-Bernstein-Institute for Mathematical Statistics in the Biosciences, University of Göttingen, Göttingen, Germany

Knut Hüper Institute of Mathematics, Julius-Maximilians-Universität Würzburg, Würzburg, Germany

Mor Joseph-Rivlin Faculty of Electrical Engineering, Technion, Haifa, Israel

Anis Kacem IMT Lille-Douai, University of Lille, CNRS, UMR 9189, CRIStAL, Lille, France

Krzysztof A. Krakowski Wydział Matematyczno-Przyrodniczy, Uniwersytet Kardynała Stefana Wyszyńskiego, Warsaw, Poland

Ron Kimmel Faculty of Computer Science, Technion, Haifa, Israel

Sebastian Kurtek Department of Statistics, The Ohio State University, Columbus, OH, USA

Jan Lellmann Institute of Mathematics and Image Computing, University of Lübeck, Lübeck, Germany

Maurício S. Louzeiro TU Chemnitz, Fakultät für Mathematik, Chemnitz, Germany

James Matuk Department of Statistics, The Ohio State University, Columbus, OH, USA

Shariq Mohammed Department of Computational Medicine and Bioinformatics, University of Michigan, Ann Arbor, MI, USA

Gautam Pai Faculty of Computer Science, Technion, Haifa, Israel

Xavier Pennec Université Côte d'Azur, Nice, France
Inria, Sophia-Antipolis, France

Leandro F. Prudente Universidade Federal de Goiás, Institute of Mathematics Statistics, Goiânia, GO, Brazil

Vakhtang Putkaradze University of Alberta, Edmonton, AB, Canada
ATCO SpaceLab, SW Calgary, AB, Canada

Karthikeyan Natesan Ramamurthy IBM Research, Yorktown Heights, NY, USA

Oliver Sander Technische Universität Dresden, Institut für Numerische Mathematik, Dresden, Germany

Christoph Schnörr Institute of Applied Mathematics, Heidelberg University, Heidelberg, Germany

Henrik Schumacher RWTH Aachen University, Institute for Mathematics, Aachen, Germany

Fátima Silva Leite Department of Mathematics and Institute of Systems and Robotics, University of Coimbra, Coimbra, Portugal

Nir Sochen Department of Applied Mathematics, Tel-Aviv University, Tel Aviv-Yafo, Israel

Anirudh Som School of Electrical, Computer and Energy Engineering, School of Arts, Media and Engineering, Arizona State University, Tempe, AZ, USA

Suvrit Sra Massachusetts Institute of Technology, Cambridge, MA, USA

Anuj Srivastava Florida State University, Tallahassee, FL, USA

Evgeny Strekalovskiy Technical University Munich, Garching, Germany
Google Germany GmbH, Munich, Germany

Jingyong Su Harbin Institute of Technology (Shenzhen), Shenzhen, China

Alain Trouvé Centre de Mathématiques et leurs Applications, ENS Paris-Saclay, Cachan, France

Pavan Turaga School of Electrical, Computer and Energy Engineering, School of Arts, Media and Engineering, Arizona State University, Tempe, AZ, USA

André Uschmajew Max Planck Institute for Mathematics in the Sciences, Leipzig, Germany

Bart Vandereycken Section of Mathematics, University of Geneva, Geneva, Switzerland

Thomas Vogt Institute of Mathematics and Image Computing, University of Lübeck, Lübeck, Germany

Johannes Wallner TU Graz, Graz, Austria

Max Wardetzky Institute of Numerical and Applied Mathematics, University of Göttingen, Göttingen, Germany

Andreas Weinmann Department of Mathematics and Natural Sciences, Hochschule Darmstadt, Darmstadt, Germany

Benedikt Wirth Westfälische Wilhelms-Universität Münster, Applied Mathematics Münster, Münster, Germany

Zhipeng Yang Sichuan University, Chengdu, China

Laurent Younes Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, MD, USA

Xinru Yuan Department of Mathematics, Florida State University, Tallahassee, FL, USA

About the Editors

Philipp Grohs was born on July 7, 1981, in Austria and has been a professor at the University of Vienna since 2016. In 2019, he also became a group leader at RICAM, the Johann Radon Institute for Computational and Applied Mathematics in the Austrian Academy of Sciences in Linz. After studying, completing his doctorate and working as a postdoc at TU Wien, Grohs transferred to King Abdullah University of Science and Technology in Thuwal, Saudi Arabia, and then to ETH Zürich, Switzerland, where he was an assistant professor from 2011 to 2016. Grohs was awarded the ETH Zurich Latsis Prize in 2014. In 2020 he was selected for an Alexander-von-Humboldt-Professorship award, the highest endowed research prize in Germany. He is a member of the board of the Austrian Mathematical Society, a member of IEEE Information Theory Society, and on the editorial boards of various specialist journals.

Martin Holler was born on May 21, 1986, in Austria. He received his MSc (2010) and his PhD (2013) with a "promotio sub auspiciis praesidentis rei publicae" in Mathematics from the University of Graz. After research stays at the University of Cambridge, UK, and the Ecole Polytechnique, Paris, he currently holds a University Assistant position at the Institute of Mathematics and Scientific Computing of the University of Graz. His research interests include inverse problems and mathematical image processing, in particular the development and analysis of mathematical models in this context as well as applications in biomedical imaging, image compression, and beyond.

Andreas Weinmann was born on July 18, 1979, in Augsburg, Germany. He studied mathematics with minor in computer science at TU Munich and received his diploma degree in mathematics and computer science from TU Munich in 2006 (with highest distinction). He was assistant at the Institute of Geometry, TU Graz. He obtained his Ph.D. degree from TU Graz in 2010 (with highest distinction). Then he worked as a researcher at Helmholtz Center Munich and TU Munich. Since 2015 he holds a position as Professor of Mathematics and Image Processing at Hochschule Darmstadt. He received his habilitation in 2018 from the University of Osnabrück. Andreas's research interests include applied analysis, in particular variational methods, nonlinear geometric data spaces, inverse problems as well as computer vision, signal and image processing, and imaging applications, in particular magnetic particle imaging.

Part I
Processing Geometric Data

Chapter 1
Geometric Finite Elements

Hanne Hardering and Oliver Sander

Contents

H. Hardering (✉) · O. Sander
Technische Universität Dresden, Institut für Numerische Mathematik, Dresden, Germany
e-mail: hanne.hardering@tu-dresden.de; oliver.sander@tu-dresden.de

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_1

Abstract Geometric finite elements (GFE) generalize the idea of Galerkin methods to variational problems for unknowns that map into nonlinear spaces. In particular, GFE methods introduce proper discrete function spaces that are conforming in the sense that values of geometric finite element functions are in the codomain manifold $\mathcal{M}$ at any point. Several types of such spaces have been constructed, and some are even completely intrinsic, i.e., they can be defined without any surrounding space. GFE spaces enable the elegant numerical treatment of variational problems posed in Sobolev spaces with nonlinear codomain space. Indeed, as GFE spaces are geometrically conforming, such variational problems have natural formulations in GFE spaces. These correspond to the discrete formulations of classical finite element methods. Also, the canonical projection onto the discrete maps commutes with the differential for a suitable notion of the tangent bundle as a manifold, and we therefore also obtain natural weak formulations. Rigorous results exist that show the optimal behavior of the a priori L^2 and H^1 errors under reasonable smoothness assumptions. Although the discrete function spaces are no vector spaces, their elements can nevertheless be described by sets of coefficients, which live in the codomain manifold. Variational discrete problems can then be reformulated as algebraic minimization problems on the set of coefficients. These algebraic problems can be solved by established methods of manifold optimization. This text will explain the construction of several types of GFE spaces, discuss the corresponding test function spaces, and sketch the a priori error theory. It will also show computations of the harmonic maps problem, and of two example problems from nanomagnetics and plate mechanics.

1.1 Introduction

A number of interesting physical phenomena is described by partial differential equations for functions that map into a nonlinear space $\mathcal{M}$. Examples are equations used to describe liquid-crystal dynamics [23], models of the microscopic behavior of ferromagnetic materials [48], finite-strain plasticity [51], image processing [14, 66], and non-standard models of continuum mechanics [54]. In these examples, the nonlinear degrees of freedom frequently represent a direction or an orientation, and the target space $\mathcal{M}$ is therefore the unit sphere S^2, the projective plane, or the special orthogonal group $\mathrm{SO}(3)$. The processing of diffusion tensor images involves functions with values in the symmetric positive definite matrices [9]. Numerical simulations of general relativity deal with fields of metric tensors with a given signature [12], and in particle physics, sigma models use fields in a variety of different spaces [45]. At the same time, maps into nonlinear spaces have also received much interest from the mathematical community, see, e.g., the works on geometric wave equations [60] or the large body of work on harmonic maps [41]. In that latter case in particular, there is no restriction to particular target spaces.

For a long time, numerical analysis has struggled with the non-Euclidean structure of these problems. Indeed, as sets of functions with nonlinear codomain cannot form vector spaces, standard finite element methods are ruled out immediately, because the very notion of piecewise polynomial functions has no meaning in nonlinear spaces. Similarly, the wealth of tools from linear functional analysis traditionally used to analyze finite element methods is not available if the codomain is nonlinear.

Various approaches have been proposed in the literature to overcome these difficulties. Without trying to be comprehensive we mention the nonconforming methods of Bartels and Prohl [11], and Alouges and Jaisson [6], finite difference methods used in image processing [66] and the simulation of harmonic maps [5], and various ad hoc methods used in computational mechanics [61, 67]. All these methods have in common that they avoid the explicit construction of discrete conforming function spaces, and that theoretical results concerning their approximation properties are either suboptimal or lacking completely.

The geometric finite element method, in contrast, strives to follow the idea of traditional Galerkin methods more closely. It is based on explicit constructions of discrete function spaces that generalize the piecewise polynomials of standard finite element methods, but which are geometrically conforming in the sense of mapping into the codomain space $\mathcal{M}$ everywhere. These spaces can then be assessed regarding their approximation properties, and a priori bounds of the discretization error as a function of the element size and solution smoothness can be shown. The error rates thus obtained correspond to what is expected from the linear theory, and this optimality is also confirmed by numerical tests.

In this exposition, the target space $\mathcal{M}$ is mostly assumed to have the structure of a Riemannian manifold. However, some of the constructions of Sect. 1.2 also work in more general settings such as pseudo-Riemannian manifolds, affine manifolds, and general metric spaces. One subject of ongoing research is the generalization of existing approximation results to these more general settings.

1.2 Constructions of geometric Finite Elements

The finite element method solves partial differential equations (PDEs) by approximating Sobolev functions by piecewise polynomials with certain continuity properties. For this, the domain $\Omega \subset \mathbb{R}^d$ is covered with a grid $\mathcal{G}$, i.e., a partition of essentially nonoverlapping (deformations of) convex polytopes T_h, the so-called elements. In this chapter, grids will always be conforming, i.e., the intersection of two grid elements is a common face or empty. The spaces of polynomials on the elements are typically described by giving particular bases, and the basis vectors are called shape functions [20].

The main ingredient of geometric finite element methods is a way to generalize spaces of polynomials on one grid element to manifold-valued functions. Under certain conditions on the behavior of these functions on the element boundary, the generalized polynomials can then be combined to globally continuous functions. This chapter describes four ways to construct geometric finite elements.

In the following, if not specified otherwise, $\mathcal{M}$ will denote a smooth n-dimensional Riemannian manifold with metric g, tangent bundle $T\mathcal{M}$, and Levi–Civita connection ∇. All derivatives of vector fields will be covariant derivatives with respect to this connection; in particular, for a vector field $V : \Omega \to u^{-1}T\mathcal{M}$ along a map $u \in C(\Omega, \mathcal{M})$, we denote by $\nabla^k V$ all k-th order covariant derivatives along u. For $k = 1$, in coordinates around $u(x)$, these are given by

$$\nabla V^l_\alpha(x) := \frac{\partial}{\partial x_\alpha} V^l(x) + \Gamma^l_{ij}(u(x)) V^i(x) \frac{\partial u^j}{\partial x_\alpha}(x),$$

where $i, j, l = 1, \dots, n$, $\alpha = 1, \dots, d$, the Γ^l_{ij} denote the Christoffel symbols of ∇, and we sum over repeated indices.

We assume that $\mathcal{M}$ is complete as a metric space with distance $d(\cdot, \cdot)$, and has bounds on the Riemannian curvature tensor Rm and its derivatives, i.e.,

$$\|\mathrm{Rm}\|_{L^\infty} := \sup_{p\in\mathcal{M}} \sup_{\substack{V_i\in T_p\mathcal{M}\\ i=1,\dots,4}} \frac{|\mathrm{Rm}(p)(V_1, V_2, V_3, V_4)|}{\|V_1\|_{g(p)}\|V_2\|_{g(p)}\|V_3\|_{g(p)}\|V_4\|_{g(p)}} \le C,$$

$$\|\nabla\mathrm{Rm}\|_{L^\infty} := \sup_{p\in\mathcal{M}} \sup_{\substack{V_i\in T_p\mathcal{M}\\ i=1,\dots,5}} \frac{|\nabla\mathrm{Rm}(p)(V_1, V_2, V_3, V_4, V_5)|}{\|V_1\|_{g(p)}\|V_2\|_{g(p)}\|V_3\|_{g(p)}\|V_4\|_{g(p)}\|V_5\|_{g(p)}} \le C.$$

We will denote the exponential map by $\exp_p : T_p\mathcal{M} \to \mathcal{M}$. Its inverse, where defined, will be denoted by $\log_p : \mathcal{M} \to T_p\mathcal{M}$, $\log_p q := \exp_p^{-1} q$. The differential of the logarithm $\log_p$ is denoted by

$$d\log_p q : \; T_q\mathcal{M} \to T_p\mathcal{M} \qquad d\log_p q(V) := \frac{d}{dt}\bigg|_{t=0} \log_p c_{(q,V)}(t),$$

where $p, q \in \mathcal{M}$ are such that $\log_{(\cdot)}(\cdot)$ is defined in a neighborhood of these points, and $c_{(p,V)}(\cdot)$ denotes a curve with $c_{(p,V)}(0) = p$ and $\dot{c}_{(p,V)}(0) = V$. The injectivity radius of $\mathcal{M}$ is

$$\mathrm{inj}(\mathcal{M}) := \inf_{p\in\mathcal{M}} \sup_{\rho>0} \left\{\exp_p \text{ is defined on } B_\rho(0) \subset T_p\mathcal{M} \text{ and injective}\right\}.$$

1.2.1 Projection-Based Finite Elements

Our first construction of geometric finite elements uses an embedding space of the manifold $\mathcal{M}$. It has been described in detail in [33]. Independently, Gawlik and Leok [28] investigated the case $d = 1$, $\mathcal{M} = \mathrm{SO}(3)$.

1.2.1.1 Construction

Let T_h be an element of a finite element grid $\mathcal{G}$ for Ω. On T_h we consider a scalar-valued r-th order Lagrange basis $\varphi_1, \dots, \varphi_m \colon T_h \to \mathbb{R}$ with associated Lagrange points $\xi_1, \dots, \xi_m \in T_h$. Suppose that $\mathcal{M}$ is embedded smoothly into a Euclidean space $\mathbb{R}^N$ by a map $\iota : \mathcal{M} \to \mathbb{R}^N$.

We will define the space $S_h^{\text{proj}}(T_h, \mathcal{M})$ of projection-based finite elements on T_h as the image of an interpolation rule. Let $v = (v_1, \dots, v_m) \in \mathcal{M}^m$ be a set of values associated to the Lagrange points. First we consider the canonical Lagrange interpolation operator $I_{\mathbb{R}^N}$ of values embedded into $\mathbb{R}^N$

$$I_{\mathbb{R}^N} \ : \mathcal{M}^m \times T_h \to \mathbb{R}^N$$

$$I_{\mathbb{R}^N}(v, \xi) := \sum_{i=1}^{m} \iota(v_i)\varphi_i(\xi).$$

Even though the v_i are elements of $\mathcal{M}$, the values of $I_{\mathbb{R}^N}(v, \cdot)$ will in general not be in $\mathcal{M}$ away from the Lagrange points ξ_i. To get $\mathcal{M}$-valued functions we compose $I_{\mathbb{R}^N}$ pointwise with the closest-point projection

$$P : \mathbb{R}^N \to \mathcal{M}, \qquad P(q) := \operatorname*{arg\,min}_{p \in \mathcal{M}} \|\iota(p) - q\|_{\mathbb{R}^N},$$

where $\| \cdot \|_{\mathbb{R}^N}$ denotes the Euclidean distance. While the closest-point projection is usually not well defined for all $q \in \mathbb{R}^N$, if $\mathcal{M}$ is regular enough it is well defined in a neighborhood $U_\delta \subset \mathbb{R}^N$ of $\mathcal{M}$ [1]. We then define $\mathcal{M}$-valued projected interpolation by composition of $I_{\mathbb{R}^N}$ and P, restricted to sets of input values for which the projection is well defined.

Definition 1.1 Let $T_h \subset \mathbb{R}^d$ be a grid element. Let $\varphi_1, \dots, \varphi_m$ be a set of r-th order scalar Lagrangian shape functions, and let $v = (v_1, \dots, v_m) \in M^m$ be values at the corresponding Lagrange nodes. We call

$$I^{\text{proj}} \ : \ \mathcal{M}^m \times T_h \to \mathcal{M}$$

$$I^{\text{proj}}(v, \xi) := P\Big(\sum_{i=1}^{m} \iota(v_i)\varphi_i(\xi)\Big)$$

(where defined) r-th order projection-based interpolation on $\mathcal{M}$.

Well-posedness is guaranteed if the values v_i are close to each other on the manifold, such that $I_{\mathbb{R}^N}(v, \xi) \in U_\delta$ for all $\xi \in T_h$.

Lemma 1.2 *There exists a* $\rho > 0$ *such that if*

$$\operatorname{diam}_{\mathcal{M}} \big\{ v_i \ : \ i = 1, \dots, m \big\} < \rho,$$

the projection-based interpolation $I^{proj}(v, \xi)$ *will well defined for all* $\xi \in T_h$.

Such a requirement of locality is common to all geometric finite element constructions. Here the closeness parameter ρ depends on the Lagrangian shape functions and the extrinsic curvature of $\mathcal{M}$.

Projection-based finite elements on T_h are defined as the range of this interpolation operator.

Definition 1.3 Let $T_h \subset \mathbb{R}^d$, $\mathcal{M} \subset \mathbb{R}^N$ an embedded submanifold, and $P\colon U_\delta \subset \mathbb{R}^N \to \mathcal{M}$ the closest-point projection. For a given set of Lagrange basis functions $\varphi_1, \dots, \varphi_m\colon T_h \to \mathbb{R}$ we define

$$S_h^{\text{proj}}(T_h, \mathcal{M}) := \Big\{ v_h \in C(T_h, \mathcal{M}) \;:\; \exists\, v \in \mathcal{M}^m \text{ such that } v_h = I^{\text{proj}}(v, \cdot) \Big\}.$$

The element-wise definition can be used to construct projection-based finite element spaces on an entire conforming grid $\mathcal{G}$. We define a global geometric Lagrange finite element space by joining projected polynomials on each element, and requesting global continuity

$$S_h^{\text{proj}}(\Omega, \mathcal{M}) := \Big\{ v_h \in C(\Omega, \mathcal{M}) \;:\; v_h|_{T_h} \in S_h^{\text{proj}}(T_h, \mathcal{M}) \quad \forall T_h \in \mathcal{G} \Big\}.$$

As the operator I^{proj} only uses the values at the Lagrange points, we have the equivalent definition

$$S_h^{\text{proj}}(\Omega, \mathcal{M}) := \Big\{ v_h \in C(\Omega, \mathcal{M}) : \exists (v_i)_{i \in I} \subset \mathcal{M} \text{ such that } v_h = P\Big(\textstyle\sum_{i \in I} v_i \varphi_i(\cdot)\Big) \Big\}, \tag{1.1}$$

where $(v_i)_{i \in I}$ and $(\varphi_i)_{i \in I}$ are now global sets of coefficients and Lagrange basis functions, respectively. Note that the definition of Discontinuous-Galerkin-type spaces without continuity constraints is straightforward.

1.2.1.2 Properties

We list a few properties of projection-based finite elements. Firstly, projection-based finite element functions are $W^{1,p}$-conforming, if the projection operator P is sufficiently smooth:

Lemma 1.4 ([33, Sec. 1.1]) *If the point-wise operator norm of the differential* $dP(x) : \mathbb{R}^N \to T_{P(x)}\mathcal{M}$ *is globally bounded, then*

$$S_h^{\text{proj}}(\Omega, \mathcal{M}) \subset W^{1,p}(\Omega, \mathcal{M})$$

for all $p \in [1, \infty]$.

Different ways to define the $\mathcal{M}$-valued Sobolev spaces are discussed in Sect. 1.4.1. The required smoothness of P holds at least in a neighborhood of $\mathcal{M}$, if $\mathcal{M}$ is embedded smoothly in $\mathbb{R}^N$ [49]. The proof of Lemma 1.4 follows from the chain rule, and the piecewise smoothness of the Lagrange basis.

Next, note that for a fixed grid $\mathcal{G}$, finite element spaces of different polynomial orders are nested.

Lemma 1.5 *Let $S^{\text{proj}}_{h,r_1}(\Omega, \mathcal{M})$ and $S^{\text{proj}}_{h,r_2}(\Omega, \mathcal{M})$ be two projection-based finite element spaces with shape functions of orders r_1 and r_2, respectively. If $r_1 \leq r_2$ then*

$$S^{\text{proj}}_{h,r_1}(\Omega, \mathcal{M}) \subset S^{\text{proj}}_{h,r_2}(\Omega, \mathcal{M}).$$

Thirdly, as projection-based finite elements as defined here are based on standard Lagrange finite elements, they form affine families in the sense of Ciarlet [20]. This means that if $S^{\text{proj}}_h(T_h, \mathcal{M})$ is a given space of projection-based finite element functions on an element T_h, then the corresponding space on a different element $\widetilde{T}_h$ is equal to

$$S^{\text{proj}}_h(\widetilde{T}_h, \mathcal{M}) = \Big\{\tilde{v}_h \in C(\widetilde{T}_h, \mathcal{M}) \ : \ \tilde{v}_h = v_h \circ F^{-1}, v_h \in S^{\text{proj}}_h(T_h, \mathcal{M})\Big\},$$

where F is an affine map from T_h to $\widetilde{T}_h$. This means that all projection-based finite elements can be implemented as push-forwards of projection-based finite elements on a reference element T_{ref}.

Finally, we investigate a crucial symmetry property. It is desirable for any finite element discretization that the interpolation operator I mapping coefficients to discrete functions be equivariant under isometries of the codomain, i.e.,

$$QI(v, \cdot) = I(Qv, \cdot)$$

for elements Q of the isometry group of $\mathcal{M}$. For the standard case $\mathcal{M} = \mathbb{R}$, this means that

$$I(v, \cdot) + \alpha = I(v + \alpha, \cdot)$$

for all $\alpha \in \mathbb{R}$, which does indeed hold for interpolation by piecewise polynomials. In mechanics, where usually $\mathcal{M} = \mathbb{R}^3$ and the corresponding isometries are the special Euclidean group $\mathbb{R}^3 \rtimes \text{SO}(3)$, equivariance implies the desirable property that discretizations of frame-invariant problems are again frame-invariant. Unfortunately, for projection-based finite elements this equivariance only holds under special circumstances.

Lemma 1.6 ([33, Thm. 5]) *Let $Q \colon \mathcal{M} \to \mathcal{M}$ be an isometry that is extendable in the sense that there exists an isometry $\widetilde{Q} \colon \mathbb{R}^N \to \mathbb{R}^N$ with $\widetilde{Q}(p) = Q(p)$ for all $p \in \mathcal{M}$. Then*

$$QI^{\text{proj}}(v, \xi) = I^{\text{proj}}(Qv, \xi)$$

for all $\xi \in T_h$.

However, the assumption of the lemma is restrictive. Indeed, in order to be extendable, Q needs to be the restriction of a rigid body motion of $\mathbb{R}^N$. This means that projection-based interpolation has the desired equivariance only in some special situations, like the cases $\mathcal{M} = S^n$ and $\mathcal{M} = \mathrm{SO}(n)$. These are, on the other hand, relevant for applications.

1.2.2 Geodesic Finite Elements

Projection-based interpolation requires an embedding space. The following alternative construction uses weighted geodesic means, and is therefore purely intrinsic [56, 57].

1.2.2.1 Construction

Let again T_h be a grid element, and r-th order scalar Lagrange functions $\varphi_1, \dots, \varphi_m$ with Lagrange nodes $\xi_1, \dots, \xi_m$ as previously. The definition of geodesic finite element functions is based on the fact that for values v_i in a vector space, Lagrangian interpolation has a minimization formulation

$$\sum_{i=1}^{m} \varphi_i(\xi) v_i = \arg\min_{q \in \mathbb{R}} \sum_{i=1}^{m} \varphi_i(\xi) \| v_i - q \|^2,$$

which is typically interpreted as a weighted average. In a general metric space $(\mathcal{M}, d)$, the proper generalization of such averages is the Fréchet mean [18, 27]. For Riemannian (and Finsler) manifolds it is also known as the Riemannian center of mass or Karcher mean [44]. We will follow [4] and call the averages with respect to the geodesic distance weighted geodesic means.

Definition 1.7 (Weighted Geodesic Mean) A weighted geodesic mean of points $v_1, \dots, v_m \in \mathcal{M}$ for weights $w_1, \dots, w_m \in \mathbb{R}$ with $\sum_{i=1}^m w_i = 1$ is any point $q \in \mathcal{M}$ minimizing $q \mapsto \sum_{i=1}^m w_i \, \mathrm{d}(v_i, q)^2$ in $\mathcal{M}$. If the minimizer exists and is unique, we denote it by

$$\mathrm{av}\big[(v_1, \dots, v_m), (w_1, \dots, w_m)\big].$$

Geodesic finite elements are defined as geodesic means with Lagrangian shape functions as weights.

Definition 1.8 Let $T_h \subset \mathbb{R}^d$ be a grid element, $\varphi_1, \dots, \varphi_m$ a set of r-th order scalar Lagrangian shape functions, and let $v = (v_1, \dots, v_m) \in \mathcal{M}^m$ be values at the corresponding Lagrange nodes. We call

$$\begin{aligned} I^{\mathrm{geo}} &: \mathcal{M}^m \times T_h \to \mathcal{M} \\ I^{\mathrm{geo}}(v, \xi) &:= \mathrm{av}\big[(v_1, \dots, v_m), (\varphi_1(\xi), \dots, \varphi_m(\xi))\big] \end{aligned} \tag{1.2}$$

(where defined) r-th order geodesic interpolation on $\mathcal{M}$ (Fig. 1.1).

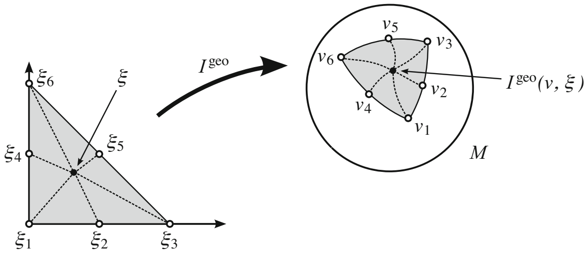


Fig. 1.1 Second-order geodesic interpolation from a triangle into a sphere

We define $S_h^{\text{geo}}(T_h, \mathcal{M})$ to be the space of all $\mathcal{M}$-valued functions on T_h that can be constructed in this way. Obviously, the construction (1.2) produces an interpolation function of the values v_i.

The definition of I^{geo} can be generalized in a number of ways. First of all, Definition 1.8 uses Lagrange shape functions mainly for simplicity. In a similar fashion, [17, 31] have used splines to achieve higher-order approximation, and [4] use Bernstein polynomials. Then, as the construction uses only metric information of $\mathcal{M}$, it is possible to define I^{geo} in the more general context of metric spaces. On the other hand, for a Riemannian manifold $\mathcal{M}$ it is known [43, Thm 5.6.1] that at minimizers $q \in \mathcal{M}$ of (1.2) we have

$$\sum_{i=1}^{m} \varphi_i(\xi) \log_q(v_i) = 0. \tag{1.3}$$

This can be used as the definition of geodesic interpolation for manifolds with an affine connection that have an exponential map but no compatible Riemannian metric.

Both definitions (1.2) and (1.3) lead to well-defined maps, but a quantitative result is available only for the minimization formulation:

Lemma 1.9

1. *If the nodal values $v_1, \dots, v_m$ are contained in a geodesic ball $B \subset \mathcal{M}$ with radius less than a threshold depending on the injectivity radius and the Lagrangian shape functions, then the minimization problem* (1.2) *has a unique solution for all ξ.*
2. *Around any $p \in \mathcal{M}$ there exists a neighborhood B, such that for nodal values v_i, $i = 1, \dots, m$ in B there exists a unique map $q : T_h \to \mathcal{M}$ solving* (1.3).

The first assertion is proved in [36]. The second one follows from the implicit function theorem as $\sum_{i=1}^{m} \varphi_i(\xi) \log_p(p) = 0$ for all ξ, and $\partial_q|_{q=p} \sum_{i=1}^{m} \varphi_i(\xi) \log_q(p) = \text{Id} : T_p\mathcal{M} \to T_p\mathcal{M}$ and hence invertible.

As in the previous section we now piece together the local interpolation rule to construct function spaces on an entire grid.

Definition 1.10 Let $\mathcal{G}$ be a grid for Ω. The space of global geodesic finite element functions is

$$S_h^{\mathrm{geo}}(\Omega, \mathcal{M}) := \Big\{ v_h \in C(\Omega, \mathcal{M}) \ : \ v_h|_{T_h} \in S_h^{\mathrm{geo}}(T_h, \mathcal{M}) \text{ for all } T_h \in \mathcal{G} \Big\}.$$

As the values of geodesic interpolation functions on a face of the boundary of an element T_h only depend on the values corresponding to Lagrange points on that face [57, Lem. 4.2], continuity across adjacent elements can be enforced by requiring the Lagrange coefficients on the common boundary to be equal. We therefore have a definition like (1.1) also for geodesic finite elements. Beware, however, that the relation between geodesic finite element functions and values at the Lagrange points is not an isomorphism. See [57] for details.

1.2.2.2 Properties

We now discuss various properties of the geodesic interpolation (1.2). Arguably the most important one for finite element applications is smoothness in ξ and the coefficients v_i. This follows from local smoothness of the distance $d(\cdot, \cdot)$, and the implicit function theorem.

Theorem 1.11 ([57, Thm. 4.1]) *Let $\mathcal{M}$ be a complete Riemannian manifold, and let $v = (v_1, \dots, v_m)$ be coefficients on $\mathcal{M}$ with respect to a Lagrange basis $\{\varphi_i\}$ on a domain T_h. Assume that the situation is such that the assumptions of Lemma 1.9(1) hold. Then the function*

$$I^{\mathrm{geo}} \ : \ \mathcal{M}^m \times T_h \to \mathcal{M} \qquad (v, \xi) \mapsto I^{\mathrm{geo}}(v, \xi)$$

is infinitely differentiable with respect to the v_i and ξ.

Actually computing derivatives is possible using the following trick, explained for derivatives with respect to ξ. To compute the derivative

$$\frac{\partial}{\partial \xi} I^{\mathrm{geo}}(v, \xi) \ : \ T_\xi T_h \to T_{I^{\mathrm{geo}}(v,\xi)} \mathcal{M}$$

(where $T_\xi T_h$ is the tangent space at ξ of the grid element T_h interpreted as a manifold), we recall that values q^* of I^{geo} are minimizers of the functional

$$f_{v,\xi} : \mathcal{M} \to \mathbb{R} \qquad f_{v,\xi}(q) := \sum_{i=1}^m \varphi_i(\xi) \, \mathrm{d}(v_i, q)^2.$$

Hence, they fulfill the first-order optimality condition

$$F(v_1, \dots, v_m; \xi, q^*) := \left.\frac{\partial f_{v,\xi}(q)}{\partial q}\right|_{q=q^*} = 0.$$

Taking the total derivative of this with respect to ξ gives, by the chain rule,

$$\frac{dF}{d\xi} = \frac{\partial F}{\partial \xi} + \frac{\partial F}{\partial q} \cdot \frac{\partial I^{\text{geo}}(v, \xi)}{\partial \xi} = 0,$$

with

$$\frac{\partial F}{\partial \xi} = \sum_{i=1}^{m} \nabla \varphi_i(\xi) \frac{\partial}{\partial q} \operatorname{d}(v_i, q)^2 \tag{1.4}$$

and

$$\frac{\partial F}{\partial q} = \sum_{i=1}^{m} \varphi_i(\xi) \frac{\partial^2}{\partial q^2} \operatorname{d}(v_i, q)^2. \tag{1.5}$$

By [57, Lemma 3.11] the matrix $\partial F/\partial q$ is invertible. Hence the derivative $\partial I^{\text{geo}}/\partial v_i$ can be computed as a minimization problem to obtain the value $I^{\text{geo}}(v, \xi)$ and the solution of the linear system of equations

$$\frac{\partial F}{\partial q} \cdot \frac{\partial}{\partial \xi} I^{\text{geo}}(v, \xi) = -\frac{\partial F}{\partial \xi}.$$

The expressions $\frac{\partial}{\partial q} \operatorname{d}(v_i, q)^2$ and $\frac{\partial^2}{\partial q^2} \operatorname{d}(v_i, q)^2$ that appear in (1.4) and (1.5), respectively, encode the geometry of $\mathcal{M}$. Closed-form expressions for both are given in [56] for the case of $\mathcal{M}$ being the unit sphere. For $\mathcal{M} = \mathrm{SO}(3)$, the first and second derivatives of $\operatorname{d}(v, \cdot)^2$ with respect to the second argument have been computed in [59].

Unlike projection-based finite elements, geodesic finite elements do not form nested spaces except in the following special case [57, Lem. 4.3].

Lemma 1.12 (Nestedness) *Let $d = 1$ and $I_1^{\text{geo}}(v^1, \cdot) : [0, 1] \to \mathcal{M}$ be a first-order geodesic interpolation function between two values $v_1^1, v_2^1 \in \mathcal{M}$. Correspondingly, let $I_r^{\text{geo}}(v^r, \cdot) : [0, 1] \to \mathcal{M}$ be an r-th order geodesic interpolation function of values $v_1^r, \dots, v_{r+1}^r$. We assume that I_r^{geo} interpolates I_1^{geo} in the sense that*

$$v_i^r = I_r^{\text{geo}}(v^r, \xi^r) = I_1^{\text{geo}}(v^1, \xi^r)$$

for all Lagrange nodes $\xi_i^r \in [0, 1]$ of I_r^{geo}. Then

$$I_r^{\text{geo}}(v^r, \xi) = I_1^{\text{geo}}(v^1, \xi) \qquad \textit{for all } \xi \in [0, 1].$$

Equivariance of the interpolation under isometries of $\mathcal{M}$, on the other hand, holds under very general circumstances:

Lemma 1.13 *Let $\mathcal{M}$ be a Riemannian manifold and G a group that acts on $\mathcal{M}$ by isometries. Let $v_1, \dots, v_m \in \mathcal{M}$ be such that the assumptions of Lemma 1.9(1) hold. Then*

$$QI^{\text{geo}}(v_1, \dots, v_m; \xi) = I^{\text{geo}}(Qv_1, \dots, Qv_m; \xi)$$

for all $\xi \in T_h$ and $Q \in G$.

As geodesic interpolation is defined using metric quantities only, this result is straightforward. We therefore omit the proof and refer the reader to the corresponding proof for the first-order case given in [56, Lem. 2.6], which can be adapted easily.

1.2.2.3 Relationship to Projection-Based Finite Elements

The close relationship between geodesic and projection-based finite elements has been noted by several authors [28, 33, 62]. If one considers weighted Fréchet means with respect to the Euclidean distance of the embedding space $\mathbb{R}^N$ instead of the geodesic distance, this agrees exactly with the definition of projection-based interpolation:

$$\arg\min_{q\in\mathcal{M}} \sum_{i=1}^{m} \varphi_i(\xi)\|v_i - q\|^2_{\mathbb{R}^N} = \arg\min_{q\in\mathcal{M}} \Big\| q - \sum_{i=1}^{m} \varphi_i(\xi) v_i \Big\|^2_{\mathbb{R}^N} = P\Big(\sum_{i=1}^{m} \varphi_i(\xi) v_i\Big).$$

This does not mean that projection-based finite elements are equal to geodesic finite elements for embedded manifolds. Instead, projection-based interpolation can be interpreted as an example of geodesic finite elements for a general metric space $(\mathcal{M}, d)$ with a non-intrinsic metric d. In particular cases, the Euclidean distance can be intrinsic, as we see in the following example:

Remark 1.14 For the sphere $\mathcal{M} = S^n$, projection-based interpolation with respect to the standard embedding into $\mathbb{R}^{n+1}$ solves the intrinsic pointwise minimization problem

$$I^{\text{proj}}(v, \xi) = \arg\min_{q\in\mathcal{M}} \sum_{i=1}^{m} \varphi_i(\xi)\big[1 - \cos(d(v_i, q))\big],$$

where $d(\cdot, \cdot)$ denotes the intrinsic distance on the sphere. It therefore coincides with the center of mass construction with a curvature adapted distance function suggested in [44, p. 511]. This construction is well-defined in larger balls than the weighted geodesic means used to define geodesic finite elements, which leads to more robustness. This effect is observable in practice (see Sect. 1.5.1).

1.2.3 Geometric Finite Elements Based on de Casteljau's Algorithm

The following two ways to construct manifold-valued generalizations of polynomials have been proposed by Absil et al. [4], who call them generalized Bézier surfaces.

1.2.3.1 Construction

The definition of generalized Bézier surfaces is based on the weighted geodesic mean (Definition 1.7). Following [4], we define them for two-dimensional domains only. The generalization to higher domains is straightforward.

Let T_h be the image of the unit quadrilateral $T_{\text{ref}} = [0, 1]^2$ under a multilinear transformation, with local coordinates $(t_1, t_2) \in T_{\text{ref}}$.

Definition 1.15 (Generalized Bézier Surface) Given control points $v_{ij} \in \mathcal{M}$, $i, j = 0, \dots, r$, we define a corresponding generalized Bézier surface of type II by

$$I^{\text{Bézier,II}}(t_1, t_2; (v_{ij})_{i,j=0,\dots,r}) = \beta_r(t_1; (\beta_r(t_2; (v_{ij})_{j=0,\dots,r}))_{i=0,\dots,r}),$$

where $\beta_r(\cdot; (b_m)_{m=0,\dots,r}) = \mathfrak{b}_0^r$ denotes a Bézier curve in $\mathcal{M}$ defined by de Casteljau's algorithm

$$\begin{aligned}
\mathfrak{b}_i^0 &= b_i, \qquad i = 0, \dots, r, \\
\mathfrak{b}_i^k &= \operatorname{av}\left[(\mathfrak{b}_i^{k-1}, \mathfrak{b}_{i+1}^{k-1}, ((1-t_1), t_1)\right], \quad k = 1, \dots, r, \quad i = 0, \dots, r-k.
\end{aligned}$$

Likewise, we define a generalized Bézier surface of type III by

$$I^{\text{Bézier,III}}(t_1, t_2; (v_{ij})_{i,j=0,\dots,r}) = \mathfrak{b}_{00}^r,$$

where $\mathfrak{b}_{00}^r$ is defined recursively via the two-dimensional de Casteljau algorithm,

$$\begin{aligned}
\mathfrak{b}_{ij}^0 &= b_{ij}, \qquad i, j = 0, \dots, r, \\
\mathfrak{b}_{ij}^k &= \operatorname{av}\left[(\mathfrak{b}_{ij}^{k-1}, \mathfrak{b}_{i,j+1}^{k-1}, \mathfrak{b}_{i+1,j}^{k-1}, \mathfrak{b}_{i+1,j+1}^{k-1}), (w_{00}, w_{01}, w_{10}, w_{11})\right], \\
&\qquad k = 1, \dots, r, \qquad i, j = 0, \dots, r-k,
\end{aligned}$$

with the first-order Lagrange shape functions as weights

$$w_{00} = (1-t_1)(1-t_2), \quad w_{01} = (1-t_1)t_2, \quad w_{10} = t_1(1-t_2), \quad w_{11} = t_1 t_2.$$

It is demonstrated in [4] that these two definitions do not produce the same maps in general.

The type I Bézier surfaces also proposed in [4] correspond to the geodesic interpolation of Definition 1.8 with the standard Bernstein polynomials as weight functions.

Bézier surfaces are well-defined if the coefficients are close to each other in a specific sense. This is a consequence of the following Lemma.

Lemma 1.16 *For every $q \in \mathcal{M}$ there exists a neighborhood $U \subset \mathcal{M}$ that is both multigeodesically convex, i.e., it contains any weighted geodesic average of any of its points with weights in* $[0, 1]$*, and proper, i.e., the averages of finitely many points with weights in* $[0, 1]$ *are unique, and depend smoothly on the points and the weights.*

Details are given in [4, Sec. 3.2].

Note that the Bézier surface functions are not interpolatory, except at the domain corners. Nevertheless, we use them to define global finite element spaces on grids where all elements are multilinear images of d-dimensional cubes

$$S_h^{\text{Bézier},Y}(\Omega, \mathcal{M}) := \Big\{ v_h \in C(\Omega, \mathcal{M}) \;:\; v_h|_{T_h} \text{ is a type } Y \text{ Bézier surface } \forall T_h \in \mathcal{G} \Big\}.$$

Here, the symbol Y replaces either II or III.

It is shown in [4] that piecewise type III surfaces are continuous for all $d \geq 1$ if the control values match at the element boundaries. For piecewise type II surfaces the situation is more complicated. Unlike type III surfaces (and unlike all previous constructions of geometric finite elements), type II surfaces are not invariant under isometries of the domain. More specifically, the natural condition

$$I^{\text{Bézier,II}}(t_1, t_2; (v_{ij})_{i,j=0,\dots,r}) = I^{\text{Bézier,II}}(t_2, t_1; (v_{ij})^T_{i,j=0,\dots,r})$$

does not hold. As a consequence, for $d \geq 3$, the restriction to an element facet depends on the control values on that facet and on the ordering of the local coordinate axes there. As these typically do not match across neighboring elements, identical control values on a common boundary do not necessarily imply continuity of the function there.

An exception to this is the case $d = 2$. There, the element boundaries are one-dimensional, and there is only one possible coordinate system (and one-dimensional Bézier curves are invariant under the coordinate changes $t \mapsto 1 - t$). We therefore conjecture that type II surfaces are a competitive choice of finite elements mainly for problems with two-dimensional domains. Their advantage is that they only use averages between pairs of points, i.e., interpolation along geodesics between two given points in $\mathcal{M}$. For this, closed-form expressions are known for a few important spaces like the spheres (where it is simply interpolation along arcs of great circles), the three-dimensional special orthogonal group [55], and the hyperbolic half-spaces [63].

1.2.3.2 Properties

Wie discuss a few properties of (piecewise) generalized Bézier surfaces of types II and III. First of all, smoothness of generalized Bézier surfaces is a direct consequence of Lemma 1.16 (cf. [4, Sec. 3.2]), but note that no quantitative measure of the required locality seems to be available. Combining the smoothness on each element with global continuity yields the following conformity result:

Lemma 1.17 $S_h^{\text{Bézier},Y}(\Omega, \mathcal{M}) \subset W^{1,p}(\Omega, \mathcal{M})$ *for* $Y \in \{\text{II}, \text{III}\}$ *and all* $p \in [1, \infty]$.

Concerning nestedness, when the codomain $\mathcal{M}$ is linear, then generalized Bézier surfaces degenerate to polynomials, and constructions with different numbers of control points lead to nested spaces. The explicit construction of the embeddings are called degree elevation formulas [24]. Corresponding results for nonlinear $\mathcal{M}$ are unknown.

As Bézier finite elements are defined using the metric structure of $\mathcal{M}$ alone, they satisfy the same favorable equivariance properties as geodesic finite elements:

Lemma 1.18 *Let* $\mathcal{M}$ *be a Riemannian manifold and* Q *an isometry of* $\mathcal{M}$*. Then*

$$QI^{\text{Bézier},Y}((v_{ij}); \xi) = I^{\text{Bézier},Y}((Qv_{ij}); \xi)$$

for $Y \in \{\text{II}, \text{III}\}$ *and all* $\xi \in [0, 1]^d$*, provided these expressions are well defined.*

Hence, discretizations of frame-invariant problems are frame-invariant as well.

1.2.4 Interpolation in Normal Coordinates

As manifolds are locally diffeomorphic to $\mathbb{R}^n$ by definition, yet another approach to obtain an interpolation method is to interpolate in a coordinate chart in $\mathbb{R}^n$. A canonical choice is to use normal coordinates around a point $p \in \mathcal{M}$, i.e., use $\exp_p$ and $\log_p$ to identify points in $\mathcal{M}$ with their coordinates

$$I_p(v, \xi) := \exp_p \left(\sum_{i=1}^{m} \varphi_i(\xi) \log_p v_i \right).$$

This interpolation depends heavily on the choice of the point $p \in \mathcal{M}$, and unless p is fixed once and for all, it is not obvious how to choose the p for each element of a grid to obtain globally continuous finite elements from this local interpolation.

Nevertheless, this construction is used by several authors [29, 52], especially for interpolation in symmetric spaces. Symmetric spaces are smooth manifolds that have an inversion symmetry about every point [43, Ch. 6]. Examples are SO(3), the space of Lorentzian metrics, the spaces of symmetric positive matrices, and the

Grassmannians. As symmetric spaces are in one-to-one correspondence with Lie triple systems, they have the advantage that the Lie group exponential can be used to construct the coordinate chart. The dependence on the choice of p, however, remains.

1.2.4.1 Construction

We follow the construction in [29]. Let $\mathcal{S}$ be a symmetric space, $\mathfrak{g}$ its Lie algebra of Killing vector fields, and $p \in \mathcal{S}$. Then $T_p\mathcal{S}$ is isomorphic to the vector space

$$\mathfrak{p} := \{X \in \mathfrak{g} \ : \ \nabla X(p) = 0\}.$$

Let G be the isometry group of $\mathcal{S}$, and let $\exp : \mathfrak{g} \to G$ denote the Lie group exponential map.

Fixing an element $\overline{g} \in G$, a diffeomorphism from a neighborhood of $0 \in \mathfrak{p}$ to a neighborhood of $\overline{g} \cdot p \in \mathcal{S}$ can be defined by

$$F_{\overline{g}}(P) := \overline{g} \cdot \exp(P) \cdot p.$$

Definition 1.19 Let T_h be a grid element, and $\mathcal{S}$ a symmetric space with $F_{\overline{g}}$ defined as above. Let $\varphi_1, \dots, \varphi_m$ be a set of r-th order scalar Lagrangian shape functions, and let $v = (v_1, \dots, v_m) \in \mathcal{S}^m$ be values at the corresponding Lagrange nodes. We call

$$I_{\overline{g}}^{\exp} \ : \ \mathcal{M}^m \times T_h \to \mathcal{M}$$

$$I_{\overline{g}}^{\exp}(v, \xi) := F_{\overline{g}}\biggl(\sum_{i=1}^{m} \varphi_i(\xi) F_{\overline{g}}^{-1} v_i\biggr) \tag{1.6}$$

r-th order normal-coordinate interpolation on $\mathcal{S}$.

Here it is assumed that the interpolation values v_i belong to the range of $F_{\overline{g}}$; therefore the interpolation is only suitable for values in a neighborhood of $\overline{g} \cdot p \in \mathcal{S}$. Note that different choices of $\overline{g}$ essentially correspond to different choices of the point $p \in \mathcal{S}$. Indeed, it is shown in [29] that for $h \in G^p := \{g \in G \ : \ g \cdot p = p\}$ (the stabilizer subgroup), the interpolation is invariant under post-multiplication, i.e.,

$$I_{\overline{g}h}^{\exp}(v, \xi) = I_{\overline{g}}^{\exp}(v, \xi).$$

Example 1.20 If we consider $\mathcal{S} = \mathrm{SO}(3)$, $p = \mathrm{Id}$, and $\overline{g} = \mathrm{Id}$, we obtain for values $v \in \mathrm{SO}(3)^m$ that are in the range of the matrix exponential exp

$$I_{\mathrm{Id}}^{\exp}(v, \xi) = \exp\biggl(\sum_{i=1}^{m} \varphi_i(\xi) \log v_i\biggr).$$

For $\mathcal{S}$ the space of symmetric positive definite matrices, this is the (weighted) Log-Euclidean mean proposed by Arsigny et al. [10].

For interpolation with respect to a different $\overline{g} \in \mathrm{SO}(3)$, we obtain for values $v \in \mathrm{SO}(3)^m$ such that the $\overline{g}^{-1}v_i$ are in the range of exp

$$I_{\overline{g}}^{\exp}(v,\xi) = \overline{g}\exp\left(\sum_{i=1}^{m} \varphi_i(\xi)\log(\overline{g}^{-1}v_i)\right).$$

Münch [52] heuristically switched between different choices of p for a discretization of SO(3)-valued functions.

As in the previous sections we use the local interpolation rule to define a global discrete space.

Definition 1.21 Let $\mathcal{G}$ be a grid for Ω. The space of global normal-coordinate finite element functions for a symmetric space $\mathcal{S}$ is

$$S_h^{\exp}(\Omega,\mathcal{S}) := \left\{v_h \in C(\Omega,\mathcal{M}) \ : \ \exists (v_i)_{i\in I} \subset \mathcal{S},\ \overline{g} \in G \text{ s.t. } v_h|_{T_h} = I_{\overline{g}}^{\exp}(v,\cdot)\right\}.$$

As pointed out in the beginning of this section, it is not clear that there exists a natural choice for the $\overline{g}$ on each element T_h such that this is non-empty. Engineering-oriented works such as [52] that switch between several p appear to be tacitly working with spaces of discontinuous functions.

In [29] it is possible to let $\overline{g}$ vary with ξ. If one chooses $\overline{g}(\xi)$ such that

$$\overline{g}(\xi)\cdot p = I_{\overline{g}(\xi)}^{\exp}(v,\xi),$$

this is equivalent to

$$\sum_{i=1}^{m} \varphi_i(\xi)F_{\overline{g}(\xi)}^{-1}(v_i) = 0.$$

If G is equipped with a left-invariant Riemannian metric, then this corresponds to the definition of geodesic finite elements by first variation (1.3).

1.2.4.2 Properties

Once the choice of $p \in \mathcal{S}$ is fixed, the normal coordinate interpolation has several nice properties.

Lemma 1.22 *The interpolation operator $I_{\overline{g}}^{\exp}$ defined by* (1.6) *is infinitely differentiable with respect to the v_i and ξ.*

For fixed $\overline{g} \in G$ this follows from the chain rule, and smoothness of the exponential map. In [29] explicit formulas for the first and second derivatives with respect to the ξ are given. For varying $\overline{g}(\xi)$, the implicit function theorem implies the result.

It is easy to see that normal-coordinate interpolation produces nested spaces if $\overline{g}$ remains fixed for all orders:

Lemma 1.23 (Nestedness) *If $S^{\exp}_{h,r_1}(\Omega, \mathcal{S})$ and $S^{\exp}_{h,r_2}(\Omega, \mathcal{S})$ are two global finite element spaces of polynomial orders $r_1 \leq r_2$, defined by normal-coordinate interpolation with respect to the same $\overline{g}$ on each element, then*

$$S^{\exp}_{h,r_1}(\Omega, \mathcal{S}) \subset S^{\exp}_{h,r_2}(\Omega, \mathcal{S}).$$

Note that this is no longer true for varying $\overline{g}$, as illustrated, e.g., by the lack of nestedness of geodesic finite elements.

After the choice of $p \in \mathcal{S}$ there is no more coordinate dependence, i.e., the method is frame-invariant:

Lemma 1.24 ([29]) *Let $\mathcal{S}$ be a symmetric space, G its isometry group, and $v_1, \ldots, v_m \in \mathcal{S}$, $\overline{g} \in G$ such that (1.6) is well-defined. Then*

$$Q \cdot I^{\exp}_{\overline{g}} v(\xi) = I^{\exp}_{Q\overline{g}}(Q \cdot v)(\xi)$$

holds for every $\xi \in T_h$ and every $Q \in G$.

Note that on the right hand side, the isometry Q not only acts on the coefficients $v = (v_1, \ldots, v_m)$, but also on $\overline{g}$.

1.3 Discrete Test Functions and Vector Field Interpolation

The standard finite element method uses the word "test functions" to denote the variations of a function in variational problems. As the function spaces are linear or affine, these variations are independent of the functions they act on. Even more, in many cases the spaces of variations coincide with the function spaces that contain the functions themselves.

This changes in nonlinear function spaces like the ones used by geometric finite element methods. Here, test functions have to be defined by admissible variations of functions, and the spaces of test functions therefore do depend on the function that is being tested. Consider the set $C^\infty(T_h, \mathcal{M})$ of differentiable maps into a manifold $\mathcal{M}$. An admissible variation of $c \in C^\infty(T_h, \mathcal{M})$ is a family $\Gamma(\cdot, s) \in C^\infty(T_h, \mathcal{M})$ for all $s \in (-\delta, \delta)$ with $\Gamma(\cdot, 0) = c$. The variational field of this variation, i.e., the vector field $V = \partial_s \Gamma(\cdot, 0) \in C^\infty_0(T_h, c^{-1}T\mathcal{M})$, is the generalization of a test function in this context. This construction carries over to functions of lesser smoothness.

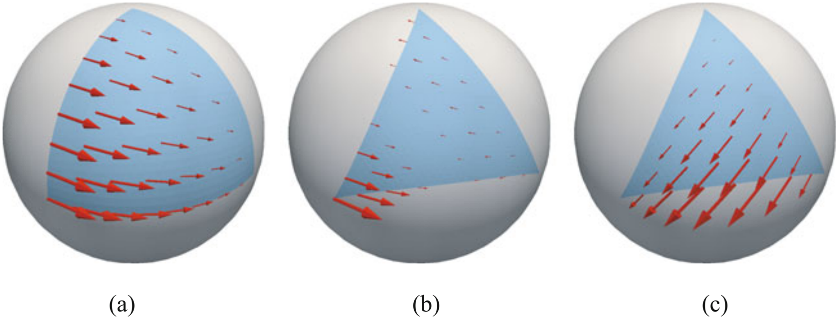


Fig. 1.2 Test functions along geodesic interpolation functions from a triangle into the sphere S^2. These vector fields correspond to the shape functions normally used for Lagrangian finite element methods, because they are zero on all but one Lagrange point. Note how the second-order vertex vector field (center) partially point "backwards", because the corresponding scalar shape function has negative values on parts of its domain. (**a**) First order, (**b**) second order: vertex degree of freedom, (**c**) second order: edge degree of freedom

Analogously, we can define discrete test vector fields by variations of discrete maps. Let $S_h(\Omega, \mathcal{M}) \subset C(\Omega, \mathcal{M})$ be a set of geometric finite element functions.

Definition 1.25 For any $u_h \in S_h(\Omega, \mathcal{M})$, we denote by $S_h(\Omega, u_h^{-1}T\mathcal{M})$ the set of all vector fields $V_h \in C(\Omega, u_h^{-1}T\mathcal{M})$ such that there exists a family $v_h(s) \in S_h(\Omega, \mathcal{M})$, $s \in (-\delta, \delta)$, with $v_h(0) = u_h$ and $\frac{d}{ds} v_h(0) = V_h$.

Figure 1.2 shows three such vector fields along functions from $S_h^{\text{geo}}(T_h, S^2)$. If we set $\mathcal{M} = \mathbb{R}$, the standard Lagrangian shape functions are obtained. Hence, Definition 1.25 is a direct generalization of the test functions normally used in the finite element method.

1.3.1 Algebraic Representation of Test Functions

Geometric finite element test functions have an algebraic representation. Unlike for geometric finite element functions themselves, where the relationship between discrete functions and algebraic representations is complicated, for test functions the two are isomorphic.

Lemma 1.26 ([58, Lem. 3.1]) *Let T_h be a grid element, and let I be a geometric interpolation function for values $v_1, \dots, v_m \in \mathcal{M}$. The admissible variations of the function $I(v, \cdot) : T_h \to \mathcal{M}$ form a vector space $S_h(T_h, v^{-1}T\mathcal{M})$, which is isomorphic to $\prod_{i=1}^{m} T_{v_i}\mathcal{M}$. The isomorphism has an explicit representation as*

$$\mathcal{T} : \prod_{i=1}^{m} T_{v_i}\mathcal{M} \to V_{h,v}$$

$$\mathcal{T}[b_1, \dots, b_m](\xi) = \sum_{i=1}^{m} \frac{\partial I(v_1, \dots, v_m; \xi)}{\partial v_i} \cdot b_i$$

for all $b_i \in T_{v_i}\mathcal{M}$*,* $i = 1, \dots, m$*, and* $\xi \in T_h$*.*

In other words, any test function along a geometric finite element function $I(v, \cdot) : T_h \to \mathcal{M}$ can be uniquely characterized by a set of tangent vectors $b_i \in T_{v_i}\mathcal{M}$, $i = 1, \dots, m$.

Practical computation of the quantities $\partial I(v, \xi)/\partial v_i$ depends on the specific interpolation rule. It is straightforward for rules that are given in closed form such as projection-based interpolation (Definition 1.1) and interpolation in normal coordinates with respect to a fixed point (Definition 1.19). For geodesic interpolation, the trick to compute $\partial I^{\text{geo}}/\partial \xi$ described in Sect. 1.2.2.2 can be modified to obtain the differentials with respect to the v_i as well. Details can be found in [56].

1.3.2 Test Vector Fields as Discretizations of Maps into the Tangent Bundle

To analyze test vector fields further, we consider an implicit definition of geometrically conforming finite element functions. Note that both geodesic (Example 1.27) as well as projection-based finite elements (Example 1.28) can be viewed in this way. Let $S_h(T_h, \mathcal{M})$ be defined implicitly by a differentiable mapping

$$F_{\mathcal{M}} : T_h \times \mathcal{M}^m \times \mathcal{M} \to T_{\mathcal{M}}, \quad (\xi, v_1, \dots, v_m; q) \mapsto F_{\mathcal{M}}(\xi, v_1, \dots, v_m; q) \in T_q\mathcal{M}.$$

We assume that $F_{\mathcal{M}}(\xi, \mathbf{p}; p) = 0$ for all $p \in \mathcal{M}$ (with $\mathbf{p}=(p, \dots, p) \in \mathcal{M}^m$), and that $\partial_q F_{\mathcal{M}}(\xi, \mathbf{p}; q)\big|_{q=p} : T_p\mathcal{M} \to T_{F_{\mathcal{M}}(\xi,\mathbf{p};p)}T_p\mathcal{M} = T_p\mathcal{M}$ is invertible. Then we can define the discrete maps $v_h \in S_h(T_h, \mathcal{M})$ by

$$F_{\mathcal{M}}(\xi, v_1, \dots, v_m, v_h(\xi)) = 0 \in T_{v_h(\xi)}\mathcal{M}, \tag{1.7}$$

as long as the $v_i \in \mathcal{M}$, $i = 1, \dots, m$, are close enough for the implicit function theorem to hold.

Example 1.27 For geodesic finite elements into an affine manifold $(\mathcal{M}, \nabla)$ the mapping $F_{\mathcal{M}}$ is given by

$$F_{\mathcal{M}}(\xi, v_1, \dots, v_m; q) = \sum_{i=1}^{m} \varphi_i(\xi) \log_q v_i.$$

Example 1.28 Let $\mathcal{M}$ be a manifold embedded smoothly into $\mathbb{R}^N$. For any $q \in \mathcal{M}$ denote by $P_q : T_q\mathbb{R}^N \to T_q\mathcal{M}$ the orthogonal projection. For projection-based finite elements, the mapping $F_\mathcal{M}$ is then given by

$$F_\mathcal{M}(\xi, v_1, \dots, v_m; q) = P_q\biggl(\sum_{i=1}^m \varphi_i(\xi)v_i - q\biggr).$$

Definition 1.25 defines discrete test vector fields as variations. Implicit differentiation of v_h defined in (1.7) thus leads to a natural extension of the definition of $S_h(\Omega, \mathcal{M})$ to maps from Ω to the tangent bundle $T\mathcal{M}$.

Lemma 1.29 *Let $S_h(T_h, \mathcal{M})$ be defined implicitly by (1.7), and define $\hat{F} : T_h \times T\mathcal{M}^m \times T\mathcal{M} \to TT\mathcal{M}$ by*

$$\begin{aligned}&\hat{F}(\xi, (v_1, V_1), \dots, (v_m, V_m); (q, W))\\ &\quad := \begin{pmatrix} F_\mathcal{M}(\xi, v_1, \dots, v_m; q) \\ \sum_{j=1}^m \partial_{v_j} F_\mathcal{M}(\xi, v_1, \dots, v_m; q)(V_j) + \partial_q F_\mathcal{M}(\xi, v_1, \dots, v_m; q)(W)\end{pmatrix}\end{aligned}$$

with the standard identification $T_{(q,W)}T\mathcal{M} = (T_q\mathcal{M})^2$. Then the discrete test vector fields (v_h, V_h) with $V_h \in S_h(T_h, v_h^{-1}T\mathcal{M})$ as defined in Definition 1.25 agree with the implicit definition

$$\hat{F}(\xi, (v_1, V_1), \dots, (v_m, V_m), (v_h(\xi), V_h(\xi))) = 0.$$

Proof For $i = 1, \dots, m$ consider curves $c_i : (-\delta, \delta) \to \mathcal{M}$ with $c_i(0) = v_i$, $\dot{c}_i(0) = V_i$, and a curve $\gamma : (-\delta, \delta) \to \mathcal{M}$ with $\gamma(0) = q$, $\dot{\gamma}(0) = W$. Inserting these as arguments for $F_\mathcal{M}$ and differentiating yields

$$\begin{aligned}&\frac{d}{ds}\bigg|_{s=0} F_\mathcal{M}(\xi, c_1(s), \dots, c_m(s), \gamma(s)(\xi))\\ &\qquad\qquad = \hat{F}_2\bigl(\xi, (v_1, V_1), \dots, (v_m, V_m); (q, W)\bigr) \in T_q\mathcal{M},\end{aligned}$$

where $\hat{F}_2$ denotes the second component of $\hat{F}$.

Let now $w_h : (-\delta, \delta) \to S_h(\Omega, \mathcal{M})$ be a family of discrete maps with $w_h(0) = v_h$ and $\frac{d}{ds}w_h(0) = V_h$, and let $V_j \in T_{v_j}\mathcal{M}$ denote the values of V_h at v_j for $j = 1, \dots, m$. Then differentiation of (1.7) yields

$$\hat{F}_2(\xi, (v_1, V_1), \dots, (v_m, V_m); (v_h(\xi), V_h(\xi))) = 0 \in T_{v_h(\xi)}\mathcal{M}.$$

Thus, a discrete test vector field fulfills the implicit definition.

Further note that at $(\xi, (\mathbf{p}, \mathbf{V}); (p, V)) \in \Omega \times T\mathcal{M}^m \times T\mathcal{M}$, where $(\mathbf{p}, \mathbf{V}) := ((p, V), \dots, (p, V))$,

$$\hat{F}(\xi, (\mathbf{p}, \mathbf{V}); (p, V)) = \begin{pmatrix} F_{\mathcal{M}}(\xi, \mathbf{p}; p) \\ \frac{d}{dt}\big|_{t=0} F_{\mathcal{M}}(\xi, \exp_p(tV), \dots, \exp_p(tV); \exp_p(tV)) \end{pmatrix}$$
$$= \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

and that the invertibility of $\partial_q F_{\mathcal{M}}(\xi, \mathbf{p}; p)$ implies the invertibility of

$$\partial_{(q,W)} \hat{F}(\xi, (\mathbf{p}, \mathbf{V}); (p, V))$$
$$= \begin{pmatrix} \partial_q F_{\mathcal{M}}(\xi, \mathbf{p}; p) & 0 \\ \sum_{j=1}^m \partial_q \partial_{v_j} F_{\mathcal{M}}(\xi, \mathbf{p}; p)(V, \cdot) + \partial_q \partial_{v_j} F_{\mathcal{M}}(\xi, \mathbf{p}; p)(V, \cdot) & \partial_q F_{\mathcal{M}}(\xi, \mathbf{p}; p) \end{pmatrix}.$$

Thus, the implicit definition for vector fields is well posed. □

An alternative way to define vector field discretizations is to consider $T\mathcal{M}$ as a manifold and use the finite elements $S_h(\Omega, T\mathcal{M})$ defined by

$$F_{T\mathcal{M}}(\xi, (v_1, V_1), \dots, (v_m, V_m), (v_h(\xi), V_h(\xi))) = 0.$$

If $F_{T\mathcal{M}} = \hat{F}$, then variation and discretization commute, i.e., the discretization of the variational field is the variational field of the discretization. For this, $T\mathcal{M}$ needs to be endowed with an appropriate metric structure. We give two examples, but it is an open question whether such a structure always exists.

Example 1.30 ([36, 38]) For geodesic finite elements

$$\hat{F}(\xi, (v_1, V_1), \dots, (v_l, V_l); (q, W)) = \sum_{i=1}^m \varphi_i(\xi) \begin{pmatrix} \log_q v_i \\ d \log_q v_j(V_j) + d_q \log_q v_j(W) \end{pmatrix},$$

where d_q denotes the differential of the bivariate mapping $\log : \mathcal{M} \times \mathcal{M} \to T\mathcal{M}$, $(p, q) \mapsto \log_q p$, with respect to the second variable. $\hat{F}$ corresponds to $F_{T\mathcal{M}}$ if we endow $T\mathcal{M}$ with the horizonal lift connection [15, 47]. Choosing the more commonly used Sasaki metric instead does not yield commutativity.

Example 1.31 ([33]) For projection-based finite elements with $P : \mathcal{M} \times \mathbb{R}^N \to T\mathcal{M}$, $P(q, \cdot) = P_q(\cdot)$, defined by the orthogonal projection onto $T_q\mathcal{M}$ of Example 1.28, we have

$$\begin{aligned}
&\hat{F}(\xi, (v_1, V_1), \dots, (v_l, V_l); (q, W)) \\
&= \begin{pmatrix} P_q \left(\sum_{i=1}^m \varphi_i(\xi) v_i - q\right) \\ P_q \left(\sum_{i=1}^m \varphi_i(\xi) V_i - W\right) - \partial_1 P(q, \sum_{i=1}^m \varphi_i(\xi) v_i - q)(W) \end{pmatrix} \\
&= \begin{pmatrix} P_q & 0 \\ -\partial_1 P(q, \cdot)(W) & P_q \end{pmatrix} \begin{pmatrix} \sum_{i=1}^m \varphi_i(\xi) v_i - q \\ \sum_{i=1}^m \varphi_i(\xi) V_i - W \end{pmatrix}.
\end{aligned}$$

Hence, if we choose the projection $P_{(q,W)} : \mathbb{R}^{N \times N} \to T_q \mathcal{M}^2$

$$P_{(q,W)} := \begin{pmatrix} P_q & 0 \\ -\partial_1 P(q, \cdot)(W) & P_q \end{pmatrix}$$

we recover projection-based finite elements with values in $T\mathcal{M}$. Note that $P_{(q,W)}$ does not correspond to the standard projection which is

$$\tilde{P}_{(q,W)} = \begin{pmatrix} P_q & 0 \\ 0 & P_q \end{pmatrix}.$$

1.4 A Priori Error Theory

1.4.1 Sobolev Spaces of Maps into Manifolds

The analysis of Galerkin methods for the approximation of variational PDEs is written in the language of Sobolev spaces. While some of the concepts can be generalized directly to maps from $\Omega \in \mathbb{R}^d$ to a smooth complete Riemannian manifold $(\mathcal{M}, g)$, others need to treated more carefully.

Definition 1.32 Let $(\mathcal{M}, g)$ be a complete manifold with distance $d(\cdot, \cdot)$, and $1 \leq p < \infty$. We define

$$\begin{aligned} L^p(\Omega, \mathcal{M}) := \Big\{ u : \Omega \to \mathcal{M} \mid u \text{ measurable},\ u(\Omega) \text{ separable}, \\ \int_\Omega d^p(u(x), q)\, dx < \infty \text{ for some } q \in \mathcal{M} \Big\}. \end{aligned}$$

A distance d_{L^p} on $L^p(\Omega, \mathcal{M})$ is then given by

$$d_{L^p}^p(u, v) := \int_\Omega d^p(u(x), v(x))\, dx.$$

There are several ways to define the Sobolev space $W^{k,p}(\Omega, \mathcal{M})$. The most common definition (see, e.g., [34, 40, 41, 64]) uses the Nash embedding theorem, that asserts that every Riemannian manifold can be isometrically embedded into some Euclidean space.

Definition 1.33 Let $p \in [1, \infty]$, and $\iota : \mathcal{M} \to \mathbb{R}^N$ be an isometric embedding into a Euclidean space. We then define

$$W_\iota^{k,p}(\Omega, \mathcal{M}) := \Big\{ v \in W^{k,p}(\Omega, \mathbb{R}^N) \;:\; v(x) \in \iota(\mathcal{M}) \text{ a.e.} \Big\}.$$

While this definition is always possible it depends in principle on the choice of embedding ι. If $k = 1$ and $\mathcal{M}$ is compact, then the definition is independent of ι (see, e.g., [41]). For orders $k \geq 2$, the definition is no longer intrinsic.

An alternative definition [42, 46, 53] for $k = 1$ uses only the metric structure of $\mathcal{M}$.

Definition 1.34 Let $p \in [1, \infty]$. We say that $u \in W^{1,p}(\Omega, \mathcal{M})$ if for every $q \in \mathcal{M}$ the map $x \mapsto d(u(x), q)$ is in $W^{1,p}(\Omega)$, and if there exists a function $h \in L^p(\Omega)$ independent of q such that $|D(d(u(x), q))| \leq h(x)$ almost everywhere.

This provides an intrinsic definition of $W^{1,p}$-spaces that is equivalent to Definition 1.33, at least for compact $\mathcal{M}$ [35]. It also gives a notion of the Dirichlet integrand $|Du|$: it is the optimal h in the definition above, which may differ from the definition of $|Du|$ given by an isometric embedding. Using the notion of approximate continuity [7], it is possible to construct a posteriori an approximate differential almost everywhere [26]. Metric-space-based definitions of Sobolev spaces such as Definition 1.34, however, do not work for higher-order derivatives.

A third approach uses the chain rule to define a notion of weak derivatives [21, 22]: A map $u : \Omega \to \mathcal{M}$ is called colocally weakly differentiable if it is measurable and $f \circ u$ is weakly differentiable for all $f \in C^1(\mathcal{M}, \mathbb{R})$ with compact support. This defines a unique bundle morphism du via $D(f \circ u) = df \circ du$, which directly generalizes the classical notion of differential [21]. This notion extends to higher-order derivatives [22]. For Riemannian manifolds, [22] also gives a definition of a colocal weak covariant derivative. Intrinsic higher-order Sobolev maps are then defined as maps for which the corresponding weak colocal covariant derivatives are in L^p with respect to the norm induced by the Riemannian metric. While this construction has several nice properties (a notion of intrinsic weak derivative, a Rellich–Kondrachov type compactness result, a weak closure property of bounded maps), one drawback is that the chain rule does not hold in general. This can be overcome by assuming additional integrability conditions based on a k-tuple norm.

In previous works about geometric finite elements [32, 36, 39], Definition 1.33 is used to define the Sobolev space $W^{k,p}(\Omega, \mathcal{M})$. For weak derivatives, only $W^{k,p}(\Omega, \mathcal{M}) \cap C(\Omega, \mathcal{M})$ is considered, and weak covariant derivatives are defined using local coordinates. It is shown in [22] that although the homogeneous Sobolev space $\dot{W}^{k,p}(\Omega, \mathcal{M})$ defined by weak colocal covariant derivatives does not agree with $\dot{W}_\iota^{k,p}(\Omega, \mathcal{M})$, at least for compact $\mathcal{M}$ it holds that

$$\dot{W}_\iota^{k,p}(\Omega, \mathcal{M}) = \bigcap_{j=1}^{k} \dot{W}^{j,\frac{kp}{j}}(\Omega, \mathcal{M}).$$

A similar result for continuous maps using weak derivatives in local coordinates is shown in [36].

1.4.1.1 Smoothness Descriptors

Driven by a desire to retain the chain rule, previous works about geometric finite elements [32, 39] introduced the following notion of integrability condition which uses local coordinates to define weak differentiability only for continuous maps.

Definition 1.35 (Smoothness Descriptor) Let $k \geq 1$, $p \in [1, \infty]$. The homogeneous k-th order smoothness descriptor of a map $u \in W^{k,p} \cap W^{1,kp} \cap C(\Omega, \mathcal{M})$ is defined by

$$\dot{\theta}_{k,p,\Omega}(u) := \left(\int_\Omega |\nabla^k u|^p \, dx + \int_\Omega |du|^{kp} \, dx \right)^{\frac{1}{p}},$$

with the usual modifications for $p = \infty$. The corresponding inhomogeneous smoothness descriptor is

$$\theta_{k,p,\Omega}(u) := \left(\sum_{i=1}^{k} \dot{\theta}_{i,p,\Omega}^p(u) \right)^{\frac{1}{p}}.$$

We will also need the smoothness descriptor for vector fields. Although vector fields, i.e., maps $(u, V) : \Omega \to T\mathcal{M}$, are linear in the sense that $u^{-1}T\mathcal{M}$ is a vector space for each u, the set of all vector fields for all base maps u is not linear. For a direct generalization of the linear Sobolev norm, we generalize the smoothness descriptor to vector fields by taking essentially a full Sobolev norm of the linear vector field part $V : \Omega \to u^{-1}T\mathcal{M}$ weighted with covariant derivatives of u to obtain the correct scaling.

Definition 1.36 (Smoothness Descriptor for Vector Fields) Given $k \in \mathbb{N}$ and $p \in [1, \infty]$ set for $i = 1, \ldots, k$

$$\frac{1}{p_i} := \begin{cases} \frac{1}{p} & \text{for } ip > d, \\ \frac{1}{p} - \epsilon_i & \text{for } ip = d, \\ \frac{i}{d} & \text{for } ip < d, \end{cases} \qquad \frac{1}{s_i} := \frac{1}{p} - \frac{1}{p_i},$$

where $0 < \epsilon_i < \min\{\frac{1}{i+p}, \frac{1}{ip}\}$. Let $u \in W^{k,p_k} \cap C(\Omega, \mathcal{M})$, and $V \in W^{k,p}(\Omega, u^{-1}T\mathcal{M})$. We define the k-th order homogeneous smoothness descriptor for vector fields by

$$\dot{\Theta}_{k,p,\Omega}(V) := \left(\|\nabla^k V\|^p_{L^p(\Omega,u^{-1}T\mathcal{M})} + \sum_{i=1}^{k} \dot{\theta}_{i,p_i,\Omega}(u)^p \|\nabla^{k-i} V\|^p_{L^{s_i}(\Omega,u^{-1}T\mathcal{M})} \right)^{1/p}.$$

For $k = 0$, we set $\dot{\Theta}_{0,p,\Omega}(V) := \|V\|_{L^p}$, and we make the usual modifications for $p = \infty$. The inhomogeneous smoothness descriptor is defined by

$$\begin{aligned} \Theta_{k,p,\Omega}(V) &:= \left(\sum_{s=0}^{k} \dot{\Theta}_{s,p,\Omega}(V)^p \right)^{\frac{1}{p}} \\ &= \left(\|V\|^p_{W^{k,p}(\Omega,u^{-1}T\mathcal{M})} + \sum_{i=1}^{k} \dot{\theta}_{i,p_i,\Omega}(u)^p \|V\|^p_{W^{k-i,s_i}(\Omega,u^{-1}T\mathcal{M})} \right)^{\frac{1}{p}}. \end{aligned}$$

For a fixed base map u, the smoothness descriptor acts like a semi-norm on functions into the linear space $u^{-1}T\mathcal{M}$. In particular, if u maps Ω to a constant point p on $\mathcal{M}$, then the smoothness descriptor of a vector field $V : \Omega \to T_p\mathcal{M}$ coincides with the Sobolev semi-norm.

1.4.1.2 Scaling Properties

As already observed in [32], the homogeneous smoothness descriptor is subhomogeneous with respect to rescaling of the domain $\Omega \in \mathbb{R}^d$ with a parameter h.

Definition 1.37 Let T_{ref}, T_h be two domains in $\mathbb{R}^d$, and $F : T_h \to T_{\text{ref}}$ a C^∞-diffeomorphism. For $l \in \mathbb{N}_0$ we say that F scales with h of order l if we have

$$\sup_{x \in T_{\text{ref}}} \left| \partial^\beta F^{-1}(x) \right| \le C\, h^k \qquad \text{for all } \beta \in [d]^k,\ k = 0, \dots, l, \tag{i}$$

$$|\det(DF(x))| \sim h^{-d} \qquad \text{for all } x \in T_h \text{ (where } DF \text{ is the Jacobian of } F\text{)}, \tag{ii}$$

$$\sup_{x \in T_h} \left| \frac{\partial}{\partial x^\alpha} F(x) \right| \le C\, h^{-1} \qquad \text{for all } \alpha = 1, \dots, d. \tag{iii}$$

Readers familiar with finite element theory will recognize such maps F as transformations of an element of a discretization of Ω to a reference element. The smoothness descriptor scales in the following manner.

Lemma 1.38 *Let T_{ref}, T_h be two domains in $\mathbb{R}^d$, and $F : T_h \to T_{ref}$ a map that scales with h of order l. Consider $u \in W^{k,p} \cap W^{1,kp} \cap C(\Omega, \mathcal{M})$ with $1 \leq k \leq l$ and $p \in [1, \infty]$. Then*

$$\dot{\theta}_{k,p,T_{ref}}(u \circ F^{-1}) \leq C\, h^{k-\frac{d}{p}} \left(\sum_{i=1}^{k} \dot{\theta}^{p}_{i,p,T_h}(u) \right)^{\frac{1}{p}} \leq C\, h^{k-\frac{d}{p}} \theta_{k,p,T_h}(u).$$

The proof follows from the chain rule and the integral transformation formula [32].

Remark 1.39 Note that Lemma 1.38 only asserts *sub*homogeneity of the smoothness descriptor, as the homogeneous descriptor is bounded by the inhomogeneous one. This differs from the scaling behavior of Sobolev semi-norms in the Euclidean setting.

Like the smoothness descriptor for functions, the smoothness descriptor for vector fields is subhomogeneous with respect to scaling of the domain [36, 39].

Lemma 1.40 *Let T_{ref}, T_h be two domains in $\mathbb{R}^d$, and $F : T_h \to T_{ref}$ a map that scales with h of order l. Consider $u \in W^{k,p_k} \cap C(\Omega, \mathcal{M})$, and $V \in W^{k,p}(\Omega, u^{-1}T\mathcal{M})$ with $1 \leq k \leq l$ and $p \in [1, \infty]$. Then*

$$\dot{\Theta}_{k,p,T_{ref}}(V \circ F^{-1}) \leq C\, h^{k-\frac{d}{p}} \Theta_{k,p,T_h}(V).$$

Assumption (iii) of Definition 1.37 is not required for the proof of Lemmas 1.38 and 1.40. Instead, it is needed for the following 'inverse' estimate [36].

Lemma 1.41 *Let T_{ref}, T_h be two domains in $\mathbb{R}^d$, and $F : T_h \to T_{ref}$ a map that scales with h of order 1. Consider $u \in W^{1,p} \cap C(T_h, \mathcal{M})$ with $p \in [1, \infty]$. Then*

$$\dot{\theta}_{1,p,T_h}(u) \leq C\, h^{-1+\frac{d}{p}} \dot{\theta}_{1,p,T_{ref}}(u \circ F^{-1}).$$

1.4.1.3 Sobolev Distances

Based on the different definitions of Sobolev maps, the spaces $W^{1,p}(\Omega, \mathcal{M})$ can be endowed with a metric topology. Several of these have been discussed in [19] based on the definitions of [7, 42, 46, 53], with the result that while all of them are proper generalizations of the classical topology on $W^{1,p}(\Omega, \mathbb{R})$ for $1 < p < \infty$, the spaces are not complete for any of these distances.

In the context of colocal derivatives [21] so-called concordant distances are discussed, which have the form

$$\delta_{1,p}(u, v) = \left(\int_{\Omega} D^p(du(x), dv(x))\, dx \right)^{\frac{1}{p}}, \tag{1.8}$$

where D is a distance on the tangent bundle $T\mathcal{M}$ of the target manifold $\mathcal{M}$. Completeness is shown in [21], if D is the Sasaki metric. In the context of geometric finite elements, this construction was also introduced to measure the $W^{1,2}$-error of approximations [32, 36, 39]. However, instead of using the Sasaki metric directly, a different notion based on the Sasaki norm of the horizontal lift connection was introduced. This horizontal lift metric has the desirable property that $W^{1,p}$-geodesics project onto L^p-geodesics, which are themselves families of pointwise geodesics (geodesic homotopies).

Definition 1.42 Let $u, v \in W^{1,p}(\Omega, \mathcal{M}) \cap C(\Omega, B_{\mathrm{inj}(\mathcal{M})})$, and let Γ denote the geodesic homotopy connecting u to v. We set

$$D^p_{1,p}(u, v) := \sum_{\alpha=1}^{d} \int_\Omega \|\nabla_{d^\alpha u} \log_{u(x)} v(x)\|^p_{g(u(x))}\, dx,$$

and

$$d_{W^{1,p}}(u, v) := d_{L^p}(u, v) + D_{1,p}(u, v).$$

While this definition does not define a distance in the classical sense, one can show that it is locally an quasi-infra-metric, i.e., it fulfills the triangle inequality and the symmetry condition up to multiplication with a constant. The proofs for this are based on the so-called uniformity lemma [32, 39]:

Lemma 1.43 (Uniformity Lemma) *Let $q > \max\{p, d\}$, and let K and L be two constants such that $L \leq \mathrm{inj}(\mathcal{M})$ and $KL \leq \frac{1}{\|\mathrm{Rm}\|_\infty}$, where Rm is the Riemann tensor of $\mathcal{M}$. We set*

$$W^{1,q}_K := \left\{ v \in W^{1,q}(\Omega, \mathcal{M}) \ : \ \theta_{1,q,\Omega}(v) \leq K \right\},$$

and denote by $H^{1,p,q}_{K,L}$ an L-ball w.r.t. L^s in $W^{1,q}_K$, where $\frac{1}{s} := \frac{1}{p_1} - \frac{1}{q}$ for p_1 as in Definition 1.36. Let Γ be the geodesic homotopy connecting u to v in $H^{1,p,q}_{K,L}$. Consider a pointwise parallel vector field $V \in W^{1,p} \cap C(\Omega \times [0, 1], \Gamma^{-1}T\mathcal{M})$ along Γ. Then there exists a constant C depending on the curvature of $\mathcal{M}$, the Sobolev constant, and the dimension d of Ω such that

$$\frac{1}{1 + C\,t} \|V(\cdot, 0)\|_{W^{1,p}(\Omega, u^{-1}T\mathcal{M})} \leq \|V(\cdot, t)\|_{W^{1,p}(\Omega, \Gamma(\cdot,t)^{-1}T\mathcal{M})} \leq (1 + C\,t)\|V(\cdot, 0)\|_{W^{1,p}(\Omega, u^{-1}T\mathcal{M})}$$

holds for all $t \in [0, 1]$.

The proof of this lemma follows by differentiating $\|V(\cdot, t)\|_{W^{1,p}(\Omega, \Gamma(\cdot,t)^{-1}T\mathcal{M})}$ with respect to t and using the Hölder and Sobolev inequalities. Note that the uniformity lemma extends to higher-order $W^{k,p}$-Sobolev norms of vector fields as long as the smoothness descriptors of the end point maps u and v are bounded in a slightly higher-order (k, q)-smoothness descriptor ($q > \max\{kp, d\}$).

Using Lemma 1.43 one can show that $D_{1,2}$ is locally equivalent to both the concordant distance induced by the Sasaki metric (1.8), as well as to the distance induced by an embedding. This implies that it is a quasi-infra-metric [36].

Proposition 1.44 ([36]) *On $H^{1,p,q}_{K,L}$ the mapping $d_{W^{1,p}}$ is a quasi-infra-metric. If $\iota : \mathcal{M} \to \mathbb{R}^N$ denotes a smooth isometric embedding, then for all $u, v \in H^{1,p,q}_{K,L}$ there exists a constant depending on the curvature of $\mathcal{M}$, $\|\iota\|_{C^2}$, and K, such that*

$$\|\iota \circ u - \iota \circ v\|_{W^{1,p}(\Omega,\mathbb{R}^N)} \leq C\, d_{W^{1,p}}(u, v).$$

If additionally $d_{L^\infty}(u, v) \leq inj(\mathcal{M})$, then equivalence to both the distances induced by the embedding and the Sasaki metric holds.

Finally, there exists a constant C such that

$$\dot{\theta}_{1,p,\Omega}(v) \leq \dot{\theta}_{1,p,\Omega}(u) + C\, D_{1,p}(u, v).$$

For the context of finite element analysis we note the following scaling properties.

Proposition 1.45 *Let T_{ref}, T_h be two domains in $\mathbb{R}^d$, and $F : T_h \to T_{ref}$ a map that scales with h of order* 1. *Consider $u, v \in W^{1,p} \cap C(T_h, \mathcal{M})$ with $p \in [1, \infty]$. Then*

$$d_{L^p}(u, v) \leq C\, h^{\frac{d}{p}} d_{L^p}(u \circ F^{-1}, v \circ F^{-1})$$

$$D_{1,p}(u, v) \leq C\, h^{\frac{d}{p}-1} D_{1,p}(u \circ F^{-1}, v \circ F^{-1}).$$

The proof follows from the chain rule and the integral transformation formula [32].

1.4.2 Discretization of Elliptic Energy Minimization Problems

We consider the minimization of energies $\mathcal{J}$ in $H \subset W^{1,q}_{\phi}(\Omega, \mathcal{M})$, $q > \max\{2, d\}$, where $\phi : \overline{\Omega} \to \mathcal{M}$ prescribes Dirichlet boundary data and a homotopy class.

$$u \in H : \qquad \mathcal{J}(u) \leq \mathcal{J}(v) \qquad \forall v \in H. \tag{1.9}$$

We assume that the boundary data ϕ is chosen such that there exists a unique (local) solution $u \in W^{r+1,2}(\Omega, \mathcal{M})$ to (1.9). For a condensed discussion of such choices in the context of harmonic maps see, e.g., [41].

1.4.2.1 Ellipticity

The focus will be on $W^{1,2}$-elliptic energies. Following notions from metric space theory [8] we define ellipticity with respect to a class of curves and a choice of distance.

Definition 1.46 (Ellipticity) Let $\mathcal{J} : H \to \mathbb{R}$ be twice continuously differentiable along geodesic homotopies $\Gamma : [0, 1] \ni t \mapsto \Gamma(t) \in H$. We say that $\mathcal{J}$ is

1. $W^{1,2}$-coercive around $u \in H$, if there exists a constant $\lambda > 0$ such that for all geodesic homotopies Γ starting in u the map $t \mapsto \mathcal{J}(\Gamma(t))$ is λ-convex with respect to $D_{1,2}$, i.e.,

$$\mathcal{J}(\Gamma(t)) \geq (1-t)\mathcal{J}(\Gamma(0)) + t\mathcal{J}(\Gamma(1)) - \frac{\lambda}{2}t(1-t)D^2_{1,2}(\Gamma(0), \Gamma(1)),$$

2. $W^{1,2}$-bounded around u if there exists a constant $\Lambda > 0$ such that for all v

$$\mathcal{J}(v) - \mathcal{J}(u) \leq \frac{\Lambda}{2}D^2_{1,2}(v, u),$$

3. locally $W^{1,2}$-elliptic if (1) and (2) hold for all v in a neighborhood of u (the exact type of neighborhood may depend on $\mathcal{J}$).

A prototypical example of a locally elliptic energy between manifolds is the harmonic map energy, for which (1) and (2) hold around a minimizer u for maps in $H^{1,2,q}_{K,L}$ defined as in Lemma 1.43.

We consider the approximation of (1.9) by geometric finite elements. Assume that we have a conforming grid $\mathcal{G}$ on Ω.

Definition 1.47 We say that a conforming grid $\mathcal{G}$ for the domain $\Omega \subset \mathbb{R}^d$ is of width h and order r, if for each element T_h of $\mathcal{G}$ there exists a C^∞-diffeomorphism $F_h : T_h \to T_{\text{ref}}$ to a reference element $T_{\text{ref}} \subset \mathbb{R}^d$ that scales with h of order r.

Let $S_{h,r}(\Omega, \mathcal{M}) \subset H$ be a conforming discrete approximation space for a grid $\mathcal{G}$ on Ω of width h and order r. This can be, e.g., any of the constructions presented in Sect. 1.2.

Remark 1.48 The conformity assumption $S_{h,r}(\Omega, \mathcal{M}) \subset H$ implies that the boundary data $\phi|_{\partial\Omega}$ can be represented exactly in $S_{h,r}(\Omega, \mathcal{M})$. This part of the assumption may be waived and replaced by a standard approximation argument for boundary data [20, 32].

Consider the discrete approximation of (1.9)

$$u_h \in S_{h,r}(\Omega, \mathcal{M}) : \qquad \mathcal{J}(u_h) \leq \mathcal{J}(v_h) \qquad \forall v_h \in S_{h,r}(\Omega, \mathcal{M}). \tag{1.10}$$

Variations of discrete functions provide a notion of discrete test vector fields $S_{h,r}(\Omega, u_h^{-1}T\mathcal{M})$ along a discrete map $u_h \in S_{h,r}(\Omega, \mathcal{M})$ (see Definition 1.25). We denote by $S_{h,r;0}(\Omega, u_h^{-1}T\mathcal{M})$ the set of all vector fields $V_h \in S_{h,r}(\Omega, u_h^{-1}T\mathcal{M})$ with boundary values equal to zero.

1.4.2.2 Approximability Conditions

In order to control the error between u and u_h, we formulate approximability conditions on the discrete space $S_{h,r}(\Omega, \mathcal{M})$ [32, 39]. This allows to obtain discretization error estimates for all approximation spaces that fulfill these conditions. The conditions will be discussed for geodesic and projection-based finite elements in Sect. 1.4.3.

The first condition is an estimate for the best approximation error in $S_{h,r}(\Omega, \mathcal{M})$, similar to what is used in the Euclidean setting [20].

Condition 1.49 *Let $kp > d$, $r \geq k-1$, and $u \in W^{k,p}(\Omega, \mathcal{M})$ with $u(T_h) \subset B_\rho \subset \mathcal{M}$, where $\rho \leq \operatorname{inj}(\mathcal{M})$, for all elements $T_h \in \mathcal{G}$. For small enough h there exists a map $u_I \in S_h^r$ and constants C with*

$$\dot{\theta}_{l,q,T_h}(u_I) \leq C\, \dot{\theta}_{l,q,T_h}(u) \tag{1.11}$$

for all $k - \frac{d}{p} \leq l \leq k$ and $q \leq \frac{pd}{d-p(k-l)}$, that fulfills on each element $T_h \in \mathcal{G}$ the estimate

$$d_{L^p}(u, u_I) + h\, D_{1,p}(u, u_I) \leq C\, h^k\, \theta_{k,p,T_h}(u). \tag{1.12}$$

Note that the discrete functions in $S_{h,r}(\Omega, \mathcal{M}) \subset H$ are globally only of $W^{1,q}$-smoothness. In analogy to the Euclidean theory we define grid-dependent smoothness descriptors.

Definition 1.50 Let $k \geq 1$, $p \in [1, \infty]$, and $u \in C(\Omega, \mathcal{M})$ with $u_{|T_h} \in W^{k,p}(T_h, \mathcal{M})$ for all elements T_h from the grid $\mathcal{G}$. We define the grid-dependent smoothness descriptor of u by

$$\dot{\theta}_{k,p,\mathcal{G}}(u) := \left(\sum_{T_h \in \mathcal{G}} \dot{\theta}^p_{k,p,T_h}(u) \right)^{\frac{1}{p}}.$$

Analogously, for a function $v \in C(\Omega, \mathcal{M})$ with $v_{|T_h} \in W^{k,p_k}(T_h, \mathcal{M})$ for all elements $T_h \in \mathcal{G}$, p_k as in Definition 1.36, and a vector field V along v such that $V \in W^{k,p}(T_h, v^{-1}T\mathcal{M})$ for all $T_h \in \mathcal{G}$, we define the grid-dependent smoothness descriptor of V by

$$\dot{\Theta}_{k,p,\mathcal{G}}(V) := \left(\sum_{T_h \in \mathcal{G}} \dot{\Theta}^p_{k,p,T_h}(V) \right)^{\frac{1}{p}}.$$

By summation over all elements, estimates like (1.11) and (1.12) imply corresponding global bounds involving grid-dependent smoothness descriptors.

As we need to approximate the generalized test functions as well, we also assume a best approximation error estimate between general $W^{2,2}$ vector fields and variations of discrete maps.

Condition 1.51 *For any element T_h from the grid $\mathcal{G}$, let $S_h^r(T_h, \mathcal{M}) \subset W^{2,p_2}(T_h, \mathcal{M})$ for p_2 as in Definition 1.36 with $p = k = 2$. Given any $u_h \in S_h^r(T_h, \mathcal{M})$ and $V \in W^{2,2}(T_h, u_h^{-1}T\mathcal{M})$, there exists a family of maps $v_h(t) \in S_h^r(T_h, \mathcal{M})$ with $v_h(0) = u_h$ and constants C such that for $V_I = \frac{d}{dt}v_h(0)$ the estimates*

$$\Theta_{2,2,T_h}(V_I) \leq C\ \Theta_{2,2,T_h}(V)$$

and

$$\|V - V_I\|_{W^{1,2}(T_h, u_h^{-1}T\mathcal{M})} \leq Ch\ \Theta_{2,2,T_h}(V)$$

hold.

Conditions 1.49 and 1.51 are sufficient to prove approximation errors locally close to the exact solution u of (1.9). This locality can be achieved by restricting the set of discrete functions $S_{h,r}(\Omega, \mathcal{M})$ in (1.10) by additional bounds. We still obtain a meaningful result if we can show that the discrete solution u_h stays away from these bounds. In order to do this, we need the following inverse estimate.

Condition 1.52 *On a grid $\mathcal{G}$ of width h and order r, under the additional assumption that $F_h^{-1} : T_{ref} \to T_h$ scales with order 2 for every $T_h \in \mathcal{G}$, for $p, q \in [1, \infty]$ there exists a constant C such that*

$$\dot{\theta}_{1,p,T_h}(v_h) \leq C\ \ h^{-d\max\left\{0,\frac{1}{q}-\frac{1}{p}\right\}}\ \dot{\theta}_{1,q,T_h}(v_h)$$

$$\dot{\theta}_{2,p,T_h}(v_h) \leq C\ \dot{\theta}^2_{1,2p,T_h}(v_h) + C\ h^{-1-d\max\left\{0,\frac{1}{q}-\frac{1}{p}\right\}}\ \dot{\theta}_{1,q,T_h}(v_h)$$

for any $v_h \in S_{h,r}(T_h, \mathcal{M})$ with $v(T_h) \subset B_\rho$ for ρ small enough.

Once we can show that the discrete solution is indeed close to the continuous one, we can infer even stronger bounds on higher derivatives of it from the exact solution. In order to do so, we need inverse estimates on differences of discrete functions, i.e., vector fields of the form $\log_{v_h} w_h$ with $v_h, w_h \in S_{h,r}(T_h, \mathcal{M})$. Note that these are not discrete vector fields in the sense of Definition 1.25 themselves.

Condition 1.53 *On a grid $\mathcal{G}$ of width h and order r, under the additional assumption that $F_h^{-1} : T_{ref} \to T_h$ scales with order 2 for every $T_h \in \mathcal{G}$, for $p, q \in [1, \infty]$ there exist constants $C, \hat{C}$ such that*

$$\|\log_{v_h} w_h\|_{L^p(T_h, v_h^{-1}T\mathcal{M})} \le C\, h^{-d\max\left\{0,\frac{1}{q}-\frac{1}{p}\right\}} \|\log_{v_h} w_h\|_{L^q(T_h, v_h^{-1}T\mathcal{M})} \tag{1.13}$$

$$\begin{aligned}|\log_{v_h} w_h|_{W^{2,p}(T_h, u_h^{-1}T\mathcal{M})} \le{}& C\dot{\theta}_{2,p,T_h}(v_h) + C\dot{\theta}^2_{1,2p,T_h}(w_h) \\ &+ Ch^{-1-d\max\left\{0,\frac{1}{q}-\frac{1}{p}\right\}} \|\log_{v_h} w_h\|_{W^{1,q}(T_h, v_h^{-1}T\mathcal{M})} \\ &+ \hat{C}h^{-2}\|\log_{v_h} w_h\|_{L^p(T_h, v_h^{-1}T\mathcal{M})}\end{aligned} \tag{1.14}$$

for any $v_h, w_h \in S_{h,r}(T_h, \mathcal{M})$ with $w(T_h), v(T_h) \subset B_\rho$ for ρ small enough. For $r = 1$, the constant $\hat{C}$ must be zero.

1.4.2.3 $W^{1,2}$-Error Bounds

We recall the $W^{1,2}$-discretization error bounds given in [32], in particular the generalized Céa Lemma:

Lemma 1.54 *Assume that $u \in H$ is a minimizer of $\mathcal{J} : H \to \mathbb{R}$ w.r.t. variations along geodesic homotopies in H, and that $\mathcal{J}$ is elliptic along geodesic homotopies starting in u.*
For $K > \theta_{1,q,\Omega}(u)$, $L \le \operatorname{inj}(\mathcal{M})$, and $KL \le \frac{1}{|\mathrm{Rm}|_\infty}$ let $H^{1,2,q}_{K,L}$ be defined as in Lemma 1.43, and consider a subset $V_h \subset H \cap H^{1,2,q}_{K,L}$ such that

$$w = \operatorname*{arg\,min}_{v \in V_h} \mathcal{J}(v)$$

exists. Then

$$D_{1,2}(u, w) \le (1 + C)^2 \sqrt{\frac{\Lambda}{\lambda}} \inf_{v \in V_h} D_{1,2}(u, v)$$

holds, where C is the constant appearing in Lemma 1.43.

A combination of this version of Céa's lemma with Condition 1.49 yields the $W^{1,2}$-error estimate shown in [32].

Theorem 1.55 *Let $2(r + 1) > d$, and $r \ge 1$. Assume that $u \in W^{r+1,2}_\phi(\Omega, \mathcal{M})$ is a minimizer of $\mathcal{J} : H \to \mathbb{R}$ w.r.t. variations along geodesic homotopies in H, and that $\mathcal{J}$ is elliptic along geodesic homotopies starting in u.*

- *For a conforming grid $\mathcal{G}$ of width h and order r set $V_h := H \cap S_{h,r}$. Assume that the boundary data $\phi_{|\partial\Omega}$ is such that V_h is not empty.*
- *Let $u_I \in S^r_h$ be the approximating map from Condition 1.49, and let K be a constant such that*

$$K \ge \max\{\theta_{1,q,\Omega}(u_I), \theta_{1,q,\Omega}(u)\}.$$

- *Assume that h is small enough such that $u_I \in H^{1,2,q}_{K,L}$, where $H^{1,2,q}_{K,L}$ is defined as in Lemma 1.43.*

  Then the discrete minimizer

$$u_h := \operatorname*{arg\,min}_{v_h \in V_h \cap H^{1,2,q}_{K,L}} \mathcal{J}(v_h)$$

  fulfills the a priori error estimate

$$D_{1,2}(u, u_h) \leq Ch^r \theta_{r+1,2,\Omega}(u).$$

Note that the proofs of Céa's Lemma 1.54 and Theorem 1.55 do not use a first variation of the discrete problem. Thus, they do not conflict with the additional K- and L-bounds on the discrete functions. We can show using Condition 1.52 that we can choose q such that the restricted solution $u_h \in V_h \cap H^{1,2,q}_{K,L}$ stays away from the K and L bounds and is thus indeed a local solution in V_h:

Lemma 1.56 ([36]) *Assume that Condition 1.52 holds, and that the grid $\mathcal{G}$ fulfills the additional assumption that $F_h^{-1} : T_{ref} \to T_h$ scales with order 2 for every $T_h \in \mathcal{G}$. Let $u \in W^{r+1,2}_{\phi}(\Omega, \mathcal{M})$, $2(r+1) > d$, $u_h \in H \cap S^r_h$ with $\dot{\theta}_{1,q,\Omega}(u_h) \leq K$, and*

$$D_{1,2}(u, u_h) \leq C\ h^r \theta_{r+1,2,\Omega}(u).$$

Then we can choose $q > \max\{d, 2\}$ such that

$$D_{1,q}(u, u_h) \leq C\ h^\delta$$

holds with some $\delta > 0$ and a constant C depending on K and u. Note that this also implies

$$d_{L^\infty}(u, u_h) \leq C\ h^\delta,$$

as well as $\theta_{1,q,\Omega}(u_h) < K$, as long as $K > \theta_{1,q,\Omega}(u)$ and h small enough.

The proof follows from L^p-interpolation for q between $\max\{2, d\}$ and s_r as in Definition 1.36, with $p = 2$ and $k = r + 1$.

1.4.2.4 L^2-Error Bounds

A generalization of the Aubin–Nitsche lemma has been given in [39]. It is applicable for energies that are "predominantly quadratic", by which we mean that there is a bound on the third variation of the energy. We will in the following denote the k-th variation of the energy (cf. e.g. [30, 68]) by $\delta^k \mathcal{J}$, omitting k for $k = 1$.

Definition 1.57 Let $q > \max\{d, 2\}$ and $\mathcal{J} : H \to \mathbb{R}$ be an energy functional. We say that $\mathcal{J}$ is predominantly quadratic if $\mathcal{J}$ is C^3 along geodesic homotopies, and if for any $v \in H \cap W_K^{1,q}$, and vector fields U, V along v

$$\begin{aligned}&|\delta^3\mathcal{J}(v)(U, V, V)|\\ &\quad\le C(K, \mathcal{M})\|U\|_{W^{1,p}(\Omega, v^{-1}T\mathcal{M})}\|V\|_{W^{1,2}(\Omega, v^{-1}T\mathcal{M})}\|V\|_{W^{o,t}(\Omega, v^{-1}T\mathcal{M})},\end{aligned}$$

with

$$\begin{aligned}&p = \infty && \text{if } d = 1,\\ &p \in [1, \infty) && \text{if } d = 2,\\ &\frac{1}{p} = \frac{1}{2} - \frac{1}{d} && \text{if } d > 2,\end{aligned}$$

and either $(o, t) = (1, 2)$, or $o = 0$ and $t \le d$.

Note that as long as the coefficient functions of a semilinear PDE coming from a minimization problem are smooth enough and bounded, the corresponding energy is predominantly quadratic [37]. A prototypical example is again the harmonic map energy.

We consider the variational formulation of the problems (1.9) and (1.10)

$$u \in H: \qquad \delta\mathcal{J}(u)(V) = 0 \qquad \forall V \in W_0^{1,2}(\Omega, u^{-1}TM), \tag{1.15}$$

$$u_h \in S_h^r(\Omega, \mathcal{M}): \qquad \delta\mathcal{J}(u_h)(V_h) = 0 \qquad \forall V_h \in S_{h;0}^r(\Omega, u_h^{-1}T\mathcal{M}). \tag{1.16}$$

In the Euclidean setting, Galerkin orthogonality is an important tool for proving L^2-error estimates. It is obtained by inserting a discrete test function into the continuous problem and subtracting the variational formulations. As in the manifold setting the test spaces depend on the base functions, we obtain only an integrated type of Galerkin orthogonality with respect to a transport of the test vector field:

Proposition 1.58 *Let u and u_h be solutions to* (1.15) *and* (1.16)*, respectively, and let Γ be the geodesic homotopy joining u and u_h. Then, for any transport V_h : $[0, 1] \to W_0^{1,2}(\Omega, \Gamma(t)^{-1}T\mathcal{M})$ along Γ of any discrete vector field $V_h(1) = V_{h,1} \in S_h(\Omega, u_h^{-1}T\mathcal{M})$ holds*

$$\int_0^1 \delta^2\mathcal{J}(\Gamma(t))(V_h(t), \dot{\Gamma}(t)) + \delta\mathcal{J}(\Gamma(t))(\nabla_t V_h(t))\, dt = 0.$$

While there are different choices of transports possible, we employ parallel transport to let the second term on the left hand side vanish. A second tool is the definition of the so-called adjoint problem. For nonlinear energies this is a linearization of problem (1.15) with a right hand side that is given by the difference of the solutions u and u_h to (1.15) and (1.16), respectively. In the context of Riemannian manifolds we call this linearization the deformation problem, as it acts on vector fields:

Find $W \in W^{1,2}(\Omega, u^{-1}T\mathcal{M})$ such that

$$\delta^2 \mathcal{J}(u)(W, V) = -(V, \log_u u_h)_{L^2(\Omega, u^{-1}T\mathcal{M})} \quad \forall V \in W_0^{1,2}(\Omega, u^{-1}T\mathcal{M}). \tag{1.17}$$

Based on generalizations of the standard tools in the Aubin–Nitsche lemma to Riemannian manifold codomains, an L^2-error estimate has been proven in [39]:

Theorem 1.59 *Let the assumptions of Theorem 1.55 and Lemma 1.56 be fulfilled with $q > \max\{d, 4\}$ chosen as in Lemma 1.56 for a predominantly quadratic energy $\mathcal{J}$, and a discrete set S_h^r fulfilling Condition 1.51.*

Let u_h be a minimizer of $\mathcal{J}$ in $S_h^r \cap H_{K,L}^{1,2,q}$ under the boundary and homotopy conditions. We assume that

$$\theta_{2,p_2,G}(u_h) \leq K_2 \tag{1.18}$$

for a constant K_2 and p_2 as in Definition 1.36 with $p = k = 2$. Finally, suppose that the deformation problem (1.17) is H^2-regular, i.e., that its solution W fulfills

$$\|W\|_{W^{2,2}(\Omega, u^{-1}T\mathcal{M})} \leq C \, \|\log_u u_h\|_{L^2(\Omega, u^{-1}T\mathcal{M})}.$$

Then there exists a constant C such that

$$d_{L^2}(u, u_h) \leq C \, h^{r+1} \left(\theta_{r+1,2,\Omega}(u) + \theta_{r+1,2,\Omega}^2(u)\right).$$

The proof follows the basic idea of the Aubin–Nitsche trick: The vector field describing the error $\log_u u_h$ is inserted as a test vector field in the deformation problem:

$$d_{L^2}(u, u_h) = -\delta^2 \mathcal{J}(u)(W, \log_u u_h). \tag{1.19}$$

Further, we transport the solution W of the deformation problem to the discrete solution u_h, and interpolate along u_h. This produces a discrete vector field W_I for which the integrated Galerkin orthogonality holds. Adding this to the right hand side of (1.19), we integrate again and obtain

$$\begin{aligned} d_{L^2}^2(u, u_h) = &\int_0^1 \int_0^t \delta^3 \mathcal{J}(\Gamma(s)) \left(\frac{s}{t} W_I(s) + \left(1 - \frac{s}{t}\right) W(s), \dot{\Gamma}(s), \dot{\Gamma}(s)\right) \, ds \, dt \\ &+ \int_0^1 \int_0^t \frac{1}{t} \delta^2 \mathcal{J}(\Gamma(s))(W_I(s) - W(s), \dot{\Gamma}(s)) \, ds \, dt. \end{aligned}$$

The first integral can be estimated by norms of W and $\dot{\Gamma}$ since $\mathcal{J}$ is assumed to be predominantly quadratic. The second integral is bounded using the ellipticity assumption. The arising norms are estimated using Theorem 1.55, Condition 1.51 for vector field interpolation, and the H^2-regularity.

Note that in order to transport H^2-norms of vector fields the a priori bound (1.18) is needed. This additional a priori bound needs either to be proven directly for the discrete solution u_h, or can be shown in general if Condition 1.53 is fulfilled.

Proposition 1.60 ([39]) *Let $r \geq 1$ and $2(r+1) > d$. Define s_i, p_i as in Definition 1.36 with $p = 2$ and $k = r+1$, and let $q > \max\{4, d\}$ fulfill $2p_2 \leq q < s_r$. Suppose that the grid $\mathcal{G}$ on Ω is of width h and order r, that $F_h^{-1} : T_{ref} \to T_h$ scales with order 2 for all elements T_h, and that S_h^r fulfills Conditions 1.49, 1.52, and 1.53. For $v \in W^{r+1,2}(\Omega, \mathcal{M})$, and $v_h \in S_{h,r} \cap H^{1,2,q}_{K,L}$ with $v_I|_{\partial\Omega} = v_h|_{\partial\Omega}$, we assume the relation*

$$D_{1,2}(v, v_h) \leq C\, h^r \theta_{r+1,2,\Omega}(v).$$

Then there exists a constant K_2 depending on v and K but independent of h such that

$$\theta_{2,p_2,G}(v_h) \leq K_2$$

if h is small enough.

1.4.3 Approximation Errors

In this section we discuss Conditions 1.49–1.53 for two of the geometric methods—geodesic and projection-based finite elements.

1.4.3.1 Geodesic Finite Elements

Detailed proofs that geodesic finite elements fulfill the Conditions 1.49–1.53 have appeared in [32] and [38, Section 1.3].

The best-approximation condition 1.49 can be shown for $u_I \in S^{\text{geo}}_{h,r}(\Omega, \mathcal{M})$ being the geodesic interpolant of u. In particular, the a priori bound (1.11) can be shown by contradiction. The interpolation estimate (1.12) has appeared in [32], and is based on the first-order optimality condition (1.3) using a Taylor expansion of the vector field $\log_{u_I(x)} u(y)$. Both estimates are done on a reference element, and use the scaling properties in Sect. 1.4.1.2.

The proof of best-approximation estimates for vector fields, Condition 1.51, follows for interpolations of continuous vector fields analogously to the proof of (1.12) for geodesic finite elements, as discrete vector fields are themselves geodesic interpolations for the pseudo-Riemannian codomain $T\mathcal{M}$ with the horizontal lift metric (cf. Sect. 1.3.2). If the vector fields are discontinuous, so that geodesic interpolation is not well-defined, a weakened version of Condition 1.51 can be proven using mollifier smoothing [38].

The inverse estimates, Condition 1.52, follow from differentiation of (1.3), properties of the exponential map, and norm equivalence in $\mathbb{R}^m$, where m is the number of degrees of freedom on T_h. Note that they do not hold for higher order smoothness descriptors.

The inverse estimate (1.13) for L^p-differences of geodesic finite elements in Condition 1.53 follows from the fact that the point-wise difference of two maps in $u, v \in S_h^{\text{geo}}(\Omega, \mathcal{M})$ can be estimated from above and below by the norm of a vector field of the form

$$\sum_{i=1}^{m} \varphi_i(\xi) d \log_{u(\xi)} u_i(W_i).$$

The class of these vector fields is isomorphic to the finite-dimensional product space $\Pi_{i=1}^m T_{u_i}\mathcal{M}$. Thus, standard arguments for norm equivalences can be applied on a reference element and scaled to the grid. The higher-order estimate (1.14) follows from a similar argument using the differentiated versions of (1.3) and the scaling of the Lagrange basis functions φ_i and their derivatives.

1.4.3.2 Projection-Based Finite Elements

Detailed proofs that projection-based finite elements fulfill the a priori conditions 1.49–1.53 have appeared in [33].

Introduce the Lagrange interpolation operator

$$Q_{\mathbb{R}^N} : C(\Omega, \mathbb{R}^N) \to C(\Omega, \mathbb{R}^N), \qquad Q_{\mathbb{R}^N} v = \sum_{i \in I} v(\xi_i)\varphi_i,$$

and the projected operator

$$Q_{\mathcal{M}} : C(\Omega, \mathbb{R}^N) \to C(\Omega, \mathcal{M}), \qquad Q_{\mathcal{M}} = \mathcal{P} \circ Q_{\mathbb{R}^N},$$

where $\mathcal{P}$ is the superposition operator induced by the closest-point projection P from $\mathbb{R}^N$ onto $\mathcal{M}$. One can bound the discrete maps $Q_{\mathcal{M}} v$ in terms of smoothness descriptors of $Q_{\mathbb{R}^n} v$ using boundedness of the projection operator, and the Galiardo–Nirenberg–Sobolev inequality for higher-order derivatives. This implies the a priori bound (1.11) of Condition (1.49) as well as the inverse estimates in Condition 1.52 and Condition 1.53 using inverse estimates for $Q_{\mathbb{R}^n}$. The interpolation error (1.12) follows from properties of the standard Lagrange interpolation $Q_{\mathbb{R}^n}$ using the triangle inequality, and the identity $Q_{\mathbb{R}^n} \circ Q_{\mathcal{M}} = Q_{\mathbb{R}^n}$. Analogous but even simpler to prove is Condition 1.51, as the projection operator for vector fields is actually linear.

1.5 Numerical Examples

We close by showing three numerical examples that demonstrate the use of geometric finite elements for different standard problems. All these are problems with optimization formulations. After discretization, they can be written as algebraic minimization problems on the product manifold $\mathcal{M}^{|I|}$ (but see [57] for a discussion of the subtleties involved), where $|I|$ is the global number of Lagrange points or control points. This minimization problem is solved using the Riemannian trust-region method introduced in [2] together with the inner monotone multigrid solver described in [56]. Gradient and Hessian of the energy functional are computed using the ADOL-C automatic differentiation software [65], and the formula derived in [3] to obtain the Riemannian Hessian matrix from the Euclidean one.

1.5.1 Harmonic Maps into the Sphere

We first show measurements of the L^2 and H^1 discretization errors of projection-based and geodesic finite elements for harmonic maps into S^2, confirming the theoretical predictions of Chapter 1.4. The results of this section have previously appeared in [33].

As domain we use the square $\Omega = (-5, 5)^2$, and we prescribe Dirichlet boundary conditions. As Dirichlet values we demand the values of the inverse stereographic projection function

$$p_{\mathrm{st}} : \mathbb{R}^2 \to S^2, \qquad\qquad p_{\mathrm{st}}(x) := \left(\frac{2x_1}{|x|^2+1}, \frac{2x_2}{|x|^2+1}, \frac{|x|^2-1}{|x|^2+1} \right)^T,$$

restricted to $\partial\Omega$. This function is in C^∞, and we can therefore hope for optimal discretization error orders. Indeed, it is shown in [13] (see also [50]) that this function is actually a minimizer of the harmonic energy in the set of functions that are connected to p_{st} by continuous deformations, and we therefore have a closed-form reference solution to compare with.

We discretize the domain with the grid shown in Fig. 1.3, and create a sequence of grids by refining the initial grid uniformly up to six times. We then compute harmonic maps in spaces of projection-based and geometric finite elements of orders $r = 1, 2, 3$ using the algorithm described above.

The Riemannian trust-region solver is set to iterate until the maximum norm of the correction drops below 10^{-6}. We then compute errors

$$e_r^k = \|v_r^k - p_{\mathrm{st}}\|, \qquad k = 0, \dots, 6, \qquad r = 1, 2, 3,$$

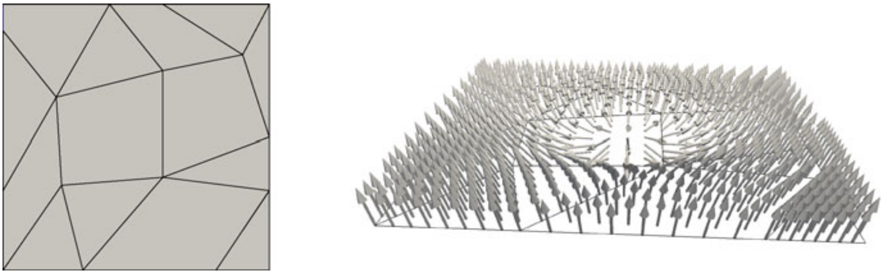


Fig. 1.3 Approximating harmonic maps from a square into the unit sphere in $\mathbb{R}^3$. Left: coarsest grid. Right: function values

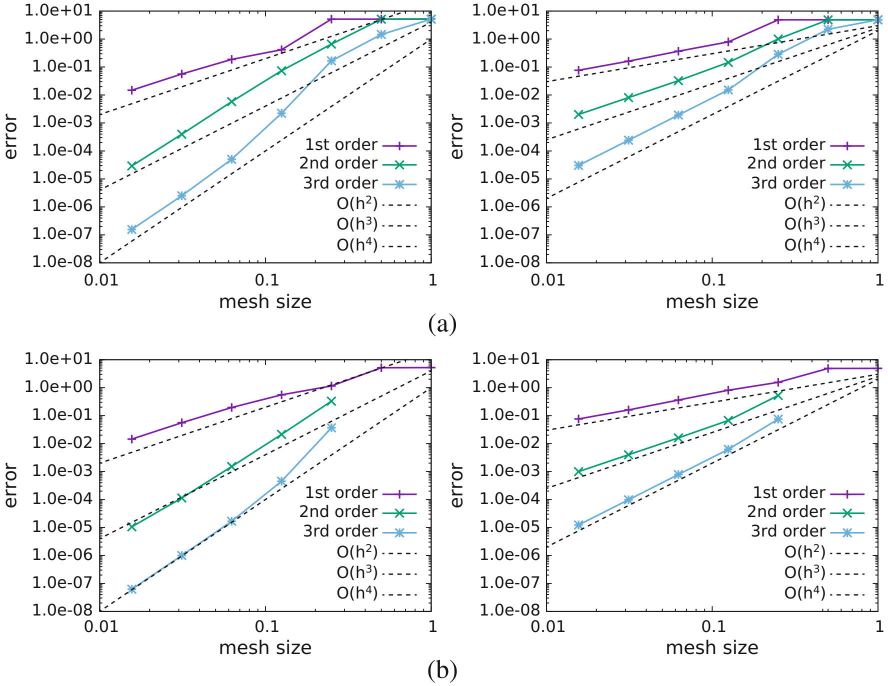
where v_r^k is the discrete solution, and k is the number of grid refinement steps. The norm $\|\cdot\|$ is either the norm in $L^2(\Omega, \mathbb{R}^3)$, or the semi-norm in $H^1(\Omega, \mathbb{R}^3)$; in other words, we interpret the functions as Sobolev functions in the sense of Definition 1.33.

Sixth-order Gaussian quadrature rules are used for the integrals, but note that since geometric finite element functions are not piecewise polynomials in $\mathbb{R}^3$, a small additional error due to numerical quadrature remains.

Figure 1.4 shows the errors e_r^k as functions of the normalized mesh size h both for projection-based finite elements (Fig. 1.4a) and for geodesic finite elements (Fig. 1.4b). We see that for r-th order finite elements the L^2-error decreases like h^{r+1}, and the H^1-error decreases like h^r. Hence we can reproduce the optimal convergence behavior predicted by Theorems 1.55 and 1.59.

Comparing the two discretizations, one can see that while the same asymptotic orders are obtained, the constant is slightly better for geodesic finite elements. On the other hand, one can see that the graphs in Fig. 1.4b do not contain values for the two coarsest grids and approximation orders 2 and 3. This is because the minimization problem that defines geodesic interpolation was actually ill-defined on at least one grid element in these cases. The problem does not happen for projection-based finite elements for this example. This is an effect of the increased radius of well-posedness for projection-based finite elements into S^2, briefly explained in Remark 1.14.

The decisive argument for projection-based finite elements for this scenario, however, is run-time. Figure 1.5 plots the total time needed to compute the harmonic energy for the different finite element spaces and grid resolutions, computed on a standard laptop computer. Projection-based finite elements need only about 10 % of the time of geodesic finite elements. This is of course because projection-based interpolation is given by a simple closed-form formula in the case of $\mathcal{M} = S^2$, whereas for geodesic finite elements it involves numerically solving a small minimization problem (1.2). In practical applications of sphere-valued problems, projection-based finite elements are therefore typically preferable to geodesic finite elements. Note, however, that the run-time difference very much depends on the target space $\mathcal{M}$. In [33] it is shown, e.g., that there is hardly any difference if $\mathcal{M} = \mathrm{SO}(3)$.

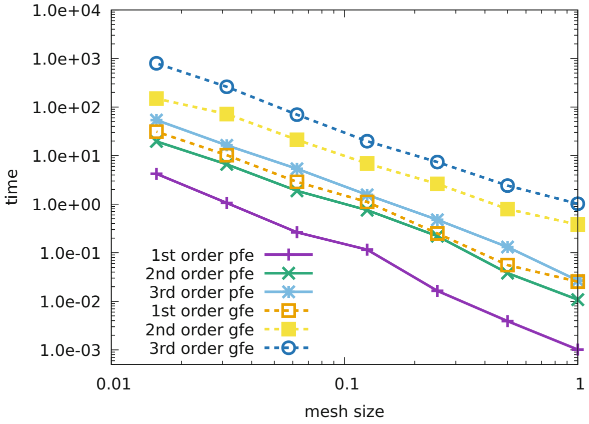


(a)

(b)

Fig. 1.4 Discretization errors for a harmonic map into S^2 as a function of the normalized grid edge length. Left: L^2-norm. Right: H^1-semi-norm. The black dashed reference lines are at the same positions for both discretizations. (**a**) Projection-based finite elements. (**b**) Geodesic finite elements

Fig. 1.5 Wall-time needed to compute the harmonic energy on seven different grids and with approximation orders $r = 1, 2, 3$. Solid: projection-based finite elements. Dashed: geodesic finite elements



1.5.2 Magnetic Skyrmions in the Plane

In the second example we consider a model of chiral magnetic Skyrmions in the plane as investigated, e.g., by Melcher [50]. Magnetic Skyrmions are quasi-particles that appear as soliton solutions in micromagnetic descriptions of ferromagnetic materials [25]. These particles are of considerable technological interest, because they allow to describe magnetization patterns that are kept stable by topological restrictions.

Following [50], we consider a two-dimensional domain Ω and a field of magnetization vectors $\mathbf{m} : \Omega \to S^2$. Fixing the values of $\mathbf{m}$ on the entire boundary of Ω splits the set of continuous functions $\mathbf{m}$ into pairwise disconnected homotopy classes. For any such class we are interested in finding a local minimizer of the energy functional

$$E(\mathbf{m}) = \int_\Omega e(\mathbf{m})\,dx = \int_\Omega \left[\frac{1}{2}|\nabla \mathbf{m}|^2 + \kappa \mathbf{m} \cdot (\nabla \times \mathbf{m}) + \frac{h}{2}|\mathbf{m} - \mathbf{e}_3|^2\right] dx$$

in that class. The two material parameters $\kappa \neq 0$ and $h > 0$ are the normalized Dzyaloshinskii–Moriya constant and the normalized magnetic field strength, respectively.

Inspired by the works of [16], who investigated lattices of Skyrmions, we pick Ω to be a regular hexagon with edge length 1. As in [16], we prescribe Dirichlet boundary values

$$\mathbf{m}(x) = (1, 0, 0)^T \qquad \text{on } \partial\Omega,$$

and we are looking for minimizers in the class of functions with topological charge $+1$.

We discretize the domain by six equilateral triangles, and create a hierarchy of grid by uniform refinement. For the discretization we pick projection-based finite elements, for their advantage in speed. Figure 1.6 shows that the solution really is a soliton. Its diameter can be controlled with the parameter h/κ^2.

Note that the Skyrmion problem is covered by our a priori error theory, as long as $\kappa^2 < h$, and the continuous minimizer solution $\mathbf{m}^\star$ fulfills

$$\left(\int_\Omega e(\mathbf{m}^\star)^q\,dx\right)^{\frac{1}{q}} < \frac{1}{2C^2(q,\Omega)}\left(1 - \frac{\kappa^2}{h}\right),$$

where $q > 1$, and $C(q,\Omega)$ denotes the constant of the embedding $H_0^1(\Omega) \hookrightarrow L^{\frac{2q}{q-1}}(\Omega)$.

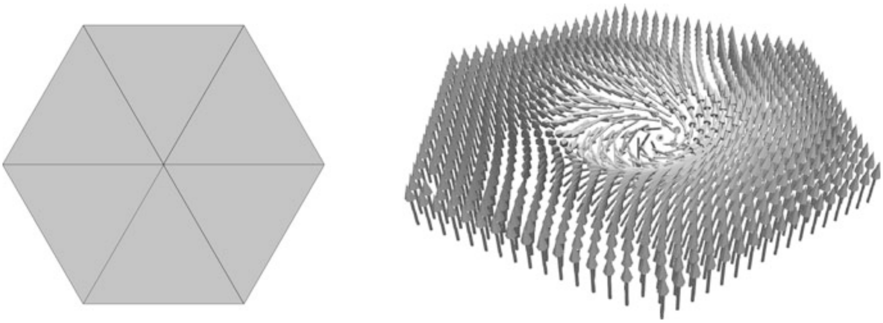


Fig. 1.6 Left: coarse grid; Right: Skyrmion solution

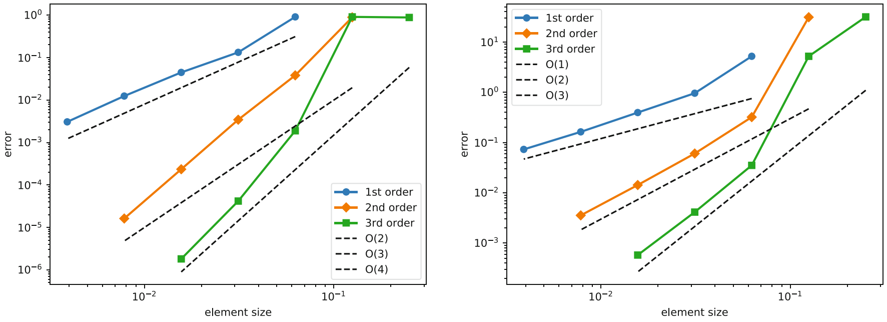


Fig. 1.7 Discretization error of the Skyrmion example. Left: error in the L^2 norm. Right: error in the H^1 semi-norm

We numerically determine the discretization error behavior by comparing with a numerical reference solution obtained by 10, 9, and 8 steps of uniform refinement for approximation orders 1, 2, and 3, respectively. Figure 1.7 plots the errors in the L^2 norm and the H^1 semi-norm with respect to the standard embedding of S^2 into $\mathbb{R}^3$, for finite elements with approximation orders 1, 2, and 3. These figures show that optimal orders are obtained in practice. The reason for the slightly superoptimal behavior of the L^2 error is unclear.

1.5.3 Geometrically Exact Cosserat Plates

We finally show an example with the target space $\mathcal{M} = \mathrm{SO}(3)$, which has appeared previously in [59]. For this we simulate torsion of a long elastic strip, which is modeled by a field $m : \Omega \to \mathbb{R}^3$ of midsurface displacements, and a field $R : \Omega \to \mathrm{SO}(3)$ of microrotations. Stable configurations are described as minimizers of an energy involving the first derivatives of m and R. One short edge is clamped, and using prescribed displacements, the other short edge is then rotated around the center line of the strip, to a final position of three full revolutions. The example hence demonstrates that geometric finite elements can discretize Cosserat materials with arbitrarily large rotations.

Let $\Omega = (0, 100)\,\mathrm{mm} \times (-5, 5)\,\mathrm{mm}$ be the parameter domain, and γ_0 and γ_1 be the two short ends. Let R_3 be the third column of the matrix R. We clamp the shell on γ_0 by requiring

$$m(x, y) = (x, y, 0), \qquad R_3 = (0, 0, 1)^T \qquad \text{on } \gamma_0,$$

and we prescribe a parameter dependent displacement

$$m_t(x, y) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos 2\pi t & -\sin 2\pi t \\ 0 & \sin 2\pi t & \cos 2\pi t \end{pmatrix} \begin{pmatrix} x \\ y \\ 0 \end{pmatrix} \qquad (R_t)_3 = \begin{pmatrix} 0 \\ -\sin 2\pi t \\ \cos 2\pi t \end{pmatrix} \qquad \text{on } \gamma_1.$$

For each increase of t by 1 this models one full revolution of γ_1 around the shell central axis. Homogeneous Neumann boundary conditions are applied to the remaining boundary degrees of freedom. The material parameters are given in Table 1.1. We discretize the domain with 10 × 1 quadrilateral elements, and use second-order geodesic finite elements to discretize the problem.

The result is pictured in Fig. 1.8 for several values of t. Having little bending stiffness, the configuration stays symmetric for the first few rounds. After about three full revolutions, a breaking of the symmetry can be observed. Figure 1.9 shows one configuration of the plate with the microrotation field.

Table 1.1 Material parameters for the twisted strip

h [mm]	μ [N/m^2]	λ [N/m^2]	μ_c [N/m^2]	L_c [mm]	q [1]
2	$5.6452 \cdot 10^9$	$2.1796 \cdot 10^9$	0	$2 \cdot 10^{-3}$	2

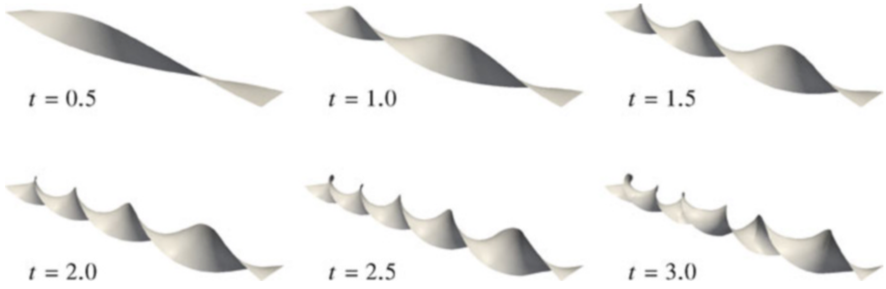


Fig. 1.8 Twisted rectangular strip at different parameter values t, with t equal to the number of revolutions

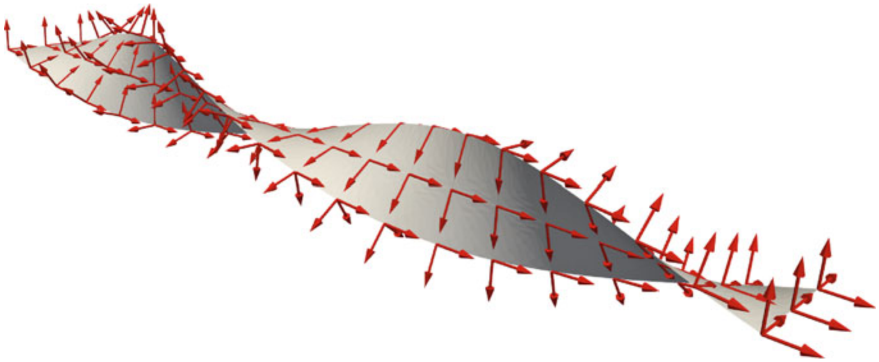


Fig. 1.9 Twisted strip with the microrotation field R visualized as orthonormal director frame field

References

1. Abatzoglou, T.J.: The minimum norm projection on C^2-manifolds in $\mathbb{R}^n$. Trans. Am. Math. Soc. **243**, 115–122 (1978)
2. Absil, P.A., Mahony, R., Sepulchre, R.: Optimization Algorithms on Matrix Manifolds. Princeton University Press, Princeton (2008)
3. Absil, P.A., Mahony, R., Trumpf, J.: An extrinsic look at the Riemannian Hessian. In: Geometric Science of Information. Lecture Notes in Computer Science, vol. 8085, pp. 361–368. Springer, Berlin (2013)
4. Absil, P.A., Gousenbourger, P.Y., Striewski, P., Wirth, B.: Differentiable piecewise-Bézier surfaces on Riemannian manifolds. SIAM J. Imaging Sci. **9**(4), 1788–1828 (2016)
5. Alouges, F.: A new algorithm for computing liquid crystal stable configurations: the harmonic mapping case. SIAM J. Numer. Anal. **34**(5), 1708–1726 (1997)
6. Alouges, F., Jaisson, P.: Convergence of a finite element discretization for the landau–lifshitz equations in micromagnetism. Math. Models Methods Appl. Sci. **16**(2), 299–316 (2006)
7. Ambrosio, L.: Metric space valued functions of bounded variation. Ann. Sc. Norm. Super. Pisa Cl. Sci. **17**(3), 439–478 (1990)
8. Ambrosio, L., Gigli, N., Savaré, G.: Gradient Flows in Metric Spaces and in the Space of Probability Measures. Springer, Berlin (2006)
9. Arsigny, V., Fillard, P., Pennec, X., Ayache, N.: Log-Euclidean metrics for fast and simple calculus on diffusion tensors. Magn. Reson. Med. **56**(2), 411–421 (2006)
10. Arsigny, V., Fillard, P., Pennec, X., Ayache, N.: Geometric means in a novel vector space structure on symmetric positive-definite matrices. SIAM J. Matrix Anal. Appl. **29**(1), 328–347 (2007)
11. Bartels, S., Prohl, A.: Constraint preserving implicit finite element discretization of harmonic map flow into spheres. Math. Comput. **76**(260), 1847–1859 (2007)
12. Baumgarte, T.W., Shapiro, S.L.: Numerical Relativity – Solving Einstein's Equations on the Computer. Cambridge University Press, Cambridge (2010)
13. Belavin, A., Polyakov, A.: Metastable states of two-dimensional isotropic ferromagnets. JETP Lett. **22**(10), 245–247 (1975)
14. Bergmann, R., Laus, F., Persch, J., Steidl, G.: Processing manifold-valued images. SIAM News **50**(8), 1,3 (2017)
15. Berndt, J., Boeckx, E., Nagy, P.T., Vanhecke, L.: Geodesics on the unit tangent bundle. Proc. R. Soc. Edinb. A Math. **133**(06), 1209–1229 (2003)

16. Bogdanov, A., Hubert, A.: Thermodynamically stable magnetic vortex states in magnetic crystals. J. Magn. Magn. Mater. **138**, 255–269 (1994)
17. Buss, S.R., Fillmore, J.P.: Spherical averages and applications to spherical splines and interpolation. ACM Trans. Graph. **20**, 95–126 (2001)
18. Cartan, E.: Groupes simples clos et ouverts et géométrie riemannienne. J. Math. Pures Appl. **8**, 1–34 (1929)
19. Chiron, D.: On the definitions of Sobolev and BV spaces into singular spaces and the trace problem. Commun. Contemp. Math. **9**(04), 473–513 (2007)
20. Ciarlet, P.G.: The Finite Element Method for Elliptic Problems. Elsevier, Amsterdam (1978)
21. Convent, A., Van Schaftingen, J.: Intrinsic colocal weak derivatives and Sobolev spaces between manifolds. Ann. Sc. Norm. Super. Pisa Cl. Sci. **16**(5), 97–128 (2016)
22. Convent, A., Van Schaftingen, J.: Higher order weak differentiability and Sobolev spaces between manifolds (2017). arXiv preprint 1702.07171
23. de Gennes, P., Prost, J.: The Physics of Liquid Crystals. Clarendon Press, Oxford (1993)
24. Farin, G.: Curves and Surfaces for Computer Aided Geometric Design, 2nd edn. Academic, Boston (1990)
25. Fert, A., Reyren, N., Cros, V.: Magnetic skyrmions: advances in physics and potential applications. Nat. Rev. Mater. **2**(17031) (2017)
26. Focardi, M., Spadaro, E.: An intrinsic approach to manifold constrained variational problems. Ann. Mat. Pura Appl. **192**(1), 145–163 (2013)
27. Fréchet, M.: Les éléments aléatoires de nature quelconque dans un espace distancié. Ann. Inst. Henri Poincaré **10**(4), 215–310 (1948)
28. Gawlik, E.S., Leok, M.: Embedding-based interpolation on the special orthogonal group. SIAM J. Sci. Comput. **40**(2), A721–A746 (2018)
29. Gawlik, E.S., Leok, M.: Interpolation on symmetric spaces via the generalized polar decomposition. Found. Comput. Math. **18**(3), 757–788 (2018)
30. Giaquinta, M., Hildebrandt, S.: Calculus of Variations I. Grundlehren der mathematischen Wissenschaften. Springer, Berlin (2004). https://books.google.de/books?id=4NWZdMBH1fsC
31. Grohs, P.: Quasi-interpolation in Riemannian manifolds. IMA J. Numer. Anal. **33**(3), 849–874 (2013)
32. Grohs, P., Hardering, H., Sander, O.: Optimal a priori discretization error bounds for geodesic finite elements. Found. Comput. Math. **15**(6), 1357–1411 (2015)
33. Grohs, P., Hardering, H., Sander, O., Sprecher, M.: Projection-based finite elements for nonlinear function spaces. SIAM J. Numer. Anal. **57**(1), 404–428 (2019)
34. Hajłasz, P.: Sobolev mappings between manifolds and metric spaces. In: Sobolev Spaces in Mathematics I. International Mathematical Series, vol. 8, pp. 185–222. Springer, Berlin (2009)
35. Hajlasz, P., Tyson, J.: Sobolev peano cubes. Michigan Math. J. **56**(3), 687–702 (2008)
36. Hardering, H.: Intrinsic discretization error bounds for geodesic finite elements. Ph.D. thesis, Freie Universität Berlin (2015)
37. Hardering, H.: The Aubin–Nitsche trick for semilinear problems (2017). arXiv e-prints arXiv:1707.00963
38. Hardering, H.: L^2-discretization error bounds for maps into Riemannian manifolds (2018). ArXiv preprint 1612.06086
39. Hardering, H.: L^2-discretization error bounds for maps into Riemannian manifolds. Numer. Math. **139**(2), 381–410 (2018)
40. Hélein, F.: Harmonic Maps, Conservation Laws and Moving Frames, 2nd edn. Cambridge University Press, Cambridge (2002)
41. Hélein, F., Wood, J.C.: Harmonic maps. In: Handbook of Global Analysis, pp. 417–491. Elsevier, Amsterdam (2008)
42. Jost, J.: Equilibrium maps between metric spaces. Calc. Var. Partial Differ. Equ. **2**(2), 173–204 (1994)
43. Jost, J.: Riemannian Geometry and Geometric Analysis, 6th edn. Springer, New York (2011)
44. Karcher, H.: Riemannian center of mass and mollifier smoothing. Commun. Pure Appl. Math. **30**, 509–541 (1977)

45. Ketov, S.V.: Quantum Non-linear Sigma-Models. Springer, Berlin (2000)
46. Korevaar, N.J., Schoen, R.M.: Sobolev spaces and harmonic maps for metric space targets. Commun. Anal. Geom. **1**(4), 561–659 (1993)
47. Kowalski, O., Sekizawa, M.: Natural transformations of Riemannian metrics on manifolds to metrics on tangent bundles – a classification. Bull. Tokyo Gakugei Univ. **40**, 1–29 (1997)
48. Kružík, M., Prohl, A.: Recent developments in the modeling, analysis, and numerics of ferromagnetism. SIAM Rev. **48**(3), 439–483 (2006)
49. Lee, J.M.: Introduction to Smooth Manifolds. Springer, New York (2003)
50. Melcher, C.: Chiral skyrmions in the plane. Proc. R. Soc. A **470**(2172) (2014)
51. Mielke, A.: Finite elastoplasticity Lie groups and geodesics on SL(d). In: Newton, P., Holmes, P., Weinstein, A. (eds.) Geometry, Mechanics, and Dynamics, pp. 61–90. Springer, New York (2002)
52. Münch, I.: Ein geometrisch und materiell nichtlineares Cosserat-Modell – Theorie, Numerik und Anwendungsmöglichkeiten
53. Reshetnyak, Y.G.: Sobolev classes of functions with values in a metric space. Sib. Mat. Zh. **38**(3), 657–675 (1997)
54. Rubin, M.: Cosserat Theories: Shells, Rods, and Points. Springer, Dordrecht (2000)
55. Sander, O.: Geodesic finite elements for Cosserat rods. Int. J. Numer. Methods Eng. **82**(13), 1645–1670 (2010)
56. Sander, O.: Geodesic finite elements on simplicial grids. Int. J. Numer. Methods Eng. **92**(12), 999–1025 (2012)
57. Sander, O.: Geodesic finite elements of higher order. IMA J. Numer. Anal. **36**(1), 238–266 (2016)
58. Sander, O.: Test function spaces for geometric finite elements (2016). ArXiv e-prints 1607.07479
59. Sander, O., Neff, P., Bîrsan, M.: Numerical treatment of a geometrically nonlinear planar Cosserat shell model. Comput. Mech. **57**(5), 817–841 (2016)
60. Shatah, J., Struwe, M.: Geometric Wave Equations. American Mathematical Society, Providence (2000)
61. Simo, J., Fox, D., Rifai, M.: On a stress resultant geometrically exact shell model. Part III: Computational aspects of the nonlinear theory. Comput. Methods Appl. Mech. Eng. **79**(1), 21–70 (1990)
62. Sprecher, M.: Numerical methods for optimization and variational problems with manifold-valued data. Ph.D. thesis, ETH Zürich (2016)
63. Stahl, S.: The Poincaré Half-Plane – A Gateway to Modern Geometry. Jones and Bartlett Publishers, Burlington (1993)
64. Struwe, M.: On the evolution of harmonic mappings of Riemannian surfaces. Comment. Math. Helv. **60**(1), 558–581 (1985)
65. Walther, A., Griewank, A.: Getting started with ADOL-C. In: Naumann, U., Schenk, O. (eds.) Combinatorial Scientific Computing. Computational Science, pp. 181–202. Chapman-Hall CRC, Boca Raton (2012)
66. Weinmann, A., Demaret, L., Storath, M.: Total variation regularization for manifold-valued data. SIAM J. Imaging Sci. **7**(4), 2226–2257 (2014)
67. Wriggers, P., Gruttmann, F.: Thin shells with finite rotations formulated in Biot stresses: theory and finite element formulation. Int. J. Numer. Methods Eng. **36**, 2049–2071 (1993)
68. Zeidler, E.: Nonlinear Functional Analysis and its Applications, vol. 1. Springer, New York (1986)

Chapter 2
Non-smooth Variational Regularization for Processing Manifold-Valued Data

Martin Holler and Andreas Weinmann

Contents

Abstract Many methods for processing scalar and vector valued images, volumes and other data in the context of inverse problems are based on variational formulations. Such formulations require appropriate regularization functionals that model expected properties of the object to reconstruct. Prominent examples of regularization functionals in a vector-space context are the total variation (TV) and the Mumford-Shah functional, as well as higher-order schemes such as total generalized variation models. Driven by applications where the signals or data

M. Holler (✉)
Institute of Mathematics and Scientific Computing, University of Graz, Graz, Austria
e-mail: martin.holler@uni-graz.at

A. Weinmann
Department of Mathematics and Natural Sciences, Hochschule Darmstadt, Darmstadt, Germany
e-mail: andreas.weinmann@h-da.de

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_2

live in nonlinear manifolds, there has been quite some interest in developing analogous methods for nonlinear, manifold-valued data recently. In this chapter, we consider various variational regularization methods for manifold-valued data. In particular, we consider TV minimization as well as higher order models such as total generalized variation (TGV). Also, we discuss (discrete) Mumford-Shah models and related methods for piecewise constant data. We develop discrete energies for denoising and report on algorithmic approaches to minimize them. Further, we also deal with the extension of such methods to incorporate indirect measurement terms, thus addressing the inverse problem setup. Finally, we discuss wavelet sparse regularization for manifold-valued data.

2.1 Introduction

Any measurement process, either direct or indirect, produces noisy data. While for some setups, the noise can safely be ignored, for many others it severely hinders an interpretation or further processing of the data of interest. In addition, measurements might also be incomplete such that again direct usability of the measured data is limited.

Variational regularization, i.e., a postprocessing or reconstruction of the quantity of interest via the minimization of an energy functional, often allows to reduce data corruption significantly. The success of such methods heavily relies on suitable regularization functionals and, in particular in the broadly relevant situation that the quantity of interest is sparse in some sense, non-smooth functionals are known to perform very well. Prominent and well-established examples of non-smooth regularization functional in the context of vector-space data are for instance the total variation functional and higher-order extensions such as total generalized variation, the Mumford-Shah functional and the ℓ^1- or ℓ^0-penalization of coefficients w.r.t. some wavelet basis.

When it comes to data in a non-linear space such as a manifold, the situation is different and the development of appropriate analogues of non-smooth regularization functionals in this setting is currently an active topic of research with many challenges still to be overcome. Most of these challenges are related to the nonlinearity of the underlying space, which complicates the transfer of concepts from the context of vector-space regularizers, such as measure-valued derivatives or basis transforms, but also their numerical realization.

On the other hand, applications where the underlying data naturally lives in a non-linear space are frequent and relevant. A prominent example is diffusion tensor imaging (DTI), which is a technique to quantify non-invasively the diffusional characteristics of a specimen [10, 69]. Here the underlying data space is the set of positive (definite) matrices, which becomes a Cartan-Hadamard manifold when equipped with the corresponding Fisher-Rao metric. Another example is interferometric synthetic aperture radar (InSAR) imaging which is an important airborne imaging modality for geodesy [76]. Often the InSAR image has the interpretation of a wrapped periodic version of a digital elevation model [87] and the underlying data space is the unit circle $\mathbb{S}^1$. Further examples are nonlinear color spaces for image processing, as for instance the LCh, HSV and HSL color spaces (where the underlying manifold is the cylinder $\mathbb{R}^2 \times \mathbb{S}^1$) and chromaticity-based color spaces where the underlying manifold is $\mathbb{S}^2 \times \mathbb{R}$, see [37]. Also, the rotation group SO(3) appears as data space in the context of aircraft orientations and camera positions [105], protein alignments [60], and the tracking of 3D rotational data arising in robotics [44]. Data in the euclidean motion group SE(3) may represent poses [89] and sphere-valued data appear as orientation fields of three dimensional images [86]. Finally, shape-space data [16, 77] constitutes manifold-valued data as well.

Motivated by such applications, we review existing non-smooth regularization techniques for non-linear geometric data and their numerical realization in this chapter. Following the majority of existing approaches, we will concentrate on discrete signals in a finite difference setting, which is appropriate particularly for image processing tasks due to the mostly Cartesian grid domains of images. We start with total variation regularization in Sect. 2.2, which can be transferred to a rather simple yet effective approach for non-linear data with different possibilities for a numerical realization. With the aim of overcoming well-known drawbacks of TV regularization, in particular so-called staircasing effects, we then move to higher-order functionals in Sect. 2.3, where the goal is to provide a model for piecewise smooth data with jumps. Next, having a similar scope, we discuss different models for Mumford-Shah regularization and their algorithmic realization (using concepts of dynamic programming) in Sect. 2.4. Indirect measurements in the context of manifold valued data are then the scope of Sect. 2.5, where we consider a regularization framework and algorithmic realization that applies to the previously defined approaches. Finally, we deal with wavelet sparse regularization of manifold valued data in Sect. 2.6 where we consider ℓ^1 and ℓ^0 type models and their algorithmic realization.

2.2 Total Variation Regularization of Manifold Valued Data

For scalar data, total variation regularization was early considered by Rudin et al. [90] and by Chambolle and Lions [33] in the 1990s. A major advantage of total variation regularization compared to classical Tikhonov regularization is that it preserves sharp edges [59, 101] which is the reason for a high popularity of TV regularization in particular in applications with image-related data. The most direct application of TV regularization is denoising, where ℓ^2 data terms have originally been used in [90] (and are well-suited in case of Gaussian noise) and ℓ^1 data terms are popular due to robustness against outliers and some favorable analytical properties [3, 36, 81]. An extension of TV for vector-valued data has early been considered in [91] and we refer to [45] for an overview of different approaches.

This section reviews existing extensions of TV regularization to manifold-valued data. In the continuous setting, such an extension has been considered analytically in

[56, 57], where [57] deals with the $\mathbb{S}^1$ case and [56] deals with the general case using the notion of cartesian currents. There, in particular, the existence of minimizers of certain TV-type energies in the continuous domain setup has been shown. In a discrete, manifold-valued setting, there is a rather straight forward definition of TV. Here, the challenge is more to develop appropriate algorithmic realizations. Indeed, many of the successful numerical algorithms for TV minimization in the vector space setting, such as [32, 34, 58] and [82] for ℓ^1-TV, rely on smoothing or convex duality, where for the latter no comprehensive theory is available in the manifold setting.

2.2.1 Models

For univariate data of length N in a finite dimensional Riemannian manifold $\mathcal{M}$, the (discrete) TV denoising problem with ℓ^q-type data fidelity term reads as

$$\operatorname{argmin}_{x \in \mathcal{M}^N} \Big\{ \frac{1}{q} \sum_{i=1}^{N} \mathrm{d}(x_i, f_i)^q + \alpha \sum_{i=1}^{N-1} \mathrm{d}(x_i, x_{i+1}) \Big\}. \tag{2.1}$$

Here, $f = (f_i)_{i=1}^N$ denotes the observed data and $x = (x_i)_{i=1}^N$ is the argument to optimize for. Further, $q \in [1, \infty)$ is a real number and $\alpha > 0$ is a regularization parameter controlling the trade of between data fidelity and the regularity. The symbol $\mathrm{d}(y, z)$ denotes the distance induced by the Riemannian metric on the manifold $\mathcal{M}$. We note that in the euclidean case $\mathcal{M} = \mathbb{R}^d$, the above distance to the data f corresponds to the ℓ^q norm. For noise types with heavier tails (such as Laplacian noise in the euclidean case,) $q = 1$ is a good choice. We further point out that, in the scalar case $\mathcal{M} = \mathbb{R}$, the expression $\sum_{i=1}^{N-1} d(x_i, x_{i+1})$ defines the total variation of the sequence x interpreted as a finite sum of point measures.

In the bivariate case, a manifold version of TV denoising for signals in $\mathcal{M}^{N \times M}$ is given by

$$\begin{aligned} \operatorname{argmin}_{u \in M^{N \times M}} \Big\{ & \tfrac{1}{q} \textstyle\sum_{i,j} \mathrm{d}(x_{i,j}, f_{i,j})^q \\ & + \alpha \textstyle\sum_{i,j} \big(\mathrm{d}(x_{i,j}, x_{i+1,j})^p + \mathrm{d}(x_{i,j}, x_{i,j+1})^p \big)^{1/p} \Big\}. \end{aligned} \tag{2.2}$$

Note that here and in the following, we will frequently omit the index bounds in finite-length signals and sums for the sake of simplicity, and always implicitly set all scalar-valued summands containing out-of-bound indices to 0. In (2.2), the cases $p = 1$ and $p = 2$ are most relevant, where $p = 1$ has computational advantages due to a separable structure and $p = 2$ is often used because is corresponds to an isotropic functional in the continuous, vector-space case. We note however that, in the TV case, the effects resulting from anisotropic discretization are not severe. Moreover, they can be almost completely eliminated by including further difference directions, such as diagonal differences. For details on including further difference directions we refer to Sect. 2.4 (discussing reduction of anisotropy effects for the Mumford-Shah case in which case such effects are more relevant.)

Note that if we replace the distance term in the TV component of (2.1) by the squared distance (or remove the square root for $p = 2$ in (2.2)), we end up with a discrete model of classical H^1 regularization. Further, we may also replace the distance term in the regularizer by $h \circ \mathrm{d}$ where h can for instance be the so-called Huber function which is a parabola for small arguments smoothly glued with two linear functions for larger arguments. Using this, we end up with models for Huber regularization, see [114] for details.

2.2.2 Algorithmic Realization

As mentioned in the introduction to this section, the typical methods used for TV regularization in vector spaces are based on convex duality. The respective concepts are not available in a manifold setting. However, there are different strategies to solve (2.1) and (2.2), and we briefly review some relevant strategies in the following.

The authors of [40, 100] consider TV regularization for $\mathbb{S}^1$-valued data and develop a lifting approach, i.e., they lift functions with values in $\mathbb{S}^1$ to functions with values in the universal covering $\mathbb{R}$ of $\mathbb{S}^1$, lifting the involved functionals at the same time such that the periodicity of the data is respected. This results in a nonconvex problem for real valued data (which still reflects the original $\mathbb{S}^1$ situation), which can then algorithmically be approached by using convex optimization techniques on the convex relaxation of the nonconvex vector space problem. We note that the approach is a covering space approach which relies on the fact that the covering space is a vector space which limits its generalization to general manifolds. In connection with $\mathbb{S}^1$ valued data we also point out the paper [98] which provides an exact solver for the univariate L^1-TV problem (2.1) with $q = 1$.

For general manifolds there are three conceptually different approaches to TV regularization. The authors of [74] reformulate the TV problem as a multi-label optimization problem. More precisely, they consider a lifted reformulation in a vector-space setting, where the unknown takes values in the space of probability measures on the manifold (rather than the manifold itself), such that it assigns a probability for each given value on the manifold. Constraining the values of the unknown to be delta peaks, this would correspond to an exact reformulation whereas dropping this constraint yields a convex relaxation. After discretization, the unknown takes values in the unit simplex assigning a probability to each element of a discrete set of possible values. This corresponds to a lifting of the problem to higher dimensions, where the number of values the unknown is allowed to attain defines the dimensionality of the problem. Having a vector-space structure available, the lifted problem is then solved numerically using duality-based methods. We refer to [74] for details and to Chapter 3 of this book for an overview of research in that direction and extensions.

Another approach can be found in the paper [63]. There, the authors employ an iteratively reweighted least squares (IRLS) algorithm to the isotropic discrete TV functional (2.2). The idea of the IRLS is to replace the distance terms in the TV regularizer by squared distance terms and to introduce a weight for each summand of the regularizer. Then, fixing the weights, the problem is differentiable and can be solved using methods for differentiable functions such as a gradient descent scheme. In a next step, the weights are updated where a large residual part of a summand results in a small weight, and the process is iterated. This results in an alternating minimization algorithm. The authors show convergence in the case of Hadamard spaces and for data living in a half-sphere. We mention that IRLS minimization is frequently applied for recovering sparse signals and that it has been also applied to scalar TV minimization in [88]. In connection with this, we also mention the paper [13] which considers half-quadratic minimization approaches that are generalizations of [63].

Finally, the approach of [114] to TV regularization employs iterative geodesic averaging to implement cyclic and parallel proximal point algorithms. The main point here is that the appearing proximal mappings can be analytically computed and the resulting algorithms exclusively perform iterative geodesic averaging. This means that only points on geodesics have to be computed. We will elaborate on this approach in the following. In connection with this, we also mention the paper [12] where a generalized forward-backward type algorithm is proposed to solve a related problem in the context of DTI; see also [11, 92] in the context of shape spaces.

The approach of [114] relies on the concepts of cyclic proximal point algorithms (CPPAs) and parallel proximal point algorithms (PPPA) in a manifold. A reference for cyclic proximal point algorithms in vector spaces is [19]. In the context of nonlinear spaces, the concept of CPPAs was first proposed in [7], where it is employed to compute means and medians in Hadamard spaces. In the context of variational regularization methods for nonlinear, manifold-valued data, they were first used in [114], which also proposed the PPPA in the manifold setting.

CPPAs and PPPAs

The idea of both CPPAs and PPPAs is to decompose a functional $F : \mathcal{M}^N \to \mathbb{R}$ to be minimized into basic atoms $(F_i)_{i=1}^K$ and then to compute the proximal mappings of the atoms F_i iteratively. For a CPPA, this is done in a cyclic way, and for a PPPA, in a parallel way. More precisely, assume that

$$F = \sum\nolimits_{i=1}^{K} F_i \tag{2.3}$$

and consider the proximal mappings [6, 48, 78] $\operatorname{prox}_{\lambda F_i} : \mathcal{M}^N \to \mathcal{M}^N$ given as

$$\operatorname{prox}_{\lambda F_i}(x) = \operatorname{argmin}_y \, F_i(y) + \tfrac{1}{2\lambda} \sum\nolimits_{j=1}^{N} \mathrm{d}(x_j, y_j)^2. \tag{2.4}$$

Algorithm 1 CPPA for solving $\min_x F(x)$ with $F = \sum_{j=1}^K F_j$

```
CPPA($x^0$, $(\lambda_k)_k$, $(\sigma(j))_{j=1}^K$)
$k = 0$, $x_0^0 = x^0$
    repeat until stopping criterion fulfilled
        for $j = 1, \ldots, K$
            $x_j^k = \operatorname{prox}_{\lambda_k F_{\sigma(j)}}(x_{j-1}^k)$
        $x_0^{k+1} = x_K^k, \quad k \leftarrow k+1$
return $x_0^k$
```

Algorithm 2 PPPA for solving $\min_x F(x)$ with $F = \sum_{j=1}^K F_j$

```
PPPA($x^0$, $(\lambda_k)_k$)
$k = 0$,
    repeat until stopping criterion fulfilled
        for $j = 1, \ldots, K$
            $x_j^{k+1} = \operatorname{prox}_{\lambda_k F_j}(x^k)$
        $x^{k+1} = \operatorname{mean}_j\left(x_j^{k+1}\right), \quad k \leftarrow k+1$
return $x^k$
```

One cycle of a CPPA then consists of applying each proximal mapping $\operatorname{prox}_{\lambda F_i}$ once in a prescribed order, e.g., $\operatorname{prox}_{\lambda F_1}$, $\operatorname{prox}_{\lambda F_2}$, $\operatorname{prox}_{\lambda F_3}, \ldots,$ or, generally, $\operatorname{prox}_{\lambda F_{\sigma(1)}}$, $\operatorname{prox}_{\lambda F_{\sigma(2)}}$, $\operatorname{prox}_{\lambda F_{\sigma(3)}}, \ldots,$ where the symbol σ is employed to denote a permutation. The cyclic nature is reflected in the fact that the output of $\operatorname{prox}_{\lambda F_{\sigma(i)}}$ is used as input for $\operatorname{prox}_{\lambda f_{\sigma(i+1)}}$. Since the ith update is immediately used for the $(i+1)$th step, it can be seen as a Gauss-Seidel-type scheme. We refer to Algorithm 1 for its implementation in pseudocode.

A PPPA consists of applying the proximal mapping to each atom F_i to the output of the previous iteration x^k in parallel and then averaging the results, see Algorithm 2. Since it performs the elementary update steps, i.e., the evaluation of the proximal mappings, in parallel it can be seen as update pattern of Jacobi type. In Algorithm 2, the symbol *mean* denotes the generalization of the arithmetic average to a Riemannian manifold, which is the well known intrinsic mean, i.e., given $z_1, \ldots, z_K$ in $\mathcal{M}$, a mean $z^* \in \mathcal{M}$ is defined by (cf. [50, 70, 71, 83])

$$z^* = \operatorname{mean}_j\left(z_j\right) = \operatorname{argmin}_{z \in \mathcal{M}} \sum\nolimits_{j=1}^K \mathrm{d}(z, z_j)^2. \tag{2.5}$$

Please note that this definition is employed component-wise for x^{k+1} in Algorithm 2. We note that, if the $(F_i)_i$ are lower semi continuous, both the minimization problem for the proximal mapping and for the mean admit a solution. On general manifolds, however, the solution is not necessarily unique. For arguments whose points are all contained in a small ball (whose radius depends on the sectional curvature $\mathcal{M}$) it is unique, see [6, 48, 70, 71] for details. This is a general issue in the context of manifolds that are—in a certain sense—a local concept involving objects that are often only locally well defined. In case of ambiguities, we hence consider the above objects as set-valued quantities.

During the iteration of both CPPA and PPPA, the parameter λ_k of the proximal mappings is successively decreased. In this way, the penalty for deviation from the previous iterate is successively increased. It is chosen in a way such that the sequence $(\lambda^k)_k$ is square-summable but not summable. Provided that this condition holds, the CPPA can be shown to converge to the optimal solution of the underlying minimization problem, at least in the context of Hadamard manifolds and convex $(F_i)_i$, see [8, Theorem 3.1]. The same statement holds for the PPPA, see [114, Theorem 4]. The mean can be computed using a gradient descent or a Newton scheme. To reduce the computation time further, it has been proposed in [114] to replace the mean by another construction (known as geodesic analogues in the subdivision context [107]) which is an approximation of the mean that is computationally less demanding. As above, in the context of Hadamard manifolds and convex $(F_i)_i$, the convergence towards a global minimizer is guaranteed, see [114, Theorem 7]. For details we refer to the above reference.

Proximal Mappings for the Atoms of the TV Functionals

Now we consider a splitting of the univariate problem (2.1) and the bivariate problem (2.2) into basic atoms such that the CPPA and the PPPA can be applied. Regarding (2.1) we use the atoms

$$F_1(x) := \tfrac{1}{q}\sum_{i=1}^{N} \mathrm{d}(x_i, f_i)^q, \quad F_2(x) = \sum_{\substack{i=1\\ i \text{ odd}}}^{N-1} \mathrm{d}(x_i, x_{i+1}), \quad F_3(x) = \sum_{\substack{i=1\\ i \text{ even}}}^{N} \mathrm{d}(x_i, x_{i+1}). \tag{2.6}$$

Regarding (2.2), we consider the case $p = 1$ and again define F_1 to be the data term, F_2 and F_3 to be a splitting of the sum $\sum_{i,j} \mathrm{d}(x_{i,j}, x_{i+1,j})$ into even and odd values of i and F_4 and F_5 to be a splitting of the sum $\sum_{i,j} \mathrm{d}(x_{i,j}, x_{i,j+1})$ into even and odd values of j. With these splittings, all summands in the atom $(F_i)_i$ decouple such that the computation of the proximal mappings reduces to a point-wise computation of the proximal mappings of

$$x \mapsto g_1(x, f) := \frac{1}{q}\mathrm{d}(x, f)^q \quad \text{and} \quad (x_1, x_2) \mapsto g_2(x_1, x_2) = \mathrm{d}(x_1, x_2). \tag{2.7}$$

From the splitting (2.6) (and its bivariate analogue below (2.6)) together with (2.7) we see that within a PPPA all proximal mappings of the basic building blocks g_1, g_2 can be computed in parallel and the computation of each mean only involves 3 points in the manifold $\mathcal{M}$ in the univariate setting and 5 points in the multivariate setting. For a CPPA we see that a cycle has length 3 and 5 in the univariate and bivariate situation, respectively, and that within each atom F_i the proximal mappings of the respective terms of the form g_1, g_2 can be computed in parallel.

For the data term, the proximal mappings $\mathrm{prox}_{\lambda g_1}$ are explicit for $q = 1$ and $q = 2$ and, as derived in [48], are given as

$$(\mathrm{prox}_{\lambda g_1(\cdot, f)})_j(x) = [x, f]_t \tag{2.8}$$

where

$$t = \tfrac{\lambda}{1+\lambda} \text{ for } q = 2, \qquad t = \min\left(\tfrac{\lambda}{\mathrm{d}(x,f)}, 1\right) \text{ for } q = 1. \tag{2.9}$$

Here, we use the symbol $[\cdot, \cdot]_t$ to denote the point reached after time t on the (non unit speed) length-minimizing geodesic starting at the first argument reaching the second argument at time 1. (Note, that up to sets of measure zero, length minimizing geodesics are unique, and in the extraordinary case of non-uniqueness we may pick one of them.)

Regarding g_2, it is shown in [114] that the proximal mappings are given in closed form as

$$\mathrm{prox}_{\lambda g_2}((x_1, x_2)) = ([x_1, x_2]_t, [x_2, x_1]_t), \quad \text{where } t = \min\left(\frac{\lambda\alpha}{\mathrm{d}(x_1, x_2)}, \frac{1}{2}\right). \tag{2.10}$$

Here, for each point, the result is a point on the geodesic segment connecting two arguments.

It is important to note that the point $p_t = [p_0, p_1]_t$ on the geodesic connecting two points p_0, p_1 is given in terms of the Riemannian exponential map exp and its inverse denoted by log or $\exp^{-1}$ by

$$p_t = [p_0, p_1]_t = \exp_{p_0}(t\ \log_{p_0} p_1). \tag{2.11}$$

Here, $v := \log_{p_0} p_1$ denotes that tangent vector sitting in p_0 such that $\exp_{p_0} v = p_1$. The tangent vector v is scaled by t, and then the application of the exp-map yields p_t. More precisely, $\exp_{p_0}$ assigns the point $p_t = \exp_{p_0} tv$ to the tangent vector tv by evaluating the geodesic starting in p_0 with tangent vector tv at time 1.

We note that also the proximal mappings of the classical Tichanov regularizers as well as of the Huber regularizers mentioned above have a closed form representation in terms of geodesic averaging as well. Further, there are strategies to approximate intrinsic means by iterated geodesic averages to speed up the corresponding computations. For details on these comments we refer to [114].

Plugging in the splittings and proximal mappings as above into the Algorithms 1 and 2 yields a concrete implementation for the TV-regularized denoising of manifold-valued data. Regarding convergence, we have the following result.

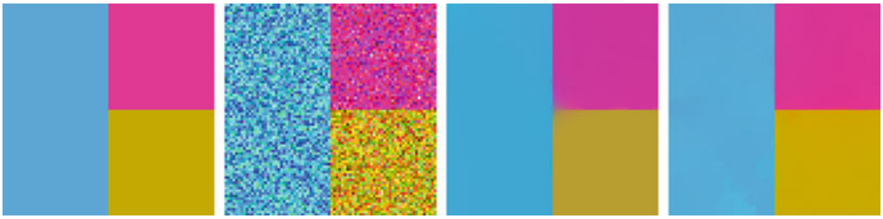


Fig. 2.1 The effect of ℓ^2-TV denoising in LCh space ($\alpha = 0.80$). *Left.* Ground truth. *Middle left.* Noisy input image corrupted by Gaussian noise on each channel. *Middle right.* The ℓ^2-TV reconstruction in linear space. *Right.* The ℓ^2-TV reconstruction in the nonlinear LCh color space. Using the distance in the non-flat LCh metric can lead to higher reconstruction quality

Theorem 2.1 *For data in a (locally compact) Hadamard space and a parameter sequence $(\lambda_k)_k$ which is squared summable but not summable, the iterative geodesic averaging algorithms for TV-regularized denosing (based on the CPPA, the PPPA, as well as the inexact approximative and fast variant of the PPPA) converge towards a minimizer of the ℓ^p-*TV *functional.*

We further remark that the statement remains true when using the Huber potential mentioned above either as data term or for the regularization, as well as when using quadratic variation instead of TV. A proof of this statement and more details on the remarks can be found in [114].

We illustrate the algorithms with some examples. First we consider denoising in the LCh color space. As explained above, the underlying manifold is $\mathbb{S}^1 \times \mathbb{R}^2$. The exponential and its inverse are given componentwise by the respective mappings on $\mathbb{R}^2$ and $\mathbb{S}^1$. By (2.11), this allows to compute the involved proximal mappings via (2.8), (2.9) and (2.10), respectively. We point out that in spite of the separability of the exponential and its inverse, the proposed algorithm is in general not equivalent to performing the algorithm on $\mathbb{R}^2$ and $\mathbb{S}^1$ separately. The reason is that the parameter t in (2.9) and (2.10) depend nonlinearly on the distance in the product manifold (except for $p, q = 2$). In Fig. 2.1 we illustrate the denoising potential of the proposed scheme in the LCh space. Here, the vector-space computation was realized using the split Bregman method for vectorial TV regularization [55, 58] and we optimized the parameters of both methods with respect to the peak signal to noise ratio.

As a second example we consider noisy data on the unit sphere $\mathbb{S}^2$ (in $\mathbb{R}^3$). In Fig. 2.2, we test the denoising potential of our algorithm on a noisy (synthetic) spherical-valued image. As noise model on $\mathbb{S}^2$, we use the von Mises-Fisher distribution having the probability density $h(x) = c(\kappa)\exp(\kappa x \cdot \mu)$. Here, $\kappa > 0$ expresses the concentration around the mean orientation $\mu \in \mathbb{S}^2$ where a higher κ indicates a higher concentration of the distribution and $c(\kappa)$ is a normalization constant. We observe in Fig. 2.2 that the noise is almost completely removed by TV minimization and that the edges are retained.

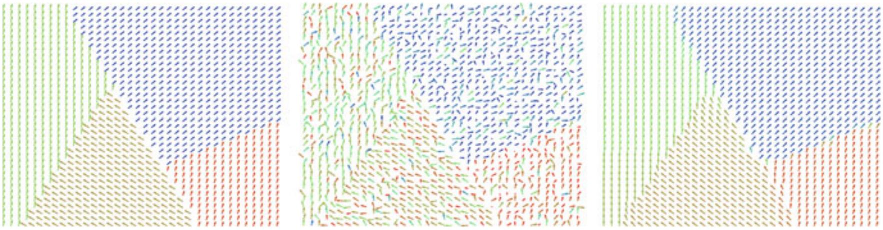


Fig. 2.2 Denoising of an $\mathbb{S}^2$-valued image. The polar angle is coded both as length of the vectors and as color (red pointing towards the reader, blue away from the reader). *Left.* Synthetic image. *Center.* Noisy data (corrupted by von Mises-Fisher noise of level $\kappa = 5.5$). *Right.* ℓ^2-TV regularization using $\alpha = 0.7$. The noise is almost completely removed whereas the jumps are preserved

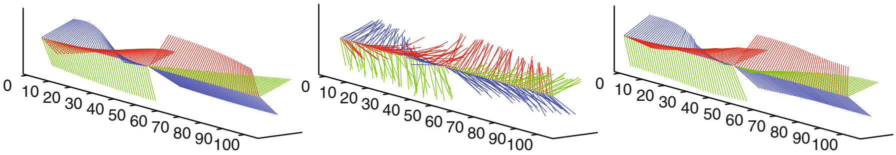


Fig. 2.3 Result (*right*) of denoising an SO(3)-valued noisy time-series (*center*) using the inexact parallel algorithm for L^2-TV regularization with $\alpha = 4.0$. (*Left:* Ground truth.) Here, an element of SO(3) is visualized by the rotation of a tripod. We observe that the noise is removed and the jump is preserved

In Fig. 2.3 we consider an univariate signal with values in the special orthogonal group SO(3) consisting of all orthogonal 3×3 matrices with determinant one. We see that the proposed algorithm removes the noise and that the jump is preserved. Finally, we consider real InSAR data [76, 87] in Fig. 2.4. InSAR images consist of phase values such that the underlying manifold is the one-dimensional sphere $\mathbb{S}^1$. The image is taken from [87]. We apply total variation denoising using ℓ^2 and ℓ^1 data terms. We observe that TV regularization reduces the noise significantly. The ℓ^1 data term seems to be more robust to outliers than the ℓ^2 data term.

2.3 Higher Order Total Variation Approaches, Total Generalized Variation

It is well known in the vector space situation (and analytically confirmed for instance in [21, 24]) that TV regularization has a tendency to produce piecewise constant results, leading to artificial jump discontinuities in case of ground truth data with smooth regions. Classical H^1 regularization avoids this effect. However, H^1 regularity does not allow for jump discontinuities, which can be seen as motivation for considering non-smooth higher order approaches. While second

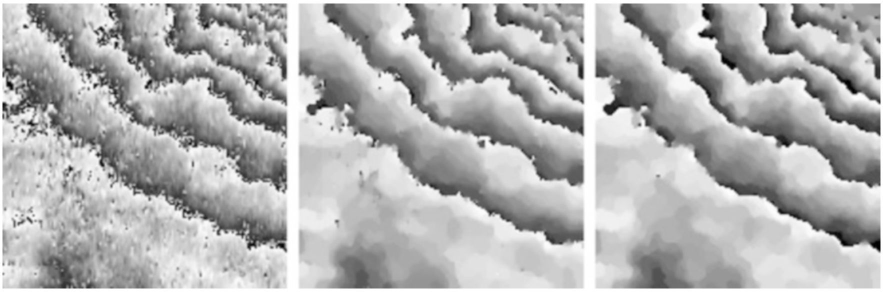


Fig. 2.4 Total variation denoising of an $\mathbb{S}^1$-valued InSAR image (real data, *left*) using L^2-TV regularization ($\alpha = 0.32$, *middle*) and L^1-TV regularization ($\alpha = 0.60$, *right*). Here, the circle $\mathbb{S}^1$ is represented as an interval with endpoints identified, i.e., white and black represent points nearby. Total variation minimization reliably removes the noise while preserving the structure of the image

order TV regularization [41, 65], i.e., penalizing the Radon norm of the second order distributional derivative of a function, is a first attempt in this direction, one can show that functions whose second order distributional derivative can be represented by a Radon measure again cannot have jumps along (smooth) hypersurfaces [25]. This disadvantage is no longer present when using a combination of first and second order TV via infimal convolution, i.e.,

$$\mathrm{IC}_\alpha(u) = \inf_v \alpha_1 \mathrm{TV}(u - v) + \alpha_0 \mathrm{TV}^2(v),$$

as originally proposed in [33]. Here, $\alpha = (\alpha_1, \alpha_0) \in (0, \infty)^2$ are two weights. Regularization with TV-TV^2 infimal convolution finds an optimal additive decomposition of the unknown u in two components, where one yields minimal cost for TV and the other one for second order TV. Extending on that, the (second order) total generalized variation (TGV) functional [27] optimally balances between first and second order derivatives on the level of the gradient rather than the function, i.e., is given as

$$\mathrm{TGV}^2_\alpha(u) = \inf_w \alpha_1 \|\nabla u - w\|_{\mathcal{M}} + \alpha_0 \|\mathcal{E} w\|_{\mathcal{M}},$$

where $\mathcal{E} w = 1/2(Jw + Jw^T)$ is a symmetrization of the Jacobian matrix field Jw and again $\alpha = (\alpha_1, \alpha_0) \in (0, \infty)^2$ are two weights. This provides a more flexible balancing between different orders of differentiation and, in particular in situations when an optimal decomposition on the image level is not possible, further reduces piecewise constancy artifacts still present with TV-TV^2 infimal convolution, see [27].

Motivated by the developments for vector spaces, and due to the challenges appearing when extending them to manifold-valued data, several works deal with developing non-smooth higher order regularization in this setting. In the following we motivate and review some existing approaches and strive to present them in a common framework. Following the existing literature, we consider a discrete setting.

2.3.1 Models

First, we define the above-mentioned higher order regularization functionals in a discrete, vector-space setting. To this aim, for $u = (u_{i,j})_{i,j} \in \mathbb{R}^{N\times M}$, let $\delta_x : \mathbb{R}^{N\times M} \to \mathbb{R}^{(N-1)\times M}$, $(\delta_x u)_{i,j} = u_{i+1,j} - u_{i,j}$ and $\delta_y : \mathbb{R}^{N\times M} \to \mathbb{R}^{N\times(M-1)}$, $(\delta_y u)_{i,j} = u_{i,j+1} - u_{i,j}$ be finite differences (on a staggered grid to avoid boundary effects) w.r.t. the first- and second component, respectively. A discrete gradient, Jacobian and symmetrized Jacobian are then given as

$$\nabla u = (\delta_x u, \delta_y u), \qquad \mathrm{J}(v^1, v^2) = (\delta_x v^1, \delta_y v^2, \delta_y v^1, \delta_x v^2),$$
$$\mathcal{E}(v^1, v^2) = (\delta_x v^1, \delta_y v^2, \tfrac{\delta_y v^1 + \delta_x v^2}{2}),$$

respectively, where $v = (v^1, v^2) \in \mathbb{R}^{(N-1)\times M} \times \mathbb{R}^{N\times(M-1)}$. We note that different components of ∇u, $\mathrm{J}v$, $\mathcal{E}v$ have different length. Using these objects, we define discrete versions of TV, second order TV, of TV-TV^2 infimal convolution and of TGV as

$$\begin{aligned} \mathrm{TV}(u) &= \|\nabla u\|_1, \quad \mathrm{TV}^2(u) = \|\mathrm{J}\nabla u\|_1, \\ \mathrm{IC}_\alpha(u) &= \min_v \alpha_1 \mathrm{TV}(u-v) + \alpha_0 \mathrm{TV}^2(v), \\ \mathrm{TGV}^2_\alpha(u) &= \min_w \alpha_1 \|\nabla u - w\|_1 + \alpha_0 \|\mathcal{E}w\|_1. \end{aligned} \tag{2.12}$$

Here, $\|\cdot\|_1$ denotes the ℓ^1 norm w.r.t. the spatial component and we take an ℓ^p norm (with $p \in [1,\infty)$) in the vector components without explicitly mentioning, e.g., $\|\nabla u\|_1 := \sum_{i,j} \left((\delta_x u)^p_{i,j} + (\delta_y u)^p_{i,j} \right)^{1/p}$, where we again replace summands containing out-of-bound indices 0. Note that the most interesting cases are $p = 1$ due to advantages for the numerical realization and $p = 2$ since this corresponds to isotropic functionals in the infinite-dimensional vector-space case, see for instance [27] for TGV. Also note that $(\mathrm{J}\nabla u)_{i,j}$ is symmetric, that the symmetric component of $\mathcal{E}w$ is stored only once and that we define $\|\mathcal{E}w\|_1 := \sum_{i,j} \left((\delta_x w^1)^p_{i,j} + (\delta_y w^2)^p_{i,j} + 2(\frac{\delta_y w^1 + \delta_x w^2}{2})^p_{i,j} \right)^{1/p}$ to compensate for that.

Now we extend these regularizers to arguments $u \in \mathcal{M}^{N\times M}$ with $\mathcal{M}$ being a complete, finite dimensional Riemannian manifold with induced distance d. For the sake of highlighting the main ideas first, we start with the univariate situation $u = (u_i)_i \in \mathcal{M}^N$.

Regarding second order TV, following [9], we observe (with δ the univariate version of δ_x) that, for $u \in (\mathbb{R}^d)^N$ and a norm $\|\cdot\|$ on $\mathbb{R}^d$,

$$\|(\delta\delta u)_i\| = \|u_{i+1} - 2u_i + u_{i-1}\| = 2\|\frac{u_{i+1}+u_{i-1}}{2} - u_i\|,$$

where the last expression only requires averaging and a distance measure, both of which is available on Riemannian manifolds. Thus, a generalization of TV^2 for $u \in \mathcal{M}^N$ can be given, for $u = (u_i)_i$, by

$$\mathrm{TV}^2(u) = \sum\nolimits_i D_c(u_{i-1}, u_i, u_{i+1}) \quad \text{where } D_c(u_-, u_\circ, u_+) = \inf_{c \in [u_-, u_+]_{\frac{1}{2}}} 2\mathrm{d}(c, u_\circ).$$

Here, D_c essentially measures the distance between the central data point $u_\circ$ and the geodesic midpoint of its neighbors, if this midpoint is unique, and the infimum w.r.t. all midpoints otherwise. Similarly, we observe for mixed derivatives and $u \in (\mathbb{R}^d)^{N\times N}$ that

$$\|(\delta_y\delta_x u)\| = 2\left\|\tfrac{u_{i+1,j}+u_{i,j-1}}{2} - \tfrac{u_{i,j}+u_{i+1,j-1}}{2}\right\|.$$

An analogue for $u \in \mathcal{M}^{M\times M}$ is hence given by

$$D_{cc}(u_{i,j}, u_{i+1,j}, u_{i,j-1}, u_{i+1,j-1}) = \inf_{c_1 \in [u_{i+1,j}, u_{i,j-1}]_{\frac{1}{2}}, c_2 \in [u_{i,j}, u_{i+1,j-1}]_{\frac{1}{2}}} 2\mathrm{d}(c_1, c_2),$$

and similarly for $\delta_x\delta_y$. Exploiting symmetry, we only incorporate $(\delta_y\delta_x u)$ and define

$$\mathrm{TV}^2(u) = \sum_{i,j} \Big(D_c(u_{i-1,j}, u_{i,j}, u_{i+1,j})^p + D_c(u_{i,j-1}, u_{i,j}, u_{i,j+1})^p + 2D_{cc}(u_{i,j}, u_{i+1,j}, u_{i,j-1}, u_{i+1,j-1})^p \Big)^{1/p}.$$

This generalizes second order TV for manifold-valued data while still relying only on point-operations on the manifold. As will be shown in Sect. 2.3.2, numerical result for TV^2 denoising show less staircasing than TV denoising. However, it tends towards oversmoothing which is expected from the underlying theory and corresponding numerical results in the vector space case.

A possible extension of TV-TV^2 infimal-convolution to manifolds is based on a representation in the linear space case given as

$$\mathrm{IC}_\alpha(u) = \inf_v \alpha_1 \mathrm{TV}(u-v) + \alpha_0 \mathrm{TV}^2(v) = (1/2) \inf_{v,w: u = \frac{v+w}{2}} \alpha_1 \mathrm{TV}(v) + \alpha_0 \mathrm{TV}^2(w),$$

where $u = (u_{i,j})_{i,j} \in (\mathbb{R}^d)^{N\times M}$. This representation was taken in [14, 15] and extended for $u = (u_{i,j})_{i,j} \in \mathcal{M}^{N\times M}$ (up to constants) via

$$\mathrm{IC}(u) = \tfrac{1}{2} \inf_{v,w} \alpha_1 \mathrm{TV}(v) + \alpha_0 \mathrm{TV}^2(w) \quad \text{s.t. } u_{i,j} \in [[v_{i,j}, w_{i,j}]]_{\frac{1}{2}},$$

where $v = (v_{i,j})_{i,j}$ and $w = (w_{i,j})_{i,j}$. Following [14], we here use the symbol $[[v_{i,j}, w_{i,j}]]$ instead of $[v_{i,j}, w_{i,j}]$, where we define the former to include also non-distance minimizing geodesics.

In order to generalize the second order TGV functional to a manifold setting, we consider (2.12) for vector spaces. This definition (via optimal balancing) requires to measure the distance of (discrete) vector fields that are in general defined in different tangent spaces. One means to do so is to employ parallel transport for vector fields in order to shift different vector fields to the same tangent space and to measure the distance there. (We note that the particular locations the vectors are shifted to is irrelevant since the values are equal.) This approach requires to incorporate more advanced concepts on manifolds. Another possibility is to consider a *discrete tangent space* of point tuples via the identification of $v = \log_a(b)$ as a point tuple $[a, b]$ (where log is the inverse exponential map), and to define a distance-type function on such point tuples. Indeed, the above identification is one-to-one except for points on the cut locus (which is a set of measure zero [67]) and allows to identify discrete derivatives $(\delta_x u)_i = (u_{i+1} - u_i) = \log_{u_i}(u_{i+1})$ as tupel $[u_i, u_{i+1}]$. Choosing appropriate distance type functions, this identification allows to work exclusively on the level of point-operations and one might say that the "level of complexity" of the latter approach is comparable with that of TV^2 and IC. Furthermore, a version of the above parallel transport variant can be realized in the tupel setting as well (still incorporating more advanced concepts). This approach was proposed in [28]; more precisely, an axiomatic approach is pursued in [28] and realizations via Schild's ladder (requiring only point operations) and parallel transport are proposed and shown to be particular instances of the axiomatic approach.

We explain the approach in more detail, where we focus on the univariate situation first. We assume for the moment that $D : \mathcal{M}^2 \times \mathcal{M}^2$ is an appropriate distance-type function for point tuples. Then, a definition of TGV^2_α for an univariate signal $u = (u_i)_i \in \mathcal{M}^N$ can be given as

$$\mathrm{TGV}^2_\alpha((u_i)_i) = \inf_{(y_i)_i} \sum_i \alpha_1 D([u_i, u_{i+1}], [u_i, y_i]) + \alpha_0 D([u_i, y_i], [u_{i-1}, y_{i-1}]).$$

Thus, one is left to determine a suitable choice of D. One possible choice is based on the Schild's ladder [72] approximation of parallel transport, which is defined as follows (see Fig. 2.5): Assuming, for the moment, uniqueness of geodesics, define $c = [v, x]_{\frac{1}{2}}$ and $y' = [u, c]_2$. Then $[x, y']$ can be regarded as approximation of the parallel transport of $w = \log_u(v)$ to x, which is exact in the vector-space case. Motivated by this, the distance of the tuples $[u, v]$ and $[x, y]$ can be defined as $\mathrm{d}(y, y')$. Incorporating non-uniqueness by minimizing over all possible points in this construction to capture also points on the cut locus, yields a distance-type function for point tuples given as

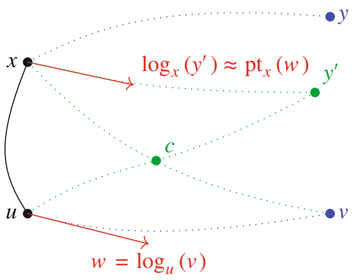


Fig. 2.5 Approximate parallel transport of $\log_u(v)$ to x via the Schild's ladder construction. Figure taken from [28]

$$D_{\mathrm{S}}([x,y],[u,v]) = \inf_{y' \in \mathcal{M}} \mathrm{d}(y,y') \quad \text{s.t. } y' \in [u,c]_2 \text{ with } c \in [x,v]_{\frac{1}{2}}.$$

In the particular case that both tuples have the same base point, i.e., $x = u$, it is easy to see that, except in the case of non-unique length-minimizing geodesics, $D_{\mathrm{S}}([x,y],[x,v]) = \mathrm{d}(v,y)$ such that we can use this as simplification and arrive at a concrete form of manifold-TGV for univariate signals $(u_i)_i$ in $\mathcal{M}$ given as

$$\text{S-TGV}^2_\alpha((u_i)_i) = \inf_{(y_i)_i} \sum_i \alpha_1 \mathrm{d}(u_{i+1}, y_i) + \alpha_0 D_{\mathrm{S}}([u_i, y_i],[u_{i-1}, y_{i-1}]).$$

We note that this operation only requires to carry out averaging and reflection followed by applying the distance in the manifold. Thus it is on the same "level of difficulty" as TV^2 or IC. For the bivariate situation, the situation is more challenging due to an additional averaging involved in the evaluation of $\mathcal{E}w$. In fact, as described in [28], there are different possibilities (of varying complexity) to generalize this to the manifold valued setting but there is a unique, rather simple one which in addition transfers fundamental properties of TGV, such as a precise knowledge on its kernel, to the manifold setting. This leads to the definition of $D_{\mathrm{S}}^{\mathrm{sym}} : (\mathcal{M}^2)^4$ which realizes the symmetrized part of $\mathcal{E}w$ in the definition of TGV and for which, for the sake of brevity, we refer to [28, Equation 20]. Using $D_{\mathrm{S}}^{\mathrm{sym}}$, a bivariate version of TGV for $u = (u_{i,j})_{i,j} \in \mathcal{M}^{N \times M}$ is given as

$$\begin{aligned}\text{S-TGV}^2_\alpha(u) = \min_{y^1_{i,j}, y^2_{i,j}} \alpha_1 \sum\nolimits_{i,j} \Big(d(u_{i+1,j}, y^1_{i,j})^p + d(u_{i,j+1}, y^2_{i,j})^p \Big)^{1/p} \\ + \alpha_0 \sum\nolimits_{i,j} \Big(D_{\mathrm{S}}\big([u_{i,j}, y^1_{i,j}],[u_{i-1,j}, y^1_{i-1,j}]\big)^p + D_{\mathrm{S}}\big([u_{i,j}, y^2_{i,j}],[u_{i,j-1}, y^2_{i,j-1}]\big)^p \\ + 2^{1-p} D_{\mathrm{S}}^{\mathrm{sym}}([u_{i,j}, y^1_{i,j}],[u_{i,j}, y^2_{i,j}],[u_{i,j-1}, y^1_{i,j-1}],[u_{i-1,j}, y^2_{i-1,j}])^p \Big)^{1/p}.\end{aligned} \tag{2.13}$$

Naturally, the above definition of S-TGV based on the Schild's ladder construction is not the only possibility to extend second order TGV to the manifold setting. As already pointed out, in [28] this was accounted for by an axiomatic approach which, for a suitable generalization, also requires additional properties such as a good description of their kernel, and we will see below that indeed this is possible for S-TGV. An alternative definition based on parallel transport presented in [28] uses, instead of D_{S} for point-tuples with different base points, the distance

$$D_{\mathrm{pt}}([x,y],[u,v]) = \left\| \log_x(y) - \mathrm{pt}_x(\log_u(v)) \right\|_x, \tag{2.14}$$

where $\mathrm{pt}_x(z)$ is the parallel transport of $z \in T\mathcal{M}$ to $T_x\mathcal{M}$, and a similar adaption of $D_{\mathrm{S}}^{\mathrm{sym}}$ for bivariate signals. It was shown in [28] that also this version suitably generalizes TGV by transferring some of its main properties to the manifold setting.

Another existing extension of TGV to the manifold setting is the one presented in [15] which is given, in the univariate setting, as

$$\widetilde{\mathrm{TGV}}_{\alpha}^{2}(u) = \inf_{(\xi_i)_i} \sum\nolimits_i \alpha_1 \| \log_{u_i}(u_{i+1}) - \xi_i \|_{u_i} + \alpha_0 \| \xi_i - P_{u_i}(\xi_{i-1}) \|_{u_i}$$

where P_{u_i} approximates the parallel transport of ξ_{i-1} to u_i by first mapping it down to a point tupel $[u_{i-1}, \exp_{u_{i-1}}(\xi_i)]$, then using the pole ladder [75] as an alternative to Schild's ladder to approximate the parallel transport to u_i and finally lifting the transported tuple again to the tangent space via the logarithmic map. In the univariate case, this also generalizes TGV and preserves its main properties such as a well defined kernel. For the bivariate version, [15] uses the standard Jacobian instead of the symmetrized derivative and it remains open to what extend the kernel of TGV is appropriately generalized, also because there is no direct, natural generalization of the kernel of $\mathrm{TGV}_{\alpha}^{2}$ (i.e., affine functions) in the bivariate setting (see the next paragraph for details).

Consistency

Given that there are multiple possibilities of extending vector-space concepts to manifolds, the question arises to what extend the extensions of TV^2, IC and TGV presented above are natural or "the correct ones." As observed in [28], the requirement of suitably transferring the kernel of the vector-space version, which consists of the set of affine functions, is a property that at least allows to reduce the number of possible generalizations. Motivated by this, we consider the zero-set of the manifold extensions of TV^2, IC_{α} and $\mathrm{TGV}_{\alpha}^{2}$. We start with the univariate situation, where a generalization of the notion of "affine" is rather natural.

Definition 2.2 (Univariate Generalization of Affine Signals) Let $u = (u_i)_i$ be a signal in $\mathcal{M}^N$. We say that u is generalized affine or *geodesic* if there exists a geodesic $\gamma : [0, L] \to \mathcal{M}$ such that all points of u are on γ at equal distance and γ is length-minimizing between two subsequent points.

The following proposition relates geodesic functions to the kernel of higher-order regularizers on manifolds. Here, in order to avoid ambiguities arising from subtle difference in the functionals depending on if length-minimizing geodesics are used or not, we assume geodesics are unique noting that the general situation is mostly analogue.

Proposition 2.3 (Consistency, Univariate) *Let $u = (u_i)_i$ in $\mathcal{M}$ be such that all points u_i, u_j are connected by a unique geodesic.*

(i) If u is geodesic, $\mathrm{TV}^2(u) = \mathrm{IC}_\alpha(u) = \text{S-TGV}^2_\alpha(u) = 0$.
(ii) Conversely, if $\mathrm{TV}^2(u) = 0$ *or* $\text{S-TGV}^2_\alpha(u) = 0$ *, then u is geodesic.*

Proof If u is geodesic, it follows that $u_i \in [u_{i-1}, u_{i+1}]_{\frac{1}{2}}$, such that $\mathrm{TV}^2(u) = 0$. In case of IC_α define $v_i = u_1$ (the first point of u) for all i and $w_i = [u_1, u_i]_{2d(u_1,u_i)}$. Then it follows that $u_i \in [v_i, w_i]_{\frac{1}{2}}$ for all i. Further, $\mathrm{TV}((v_i)_i) = 0$ and, since $(w_i)_i$ is geodesic, also $\mathrm{TV}^2((w_i)_i) = 0$ such that $\mathrm{IC}_\alpha(u) = 0$. Regarding S-TGV^2_α, we see that $\text{S-TGV}^2_\alpha(u) = 0$ follows from choosing $(y_i)_i = (u_{i+1})_i$ and noting that $D_S([u_i, u_{i+1}], [u_{i-1}, u_i]) = 0$ since $u_i \in [u_i, u_i]_{\frac{1}{2}}$ and $u_{i+1} \in [u_{i-1}, u_i]_2$. Now conversely, if $\mathrm{TV}^2(u) = 0$, it follows that $u_i \in [u_{i-1}, u_{i+1}]$ for all i such that $(u_i)_i$ is geodesic. If $\text{S-TGV}^2_\alpha(u) = 0$ we obtain $(y_i)_i = (u_{i+1})_i$ and, consequently, $u_{i+1} \in [u_i, u_{i-1}]$ which again implies that u is geodesic. □

Remark 2.4 One can observe that in Proposition 2.3, the counterparts of ii) for IC_α is missing. Indeed, an easy counterexample shows that this assertion is not true, even in case of unique geodesics: Consider $\mathcal{M} = \mathcal{S}^2 \cap ([0, \infty) \times \mathbb{R} \times [0, \infty))$ and $\varphi_1 = -\pi/4$, $\varphi_2 = 0$, $\varphi_3 = \pi/4$ and $\psi = \pi/4$ define

$$u_i = (\cos(\varphi_i)\sin(\psi), \sin(\varphi_i)\sin(\psi), \cos(\psi))$$

$$w_i = (\cos(\varphi_i), \sin(\varphi_i), 0)$$

and $v_i = (0, 0, 1)$ for all i. Then $u_i \in [v_i, w_i]_{\frac{1}{2}}$, $\mathrm{TV}(v) = 0$, $\mathrm{TV}^2(w) = 0$, hence $\mathrm{IC}_\alpha(u) = 0$ but u is not geodesic.

In the bivariate setting, a generalization of an affine function is less obvious: It seems natural that $u = (u_{i,j})_{i,j}$ being *generalized affine* or, in the notion above, *geodesic* should imply that for each k, both $(u_{i,k})_i$ and $(u_{k,j})_j$ are geodesics. However, to achieve a generalization of affine, an additional condition is necessary to avoid functions of the form $(x, y) \mapsto xy$. In [28] this condition was to require also the signal $(u_{i+k,j-k})_k$ to be geodesic for each i, j. While this has the disadvantage favoring one particular direction, it is less restrictive than to require, in addition, also $(u_{i+k,j+k})_k$ to be geodesic. As shown in the following proposition, bivariate functions that are geodesic in the sense of [28] coincide with the kernel of S-TGV^2_α. TV^2 on the other hand, gives rise to a different notion of affine.

Proposition 2.5 (Kernel, Bivariate) *Let $u = (u_{i,j})_{i,j}$ be such that all points of u are connected by a unique geodesics.*

(i) $\text{S-TGV}^2_\alpha(u) = 0$ *if and only if* $(u_{k,j_0})_k$, $(u_{i_0,k})_k$ *and* $(u_{i_0+k,j_0-k})_k$ *is geodesic for each* i_0, j_0.

(ii) $\text{TV}^2(u) = 0$ *if and only if* $(u_{k,j_0})_k$, $(u_{i_0,k})_k$ *is geodesic for each* i_0, j_0 *and* $[u_{i+1,j}, u_{i,j-1}]_{\frac{1}{2}} \cap [u_{i,j}, u_{i+1,j-1}]_{\frac{1}{2}} \neq \emptyset$ *for all* i, j.

Proof For S-TGV^2_α, a stronger version of this result is proven in [28, Theorem 2.18]. For TV^2, this follows analogously to the univariate case directly from the definition of D_c and D_{cc}. □

Higher-Order Regularized Denoising

The next proposition shows that TV^2 and S-TGV based denoising of manifold valued data is indeed well-posed.

Proposition 2.6 *Both* TV^2 *and* S-TGV^2_α *are lower semi-continuous w.r.t. convergence in* $(\mathcal{M}, \mathrm{d})$. *Further, for* $R \in \{\alpha\text{TV}^2, \text{S-TGV}^2_\alpha\}$, *where* $\alpha > 0$ *in the case of* TV^2, *the problem*

$$\inf_{u=(u_{i,j})_{i,j}} R(u) + \sum\nolimits_{i,j} d(u_{i,j}, f_{i,j})^q$$

admits a solution.

Proof The proof is quite standard: by the Hopf-Rinow theorem, it is clear that the claimed existence follows once lower semi-continuity of R can be guaranteed. For S-TGV^2_α, this is the assertion of [28, Theorem 3.1]. For TV^2, it suffices to show lower semi-continuity of D_c and D_{cc}. We provide a proof for D_c, the other case works analogously. Take $u^n = (u^n_-, u^n_\circ, u^n_+)_n$ converging to $(u_-, u_\circ, u_+)$ and take $(c_n)_n$ with $c_n \in [u^n_-, u^n_+]$ such that $d(c_n, u^n_\circ) \leq \inf_{c \in [u^n_-, u^n_+]} d(c, u^n_\circ) + 1/n$. Then, from boundedness of $(u^n)_n$ and since $d(c^n, u^n_-) \leq d(u^n_+, u^n_-)$ we obtain boundedness of $(c^n)_n$, hence a (non-relabeled) subsequence converging to some $c \in \mathcal{M}$ exists. From uniform convergence of the geodesics $\gamma^n : [0, 1] \to \mathcal{M}$ connecting u^n_- and u^n_+ such that $c^n = \gamma^n(1/2)$ (see for instance [28, Lemma 3.3]) we obtain that $c \in [u_-, u_+]_{\frac{1}{2}}$ such that $D_c(u_-, u_\circ, u_+) \leq d(c, u_\circ) \leq \liminf_n d(c_n, u^n_\circ) \leq \liminf_n D_c(u^n_-, u^n_\circ, u^\circ_+)$. □

Remark 2.7 We note that the arguments of Proposition 2.6 do not apply to IC since we cannot expect $(v_{i,j})_{i,j}$ and $(w_{i,j})_{i,j}$ with $u_{i,j} \in [v_{i,j}, w_{i,j}]_{\frac{1}{2}}$ to be bounded in general. Indeed, this is similar to vector-space infimal convolution, only that there, one can factor out the kernel of TV and still obtain existence.

2.3.2 Algorithmic Realization

Here we discuss the algorithmic realization and illustrate the results of TV^2 and S-TGV regularized denoising; for IC we refer to [14, 15]. We note that, in contrast to the TV functional, the TV^2 and the S-TGV^2_α functional are not convex on Hadamard manifolds (cf. [9, Remark 4.6]) such that we cannot expect to obtain numerical algorithms that provably converge to global optimal solutions as for TV in Hadamard spaces (cf. Theorem 2.1). Nevertheless, the cyclic proximal point algorithm and the parallel proximal point algorithm as described in Sect. 2.2.2 are applicable in practice. In the following, we discuss the corresponding splittings and proximal mappings, where we focus on the univariate case since, similar to Sect. 2.2.2, the bivariate case for $p = 1$ can be handled analogously; for details we refer to [9, 28].

For TV^2 denoising, we may use the splitting

$$\tfrac{1}{q}\sum\nolimits_i \mathrm{d}(u_i, f_i)^q + \alpha \mathrm{TV}^2((u_i)_i) = F_0(u) + F_1(u) + F_2(u) + F_3(u)$$

where $F_0(u) = \frac{1}{q}\sum_i \mathrm{d}(u_i, f_i)^q$, and

$$F_j(u) = \sum\nolimits_i D_c(u_{3i-1+j}, u_{3i+j}, u_{3i+1+j}), \qquad j = 1, \ldots, 3.$$

Due to the decoupling of the summands, the computation of the proximal maps of $(F_i)_{i=0}^3$ reduces to the computation of the proximal maps of $x \mapsto \mathrm{d}(x, f)^q$ and of $(x_1, x_2, x_3) \mapsto D_c(x_1, x_2, x_3)$. The former is given explicitly as in (2.8), while for the latter, following [9], we use a subgradient descent scheme (see for instance [47]) to iteratively solve

$$\min_{x_{k-1}, x_k, x_{k+1}} \frac{1}{2}\sum\nolimits_{l=k-1}^{k+1} \mathrm{d}^2(x_l, h_l) + \lambda D_c(x_{k-1}, x_k, x_{k+1})$$

for $(h_{k-1}, h_k, h_{k+1}) \in \mathcal{M}^3$ the given point where the proximal map needs to be computed. For this purpose, we employ Algorithm 3, which requires to compute the subgradient of D_c as well as the derivative of d. The latter is, at a point (x_{k-1}, x_k, x_{k+1}) given as $-\Big(\log_{x_{k-1}}(h_{k-1}), \log_{x_k}(h_k), \log_{x_{k+1}}(h_{k+1})\Big)^T$. Regarding the computation of D_c, in order to avoid pathological (and practically irrelevant) constellations, we assume that the arguments x_{k-1}, x_{k+1} are not cut points, such that there is exactly one length minimizing geodesics connecting x_{k-1} and x_{k+1} and the corresponding midpoint is unique. With these assumptions, the derivative w.r.t. the first component and for a point (x, y, z) with $y \neq [x, z]_{\frac{1}{2}}$ is given as

$$\partial_y D_c(x, \cdot, z)(y) = \log_y([x, z]_{\frac{1}{2}}) / \| \log_y([x, z]_{\frac{1}{2}}) \|_y.$$

Algorithm 3 Subgradient descent for solving $\min_x F(x)$

```
SGD($x_0, (\lambda_i)_i$)
$l = 0$
    repeat until stopping criterion fulfilled
        $x_{k+1} \leftarrow \exp_{x_k}(-\lambda_k \partial F(x_k))$
        $k \leftarrow k + 1$
return $x$
```

The derivative w.r.t. x is given by

$$\partial_x D_c(\cdot, y, z)(x) = \sum\nolimits_{l=1}^{d} \left\langle \log_c(y)/\| \log_c(y)\|_c, D_x c(\xi_l)\right\rangle$$

where we denote $c = [x, z]_{\frac{1}{2}}$. Here, $D_x c$ is the differential of the mapping $c : x \mapsto [x, z]_{\frac{1}{2}}$ which is evaluated w.r.t. the elements of an orthonormal basis $(\xi_l)_{l=1}^{d}$ of the tangent space at x. The derivative w.r.t. z is computed analoguosly due to symmetry and for the particular case that $y = [x, z]_{\frac{1}{2}}$ we refer to [9, Remark 3.4]. While the formulas above provide general forms of the required derivatives, a concrete realization can be done using explicit formulae for Jacobi fields in the particular manifold under consideration; we refer to [9] for explicit versions. For the bivariate case, we also refer to [9] for the derivative of D_{cc}, which can be computed with similar techniques as D_c.

For TGV, we again start with the univariate case and consider the S-TGV$_\alpha^2$ variant. We consider the splitting

$$\tfrac{1}{q} \sum\nolimits_i \mathrm{d}(u_i, f_i)^q + \mathrm{TGV}_\alpha^2((u_i)_i) = F_0(u) + F_1(u) + F_2(u) + F_3(u)$$

where $F_0(u) = \frac{1}{q} \sum_i \mathrm{d}(u_i, f_i)^q$, $F_1(u) = \sum_i \mathrm{d}(u_{i+1}, y_i)$, and

$$F_2(u) = \sum_{i:i \text{ even}} D_{\mathrm{S}}([u_i, y_i], [u_{i-1,} y_{i-1}]), \quad F_3(u) = \sum_{i:i \text{ odd}} D_{\mathrm{S}}([u_i, y_i], [u_{i-1} y_{i-1}]).$$

Here, as an advantage of this particular version of TGV that uses only points on the manifold, we see that the proximal mappings of F_1 are explicit as in (2.10) and, since again the proximal mapping of F_0 is given for $q \in \{1, 2\}$ explicitly by (2.8), we are left to compute the proximal mappings for D_{S}. To this aim, we again apply the subgradient descent method as in Algorithm 3, where the required derivatives of D_{S} are provided in [28]. In particular, again assuming uniqueness of geodesics to avoid pathological situations, we have for points $[u_i, y_i], [u_{i-1}, y_{i-1}]$ with $y_i \neq [u_{i-1}, [u_i, y_{i-1}]_{\frac{1}{2}}]_2$ that

$$\nabla_{y_i} D_{\mathrm{S}} = -\log_{y_i} S(u_{i-1}, y_{i-1}, u_i)/\left\| \log_{y_i} S(u_{i-1}, y_{i-1}, u_i)\right\| \tag{2.15}$$

where

$$S(u_{i-1}, y_{i-1}, u_i) = [u_{i-1}, [u_i, y_{i-1}]_{\frac{1}{2}}]_2 \tag{2.16}$$

denotes the result of applying the Schild's construction to the respective arguments. Derivatives w.r.t. the arguments u_{i-1} and y_{i-1} are given in abstract form as

$$\nabla_{u_{i-1}} D_{\mathrm{S}} = -T_1 \left(\log_{S(u_{i-1}, y_{i-1}, u_i)} y_i / \| \log_{S(u_{i-1}, y_{i-1}, u_i)} y_i \| \right), \tag{2.17}$$

$$\nabla_{y_{i-1}} D_{\mathrm{S}} = -T_2 \left(\log_{S(u_{i-1}, y_{i-1}, u_i)} y_i / \| \log_{S(u_{i-1}, y_{i-1}, u_i)} y_i \| \right), \tag{2.18}$$

where T_1 is the adjoint of the differential of the mapping $u_{i-1} \mapsto [u_{i-1}, [u_i, y_{i-1}]_{1/2}]_2$, and T_2 is the adjoint of the differential of the mapping $y_{i-1} \mapsto [u_{i-1}, [u_i, y_{i-1}]_{1/2}]_2$. The differential w.r.t. u_i is obtained by symmetry. Again, the concrete form of these mappings depends on the underlying manifold and we refer to [28] for details. Regarding points with $y_i \neq [u_{i-1}, [u_i, y_{i-1}]_{1/2}]_2$ we note that for instance the four-tuple consisting of the four zero-tangent vectors sitting in $u_i, u_{i-1}, [u_{i-1}, [u_i, y_{i-1}]_{1/2}]_2,\ y_{i-1}$ belongs to the subgradient of D_{S}. The algorithm for bivariate TGV-denoising can be obtained analogously, where we refer to [28] for the computation of the derivative of $D_{\mathrm{S}}^{\mathrm{sym}}$. The algorithm for TGV-denoising based on the parallel variant (2.14) employs the proximal mappings of F_0 and F_1 as well. Implementation of the proximal mappings of F_2 and F_3 based on subgradient descent can be found in [28] and [26].

Numerical Examples

We illustrate the algorithm with numerical examples. At first, we provide a comparison between TV, TV^2 and S-TGV regularization for synthetic $\mathbb{S}^2$-valued image data, taken from [28]. The results can be found in Fig. 2.6, where for each approach we optimized over the involved parameters to achieve the best ΔSNR result, with ΔSNR being defined for ground truth, noisy and denoised signals h, f and u, respectively, as $\Delta\mathrm{SNR} = 10 \log_{10} \left(\frac{\sum_i \mathrm{d}(h_i, f_i)^2}{\sum_i \mathrm{d}(h_i, u_i)^2} \right)$ dB. We observe that the TV regularization produces significant piecewise constancy artifacts (staircasing) on the piecewise smooth example. The result of TV^2 shows no visible staircasing, but smoothes the discontinuities to some extend. S-TGV is able to better reconstruct sharp interfaces while showing no visible staircasing.

As second example, Fig. 2.7 considers the denoising of interferometric synthetic aperture radar (InSAR) images. Again, it can be observed that S-TGV has a significant denoising effect while still preserving sharp interfaces.

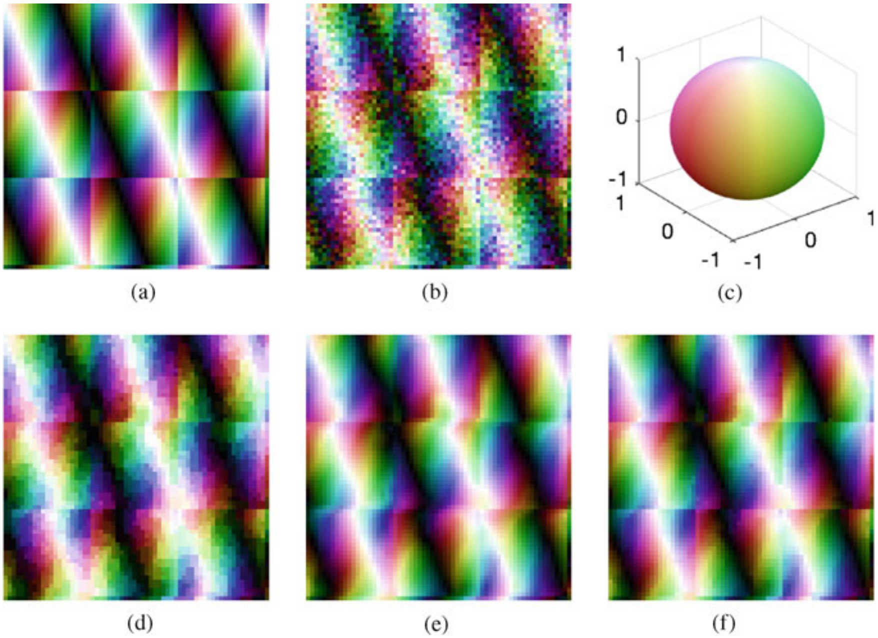


Fig. 2.6 Comparison of first and second-order total variation as well as S-TGV on an S^2-valued image from [9]. (**a**) Original. (**b**) Noisy data. (**c**) Color code for S^2. (**d**) TV, $\Delta\mathrm{SNR} = 5.7\,\mathrm{dB}$. (**e**) TV^2, $\Delta\mathrm{SNR} = 8.4\,\mathrm{dB}$. (**f**) S-TGV, $\Delta\mathrm{SNR} = 8.9\,\mathrm{dB}$. Images taken from [28]

2.4 Mumford-Shah Regularization for Manifold Valued Data

The Mumford and Shah model [79, 80], also called Blake-Zisserman model [20], is a powerful variational model for image regularization. The regularization term measures the length of the jump set and, within segments, it measures the quadratic variation of the function. The resulting regularization is a smooth approximation to the image/signal which, at the same time, allows for sharp edges at the discontinuity set. Compared with the TV regularizer, it does not penalize the jump height and, due to the quadratic variation on the complement of the edge set, no staircasing effects appear. (Please note that no higher order derivatives/differences are involved here.) The piecewise constant variant of the Mumford-Shah model (often called Potts model) considers piecewise constant functions (which then have no variation on the segments) and penalizes the length of the jump sets. Typical applications of these functionals are smoothing and the use within a segmentation pipeline. For further information considering these problems for scalar data from various perspectives (calculus of variation, stochastics, inverse problems) we refer to [4, 20, 23, 30, 51, 52, 54, 68, 85, 118] and the references therein. These references also deal with fundamental questions such as the existence of minimizers. Mumford-Shah and Potts problems are computationally challenging since the functionals are

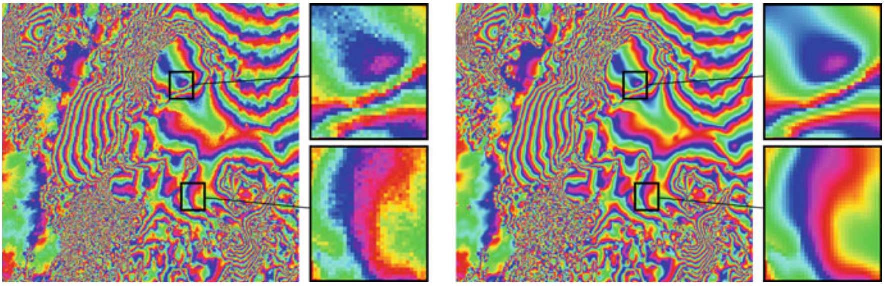


Fig. 2.7 *Left:* InSAR image from [102]. *Right:* Result of S-TGV. Image taken from [28]

non-smooth and non-convex. Even for scalar data, both problems are known to be NP-hard in dimensions higher than one [2, 22, 106]. This makes finding a (global) minimizer infeasible. Because of its importance in image processing however, many approximative strategies have been proposed for scalar- and vector valued data. Among these strategies are graduated non-convexity [20], approximation by elliptic functionals [4], graph cuts [22], active contours [104], convex relaxations [84], iterative thresholding approaches [51], and alternating direction methods of multipliers [66].

In the context of DTI, the authors of [109, 110] consider a Chan-Vese model for positive matrix-valued data, i.e., for manifold-valued data in Pos_3, as well as a piecewise smooth variant. (We recall that Chan-Vese models are variants of Potts models for the case of two labels.) Their method is based on a level-set active-contour approach. In order to reduce the computational load in their algorithms (which is due to the computation of Riemannian means for a very large number of points) the authors resort to non-Riemannian distance measures in [109, 110]. Recently, a fast recursive strategy for computing the Riemannian mean has been proposed and applied to the piecewise constant Chan-Vese model (with two labels) in [38].

We mention that for $\mathbb{S}^1$-valued data, a noniterative exact solver for the univariate Potts problem has been proposed in [99].

In this section, we consider Mumford-Shah and Potts problems for (general) manifold-valued data and derive algorithms for these problems. As in the linear case, typical applications of these functionals are smoothing and also segmentation; more precisely, they can serve as an initial step of a segmentation pipeline. In simple cases, the induced edge set may yield a reasonable segmentation directly.

2.4.1 Models

We start out with Mumford-Shah and Potts problems for univariate manifold-valued data $(f_i)_{i=1}^N \in \mathcal{M}^N$, with $\mathcal{M}$ again being a complete, finite dimensional Riemannian manifold. The discrete Mumford-Shah functional reads

$$B_{\alpha,\gamma}(x) = \frac{1}{q}\sum_{i=1}^{N} \mathrm{d}(x_i, f_i)^q + \frac{\alpha}{p}\sum_{i\notin\mathcal{J}(x)} \mathrm{d}(x_i, x_{i+1})^p + \gamma\,|\mathcal{J}(x)|\,, \tag{2.19}$$

where d is the distance with respect to the Riemannian metric in the manifold $\mathcal{M}$, $\mathcal{J}$ is the jump set of x and $p, q \in [1, \infty)$. The jump set is given by $\mathcal{J}(x) = \{i : 1 \leq i < n \text{ and } \mathrm{d}(x_i, x_{i+1}) > s\}$, and $|\mathcal{J}(x)|$ denotes the number of jumps. The jump height s and the parameter γ are related via $\gamma = \alpha s^p/p$. We rewrite (2.19) using a truncated power function to obtain the Blake-Zisserman type form

$$B_{\alpha,s}(x) = \frac{1}{q}\sum_{i=1}^{N} \mathrm{d}(x_i, f_i)^q + \frac{\alpha}{p}\sum_{i=1}^{N-1} \min(s^p, \mathrm{d}(x_i, x_{i+1})^p), \tag{2.20}$$

where s is the argument of the power function $t \mapsto t^p$, at which $\mathrm{d}(x_i, x_{i+1})^p$ is truncated at. The discrete univariate Potts functional for manifold-valued data reads

$$P_\gamma(x) = \frac{1}{q}\sum_{i=1}^{n} \mathrm{d}(x_i, f_i)^q + \gamma|\mathcal{J}(x)|, \tag{2.21}$$

where an index i belongs to the jump set of x if $x_i \neq x_{i+1}$.

In the multivariate situation, the discretization of the Mumford-Shah and Potts problem is not as straightforward as in the univariate case. A simple finite difference discretization with respect to the coordinate directions is known to produce undesired block artifacts in the reconstruction [31]. The results improve significantly when including further finite differences such as the diagonal directions [31, 93, 97]. To define bivariate Mumford-Shah and Potts functionals, we introduce the notation $\mathrm{d}^q(x, y) = \sum_{i,j} \mathrm{d}^q(x_{ij}, y_{ij})$ for the q-distance of two manifold-valued images $x, y \in \mathcal{M}^{N\times M}$. For the regularizing term, we employ the penalty function

$$\Psi_a(x) = \sum\nolimits_{i,j} \psi(x_{(i,j)+a}, x_{ij}),$$

where $a \in \mathbb{Z}^2 \setminus \{0\}$ denotes a direction, and the potentials ψ are given by

$$\psi(w, z) = \frac{1}{p}\min(s^p, \mathrm{d}(w, z)^p), \qquad \text{and} \qquad \psi(w, z) = \begin{cases} 1, & \text{if } w \neq z, \\ 0, & \text{if } w = z, \end{cases} \tag{2.22}$$

for $w, z \in \mathcal{M}$, in the Mumford-Shah case and in the Potts case, respectively. We define the discrete multivariate Mumford-Shah and Potts problems by

$$\min_{x\in\mathcal{M}^{N\times M}} \frac{1}{q}\mathrm{d}^q(x, f) + \alpha\sum_{s=1}^{R} \omega_s \Psi_{a_s}(x), \tag{2.23}$$

where the finite difference vectors $a_s \in \mathbb{Z}^2 \setminus \{0\}$ belong to a neighborhood system $\mathcal{N}$ and $\omega_1, \ldots, \omega_R$ are non-negative weights. As observed in [93], a reasonable neighborhood system is

$$\mathcal{N} = \{(1,0); (0,1); (1,1); (1,-1)\}$$

with the weights $\omega_1 = \omega_2 = \sqrt{2} - 1$ and $\omega_3 = \omega_4 = 1 - \frac{\sqrt{2}}{2}$ as in [93]. It provides a sufficiently isotropic discretization while keeping the computational load at a reasonable level. For further neighborhood systems and weights we refer to [31, 93].

For both, the univariate and multivariate discrete Mumford-Shah and Potts functionals, the following result regarding the existence of minimizers holds.

Theorem 2.8 *Let $\mathcal{M}$ be a complete Riemannian manifold. Then the univariate and multivariate discrete Mumford-Shah and Potts problems* (2.19)*,* (2.21)*, and* (2.23) *have a minimizer.*

A proof may be found in [116]. We note that most of the data spaces in applications are complete Riemannian manifolds.

2.4.2 Algorithmic Realization

We start with the univariate Mumford-Shah and Potts problems (2.19) and (2.21). These are not only important on their own, variants also appear as subproblems in the algorithms for the multivariate problems discussed below.

Dynamic Programming Scheme

To find a minimizer of the Mumford-Shah problem (2.19) and the Potts problem (2.21), we employ a general dynamic programming scheme which was employed for related scalar and vectorial problems in various contexts [30, 53, 80, 96, 115, 117]. We briefly explain the idea where we use the Mumford-Shah problem as example. We assume that we have already computed minimizers x^l of the functional $B_{\alpha,\gamma}$ associated with the partial data $f_{1:l} = (f_1, \ldots, f_l)$ for each $l = 1, \ldots, r-1$ and some $r \leq N$. (Here, we use the notation $f_{l:r} := (f_l, \ldots, f_r)$.) We explain how to compute a minimizer x^r associated to data $f_{1:r}$. For each x^{l-1} of length $l-1$, we define a candidate $x^{l,r} = (x^{l-1}, h^{l,r}) \in \mathcal{M}^r$ which is the concatenation of x^{l-1} with a vector $h^{l,r}$ of length $r-l+1$; We choose $h^{l,r}$ as a minimizer of the problem

$$\epsilon_{l,r} = \min_{h \in M^{r-l+1}} \sum\nolimits_{i=l}^{r-1} \frac{\alpha}{p} \mathrm{d}^p(h_i, h_{i+1}) + \frac{1}{q} \sum\nolimits_{i=l}^{r} \mathrm{d}^q(h_i, f_i), \tag{2.24}$$

where $\epsilon_{l,r}$ is the best approximation error on the (discrete) interval $(l, \ldots, r)$. Then we calculate

$$\min_{l=1,\ldots,r} \left\{ B_{\alpha,\gamma}(x^{l-1}) + \gamma + \epsilon_{l,r} \right\}, \tag{2.25}$$

which coincides with the minimal functional value of $B_{\alpha,\gamma}$ for data $f_{1:r}$. We obtain the corresponding minimizer $x^r = x^{l^*,r}$, where l^* is a minimizing argument in (2.25). We successively compute x^r for each $r = 1, \ldots, N$ until we end up with full data f. For the selection process, only the l^* and the $\epsilon_{l,r}$ have to be computed; the optimal vectors x^r are then computed in a postprocessing step from these data; see, e.g., [53] for further details. This skeleton (without computing the $\epsilon_{l,r}$) has quadratic complexity with respect to time and linear complexity with respect to space. In the concrete situation, it is thus important to find fast ways for computing the approximation errors $\epsilon_{l,r}$. We will discuss this below for our particular situation. In practice, the computation is accelerated significantly using pruning strategies [73, 93].

Algorithms for the Univariate Mumford-Shah and Potts Problem

To make the dynamic program work for the Mumford-Shah problem with manifold-valued data, we have to compute the approximation errors $\epsilon_{l,r}$ in (2.24). These are L^q-V^p type problems: the data term is a manifold ℓ^q distance and the second term is a pth variation; in particular, for $p = 1$ we obtain TV minimization problems. These L^q-V^p problems can be solves using the proximal point schemes discussed in Sect. 2.2.1; for details, we refer to [114] where in particular the corresponding proximal mappings are calculated in terms of geodesic averages for the important case of quadratic variation $p = 2$.

To make the dynamic program work for the Potts problem with manifold-valued data, we have to compute the approximation errors $\epsilon_{l,r}$ for the problem $\epsilon_{l,r} = \min_{h \in \mathcal{M}^{r-l+1}} \frac{1}{q} \sum_{i=l}^{r} \mathrm{d}^q(h_i, f_i)$, under the assumption that h is a constant vector. Hence we have to compute

$$\epsilon_{l,r} = \min_{h \in \mathcal{M}} \frac{1}{q} \sum\nolimits_{i=l}^{r} \mathrm{d}^q(h, f_i). \tag{2.26}$$

We observe that, by definition, a minimizer of (2.26) is given by an intrinsic mean for $q = 2$, and by an intrinsic median for $q = 1$, respectively.

As already discussed, a mean is general not uniquely defined since the minimization problem has no unique solution in general. Further, there is no closed form expression in general. One means to compute the intrinsic mean is the gradient descent (already mentioned in [70]) via the iteration

$$h^{k+1} = \exp_{h^k} \sum_{i=l}^{r} \tfrac{1}{r-l+1} \log_{h^k} f_i, \tag{2.27}$$

where again log denotes the inverse exponential map. Further information on convergence and other related topics can for instance be found in the papers [1, 50]

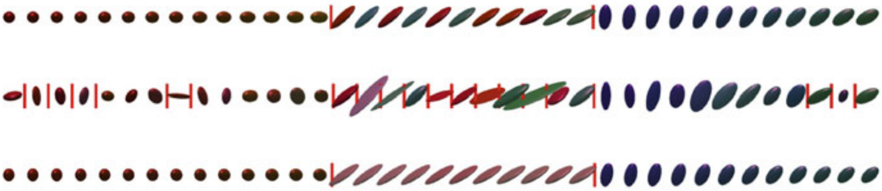


Fig. 2.8 Univariate Mumford-Shah regularization ($p, q = 1$) using dynamic programming. (The red steaks indicate jumps.) *Top.* Original signal. *Middle.* Data with Rician noise added. *Bottom.* Regularized signal. Mumford-Shah regularization removes the noise while preserving the jump

and the references given there. Newton's method was also applied to this problem in the literature; see, e.g., [49]. It is reported in the literature and also confirmed by the authors' experience that the gradient descent converges rather fast; in most cases, 5–10 iterations are enough. For general $p \neq 1$, the gradient descent approach works as well. The case $p = 1$ amounts to considering the intrinsic median together with the intrinsic absolute deviation. In this case, we may apply a subgradient descent which in the differentiable part amounts to rescaling the tangent vector given on the right-hand side of (2.27) to length 1 and considering variable step sizes which are square-integrable but not integrable; see, e.g., [5].

A speedup using the structure of the dynamic program is obtained by initializing with previous output. More precisely, when starting the iteration of the mean for data $f_{l+1:r}$, we can use the already computed mean for the data $f_{l:r}$ as an initial guess. We notice that this guess typically becomes even better the more data items we have to compute the mean for, i.e., the bigger $r - l$ is. This is important since this case is the computational more expensive part and a good initial guess reduces the number of iterations needed.

We have the following theoretical guarantees.

Theorem 2.9 *In a Cartan-Hadamard manifold, the dynamic programming scheme produces a global minimizer for the univariate Mumford-Shah problem* (2.19) *and the discrete Potts problem* (2.21)*, accordingly.*

A proof can be found in [116]. In this reference, also guarantees are obtained for Potts problems for general complete Riemannian manifold under additional assumptions; cf. [116, Theorem 3]. In Fig. 2.8, the algorithm for the univariate case is illustrated for Mumford-Shah regularization for the Cartan-Hadamard manifold of positive matrices.

Multivariate Mumford-Shah and Potts Problems

We now consider Mumford-Shah and Potts regularization for manifold-valued images. Even for scalar data, these problems are NP hard in dimensions higher than one [2, 106]. Hence, finding global minimizers is not tractable anymore in the multivariate case in general. The goal is to derive approximative strategies that perform well in practice. We present a splitting approach: we rewrite (2.23) as the constrained problem

$$\min_{x_1,\ldots,x_R} \sum_{s=1}^{R} \frac{1}{qR} \mathrm{d}^q(x_s, f) + \alpha\omega_s \Psi_{a_s}(x_s) \qquad \text{s. t. } x_s = x_{s+1},\ s \in \{1, \ldots, R\}, \tag{2.28}$$

with the convention $x_{R+1} = x_1$. We use a penalty method (see e.g. [18]) to obtain the unconstrained problem

$$\min_{x_1,\ldots,x_R} \sum\nolimits_{s=1}^{R} \omega_s q R \alpha \Psi_{a_s}(x_s) + \mathrm{d}^q(x_s, f) + \mu_k \mathrm{d}^q(x_s, x_{s+1}).$$

We use an increasing coupling sequence $(\mu_k)_k$ which fulfills the summability condition $\sum_k \mu_k^{-1/q} < \infty$. This specific splitting allows us to minimize the functional blockwise, that is, with respect to the variables $x_1, \ldots, x_R$ separately. Performing blockwise minimization yields the algorithm

$$\begin{cases} x_1^{k+1} \in \operatorname{argmin}_x q R \omega_1 \alpha \Psi_{a_1}(x) + \mathrm{d}^q(x, f) + \mu_k \mathrm{d}^q(x, x_R^k), \\ x_2^{k+1} \in \operatorname{argmin}_x q R \omega_2 \alpha \Psi_{a_2}(x) + \mathrm{d}^q(x, f) + \mu_k \mathrm{d}^q(x, x_1^{k+1}), \\ \quad \vdots \\ x_R^{k+1} \in \operatorname{argmin}_x q R \omega_R \alpha \Psi_{a_R}(x) + \mathrm{d}^q(x, f) + \mu_k \mathrm{d}^q(x, x_{R-1}^{k+1}). \end{cases} \tag{2.29}$$

We notice that each line of (2.29) decomposes into univariate subproblems of Mumford-Shah and Potts type, respectively. The subproblems are almost identical with the univariate problems above. Therefore, we can use the algorithms developed above with a few minor modification. Details may be found in [116].

There is the following result ensuring that the algorithm terminates.

Theorem 2.10 *For Cartan-Hadamard manifold-valued images the algorithm* (2.29) *for both the Mumford-Shah and the Potts problem converges.*

A proof can be found in [116].

A result of the algorithm is illustrated in Fig. 2.9 for Mumford-Shah regularization in the Cartan-Hadamard manifold of positive matrices. The data set was taken from the Camino project [39].

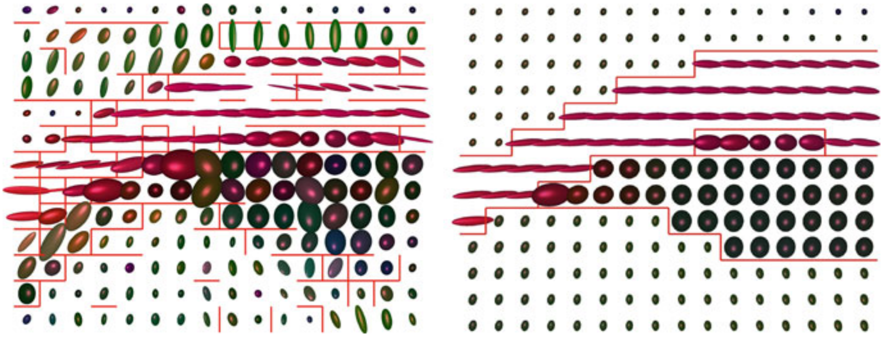


Fig. 2.9 *Left.* Part of a corpus callosum of a human brain [39]. *Right:* Mumford-Shah regularization with $p, q = 1$. The noise is significantly reduced and the edges are preserved. Here, the edge set (depicted as red lines) of the regularization yields a segmentation

2.5 Dealing with Indirect Measurements: Variational Regularization of Inverse Problems for Manifold Valued Data

In this section, we consider the situation when the data is not measured directly. More precisely, we consider the manifold valued analogue of the discrete inverse problem of reconstructing the signal u in the equation $\mathcal{A}u \approx f$, with given noisy data f. Here, $\mathcal{A} \in \mathbb{R}^{K \times N}$ is a matrix with unit row sums (and potentially negative items), and u is the objective to reconstruct. In the linear case, the corresponding variational model, given discrete data $f = (f_i)_{i=1}^K$, reads

$$\operatorname{argmin}_{u \in \mathbb{R}^N} \frac{1}{q} \sum\nolimits_{i=1}^{K} \left| \sum\nolimits_{j=1}^{N} \mathcal{A}_{i,j} u_j - f_i \right|^q + R_\alpha(u). \tag{2.30}$$

Here, the symbol R_α denotes a regularizing term incorporating prior assumption on the signal. The process of finding u given data f via minimizing (2.30) is called Tikhonov-Phillips regularization. For a general account on inverse problems and applications in imaging we refer to the books [17, 46].

In this section we consider models for variational (Tikhonov-Phillips) regularization for indirect measurement terms in the manifold setup, we present algorithms for the proposed models and we show the potential of the proposed schemes. The material is mostly taken from [94].

2.5.1 Models

We introduce models for variational (Tikhonov-Phillips) regularization of indirectly measured data in the manifold setup. The approach is as follows: Given a matrix $\mathcal{A} = (\mathcal{A}_{i,j})_{i,j} \in \mathbb{R}^{K \times N}$ with unit row sum, we replace the euclidean distance in $|\sum_j \mathcal{A}_{i,j}u_j - f_i|$ by the Riemannian distance $\mathrm{d}(\cdot, f)$ in the complete, finite dimensional Riemannian manifold $\mathcal{M}$ and the weighted mean $\sum_j \mathcal{A}_{i,j}u_j$ by the weighted Riemannian center of mass [70, 71] denoted by $\mathrm{mean}(\mathcal{A}_{i,\cdot}, u)$ which is given by

$$\mathrm{mean}(\mathcal{A}_{i,\cdot}, u) = \mathrm{argmin}_{v\in\mathcal{M}} \sum_j \mathcal{A}_{i,j}\, \mathrm{d}(v, u_j)^2. \tag{2.31}$$

We consider the manifold analogue of the variational problem (2.30) given by

$$\mathrm{argmin}_{u\in\mathcal{M}^N} \frac{1}{q} \sum_{i=1}^{K} \mathrm{d}\left(\mathrm{mean}(\mathcal{A}_{i,\cdot}, u), f_i\right)^q + R_\alpha(u). \tag{2.32}$$

Here $R_\alpha(u)$ is a regularizing term, for instance

$$R_\alpha(u) = \alpha \mathrm{TV}(u), \qquad \text{or} \qquad R_\alpha(u) = \mathrm{TGV}^2_\alpha(u), \tag{2.33}$$

where $\mathrm{TV}(u)$ denotes the total variation as discussed in Sect. 2.2, and $\mathrm{TGV}^2_\alpha(u)$ denotes the total generalized variation for the discrete manifold valued target u as discussed in Sect. 2.3, for instance, the Schild variant and the parallel transport variant of TGV.

We note that also other regularizers R such as the Mumford-Shah and Potts regularizers of Sect. 2.4 and the wavelet sparse regularizers of Sect. 2.6 may be employed.

We point out that our setup includes the manifold analogue of convolution operators (a matrix with constant entries on the diagonals), e.g., modeling blur. Further, we notice that the discussion includes the multivariate setup (by serializing).

There are the following well-posedness results for the variational problems, i.e., results on the existence of minimizers. For a general regularizer R_α, under a coercivity type condition in the manifold setup the existence of a minimizer is guaranteed as the following theorem shows.

Theorem 2.11 *We consider a sequence of signals $(u^k)_k$ in $\mathcal{M}^N$ and use the notation* $\mathrm{diam}(u^k)$ *to denote the diameter of a single element u^k (viewed as N points in $\mathcal{M}$) of the sequence $\{u^k \mid k \in \mathbb{N}\}$. If R_α is a regularizing term such that $R_\alpha(u^k) \to \infty$, as* $\mathrm{diam}(u^k) \to \infty$, *and R_α is lower semicontinuous, then the variational problem* (2.32) *with indirect measurement term has a minimizer.*

This theorem is formulated as Theorem 1 in [94] and proved there. In particular, it applies to the TV regularizers and their analogues considering qth variation instead of total variation as well as mixed first-second order regularizers of the form $\alpha_1\mathrm{TV}+\alpha_0\mathrm{TV}^2$ with $\alpha_1 > 0, \alpha_0 \geq 0$.

Theorem 2.12 *The inverse problem* (2.32) *for manifold-valued data with* TV *regularizer has a minimizer. The same statement applies to mixed first and second order regularizers of the form $\alpha_1\mathrm{TV} + \alpha_0\mathrm{TV}^2$ with $\alpha_1, \alpha_0 \in [0, \infty)$, $\alpha_1 > 0$.*

This statement is part of [94, Theorem 6] and proved there. We note that, although the TGV^2_α regularizer using either the Schild or the parallel transport variant of Sect. 2.3 is lower semicontinuous (cf. [28]) Theorem 2.11 does not apply. The same issue occurs with pure TV^2 regularization. To overcome this, results with weaker conditions on R and additional conditions on $\mathcal{A}$ have been established to ensure the existence of minimizers; cf. the discussion in [94], in particular [94, Theorem 7]. The mentioned theorem applies to TGV^2_α and pure second order TV regularizers. The conditions on $\mathcal{A}$ are in particular fulfilled if $\mathcal{A}$ is such that the data term fulfills the (significantly stronger) coercivity type condition

$$\sum\nolimits_{i=1}^{K} \mathrm{d}\left(\mathrm{mean}(\mathcal{A}_{i,\cdot}, u^n),\, f_i\right)^q \to \infty, \qquad \text{as} \qquad \mathrm{diam}\left(u^n\right) \to \infty. \tag{2.34}$$

This coercivity type condition is for instance fulfilled if $\mathcal{A}$ fulfills the manifold analogue of lower boundedness, see [94] for details. Furthermore, the conditions hold if the underlying manifold is compact. As a result we formulate the following theorem.

Theorem 2.13 *Assume that either $\mathcal{M}$ is a compact manifold, or assume that $\mathcal{A}$ fulfills the coercivity type condition* (2.34)*. Then, the inverse problem* (2.32) *for data living in $\mathcal{M}^K$ with* TGV^2_α *regularization using either the Schild or the parallel transport variant of Sect.* 2.3 *has a minimizer. The same statement applies to (pure) second order* TV^2 *regularization.*

The part of Theorem 2.13 concerning compact manifolds $\mathcal{M}$ is the statement of [94, Corollary 1], the part concerning the coercivity type condition is a special case of [94, Theorem 8, Theorem 9].

2.5.2 Algorithmic Realization

We consider the numerical solution of (2.32). For differentiable data terms ($q > 1$), we build on the concept of a generalized forward backward-scheme. In the context of DTI, such a scheme has been proposed in [12]. We discuss an extension by a trajectory method and a Gauß-Seidel type update scheme which significantly improves the performance compared to the basic scheme.

Basic Generalized Forward Backward Scheme

We denote the functional in (2.32) by $\mathcal{F}$ and decompose it into the data term $\mathcal{D}$ and the regularizer R_α which we further decompose into data atoms $(\mathcal{D}_i)_i$ and regularizer atoms $(R_\alpha)_k$, i.e.,

$$\mathcal{F}(u) = \mathcal{D}(u) + R_\alpha(u) = \sum\nolimits_{i=1}^{K} \mathcal{D}_i(u) + \sum\nolimits_{l=1}^{L} (R_\alpha)_l(u) \tag{2.35}$$

Algorithm 4 FB-splitting for solving $\min_u \mathcal{D}(u) + R_\alpha(u)$

```
1: FBS($u^0$, $(\lambda_k)_k$)
2: $k = 0$,
3:   repeat until stopping criterion fulfilled
4:     for $j = 1, \dots, N$
5:         $u_j^{k+0.5} = \exp_{u_j^k}\left(-\lambda_k \sum_{i=1}^K \nabla_{u_j}\mathcal{D}_i\left(u^k\right)\right)$
6:     for $l = 1, \dots, L$
7:         $u^{k+0.5+l/2L} = \operatorname{prox}_{\lambda_k (R_\alpha)_l}(u^{k+0.5+(l-1)/2L})$
8:     $k \leftarrow k + 1$
9: return $u^k$
```

with $\mathcal{D}_i(u) := \frac{1}{q}\mathrm{d}(\mathrm{mean}(\mathcal{A}_{i,\cdot}, u), f_i)^q$, for $i = 1, \dots, K$. Examples for decompositions $R_\alpha(u) = \sum_{l=1}^L (R_\alpha)_l(u)$ of TV and TGV_α^2 regularizers are given in Sects. 2.2 and 2.3, respectively.

The basic idea of a generalized forward-backward scheme is to perform a gradient step for the explicit term, here $\mathcal{D}$, as well as a proximal mapping step for each atom of the implicit term, here $(R_\alpha)_l$. (Concerning the computation of the corresponding proximal mappings for the TV and TGV_α^2 regularizers of Sects. 2.2 and 2.3, we refer to these sections.) We now focus on the data term $\mathcal{D}$. The gradient of $\mathcal{D}$ w.r.t. the variable u_j, $j \in \{1, \dots, N\}$, decomposes as

$$\nabla_{u_j}\mathcal{D}(u) = \sum_{i=1}^K \nabla_{u_j}\mathcal{D}_i(u). \tag{2.36}$$

The gradient of $\mathcal{D}_i$ w.r.t. u_j can then be computed rather explicitly using Jacobi fields. Performing this computation is a central topic of the paper [94]. A corresponding result is [94, Theorem 11]. The overall algorithm is summarized in Algorithm 4. Note that there, for the explicit gradient descend part, we use the kth iterate $u^k = (u_1^k, \dots, u_N^k)$ as base point for computing the gradients w.r.t. all data atoms $\mathcal{D}_i$, $i = 1, \dots, K$ and all items u_j, $j \in \{1, \dots, N\}$. This corresponds to a Jacobi type update scheme. During the iteration, the parameter $\lambda_k > 0$ is decreased fulfilling $\sum_k \lambda_k = \infty$ and $\sum_k \lambda_k^2 < \infty$. Recall that, for the regularizers $R_\alpha = \alpha\mathrm{TV}$ and $R_\alpha = \mathrm{TGV}_\alpha^2$ using either the Schild or the parallel transport variant of Sect. 2.3, the computation of line 6 in Algorithm 4 can be parallelized as explained in Sects. 2.2 and 2.3.

A Generalized Forward Backward Scheme with Gauß-Seidel Update and a Trajectory Method

A well-known issue when considering gradient descent schemes is to find a suitable parameter choice for the $(\lambda_k)_k$. Often a step size control based on line search techniques is employed. Above, there are two particular issues when employing an adaptive step size strategy: First, a single data atom $\mathcal{D}_{i'}$ may require a low step size

Algorithm 5 FB-splitting for solving $\min_u \mathcal{D}(u) + R_\alpha(u)$ using a trajectory method

```
FBSTraj($u^0$, $(\lambda_k)_k$)
$k = 0$,
  repeat until stopping criterion fulfilled
    for $i = 1, \ldots, K$
      $u^{k+i/2K} = \mathrm{traj}_{\lambda_k \mathcal{D}_i}\left(u^{k+(i-1)/2K}\right)$
    for $l = 1, \ldots, L$
      $u^{k+0.5+l/2L} = \mathrm{prox}_{\lambda_k (R_\alpha)_l}(u^{k+0.5+(l-1)/2L})$
    $k \leftarrow k+1$
return $x^k$
```

whereas the other $\mathcal{D}_i$ would allow for much larger steps, but in the standard form one has to use the small step size for all $\mathcal{D}_i$. Second, a small stepsize restriction from a single $\mathcal{D}_{i'}$ also yields a small stepsize in the proximal mapping for the regularization terms. Together, a small step size within an atom of the data term results in a small step size for the whole loop of the iteration Algorithm 4.

In order to overcome these step size issue, the paper [94] proposes to employ a Gauss-Seidel type update scheme together with a trajectory method. To explain the idea, we first replace the update of lines 4/5 of Algorithm 4 by

$$\begin{cases} \text{for } i = 1, \ldots, K \\ \quad \text{for } j = 1, \ldots, N \\ \qquad u_j^{k+i/2K} = \exp_{u_j^{k+(i-1)/2K}}\left(-\lambda_k \nabla_{u_j} \mathcal{D}_i(u^{k+(i-1)/2K})\right). \end{cases} \tag{2.37}$$

Here, the computation of the gradients is performed in a cyclic way w.r.t. the $\mathcal{D}_i$ which corresponds to a Gauß-Seidel type update scheme. This in particular has the following advantage: if we face a small step size for a particular $\mathcal{D}_{i'}$, instead of decreasing the step size for the whole loop, we may employ the following *trajectory method*. Instead of using a single geodesic line for the decay w.r.t. the atom $\mathcal{D}_i$ at iteration k, we follow a polygonal geodesic path. That is, at iteration k, we do not only carry out a single but possibly multiple successive gradient descent steps w.r.t. $\mathcal{D}_i$, where the length of each step is chosen optimal for the current direction for $\mathcal{D}_i$ (by a line search strategy) and the descent steps are iterated until the sum of the step "times" for $\mathcal{D}_i$ reaches λ_k. Details can be found in [94]. This way, a global step size choice with all atoms (potentially negatively) influencing each other, is replaced by an autonomous step size choice for each atom. We denote the resulting operator by $\mathrm{traj}_{\lambda_k \mathcal{D}_i}$ for a data atom $\mathcal{D}_i$. The overall algorithm is subsumed in Algorithm 5.

We point out that also a stochastic variant of this scheme where the atoms are chosen in a random order has been proposed in [94]. Finally, we point out that it is also possible to employ a CPPA or a PPPA as explained in Sect. 2.2. This is in particular important if the data term is not differentiable, i.e., if $q = 1$. For details on computing the proximal mappings of the atoms $\mathcal{D}_i$ we refer to the paper [94].

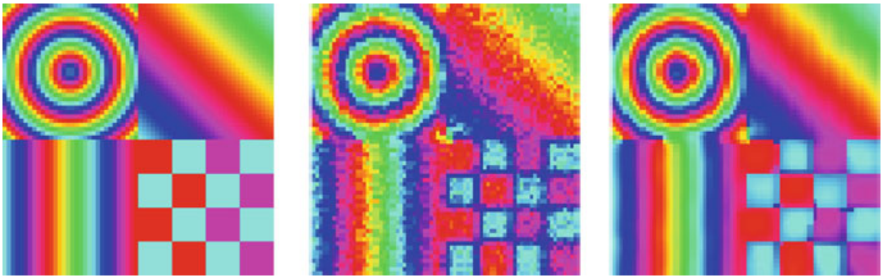


Fig. 2.10 Deconvoling an $\mathbb{S}^1$-valued image (visualized as hue values.) As input data (*center*) we use the ground truth (*left*) convolved with a Gaussian kernel (5×5 kernel with $\sigma = 1$) and corrupted by von Mises noise. We observe the denoising and deblurring capabilities of TGV regularized deconvolution (*right*)

We illustrate the results of joint deconvolution and denoising of manifold-valued data in Fig. 2.10. The data consists of an $\mathbb{S}^1$-valued image convolved with a Gaussian kernel and corrupted by von Mises noise. We employ S-TGV$^2_\alpha$ regularized deconvolution and observe good denoising and deblurring capabilities.

2.6 Wavelet Sparse Regularization of Manifold Valued Data

In contrast to TV, higher order TV type and Mumford-Shah regularizers which are all based on differences (or derivatives in the continuous setting), we here consider a variational scheme employing manifold valued interpolatory wavelets in the regularizing term. In particular, we consider a sparsity promoting ℓ^1 type term as well as an ℓ^0 type term. We obtain results on the existence of minimizers for the proposed models. We provide algorithms for the proposed models and show the potential of the proposed algorithms.

Interpolatory wavelet transforms for linear space data have been investigated by Donoho in [42]. Their analogues for manifold-valued data have been introduced by Ur Rahman, Donoho and their coworkers in [105]. Such transforms have been analyzed and developed further in [62, 64, 113]. Typically, the wavelet-type transforms employ an (interpolatory) subdivision scheme to predict the signal on a finer scale. The ‘difference’ between the prediction and the actual data on the finer scale is realized by vectors living in the tangent spaces of the predicted signal points which point to the actual signal values, i.e., they yield actual signal values after application of a retraction such as the exponential map. These tangent vectors then serve as detail coefficients. Subdivision schemes for manifold-valued data have been considered in [61, 107, 108, 112, 119]. Interpolatory wavelet transforms and subdivision are discussed in more detail in Chapter 4 of this book. All the above approaches consider explicit schemes, i.e., the measured data is processed in a forward way using the analogues of averaging rules and differences in the manifold setting. In contrast, we here consider an implicit approach based on a variational formulation.

2.6.1 Model

We discuss a model for wavelet sparse regularization for manifold-valued data. For the reader's convenience, we consider the univariate situation here. For the multivariate setup and further details we refer to [95]. Let $f \in \mathcal{M}^K$ be data in the complete, finite dimensional Riemannian manifold $\mathcal{M}$. We consider the problem

$$\operatorname{argmin}_{u \in \mathcal{M}^N} \frac{1}{q} \mathrm{d}(\mathcal{A}(u), f)^q + \mathcal{W}_{\alpha}^{\mu,p}(u). \tag{2.38}$$

Here, u denotes the argument to optimize for; it may be thought of as the underlying signal generating the response $\mathcal{A}(u) \in \mathcal{M}^K$, where $\mathcal{A}$ is an operator which models a system's response, for instance. In case of pure denoising, $\mathcal{A}$ is the identity on $\mathcal{M}^N$, $N = K$. Further instances of $\mathcal{A}$ are the manifold valued analogues of convolution operators as pointed out in Sect. 2.5. The deviation of $\mathcal{A}(u)$ from f is quantified by $\frac{1}{q}\mathrm{d}(\mathcal{A}(u), f)^q = \frac{1}{q}\sum_{i=1}^{K} \mathrm{d}(\mathcal{A}(u)_i, f_i)^q$. Further, $\alpha = (\alpha_1, \alpha_2)$ is a parameter vector regulating the trade-off between the data fidelity, and the regularizing term $\mathcal{W}_{\alpha}^{\mu,p}$ which is the central topic of this section and is given by

$$\mathcal{W}_{\alpha}^{\mu,p}(u) = \alpha_1 \cdot \sum_{n,r} 2^{rp\left(\mu + \frac{1}{2} - \frac{1}{p}\right)} \|d_{n,r}(u)\|_{\hat{u}_{n,r}}^{p} + \alpha_2 \cdot \sum_{n} \mathrm{d}(\tilde{u}_{n-1,0}, \tilde{u}_{n,0})^p. \tag{2.39}$$

We discuss the regularizing term (2.39) in more detail in the following. We start with the so-called detail coefficients $d_{n,r}$ which requires some space. The details $d_{n,r}$ at scale r of the interpolatory wavelet transform for manifold valued data are given by

$$d_{n,r} = d_{n,r}(u) = 2^{-r/2} \left(\tilde{u}_{n,r} \ominus \hat{u}_{n,r}\right), \qquad \hat{u}_{n,r} = \mathrm{S}\tilde{u}_{n,r-1}. \tag{2.40}$$

Here $\tilde{u}_{n,r-1} = u_{2^{R-r+1}n}$ and $\tilde{u}_{n,r} = u_{2^{R-r}n}$ (with R the finest level) denote the thinned out target u at scale $r-1$ and r, respectively. The coarsest level is denoted by $\tilde{u}_{n,0} = u_{2^R n}$. The symbol $\ominus$ takes the Riemannian logarithm of the first argument w.r.t. the second argument as base point. $\mathrm{S}\tilde{u}_{n,r-1}$ denotes the application of an interpolatory subdivision scheme S for manifold-valued data to the coarse level data $\tilde{u}_{\cdot,r-1}$ evaluated at the index n which serves as prediction for $\tilde{u}_{n,r}$, i.e.,

$$\mathrm{S}\tilde{u}_{n,r-1} = \operatorname{mean}(s_{n-2\,\cdot}\,, \tilde{u}_{\cdot,r-1}). \tag{2.41}$$

Here the mask s of the subdivision scheme S is a real-valued sequence such that the even as well as the odd entries sum up to 1. The even and the odd entries yield two sets of weights; in case of an interpolatory scheme $s_0 = 1$ and all other even weights equal zero. The simplest example of an interpolatory scheme is the linear interpolatory scheme for which $s_{-1} = s_1 = 1/2$ and the other odd weights equal zero. Thus, in the manifold setup, the prediction of the linear interpolatory scheme consists of the geodesic midpoint between two consecutive coarse level items. The linear interpolatory scheme is a particular example of the interpolatory Deslaurier-Dubuc schemes whose third order variant is given by the coefficients $s_{-3} = s_3 = -1/16$ as well as $s_{-1} = s_1 = 9/16$ with the remaining odd coefficients equal to zero. A reference on linear subdivision schemes is the book [29]; for manifold-valued schemes we refer to references above.

Coming back to (2.40), the detail $d_{n,r}$ quantifies the deviation between the prediction $\mathrm{S}\tilde{u}_{n,r-1}$ and the actual rth level data item $\tilde{u}_{n,r}$ by

$$d_{n,r} = \tilde{u}_{n,r} \ominus \mathrm{S}\tilde{u}_{n,r-1} = \exp^{-1}_{\mathrm{S}\tilde{u}_{n,r-1}} \tilde{u}_{n,r}$$

which denotes the tangent vector sitting in $\hat{u}_{n,r} = \mathrm{S}\tilde{u}_{n,r-1}$ pointing to $\tilde{u}_{n,r}$.

With this information on the details $d_{n,r}$, we come back to the definition of the regularizer in (2.39). We observe that the symbol $\| \cdot \|_{\hat{u}_{n,r}}$ denotes the norm induced by the Riemannian scalar product in the point $\hat{u}_{n,r}$, which is the point where the detail $d_{n,r}(u)$ is a tangent vector at; it measures the size of the detail. The parameter μ is a smoothness parameter and the parameter $p \geq 1$ stems from a norm type term. The second term measures the pth power of the distance between neighboring items on the coarsest scale.

We emphasize that the case $p = 1$, $\mu = 1$ in (2.38), corresponds to the manifold analogue of the LASSO [35, 103] or ℓ^1-sparse regularization which, in the linear case, is addressed by (iterative) soft thresholding [43]. This case is particularly interesting since it promotes solutions u which are likely to be sparse w.r.t. the considered wavelet expansion.

The manifold analogue of ℓ^0-sparse regularization which actually measures sparsity is obtained by using the regularizer

$$\mathcal{W}^0_\alpha(u) = \alpha_1 \, \#\{(n, r) \ : \ d_{n,r}(u) \neq 0\} \ + \ \alpha_2 \, \#\{n \ : \ \tilde{u}_{n-1,0} \neq \tilde{u}_{n,0}\}. \tag{2.42}$$

The operator # is used to count the number of elements in the corresponding set. Note that this way the number of non-zero detail coefficients of the wavelet expansion is penalized. Similar to the linear case [35, 43, 111], potential applications of the considered sparse regularization techniques are denoising and compression.

Concerning the existence of minimizers, we have the following results.

Theorem 2.14 *The variational problem* (2.38) *of wavelet regularization using the regularizers* $\mathcal{W}^{\mu,p}_\alpha$ *of* (2.39) *with* $\alpha_2 \neq 0$ *has a minimizer.*

Similar to the existence results in Sect. 2.5 these results are based on showing lower semicontinuity and a coercivity type condition in the manifold setting. To ensure a coercivity type condition when $\alpha_2 = 0$ we need to impose additional conditions on $\mathcal{A}$. For a precise discussion of this point we refer to [95]. As in Sect. 2.5 we here state a special case which is easier to access.

Theorem 2.15 *Let $\mathcal{M}$ be a compact manifold, or assume that $\mathcal{A}$ fulfills the coercivity type condition* (2.34)*. The variational problem* (2.38) *of wavelet regularization using the regularizers $\mathcal{W}_\alpha^{\mu,p}$ of* (2.39) *with $\alpha_2 = 0$ has a minimizer.*

Theorem 2.16 *We make the same assumptions as in Theorem 2.15. Then wavelet sparse regularization using the ℓ^0 type regularizing terms $\mathcal{W}_\alpha^0(u)$ of* (2.42) *has a minimizer.*

For proofs of these theorems (whereby Theorem 2.15 is a special case of [95, Theorem 4]) we refer to [95].

2.6.2 Algorithmic Realization

We decompose the regularizer $W_\alpha^{\mu,p}$ into atoms $\mathcal{R}_k$ with a enumerating index k by

$$\mathcal{R}_k = \alpha_1 \sum_{n,r} 2^{rp\left(\mu+\frac{1}{2}-\frac{1}{p}\right)} \|d_{n,r}(u)\|^p_{\hat{u}_{n,r}}, \qquad \text{or} \qquad \mathcal{R}_k = \alpha_2 \mathrm{d}(\tilde{u}_{n-1,0}, \tilde{u}_{n,0})^p, \tag{2.43}$$

and the data term into atoms $\mathcal{D}_k$ according to (2.35). To these atoms we may apply the concepts of a generalized forward backward-scheme with Gauss-Seidel type update and a trajectory method (explained in Sect. 2.5) as well as the concept of a CPPA or a PPPA (explained in Sect. 2.2). To implement these schemes expressions for the (sub)gradients and proximal mappings of the atoms $\mathcal{R}_k$ based on Jacobi fields have been derived in [95]. Due to space reasons, we do not elaborate on this derivation here, but refer to the mentioned paper for details. Similar to (2.43), we may decompose the ℓ^0-sparse regularizer $\mathcal{W}_\alpha^0$ into atoms we also denote by $\mathcal{D}_k$, and apply a CPPA or PPPA. For details we refer to [95]. We illustrate ℓ^1 wavelet regularization by considering a joint deblurring and denoising problem for an $\mathbb{S}^2$-valued time series in Fig. 2.11. The noisy data is convolved with the manifold-valued

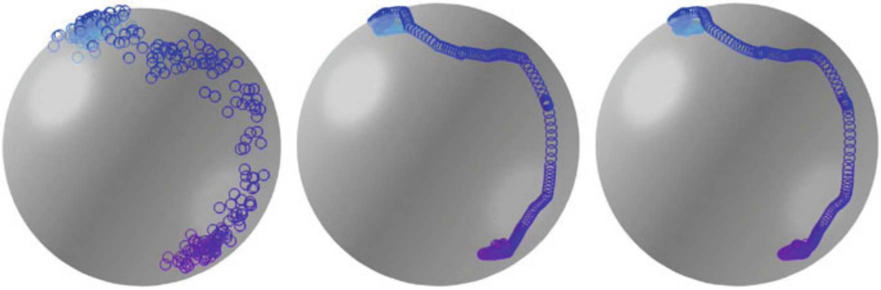


Fig. 2.11 Illustration of the proposed ℓ^1 wavelet regularization for a $\mathbb{S}^2$-valued time series. The given data (*left*) is noisy and blurred with the manifold analogue of a Gaussian kernel with $\sigma = 2$. We display the result of using the first order interpolatory wavelet (*middle*) and the third order Deslaurier-Dubuc (DD) wavelet (*right*)

analogue of a discrete Gaussian kernel. As prediction operator we employ the linear interpolatory subdivision scheme which inserts the geodesic midpoint as well as the cubic Deslaurier Dubuc scheme for manifold valued data as explained above.

Acknowledgements MH acknowledges support by the Austrian Science Fund (FWF) (Grant J 4112). AW acknowledges support by the DFG Grants WE 5886/3-1 and WE 5886/4-1.

References

1. Afsari, B., Tron, R., Vidal, R.: On the convergence of gradient descent for finding the Riemannian center of mass. SIAM J. Control Optim. **51**(3), 2230–2260 (2013)
2. Alexeev, B., Ward, R.: On the complexity of Mumford–Shah-type regularization, viewed as a relaxed sparsity constraint. IEEE Trans. Image Process. **19**(10), 2787–2789 (2010)
3. Alliney, S.: Digital filters as absolute norm regularizers. IEEE Trans. Signal Process. **40**, 1548–1562 (1992)
4. Ambrosio, L., Tortorelli, V.M.: Approximation of functional depending on jumps by elliptic functional via Γ-convergence. Commun. Pure Appl. Math. **43**(8), 999–1036 (1990)
5. Arnaudon, M., Nielsen, F.: On approximating the Riemannian 1-center. Comput. Geom. **46**(1), 93–104 (2013)
6. Azagra, D., Ferrera, J.: Proximal calculus on Riemannian manifolds. Mediterr. J. Math. **2**, 437–450 (2005)
7. Bačák, M.: Computing medians and means in Hadamard spaces. SIAM J. Optim. **24**(3), 1542–1566 (2014)
8. Bačák, M.: Convex analysis and optimization in Hadamard spaces. De Gruyter, Berlin (2014)
9. Bačák, M., Bergmann, R., Steidl, G., Weinmann, A.: A second order non-smooth variational model for restoring manifold-valued images. SIAM J. Sci. Comput. **38**(1), A567–A597 (2016)
10. Basser, P., Mattiello, J., LeBihan, D.: MR diffusion tensor spectroscopy and imaging. Biophys. J. **66**(1), 259–267 (1994)
11. Baust, M., Demaret, L., Storath, M., Navab, N., Weinmann, A.: Total variation regularization of shape signals. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2075–2083 (2015)
12. Baust, M., Weinmann, A., Wieczorek, M., Lasser, T., Storath, M., Navab, N.: Combined tensor fitting and TV regularization in diffusion tensor imaging based on a Riemannian manifold approach. IEEE Trans. Med. Imaging **35**(8), 1972–1989 (2016)
13. Bergmann, R., Chan, R.H., Hielscher, R., Persch, J., Steidl, G.: Restoration of manifold-valued images by half-quadratic minimization. Inverse Prob. Imaging **10**, 281–304 (2016)
14. Bergmann, R., Fitschen, J.H., Persch, J., Steidl, G.: Infimal convolution type coupling of first and second order differences on manifold-valued images. In: Scale Space and Variational Methods in Computer Vision 2017, pp. 447–459 (2017)
15. Bergmann, R., Fitschen, J.H., Persch, J., Steidl, G.: Priors with coupled first and second order differences for manifold-valued image processing. J. Math. Imaging Vision **60**(9), 1459–1481 (2018)
16. Berkels, B., Fletcher, P., Heeren, B., Rumpf, M., Wirth, B.: Discrete geodesic regression in shape space. In: International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition, pp. 108–122. Springer, New York (2013)
17. Bertero, M., Boccacci, P.: Introduction to Inverse Problems in Imaging. CRC Press, Boca Raton (1998)
18. Bertsekas, D.: Multiplier methods: a survey. Automatica **12**(2), 133–145 (1976)

19. Bertsekas, D.: Incremental proximal methods for large scale convex optimization. Math. Program. **129**, 163–195 (2011)
20. Blake, A., Zisserman, A.: Visual Reconstruction. MIT Press, Cambridge (1987)
21. Boyer, C., Chambolle, A., Castro, Y.D., Duval, V., De Gournay, F., Weiss, P.: On representer theorems and convex regularization. SIAM J. Optim. **29**(2), 1260–1281 (2019)
22. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. **23**(11), 1222–1239 (2001)
23. Boysen, L., Kempe, A., Liebscher, V., Munk, A., Wittich, O.: Consistencies and rates of convergence of jump-penalized least squares estimators. Ann. Stat. **37**(1), 157–183 (2009)
24. Bredies, K., Carioni, M.: Sparsity of solutions for variational inverse problems with finite-dimensional data (2018). arXiv preprint arXiv:1809.05045
25. Bredies, K., Holler, M.: Regularization of linear inverse problems with total generalized variation. J. Inverse Ill-Posed Probl. **22**(6), 871–913 (2014)
26. Bredies, K., Holler, M., Storath, M., Weinmann, A.: An observation concerning the parallel transport variant of total generalized variation for manifold-valued data. Oberwolfach Rep. **20**, 38–41 (2018)
27. Bredies, K., Kunisch, K., Pock, T.: Total generalized variation. SIAM J. Imaging Sci. **3**(3), 492–526 (2010)
28. Bredies, K., Holler, M., Storath, M., Weinmann, A.: Total generalized variation for manifold-valued data. SIAM J. Imaging Sci. **11**(3), 1785–1848 (2018)
29. Cavaretta, A.S., Dahmen, W., Micchelli, C.A.: Stationary Subdivision, vol. 453. American Mathematical Society, Providence (1991)
30. Chambolle, A.: Image segmentation by variational methods: Mumford and Shah functional and the discrete approximations. J. SIAM Appl. Math. **55**(3), 827–863 (1995)
31. Chambolle, A.: Finite-differences discretizations of the Mumford-Shah functional. ESAIM Math. Model. Numer. Anal. **33**(02), 261–288 (1999)
32. Chambolle, A.: An algorithm for total variation minimization and applications. J. Math. Imaging Vision **20**, 89–97 (2004)
33. Chambolle, A., Lions, P.L.: Image recovery via total variation minimization and related problems. Numer. Math. **76**(2), 167–188 (1997)
34. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vision **40**, 120–145 (2011)
35. Chambolle, A., De Vore, R., Lee, N., Lucier, B.: Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage. IEEE Trans. Image Process. **7**(3), 319–335 (1998)
36. Chan, T., Esedoglu, S.: Aspects of total variation regularized L^1 function approximation. J. SIAM Appl. Math. **65**, 1817–1837 (2005)
37. Chan, T., Kang, S., Shen, J.: Total variation denoising and enhancement of color images based on the CB and HSV color models. J. Vis. Commun. Image Represent. **12**, 422–435 (2001)
38. Cheng, G., Salehian, H., Vemuri, B.: Efficient recursive algorithms for computing the mean diffusion tensor and applications to DTI segmentation. In: Computer Vision–ECCV 2012, pp. 390–401. Springer, Berlin (2012)
39. Cook, P., Bai, Y., Nedjati-Gilani, S., Seunarine, K., Hall, M., Parker, G., Alexander, D.: Camino: open-source diffusion-MRI reconstruction and processing. In: 14th Scientific Meeting of the International Society for Magnetic Resonance in Medicine, p. 2759 (2006)
40. Cremers, D., Strekalovskiy, E.: Total cyclic variation and generalizations. J. Math. Imaging Vision **47**(3), 258–277 (2013)
41. Demengel, F.: Fonctionsa hessien borné. Ann. Inst. Fourier **34**, 155–190 (1984)
42. Donoho, D.: Interpolating wavelet transforms. Department of Statistics, Stanford University 2(3), 1–54 (1992). Preprint
43. Donoho, D.: De-noising by soft-thresholding. IEEE Trans. Inf. Theory **41**(3), 613–627 (1995)
44. Drummond, T., Cipolla, R.: Real-time visual tracking of complex structures. IEEE Trans. Pattern Anal. Mach. Intell. **24**, 932–946 (2002)

45. Duran, J., Möller, M., Sbert, C., Cremers, D.: Collaborative total variation: a general framework for vectorial tv models. SIAM J. Imaging Sci. **9**(1), 116–151 (2016)
46. Engl, H.W., Hanke, M., Neubauer, A.: Regularization of Inverse Problems, vol. 375. Springer Science & Business Media, New York (1996)
47. Ferreira, O., Oliveira, P.: Subgradient algorithm on Riemannian manifolds. J. Optim. Theory Appl. **97**(1), 93–104 (1998)
48. Ferreira, O., Oliveira, P.: Proximal point algorithm on Riemannian manifolds. Optimization **51**, 257–270 (2002)
49. Ferreira, R., Xavier, J., Costeira, J., Barroso, V.: Newton algorithms for Riemannian distance related problems on connected locally symmetric manifolds. IEEE J. Sel. Top. Signal Process. **7**, 634–645 (2013)
50. Fletcher, P., Joshi, S.: Riemannian geometry for the statistical analysis of diffusion tensor data. Signal Process. **87**, 250–262 (2007)
51. Fornasier, M., Ward, R.: Iterative thresholding meets free-discontinuity problems. Found. Comput. Math. **10**(5), 527–567 (2010)
52. Fornasier, M., March, R., Solombrino, F.: Existence of minimizers of the Mumford-Shah functional with singular operators and unbounded data. Ann. Mat. Pura Appl. **192**(3), 361–391 (2013)
53. Friedrich, F., Kempe, A., Liebscher, V., Winkler, G.: Complexity penalized M-estimation. J. Comput. Graph. Stat. **17**(1), 201–224 (2008)
54. Geman, S., Geman, D.: Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE Trans. Pattern Anal. Mach. Intell. **6**(6), 721–741 (1984)
55. Getreuer, P.: Rudin-Osher-Fatemi total variation denoising using split Bregman. Image Process. Line **2**, 74–95 (2012)
56. Giaquinta, M., Mucci, D.: The BV-energy of maps into a manifold: relaxation and density results. Ann. Sc. Norm. Super. Pisa Cl. Sci. **5**(4), 483–548 (2006)
57. Giaquinta, M., Modica, G., Souček, J.: Variational problems for maps of bounded variation with values in S^1. Calc. Var. Partial Differ. Equ. **1**(1), 87–121 (1993)
58. Goldstein, T., Osher, S.: The split Bregman method for L^1-regularized problems. SIAM J. Imaging Sci. **2**, 323–343 (2009)
59. Gousseau, Y., Morel, J.M.: Are natural images of bounded variation? SIAM J. Math. Anal. **33**, 634–648 (2001)
60. Green, P., Mardia, K.: Bayesian alignment using hierarchical models, with applications in protein bioinformatics. Biometrika **93**, 235–254 (2006)
61. Grohs, P.: Smoothness analysis of subdivision schemes on regular grids by proximity. SIAM J. Numer. Anal. **46**(4), 2169–2182 (2008)
62. Grohs, P.: Stability of manifold-valued subdivision schemes and multiscale transformations. Constr. Approx. **32**(3), 569–596 (2010)
63. Grohs, P., Sprecher, M.: Total variation regularization on Riemannian manifolds by iteratively reweighted minimization. Inf. Inference **5**(4), 353–378 (2016)
64. Grohs, P., Wallner, J.: Interpolatory wavelets for manifold-valued data. Appl. Comput. Harmon. Anal. **27**(3), 325–333 (2009)
65. Hinterberger, W., Scherzer, O.: Variational methods on the space of functions of bounded Hessian for convexification and denoising. Computing **76**(1–2), 109–133 (2006)
66. Hohm, K., Storath, M., Weinmann, A.: An algorithmic framework for Mumford-Shah regularization of inverse problems in imaging. Inverse Prob. **31**, 115011 (2015)
67. Itoh, J., Tanaka, M.: The dimension of a cut locus on a smooth Riemannian manifold. Tohoku Math. J. (2) **50**(4), 571–575 (1998)
68. Jiang, M., Maass, P., Page, T.: Regularizing properties of the Mumford-Shah functional for imaging applications. Inverse Prob. **30**(3), 035007 (2014)
69. Johansen-Berg, H., Behrens, T.: Diffusion MRI: From Quantitative Measurement to In-vivo Neuroanatomy. Academic, London (2009)
70. Karcher, H.: Riemannian center of mass and mollifier smoothing. Commun. Pure Appl. Math. **30**, 509–541 (1977)

71. Kendall, W.: Probability, convexity, and harmonic maps with small image I: uniqueness and fine existence. Proc. Lond. Math. Soc. **3**, 371–406 (1990)
72. Kheyfets, A., Miller, W.A., Newton, G.A.: Schild's ladder parallel transport procedure for an arbitrary connection. Int. J. Theor. Phys. **39**(12), 2891–2898 (2000)
73. Killick, R., Fearnhead, P., Eckley, I.: Optimal detection of changepoints with a linear computational cost. J. Am. Stat. Assoc. **107**(500), 1590–1598 (2012)
74. Lellmann, J., Strekalovskiy, E., Koetter, S., Cremers, D.: Total variation regularization for functions with values in a manifold. In: International Conference on Computer Vision (ICCV), pp. 2944–2951 (2013)
75. Lorenzi, M., Pennec, X.: Efficient parallel transport of deformations in time series of images: from Schild's to pole ladder. J. Math. Imaging Vision **50**(1–2), 5–17 (2014)
76. Massonnet, D., Feigl, K.: Radar interferometry and its application to changes in the earth's surface. Rev. Geophys. **36**(4), 441–500 (1998)
77. Michor, P., Mumford, D.: An overview of the Riemannian metrics on spaces of curves using the Hamiltonian approach. Appl. Comput. Harmon. Anal. **23**(1), 74–113 (2007)
78. Moreau, J.J.: Fonctions convexes duales et points proximaux dans un espace hilbertien. C. R. Acad. Sci. A Math. **255**, 2897–2899 (1962)
79. Mumford, D., Shah, J.: Boundary detection by minimizing functionals. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 17, pp. 137–154 (1985)
80. Mumford, D., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. Commun. Pure Appl. Math. **42**(5), 577–685 (1989)
81. Nikolova, M.: Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers. SIAM J. Numer. Anal. **40**, 965–994 (2002)
82. Nikolova, M.: A variational approach to remove outliers and impulse noise. J. Math. Imaging Vision **20**, 99–120 (2004)
83. Pennec, X., Fillard, P., Ayache, N.: A Riemannian framework for tensor computing. Int. J. Comput. Vis. **66**, 41–66 (2006)
84. Pock, T., Cremers, D., Bischof, H., Chambolle, A.: An algorithm for minimizing the Mumford-Shah functional. In: IEEE International Conference on Computer Vision and Pattern Recognition, pp. 1133–1140 (2009)
85. Potts, R.: Some generalized order-disorder transformations. Math. Proc. Camb. Philos. Soc. **48**(01), 106–109 (1952)
86. Rezakhaniha, R., Agianniotis, A., Schrauwen, J., Griffa, A., Sage, D., Bouten, C., Van de Vosse, F., Unser, M., Stergiopulos, N.: Experimental investigation of collagen waviness and orientation in the arterial adventitia using confocal laser scanning microscopy. Biomech. Model. Mechanobiol. **11**, 461–473 (2012)
87. Rocca, F., Prati, C., Ferretti, A.: An overview of SAR interferometry. In: Proceedings of the 3rd ERS Symposium on Space at the Service of our Environment, Florence (1997)
88. Rodriguez, P., Wohlberg, B.: An iteratively reweighted norm algorithm for total variation regularization. In: IEEE Conference on Signals, Systems and Computers, pp. 892–896 (2006)
89. Rosman, G., Bronstein, M., Bronstein, A., Wolf, A., Kimmel, R.: Group-valued regularization framework for motion segmentation of dynamic non-rigid shapes. In: Scale Space and Variational Methods in Computer Vision, pp. 725–736. Springer, Berlin (2012)
90. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D Nonlinear Phenom. **60**(1), 259–268 (1992)
91. Sapiro, G., Ringach, D.L.: Anisotropic diffusion of multivalued images with applications to color filtering. IEEE Trans. Image Process. **5**(11), 1582–1586 (1996)
92. Stefanoiu, A., Weinmann, A., Storath, M., Navab, N., Baust, M.: Joint segmentation and shape regularization with a generalized forward–backward algorithm. IEEE Trans. Image Process. **25**(7), 3384–3394 (2016)
93. Storath, M., Weinmann, A.: Fast partitioning of vector-valued images. SIAM J. Imaging Sci. **7**(3), 1826–1852 (2014)
94. Storath, M., Weinmann, A.: Variational regularization of inverse problems for manifold-valued data (2018). arXiv preprint arXiv:1804.10432

95. Storath, M., Weinmann, A.: Wavelet sparse regularization for manifold-valued data (2018). arXiv preprint arXiv:1808.00505
96. Storath, M., Weinmann, A., Demaret, L.: Jump-sparse and sparse recovery using Potts functionals. IEEE Trans. Signal Process. **62**(14), 3654–3666 (2014)
97. Storath, M., Weinmann, A., Frikel, J., Unser, M.: Joint image reconstruction and segmentation using the Potts model. Inverse Prob. **31**(2), 025003 (2014)
98. Storath, M., Weinmann, A., Unser, M.: Exact algorithms for L^1-TV regularization of real-valued or circle-valued signals. SIAM J. Sci. Comput. **38**(1), A614–A630 (2016)
99. Storath, M., Weinmann, A., Unser, M.: Jump-penalized least absolute values estimation of scalar or circle-valued signals. Inf. Inference **6**(3), 225–245 (2017)
100. Strekalovskiy, E., Cremers, D.: Total variation for cyclic structures: convex relaxation and efficient minimization. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1905–1911 (2011)
101. Strong, D., Chan, T.: Edge-preserving and scale-dependent properties of total variation regularization. Inverse Prob. **19**, S165 (2003)
102. Thiel, K., Wu, X., Hartl, P.: ERS-tandem-interferometric observation of volcanic activities in Iceland. In: ESA SP, pp. 475–480 (1997). https://earth.esa.int/workshops/ers97/papers/thiel/index-2.html
103. Tibshirani, R.: Regression shrinkage and selection via the lasso. J. R. Stat. Soc. B Methodol. **58**, 267–288 (1996)
104. Tsai, A., Yezzi Jr., A., Willsky, A.: Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification. IEEE Trans. Image Process. **10**(8), 1169–1186 (2001)
105. Ur Rahman, I., Drori, I., Stodden, V., Donoho, D., Schröder, P.: Multiscale representations for manifold-valued data. Multiscale Model. Simul. **4**(4), 1201–1232 (2005)
106. Veksler, O.: Efficient graph-based energy minimization methods in computer vision. Ph.D. thesis, Cornell University (1999)
107. Wallner, J., Dyn, N.: Convergence and C1 analysis of subdivision schemes on manifolds by proximity. Comput. Aided Geom. Des. **22**(7), 593–622 (2005)
108. Wallner, J., Yazdani, E., Weinmann, A.: Convergence and smoothness analysis of subdivision rules in Riemannian and symmetric spaces. Adv. Comput. Math. **34**(2), 201–218 (2011)
109. Wang, Z., Vemuri, B.: An affine invariant tensor dissimilarity measure and its applications to tensor-valued image segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. I228–I233 (2004)
110. Wang, Z., Vemuri, B.: DTI segmentation using an information theoretic tensor dissimilarity measure. IEEE Trans. Med. Imaging **24**(10), 1267–1277 (2005)
111. Weaver, J., Xu, Y., Healy, D., Cromwell, L.: Filtering noise from images with wavelet transforms. Magn. Reson. Med. **21**(2), 288–295 (1991)
112. Weinmann, A.: Nonlinear subdivision schemes on irregular meshes. Constr. Approx. **31**(3), 395–415 (2010)
113. Weinmann, A.: Interpolatory multiscale representation for functions between manifolds. SIAM J. Math. Anal. **44**, 162–191 (2012)
114. Weinmann, A., Demaret, L., Storath, M.: Total variation regularization for manifold-valued data. SIAM J. Imaging Sci. **7**(4), 2226–2257 (2014)
115. Weinmann, A., Storath, M., Demaret, L.: The L^1-Potts functional for robust jump-sparse reconstruction. SIAM J. Numer. Anal. **53**(1), 644–673 (2015)
116. Weinmann, A., Demaret, L., Storath, M.: Mumford–Shah and Potts regularization for manifold-valued data. J. Math. Imaging Vision **55**(3), 428–445 (2016)
117. Winkler, G., Liebscher, V.: Smoothers for discontinuous signals. J. Nonparametr. Stat. **14**(1–2), 203–222 (2002)
118. Wittich, O., Kempe, A., Winkler, G., Liebscher, V.: Complexity penalized least squares estimators: analytical results. Math. Nachr. **281**(4), 582–595 (2008)
119. Xie, G., Yu, T.: Smoothness equivalence properties of general manifold-valued data subdivision schemes. Multiscale Model. Simul. **7**(3), 1073–1100 (2008)

Chapter 3
Lifting Methods for Manifold-Valued Variational Problems

Thomas Vogt, Evgeny Strekalovskiy, Daniel Cremers, and Jan Lellmann

Contents

Abstract Lifting methods allow to transform hard variational problems such as segmentation and optical flow estimation into convex problems in a suitable higher-dimensional space. The lifted models can then be efficiently solved to a global optimum, which allows to find approximate global minimizers of the original

T. Vogt (✉) · J. Lellmann
Institute of Mathematics and Image Computing, University of Lübeck, Lübeck, Germany
e-mail: vogt@mic.uni-luebeck.de; lellmann@mic.uni-luebeck.de

E. Strekalovskiy
Technical University Munich, Garching, Germany

Google Germany GmbH, Munich, Germany
e-mail: evgeny.strekalovskiy@gmail.com

D. Cremers
Technical University Munich, Garching, Germany
e-mail: cremers@tum.de

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_3

problem. Recently, these techniques have also been applied to problems with values in a manifold. We provide a review of such methods in a refined framework based on a finite element discretization of the range, which extends the concept of sublabel-accurate lifting to manifolds. We also generalize existing methods for total variation regularization to support general convex regularization.

3.1 Introduction

Consider a variational image processing or general data analysis problem of the form

$$\min_{u:\Omega\to\mathcal{M}} F(u) \tag{3.1}$$

with $\Omega \subset \mathbb{R}^d$ open and bounded. In this chapter, we will be concerned with problems where the image u takes values in an s-dimensional *manifold* $\mathcal{M}$. Problems of this form are wide-spread in image processing and especially in the processing of manifold-valued images such as InSAR [49], EBSD [6], DTI [8], orientational/positional [58] data or images with values in non-flat color spaces such as hue-saturation-value (HSV) or chromaticity-brightness (CB) color spaces [20].

They come with an inherent non-convexity, as the space of images $u\colon \Omega \to \mathcal{M}$ is generally non-convex, with few exceptions, such as if $\mathcal{M}$ is a Euclidean space, or if $\mathcal{M}$ is a Hadamard manifold, if one allows for the more general notion of geodesic convexity [4, 5]. Except for these special cases, efficient and robust convex numerical optimization algorithms therefore cannot be applied and global optimization is generally out of reach.

The inherent non-convexity of the feasible set is not only an issue of representation. Even for seemingly simple problems, such as the problem of computing the Riemannian center of mass for a number of points on the unit circle, it can affect the energy in surprisingly intricate ways, creating multiple local minimizers and non-uniqueness (Fig. 3.1). The equivalent operation in Euclidean space, computing the weighted mean, is a simple convex (even linear) operation, with a unique, explicit solution.

The problem of non-convexity is not unique to our setting, but rather ubiquitous in a much broader context of image and signal processing: amongst others, image segmentation, 3D reconstruction, image matching, optical flow and image registration, superresolution, inpainting, edge-preserving image restoration with the Mumford-Shah and Potts model, machine learning, and many statistically or physically motivated models involve intrinsically non-convex feasible sets or energies. When applied to such non-convex problems, local optimization strategies often get stuck in local minimizers.

In *convex relaxation* approaches, an energy functional is approximated by a convex one whose global optimum can be found numerically and whose minimizers lie within a small neighborhood around the actual solution of the problem. A popular convex relaxation technique that applies to a wide range of problems from image

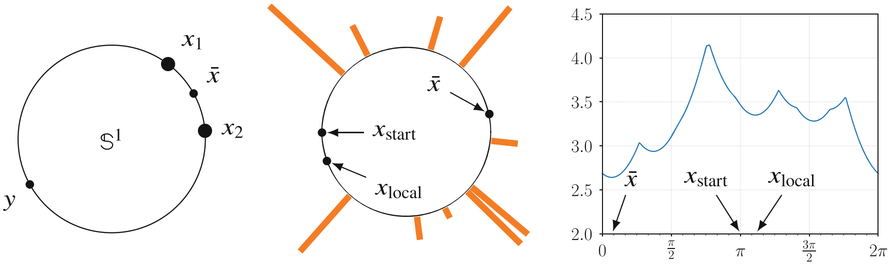


Fig. 3.1 Variational problems where the feasible set is a non-Euclidean manifold are prone to local minima and non-uniqueness, which makes them generally much harder than their counterparts in $\mathbb{R}^n$. The example shows the generalization of the (weighted) mean to manifolds: the Riemannian center of mass $\bar{x}$ of points x_i on a manifold—in this case, the unit circle $\mathbb{S}^1$—is defined as the minimizer (if it exists and is unique) of the problem $\inf_{x \in \mathbb{S}^1} \sum_i \lambda_i d(x_i, x)^2$, where d is the geodesic (angular) distance and $\lambda_i > 0$ are given weights. **Left:** Given the two points x_1 and x_2, the energy for computing their "average" has a local minimum at y in addition to the global minimum at $\bar{x}$. Compare this to the corresponding problem in $\mathbb{R}^n$, which has a strictly convex energy with the unique and explicit solution $(x_1+x_2)/2$. **Center and right:** When the number of points is increased and non-uniform weights are used (represented by the locations and heights of the orange bars), the energy structure becomes even less predictable. The objective function (right, parametrized by angle) exhibits a number of non-trivial local minimizers that are not easily explained by global symmetries. Again, the corresponding problem—computing a weighted mean—is trivial in $\mathbb{R}^n$. Starting from $x_{\text{start}} = \pi$, our functional lifting implementation finds the global minimizer $\bar{x}$, while gradient descent (a local method) gets stuck in the local minimizer x_{local}. Empirically, this behaviour can be observed for any other choice of points and weights, but there is no theoretical result in this direction

and signal processing is *functional lifting*. With this technique, the feasible set is embedded into a higher-dimensional space where efficient convex approximations of the energy functional are easier available.

Overview and Contribution In the following sections, we will give a brief introduction to the concept of functional lifting and explore its generalization to manifold-valued problems. Our aim is to provide a survey-style introduction to the area, therefore we will provide references and numerical experiments on the way. In contrast to prior work, we will explain existing results in an updated finite element-based framework. Moreover, we propose extensions to handle general regularizers other than the total variation on manifolds, and to apply the "sublabel-accurate" methods to manifold-valued problems.

3.1.1 Functional Lifting in Euclidean Spaces

The problem of finding a function $u \colon \Omega \to \Gamma$ that assigns a *label* $u(x) \in \Gamma$ from a *discrete* range Γ to each point x in a continuous domain $\Omega \subset \mathbb{R}^d$, while minimizing an energy function $F(u)$, is commonly called a *continuous multi-label (or multi-*

class labeling) problem in the image processing community [44, 53]. The name comes from the interpretation of this setting as the continuous counterpart to the fully discrete problem of assigning to each vertex of a graph one of finitely many labels $\gamma_1, \dots, \gamma_L$ while minimizing a given cost function [17, 33, 34, 36].

The prototypical application of multi-labeling techniques is multi-class image segmentation, where the task is to partition a given image into finitely many regions. In this case, the label set Γ is discrete and each label represents one of the regions so that $u^{-1}(\gamma) \subset \Omega$ is the region that is assigned label γ.

In the fully discrete setting, one way of tackling first-order multi-label problems is to look for good linear programming relaxations [17, 34, 36]. These approaches were subsequently translated to continuous domains Ω for the two-class [21], multi-class [7, 43, 53, 71], and vectorial [30] case, resulting in non-linear, but convex, relaxations. By honoring the continuous nature of Ω, they reduce metrication errors and improve isotropy [31, 61, 63, 64], see [46] for a discussion and more references.

The general strategy, which we will also follow for the manifold-valued case, is to replace the energy minimization problem

$$\min_{u\colon \Omega \to \Gamma} F(u), \tag{3.2}$$

by a problem

$$\min_{v\colon \Omega \to X} \tilde{F}(v), \tag{3.3}$$

where X is some "nice" convex set of larger dimension than Γ with the property that there is an embedding $i\colon \Gamma \hookrightarrow X$ and $F(u) \approx \tilde{F}(i \circ u)$ in some sense whenever $u\colon \Omega \to \Gamma$.

In general, the lifted functional $\tilde{F}$ is chosen in such a way that it exhibits favorable (numerical or qualitative) properties compared with the original functional F while being sufficiently close to the original functional so that minimizers of $\tilde{F}$ can be expected to have some recoverable relationship with global minimizers of F. Usually, $\tilde{F}$ is chosen to be convex when F is not, which will make the problem amenable for convex optimization algorithms and allows to find a global minimizer of the lifted problem.

While current lifting strategies generally avoid local minimizers of the original problem, they are still an approximation and they are generally not guaranteed to find the global minimizers of the original problem.

A central difficulty is that some simplifications have to be performed in the lifting process in order to make it computationally feasible, which may lose information about the original problem. As a result, global minimizers $v\colon \Omega \to X$ of the lifted problem need not be in the image of Γ under the embedding $i\colon \Gamma \hookrightarrow X$ and therefore are not directly associated with a function in the original space.

The process of projecting a solution back to the original space of functions $u\colon \Omega \to \Gamma$ is a difficult problem and, unless Γ is scalar [55], the projection cannot be expected to be a minimizer of the original functional (see the considerations in

[26, 40, 66]). These difficulties may be related to the fact that the original problems are NP-hard [23]. As in the discrete labeling setting [36], so-called rounding strategies have been proposed in the continuous case [42, 45] that come with an a priori bound for the relative gap between the minimum of the original functional and the value attained at the projected version of a minimizer to the lifted functional. For the manifold-valued case considered here, we are not aware of a similar result yet.

In addition to the case of a *discrete* range Γ, relaxation methods have been derived for dealing with a *continuous* (non-discrete) range, most notably the scalar case $\Gamma \subseteq \mathbb{R}$ [3, 55]. They typically consider *first-order* energies that depend pointwise on u and ∇u only:

$$F(u) = \int_{\Omega} f(x, u(x), \nabla u(x))\, dx. \tag{3.4}$$

The equivalent problem class in the fully discrete setting consists of the energies with only unary (depending on one vertex's label) and pairwise (depending on two vertices' labels) terms.

For the problem (3.4), applying a strategy as in (3.2)–(3.3) comes with a substantial increase in dimensions. These relaxation approaches therefore have been called *functional lifting*, starting from the paper [54] where the (non-convex) Mumford-Shah functional for edge-preserving image regularization and segmentation is lifted to a space of functions $v\colon \Omega \times \Gamma \to [0, 1]$, $\Gamma \subset \mathbb{R}$. The authors use the special "step function" lifting $X = \{v\colon \Gamma \to [0, 1]\}$ and $i(z^*) = v$ with $v(z) = 1$ if $z \leq z^*$ and 0 otherwise, which is only available in the scalar case

In this case, the integrand $f\colon \Omega \times \Gamma \times \mathbb{R}^{s,d} \to \mathbb{R}$ in (3.4) is assumed to be convex in the third component and nonnegative. The less restrictive property of polyconvexity has been shown to be sufficient [51, 69], so that also minimal surface problems fit into this framework. The continuous formulations can be demonstrated [51, 54] to have strong connections with the method of calibrations [3] and with the theory of currents [29].

In this paper, we will consider the more general case of $\Gamma = \mathcal{M}$ having a manifold structure. We will also restrict ourselves to first-order models. Only very recently, attempts at generalizing the continuous lifting strategies to models with higher-order regularization have been made—for regularizers that depend on the Laplacian [48, 66] in case of vectorial ranges $\Gamma \subset \mathbb{R}^s$ and for the total generalized variation [56, 60] in case of a scalar range $\Gamma \subset \mathbb{R}$. However, in contrast to the first-order theory, the higher-order models, although empirically useful, are still considerably less mathematically validated. Furthermore, we mention that there are models where the image *domain* Ω is replaced by a shape (or manifold) [14, 24], which is beyond the scope of this survey.

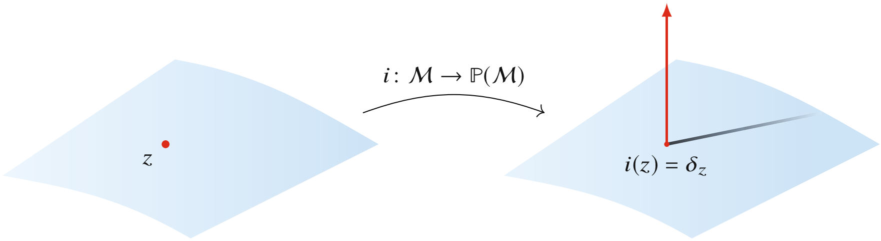


Fig. 3.2 A manifold $\mathcal{M}$ is embedded into the space $\mathbb{P}(\mathcal{M})$ of probability measures via the identification of a point $z \in \mathcal{M}$ with the Dirac point measure δ_z concentrated at z. This "lifts" the problem into a higher-dimensional *linear* space, which is much more amenable to global optimization methods

3.1.2 Manifold-Valued Functional Lifting

In this chapter, we will be concerned with problems where Γ has a manifold structure. The first step towards applying lifting methods to such problems was an application to the restoration of cyclic data [23, 62] with $\Gamma = \mathbb{S}^1$, which was later [47] generalized for the case of total variation regularization to data with values in more general manifolds. In [47], the functional lifting approach is applied to a first-order model with total variation regularizer,

$$F(u) = \int_\Omega \rho(x, u(x))dx + \lambda\, \mathrm{TV}(u), \tag{3.5}$$

for $u\colon \Omega \to \mathcal{M}$, where $\Gamma = \mathcal{M}$ is an s-dimensional manifold and $\rho\colon \Omega \times \mathcal{M} \to \mathbb{R}$ is a pointwise data discrepancy. The lifted space is chosen to be $X = \mathbb{P}(\mathcal{M})$, the space of Borel probability measures over $\mathcal{M}$, with embedding $i\colon \mathcal{M} \hookrightarrow \mathbb{P}(\mathcal{M})$, where $i(z) := \delta_z$ is the Dirac point measure with unit mass concentrated at $z \in \mathcal{M}$ (see Fig. 3.2). The lifted functional is

$$\tilde{F}(v) = \int_\Omega \langle \rho(x, \cdot), v(x)\rangle\, dx + \lambda \widetilde{\mathrm{TV}}(v), \tag{3.6}$$

where $\langle g, \mu\rangle := \int_\mathcal{M} g\, d\mu$ for $g \in C(\mathcal{M})$ and $\mu \in \mathbb{P}(\mathcal{M})$. Furthermore,

$$\widetilde{\mathrm{TV}}(v) := \sup \left\{ \int_\Omega \langle \operatorname{div}_x p(x, \cdot), v(x)\rangle\, dx : p\colon \Omega \times \mathcal{M} \to \mathbb{R}, \|\nabla_z p\|_\infty \leq 1 \right\}. \tag{3.7}$$

The Lipschitz constraint $\|\nabla_z p\|_\infty \leq 1$, where

$$\|\nabla_z p\|_\infty := \sup \left\{ \|\nabla_z p(x, z)\|_{\sigma,\infty} : (x, z) \in \Omega \times \mathcal{M} \right\}, \tag{3.8}$$

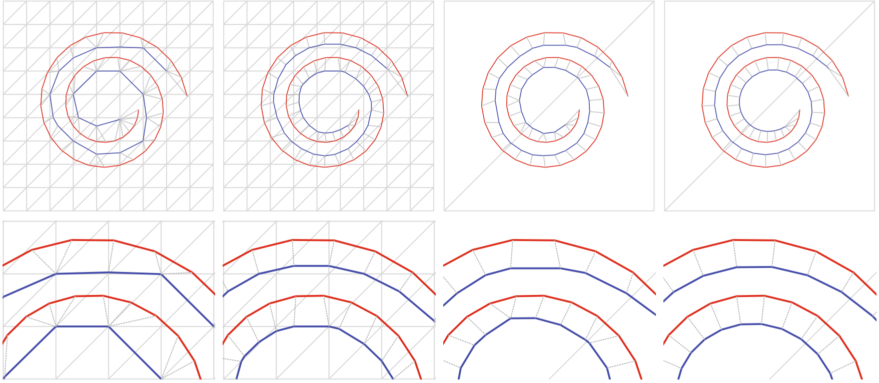
and $\|\cdot\|_{\sigma,\infty}$ the spectral (operator) norm, can be explained by a functional analytic perspective [65] on this lifting strategy: The lifted total variation functional is the vectorial total variation semi-norm for functions over Ω with values in a certain

Banach space of measures. The topological dual space of this space of measures is the space of Lipschitz continuous functions over $\mathcal{M}$. However, this interpretation does not generalize easily to other regularizers. We will instead base our model for general convex regularizers on the theory of currents as presented in [51].

Sublabel Accuracy While the above model comes with a fully continuous description, a numerical implementation requires the discretization of Ω as well as the range Γ. This introduces two possible causes for errors: *metrication errors* and *label bias*.

Metrication errors are artifacts related to the graph or grid representation of the spatial image domain Ω, finite difference operators, and the choice of metric thereof. They manifest mostly in unwanted anisotropy, missing rotational invariance, or blocky diagonals. They constitute a common difficulty with all variational problems and lifting approaches [37].

In contrast, *label bias* means that the discretization favors solutions that assume values at the chosen "labels" (discretization points) $Z^1, \ldots, Z^L$ in the *range* Γ (see Figs. 3.3 and 3.4). This is very desirable for discrete Γ, but in the context of manifolds, severely limits accuracy and forces a suitably fine discretization of the range.



[47], 10 × 10 labels
label bias

[38], 10 × 10 labels
no label bias

[38], 2 × 2 labels
sublabel-accurate

[38], 2 × 2 labels
exact data term

Fig. 3.3 Rudin-Osher-Fatemi (ROF) $L^2 - \text{TV}$ denoising (blue) of an (Euclidean) vector-valued signal $u \colon [0, 1] \to \mathbb{R}^2$ (red), visualized as a curve in the flat manifold $\mathcal{M} = \mathbb{R}^2$. The problem is solved by the continuous multi-labeling framework with functional lifting described in this chapter. The discretization points (labels) in the range $\mathcal{M}$, which are necessary for the implementation of the lifted problem, are visualized by the gray grid. **Left:** The method proposed in [47] does not force the solution to assume values at the grid points (labels), but still shows significant bias towards edges of the grid (blue curve). **Second from left:** With the same number of labels, the method from [38] is able to reduce label bias by improving data term discretization. **Second from right:** Furthermore, the method from [38] allows to exploit the convexity of the data term to get decent results with as little as four grid points. **Right:** Further exploiting the quadratic form of the data term even produces the numerically exact reference solution, which in this case can be precisely computed using the unlifted formulation due to the convexity of the problem. This shows that for the Euclidean fully convex case, the sublabel-accurate lifting allows to recover the exact solution with careful discretization

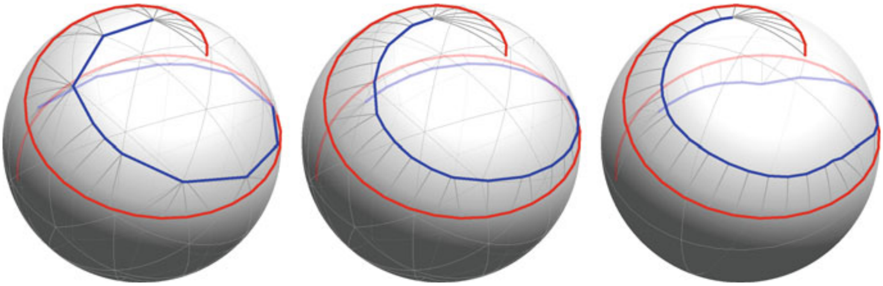


Fig. 3.4 Total Variation denoising (blue) of a signal $u\colon [0,1] \to \mathbb{S}^2$ with values in $\mathbb{S}^2$ (red), visualized as curves on the two-dimensional sphere embedded into $\mathbb{R}^3$. The problem is solved by the continuous multi-labeling framework with functional lifting described in this chapter. The discretization points (labels), that are necessary for the implementation of the lifted problem, are visualized by the gray grid. **Left:** The method proposed in [47] does not force the solution to take values at the grid points, but still shows significant grid bias. **Center:** With the same number of labels, our proposed method, motivated by [38], reduces label bias by improving data term discretization. **Right:** Furthermore, our method can get excellent results with as little as six grid points (right). Note that the typical contrast reduction that occurs in the classical Euclidean ROF can also be observed in the manifold-valued case in the form of a shrinkage towards the Fréchet mean

In more recent so-called *sublabel-accurate* approaches for scalar and vectorial ranges Γ, more emphasis is put on the discretization [38, 52, 70] to get rid of label bias in models with total variation regularization, which allows to greatly reduce the number of discretizations points for the range Γ. In a recent publication [50], the gain in sublabel accuracy is explained to be caused by an implicit application of first-order finite elements on Γ as opposed to previous approaches that can be interpreted as using zero-order elements, which naturally introduces label-bias. An extension of the sublabel-accurate approaches to arbitrary convex regularizers using the theory of currents was recently proposed in [51].

Motivated by these recent advances, we propose to extend the methods from [47] for manifold-valued images to arbitrary convex regularizers, making use of finite element techniques on manifolds [25]. This reduces label bias and thus the amount of labels necessary in the discretization.

3.1.3 Further Related Work

The methods proposed in this work are applicable to variational problems with values in manifolds of dimension $s \leq 3$. The theoretical framework applies to manifolds of arbitrary dimension, but the numerical costs increase exponentially for dimensions 4 and larger.

An alternative is to use *local* optimization methods on manifolds. A reference for the smooth case is [2]. For non-smooth energies, methods such as the cyclic proximal point, Douglas-Rachford, ADMM and (sub-)gradient descent algorithm have been applied to first and second order TV and TGV as well as Mumford-Shah and Potts regularization approaches in [9, 11, 12, 16, 67, 68]. These methods are generally applicable to manifolds of any dimension whose (inverse) exponential mapping can be evaluated in reasonable time and quite efficient in finding a local minimum, but can get stuck in local extrema. Furthermore, the use of total variation regularization in these frameworks is currently limited to anisotropic formulations; Tikhonov regularization was proposed instead for isotropic regularization [13, 67]. An overview of applications, variational models and local optimization methods is given in [13].

Furthermore, we mention that, beyond variational models, there exist statistical [27], discrete graph-based [10], wavelet-based [59], PDE-based [22] and patch-based [39] models for the processing and regularization of manifold-valued signals.

3.2 Submanifolds of $\mathbb{R}^N$

We formulate our model for submanifolds of $\mathbb{R}^N$ which is no restriction by the Whitney embedding theorem [41, Thm. 6.15]. For an s-dimensional submanifold of $\mathbb{R}^N$ and $\Omega \subset \mathbb{R}^d$ open and bounded, differentiable functions $u\colon \Omega \to \mathcal{M}$ are regarded as a subset of differentiable functions with values in $\mathbb{R}^N$. For those functions, a Jacobian $Du(x) \in \mathbb{R}^{N,d}$ in the Euclidean sense exists that can be identified with the push-forward of the tangent space $T_x\Omega$ to $T_{u(x)}\mathcal{M}$, i.e., for each $x \in \Omega$ and $\xi \in \mathbb{R}^d = T_x\Omega$, we have

$$Du(x)\xi \in T_{u(x)}\mathcal{M} \subset T_{u(x)}\mathbb{R}^N. \tag{3.9}$$

On the other hand, for differentiable maps $p\colon \mathcal{M} \to \mathbb{R}^d$, there exists an extension of p to a neighborhood of $\mathcal{M} \subset \mathbb{R}^N$ that is constant in normal directions and we denote by $\nabla p(z) \in \mathbb{R}^{N,d}$ the Jacobian of this extension evaluated at $z \in \mathcal{M}$. Since the extension is assumed to be constant in normal directions, i.e., $\nabla p(z)\zeta = 0$ whenever $\zeta \in N_z\mathcal{M}$ (the orthogonal complement of $T_z\mathcal{M}$ in $\mathbb{R}^N$), this definition is independent of the choice of extension.

3.2.1 Calculus of Variations on Submanifolds

In this section, we generalize the total variation based approach in [47] to less restrictive first-order variational problems by applying the ideas from functional lifting of vectorial problems [51] to manifold-valued problems. Most derivations will be formal; we leave a rigorous choice of function spaces as well as an analysis of well-posedness for future work. We note that theoretical work is available for the scalar-valued case in [3, 15, 55] and for the vectorial and for selected manifold-valued cases in [29].

We consider variational models on functions $u\colon \Omega \to \mathcal{M}$,

$$F(u) := \int_\Omega f(x, u(x), Du(x))\, dx, \tag{3.10}$$

for which the integrand $f\colon \Omega \times \mathcal{M} \times \mathbb{R}^{N,d} \to \mathbb{R}$ is convex in the last component. Note that the dependence of f on the full Jacobian of u spares us dealing with the tangent bundle push-forward $T\Omega \to T\mathcal{M}$ in a coordinate-free way, thus facilitating discretization later on.

Formally, the lifting strategy for vectorial problems proposed in [51] can be generalized to this setting by replacing the range Γ with $\mathcal{M}$. As the lifted space, we consider the space of probability measures on the Borel σ-Algebra over $\mathcal{M}$, $X = \mathbb{P}(\mathcal{M})$, with embedding $i\colon \mathcal{M} \to \mathbb{P}(\mathcal{M})$, where $i(z) = \delta_z$ is the Dirac point mass concentrated at $z \in \mathcal{M}$. Furthermore, we write $\Sigma := \Omega \times \mathcal{M}$ and, for $(x, z) = y \in \Sigma$, we define the coordinate projections $\pi_1 y := x$ and $\pi_2 y := z$. Then, for $v\colon \Omega \to \mathbb{P}(\mathcal{M})$, we define the lifted functional

$$\tilde{F}(v) := \sup \left\{ \int_{\Omega} \langle -\operatorname{div}_x p(x,\cdot) + q(x,\cdot), v(x) \rangle \, dx : (\nabla_z p, q) \in \mathcal{K} \right\}, \tag{3.11}$$

where $\langle g, \mu \rangle := \int_{\mathcal{M}} g \, d\mu$ is the dual pairing between $g \in C(\mathcal{M})$ and $\mu \in \mathbb{P}(\mathcal{M})$ and

$$\mathcal{K} := \left\{ (P, q) \in C(\Sigma; \mathbb{R}^{N,d} \times \mathbb{R}) : f^*(\pi_1 y, \pi_2 y, P(y)) + q(y) \le 0 \, \forall y \in \Sigma \right\}, \tag{3.12}$$

where $f^*(x, z, \zeta) := \sup_\xi \langle \zeta, \xi \rangle - f(x, z, \xi)$ is the convex conjugate of f with respect to the last variable.

In the following, the integrand $f\colon \Omega \times \mathcal{M} \times \mathbb{R}^{N,d} \to \mathbb{R}$ is assumed to decompose as

$$f(x, z, \xi) = \rho(x, z) + \eta(P_z \xi) \tag{3.13}$$

into a pointwise data term $\rho\colon \Omega \times \mathcal{M} \to \mathbb{R}$ and a convex regularizer $\eta\colon \mathbb{R}^{s,d} \to \mathbb{R}$ that only depends on an s-dimensional representation of vectors in $T_z\mathcal{M}$ given by a surjective linear map $P_z \in \mathbb{R}^{s,N}$ with $\ker(P_z) = N_z\mathcal{M}$.

This very general integrand covers most first-order models in the literature on manifold-valued imaging problems. It applies in particular to isotropic and anisotropic regularizers that depend on (matrix) norms of $Du(x)$ such as the Frobenius or spectral norm (or operator norm) where P_z is taken to be an arbitrary orthogonal basis transformation. Since $z \mapsto P_z$ is not required to be continuous, it can also be applied to non-orientable manifolds such as the Moebius strip or the Klein bottle where no continuous orthogonal basis representation of the tangent bundle $T\mathcal{M}$ exists.

Regularizers of this particular form depend on the manifold through the choice of P_z only. This is important because we approximate $\mathcal{M}$ in the course of our proposed discretization by a discrete (simplicial) manifold $\mathcal{M}_h$ and the tangent spaces $T_z\mathcal{M}$ are replaced by the linear spaces spanned by the simplicial faces of $\mathcal{M}_h$.

3.2.2 Finite Elements on Submanifolds

We translate the finite element approach for functional lifting proposed in [50] to the manifold-valued setting by employing the notation from surface finite element methods [25].

The manifold $\mathcal{M} \subset \mathbb{R}^N$ is approximated by a triangulated topological manifold $\mathcal{M}_h \subset \mathbb{R}^N$ in the sense that there is a homeomorphism $\iota\colon \mathcal{M}_h \to \mathcal{M}$ (Figs. 3.5 and 3.6). By $\mathcal{T}_h$, we denote the set of simplices that make up $\mathcal{M}_h$:

$$\bigcup_{T \in \mathcal{T}_h} T = \mathcal{M}_h. \tag{3.14}$$

For $T, \tilde{T} \in \mathcal{T}_h$, either $T \cap \tilde{T} = \emptyset$ or $T \cap \tilde{T}$ is an $(s-k)$-dimensional face for $k \in \{1, \dots, s\}$. Each simplex $T \in \mathcal{T}_h$ spans an s-dimensional linear subspace of $\mathbb{R}^N$ and there is an orthogonal basis representation $P_T \in \mathbb{R}^{s,N}$ of vectors in $\mathbb{R}^N$ to that subspace. Furthermore, for later use, we enumerate the vertices of the triangulation as $Z^1, \dots, Z^L \in \mathcal{M} \cap \mathcal{M}_h$.

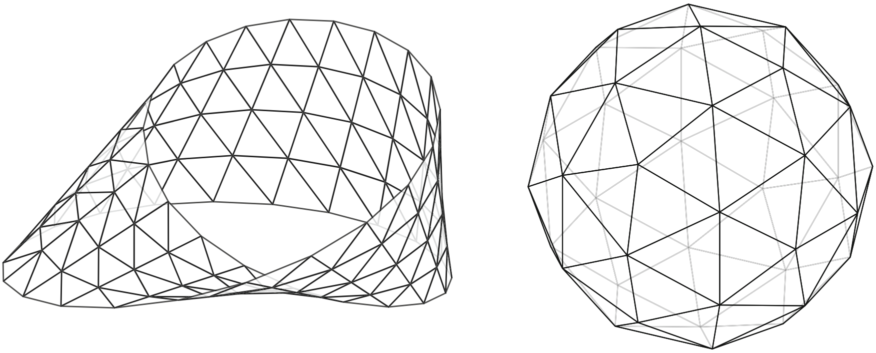
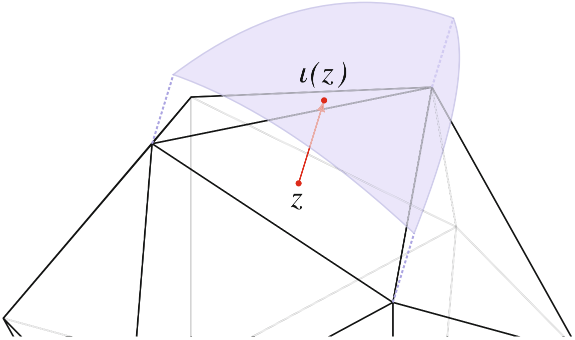


Fig. 3.5 Triangulated approximations of the Moebius strip (left) and the two-dimensional sphere (right) as surfaces embedded into $\mathbb{R}^3$

Fig. 3.6 Each simplex T in a triangulation (black wireframe plot) is in homeomorphic correspondence to a piece $\iota(T)$ of the original manifold (blue) through the map $\iota\colon \mathcal{M}_h \to \mathcal{M}$



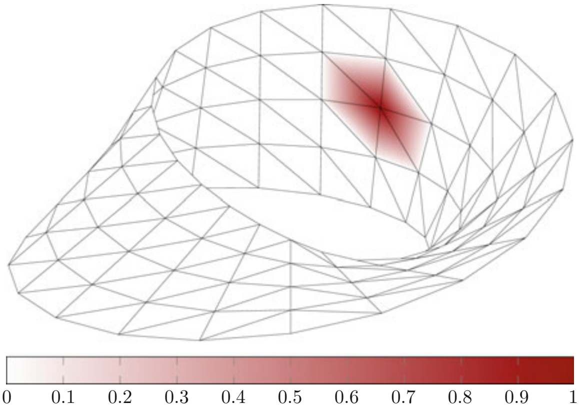


Fig. 3.7 The first-order finite element space S_h is spanned by a nodal basis $\chi_1, \dots, \chi_L$ which is uniquely determined by the property $\chi_k(Z^l) = 1$ if $k = l$ and $\chi_k(Z^l) = 0$ otherwise. The illustration shows a triangulation of the Moebius strip with a color plot of a nodal basis function

For the numerics, we assume the first-order finite element space

$$S_h := \{\phi_h \in C^0(\mathcal{M}_h) : \phi_h|_T \text{ is linear affine for each } T \in \mathcal{T}_h\}. \tag{3.15}$$

The functions in S_h are piecewise differentiable on $\mathcal{M}_h$ and we define the surface gradient $\nabla_T\phi_h \in \mathbb{R}^{N,d}$ of $\phi_h \in S_h$ by the gradient of the linear affine extension of $\phi_h|_T$ to $\mathbb{R}^N$. If L is the number of vertices in the triangulation of $\mathcal{M}_h$, then S_h is a linear space of dimension L with nodal basis $\chi_1, \dots, \chi_L$ which is uniquely determined by the property $\chi_k(Z^l) = 1$ if $k = l$ and $\chi_k(Z^l) = 0$ otherwise (Fig. 3.7).

The dual space of S_h, which we denote by $\mathfrak{M}_h(\mathcal{M}_h)$, is a space of signed measures. We identify $\mathfrak{M}_h(\mathcal{M}_h) = \mathbb{R}^L$ via dual pairing with the nodal basis $\chi_1, \dots, \chi_L$, i.e., to each $\mu_h \in \mathfrak{M}_h(\mathcal{M}_h)$ we associate the vector $(\langle\mu_h, \chi_1\rangle, \dots, \langle\mu_h, \chi_L\rangle)$. We then replace the space $\mathbb{P}(\mathcal{M})$ of probability measures over $\mathcal{M}$ by the convex subset

$$\mathbb{P}_h(\mathcal{M}_h) = \left\{\mu_h \in \mathfrak{M}_h(\mathcal{M}_h) : \mu_h \geq 0, \sum_{k=1}^{L}\langle\mu_h, \chi_k\rangle = 1\right\}. \tag{3.16}$$

The energy functional is then translated to the discretized setting by redefining the integrand f on $\mathcal{M}_h$ for any $x \in \Omega$, $z \in \mathcal{M}_h$ and $\xi \in \mathbb{R}^{N,d}$ as

$$\tilde{f}(x, z, \xi) := \rho(x, \iota(z)) + \eta(P_T\xi) \tag{3.17}$$

The epigraphical constraints in $\mathcal{K}$ translate to

$$\forall x \in \Omega \forall z \in \mathcal{M}_h : \quad \eta^*(P_T\nabla_z p(x, z)) - \rho(x, \iota(z)) + q(x, z) \leq 0, \tag{3.18}$$

for functions $p \in S_h^d$ and $q \in S_h$. The constraints can be efficiently implemented on each $T \in \mathcal{T}_h$ where $\nabla_z p$ is constant and $q(x, z) = \langle q_{T,1}(x), z\rangle + q_{T,2}(x)$ is linear affine in z:

$$\eta^*(P_T \nabla_T p(x)) + \langle q_{T,1}(x), z\rangle - \rho(x, \iota(z)) \leq -q_{T,2}(x), \tag{3.19}$$

for any $x \in \Omega$, $T \in \mathcal{T}_h$ and $z \in T$. Following the approach in [50], we define

$$\rho_T^*(x, z) := \sup_{z' \in T} \langle z, z'\rangle - \rho(x, \iota(z')), \tag{3.20}$$

and introduce auxiliary variables a_T, b_T to split the epigraphical constraint (3.19) into two epigraphical and one linear constraint for $x \in \Omega$ and $T \in \mathcal{T}_h$:

$$\eta^*(P_T \nabla_T p(x)) \leq a_T(x), \tag{3.21}$$

$$\rho_T^*(q_{T,1}(x)) \leq b_T(x), \tag{3.22}$$

$$a_T(x) + b_T(x) = -q_{T,2}(x). \tag{3.23}$$

The resulting optimization problem is described by the following saddle point form over functions $v\colon \Omega \to \mathbb{P}_h(\mathcal{M}_h)$, $p \in C^1(\Omega, S_h^{d+1})$ and $q \in C(\Omega, S_h)$:

$$\inf_{v} \sup_{p,q} \int_\Omega \langle -\operatorname{div}_x p(x,\cdot) + q(x,\cdot), v(x)\rangle \, dx \tag{3.24}$$

$$\text{subject to } \eta^*(P_T \nabla_T p(x)) \leq a_T(x), \tag{3.25}$$

$$\rho_T^*(q_{T,1}(x)) \leq b_T(x), \tag{3.26}$$

$$a_T(x) + b_T(x) + q_{T,2}(x) = 0. \tag{3.27}$$

Finally, for the fully discrete setting, the domain Ω is replaced by a Cartesian rectangular grid with finite differences operator ∇_x and Neumann boundary conditions.

3.2.3 Relation to [47]

In [47], a similar functional lifting is proposed for the special case of total variation regularization and without the finite elements interpretation. More precisely, the regularizing term is chosen to be $\eta(\xi) = \lambda\|\xi\|_{\sigma,1}$ for $\xi \in \mathbb{R}^{s,d}$, where $\|\cdot\|_{\sigma,1}$ is the matrix nuclear norm, also known as Schatten-1-norm, which is given by the sum of singular values of a matrix. It is the dual to the matrix operator or spectral norm $\|\cdot\|_{\sigma,\infty}$. If we substitute this choice of η into the discretization given above, the epigraphical constraint (3.18) translates to the two constraints

$$\|P_T \nabla_T p(x)\|_{\sigma,\infty} \leq \lambda \text{ and } q(x,z) \leq \rho(x, \iota(z)). \tag{3.28}$$

The first one is a Lipschitz constraint just as in the model from [47], but two differences remain:

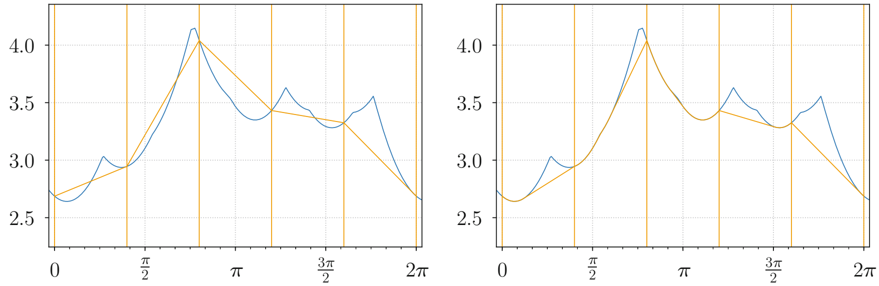


Fig. 3.8 Data term discretization for the lifting approach applied to the Riemannian center of mass problem introduced in Fig. 3.1. For each $x \in \Omega$, the data term $z \mapsto \rho(x, z)$ (blue graph) is approximated (orange graphs) between the label points Z^k (orange vertical lines). **Left:** In the lifting approach [47] for manifold-valued problems, the data term is interpolated linearly between the labels. **Right:** Based on ideas from recent scalar and vectorial lifting approaches [38, 52], we interpolate piecewise convex between the labels

1. In [47], the lifted and discretized form of the data term reads

$$\int_\Omega \sum_{k=1}^{L} \rho(x, Z^k) v(x)^k \, dx. \tag{3.29}$$

   This agrees with our setting if $z \mapsto \rho(x, \iota(z))$ is affine linear on each simplex $T \in \mathcal{T}_h$, as then $q(x, z) = \rho(x, \iota(z))$ maximizes the objective function for any p and v. Hence, the model in [47] doesn't take into account any information about ρ below the resolution of the triangulation. We improve this by implementing the epigraph constraints $\rho_T^*(q_{T,1}(x)) \leq b_T(x)$ as proposed in [38] using a convex approximation of ρ_T (see Fig. 3.8). The approximation is implemented numerically with piecewise affine linear functions in a "sublabel-accurate" way, i.e., at a resolution below the resolution of the triangulation.
2. A very specific discretization of the gradients $\nabla_T p(x)$ is proposed in [47]: To each simplex in the triangulation a mid-point $y_T \in \mathcal{M}$ is associated. The vertices $Z_T^1, \ldots, Z_T^{s+1}$ of the simplex are projected to the tangent space at y_T as $v_T^k := \log_{y_T} Z_T^k$. The gradient is then computed as the vector g in the tangent space $T_{y_T}\mathcal{M}$ describing the affine linear map on $T_{y_T}\mathcal{M}$ that takes values $p(Z_T^k)$ at the points v_T^k, $k = 1, \ldots, s+1$ (Fig. 3.9).

   This procedure aims to make up for the error introduced by the simplicial discretization and amounts to a different choice of P_T—a slight variant of our model. We did not observe any significant positive or negative effects from using either discretization; the difference between the minimizers is very small.

   In the one-dimensional case, the two approaches differ only in a constant factor: Denote by $P_T \in \mathbb{R}^{s,N}$ the orthogonal basis representation of vectors in $\mathbb{R}^N$ in the subspace spanned by the simplex $T \in \mathcal{T}_h$ and denote by $\tilde{P}_T \in \mathbb{R}^{s,N}$ the alternative approach from [47]. Now, consider a triangulation $\mathcal{T}_h$ of the circle

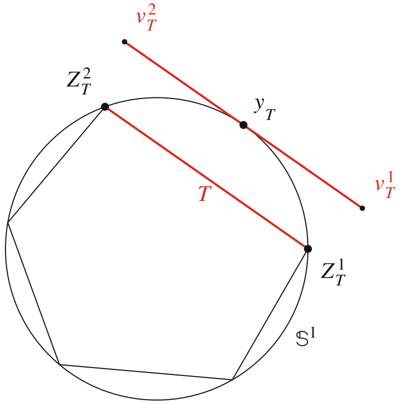


Fig. 3.9 Mapping a simplex T, spanned by $Z_T^1, \dots, Z_T^{s+1}$, to the tangent space at its center-of-mass y_T using the logarithmic map. The proportions of the simplex spanned by the mapped points $v_T^1, \dots, v_T^{s+1}$ may differ from the proportions of the original simplex for curved manifolds. The illustration shows the case of a circle $\mathbb{S}^1 \subset \mathbb{R}^2$, where the deformation reduces to a multiplication by a scalar α_T, the ratio between the geodesic (angular) and Euclidean distance between Z_T^1 and Z_T^2. The gradient $\nabla_T p$ of a finite element $p \in S_h$ can be modified according to this change in proportion in order to make up for some of the geometric (curvature) information lost in the discretization

$\mathbb{S}^1 \subset \mathbb{R}^2$ and a one-dimensional simplex $T \in \mathcal{T}_h$. A finite element $p \in S_h$ that takes values $p_1, p_2 \in \mathbb{R}$ at the vertices $Z_T^1, Z_T^2 \in \mathbb{R}^2$ that span T has the gradient

$$\nabla_T p = (p_1 - p_2)\frac{Z_T^1 - Z_T^2}{\|Z_T^1 - Z_T^2\|_2^2} \in \mathbb{R}^2 \tag{3.30}$$

and $P_T, \tilde{P}_T \in \mathbb{R}^{1,2}$ are given by

$$P_T := \frac{(Z_T^1 - Z_T^2)^\top}{\|Z_T^1 - Z_T^2\|_2}, \qquad \tilde{P}_T := \frac{(Z_T^1 - Z_T^2)^\top}{d_{\mathbb{S}^1}(Z_T^1, Z_T^2)}. \tag{3.31}$$

Hence $P_T = \alpha_T \tilde{P}_T$ for $\alpha_T = d_{\mathbb{S}^1}(Z_T^1, Z_T^2)/\|Z_T^1 - Z_T^2\|_2$ the ratio between geodesic (angular) and Euclidean distance between the vertices. If the vertices are equally spaced on $\mathbb{S}^1$, this is a constant factor independent of T that typically scales the discretized regularizer by a small constant factor. On higher-dimensional manifolds, more general linear transformations $P_T = A_T \tilde{P}_T$ come into play. For very irregular triangulations and coarse discretization, this may affect the minimizer; however, in our experiments the observed differences were negligible.

3.2.4 Full Discretization and Numerical Implementation

A prime advantage of the lifting method when applied to manifold-valued problems is that it translates most parts of the problem into Euclidean space. This allows to apply established solution strategies for the non-manifold case, which rely on non-smooth convex optimization: After discretization, the convex-concave saddle-point form allows for a solution using the primal-dual hybrid gradient method [18, 19] with recent extensions [32]. In this optimization framework, the epigraph constraints are realized by projections onto the epigraphs in each iteration step. For the regularizers to be discussed in this paper (TV, quadratic and Huber), we refer to the instructions given in [55]. For the data term ρ, we follow the approach in [38]: For each $x \in \Omega$, The data term $z \mapsto \rho(x, \iota(z))$ is sampled on a subgrid of $\mathcal{M}_h$ and approximated by a piecewise affine linear function. The quickhull algorithm can then be used to get the convex hull of this approximation. Projections onto the epigraph of ρ_T^* are then projections onto convex polyhedra, which amounts to solving many low-dimensional quadratic programs; see [38] for more details.

Following [47], the numerical solution $u\colon \Omega \to \mathbb{P}_h(\mathcal{M}_h)$, taking values in the lifted space $\mathbb{P}_h(\mathcal{M}_h)$, is projected back to a function $u\colon \Omega \to \mathcal{M}$, taking values in the original space $\mathcal{M}$, by mapping, for each $x \in \Omega$ separately, a probability measure $u(x) = (\lambda_1, \ldots, \lambda_L) = \mu_h \in \mathbb{P}_h(\mathcal{M}_h)$ to the following Riemannian center of mass on the original manifold $\mathcal{M}$:

$$\mu_h = (\lambda_1, \ldots, \lambda_L) \mapsto \operatorname*{arg\,min}_{z \in \mathcal{M}} \sum_{k=1}^{L} \lambda_k d_{\mathcal{M}}(z, Z^k)^2 \tag{3.32}$$

For $\mathcal{M} = \mathbb{R}^s$, this coincides with the usual weighted mean $\bar{z} = \sum_{k=1}^{L} \lambda_k Z_k$. However, on manifolds this minimization is known to be a non-convex problem with non-unique solutions (compare Fig. 3.1). Still, in practice the iterative method described in [35] yields reasonable results for all real-world data considered in this work: Starting from a point $z_0 := Z^k$ with maximum weight λ_k, we proceed for $i \geq 0$ by projecting the Z^k, $k = 1, \ldots, L$, to the tangent space at z_i using the inverse exponential map, taking the linear weighted mean v_i there and defining z_{i+1} as the projection of v_i to $\mathcal{M}$ via the exponential map:

$$V_i^k := \log_{z_i}(Z^k) \in T_{z_i}\mathcal{M}, \; k = 1, \ldots, L, \tag{3.33}$$

$$v_i := \sum_{k=1}^{L} \lambda_k V_i^k \in T_{z_i}\mathcal{M}, \tag{3.34}$$

$$z_{i+1} := \exp_{z_i}(v_i). \tag{3.35}$$

The method converges rapidly in practice. It has to be applied only once for each $x \in \Omega$ after solving the lifted problem, so that efficiency is non-critical.

3.3 Numerical Results

We apply our model to problems with quadratic data term $\rho(x,z) := d^2_{\mathcal{M}}(I(x),z)$ and Huber, total variation (TV) and Tikhonov (quadratic) regularization with parameter $\lambda > 0$:

$$\eta_{\text{TV}}(\xi) := \lambda \|\xi\|_2, \tag{3.36}$$

$$\eta_{\text{Huber}}(\xi) := \lambda \phi_\alpha(\xi), \tag{3.37}$$

$$\eta_{\text{quad}}(\xi) := \frac{\lambda}{2}\|\xi\|_2^2, \tag{3.38}$$

where the Huber function ϕ_α for $\alpha > 0$ is defined by

$$\phi_\alpha(\xi) := \begin{cases} \frac{\|\xi\|_2^2}{2\alpha} & \text{if } \|\xi\|_2 \leq \alpha, \\ \|\xi\|_2 - \frac{\alpha}{2} & \text{if } \|\xi\|_2 > \alpha. \end{cases} \tag{3.39}$$

Note that previous lifting approaches for manifold-valued data were restricted to total variation regularization η_{TV}.

The methods were implemented in Python 3 with NumPy and PyCUDA, running on an Intel Core i7 4.00 GHz with an NVIDIA GeForce GTX 1080 Ti 12 GB and 16 GB RAM. The iteration was stopped as soon as the relative gap between primal and dual objective fell below 10^{-5}. Approximate runtimes ranged between 5 and 45 min. The code is available from https://github.com/room-10/mfd-lifting.

3.3.1 *One-Dimensional Denoising on a Klein Bottle*

Our model can be applied to both orientable and non-orientable manifolds. Figure 3.10 shows an application of our method to Tikhonov denoising of a synthetic one-dimensional signal $u\colon [0,1] \to \mathcal{M}$ on the two-dimensional Klein surface embedded in $\mathbb{R}^3$, a non-orientable closed surface that cannot be embedded into $\mathbb{R}^3$ without self-intersections. Our numerical implementation uses a triangulation with a very low count of 5×5 vertices and 50 triangles. The resolution of the signal (250 one-dimensional data points) is far below the resolution of the triangulation and, still, our approach is able to restore a smooth curve.

3.3.2 *Three-Dimensional Manifolds: $SO(3)$*

Signals with rotational range $u\colon \Omega \to SO(3)$ occur in the description of crystal symmetries in EBSD (Electron Backscatter Diffraction Data) and in motion tracking. The rotation group $SO(3)$ is a three-dimensional manifold that can be identified with the three-dimensional unit-sphere $\mathbb{S}^3$ up to identification of antipodal points via

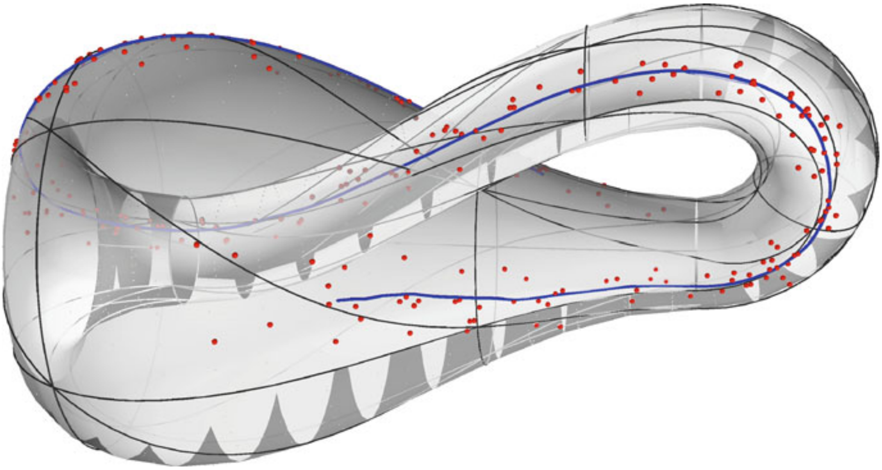


Fig. 3.10 Tikhonov (quadratic) denoising (blue) of a one-dimensional signal (red) $u\colon [0,1] \to \mathcal{M}$ with values on the two-dimensional Klein surface (commonly referred to as Klein bottle) $\mathcal{M} \subset \mathbb{R}^3$. The black wireframe lines on the surface represent the triangulation used by the discretization of our functional lifting approach. The numerical implementation recovers the denoised signal at a resolution far below the resolution of the manifold's discretization. The lifting approach does not require the manifold to be orientable

the quaternion representation of 3D rotations. A triangulation of $\mathbb{S}^3$ is given by the vertices and simplicial faces of the hexacosichoron (600-cell), a regular polytope in $\mathbb{R}^4$ akin to the icosahedron in $\mathbb{R}^3$. As proposed in [47], we eliminate opposite points in the hexacosichoron and obtain a discretization of $SO(3)$ with 60 vertices and 300 tetrahedral faces.

Motivated by Bézier surface interpolation [1], we applied Tikhonov regularization to a synthetic inpainting (interpolation) problem with added noise (Fig. 3.11). In our variational formulation, we chose $\rho(x,z) = 0$ for x in the inpainting area and $\rho(x,z) = \delta_{\{z=I(x)\}}$ (a hard constraint to the input signal $I\colon \Omega \to SO(3)$) for x in the known area.

Using the proposed sublabel-accurate handling of data terms, we obtain good results with only 60 vertices, in contrast to [47], where the discretization is refined to 720 vertices (Fig. 3.11).

3.3.3 Normals Fields from Digital Elevation Data

In digital elevation models (DEM), elevation information for earth science studies and mapping applications often includes surface normals which can be used to produce a shaded coloring of elevation maps. Normal fields $u\colon \Omega \to \mathbb{S}^2$ are defined

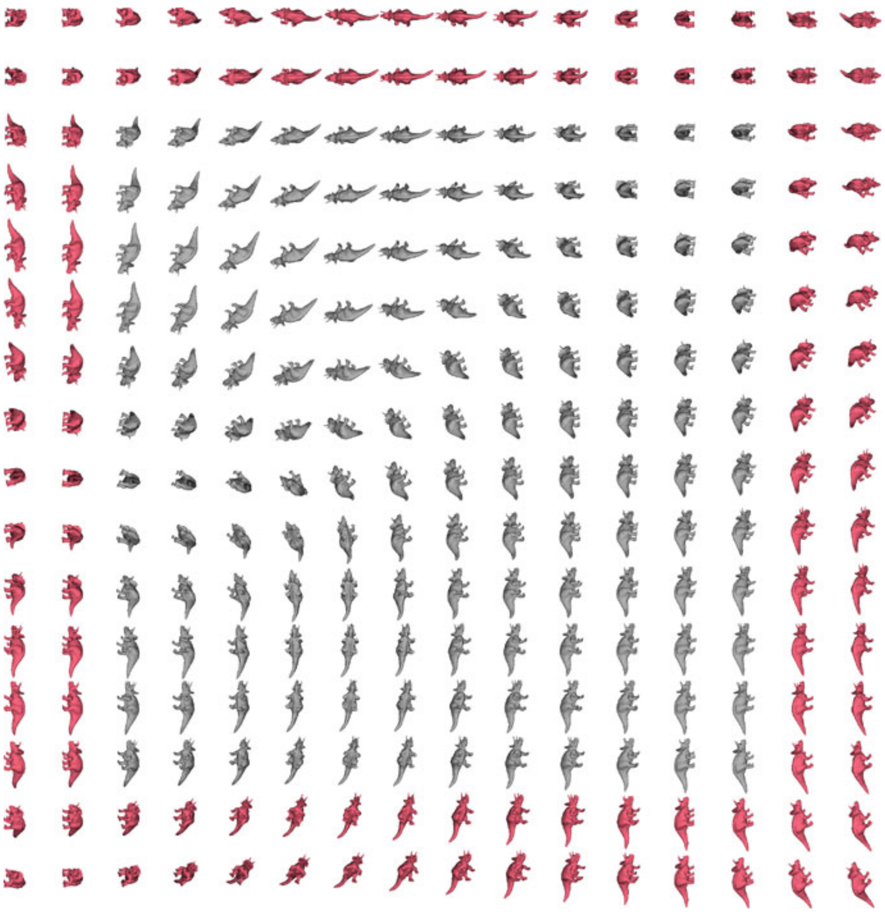


Fig. 3.11 Tikhonov inpainting of a two-dimensional signal of (e.g., camera) orientations, elements of the three-dimensional special orthogonal group of rotations $SO(3)$, a manifold of dimension $s = 3$. The masked input signal (red) is inpainted (gray) using our model with Tikhonov (quadratic) regularization. The interpolation into the central area is smooth. Shape: *Triceratops* by BillyOceansBlues (CC-BY-NC-SA, https://www.thingiverse.com/thing:3313805)

on a rectangular image domain $\Omega \subset \mathbb{R}^2$; variational processing of the normal fields is therefore a manifold-valued problem on the two-dimensional sphere $\mathbb{S}^2 \subset \mathbb{R}^3$.

Denoising using variational regularizers from manifold-valued image processing before computing the shading considerably improves visual quality (Fig. 3.12). For our framework, the sphere was discretized using 12 vertices and 20 triangles, chosen to form a regular icosahedron. The same dataset was used in [47], where the proposed lifting approach required 162 vertices—and solving a proportionally larger optimization problem—in order to produce comparable results.

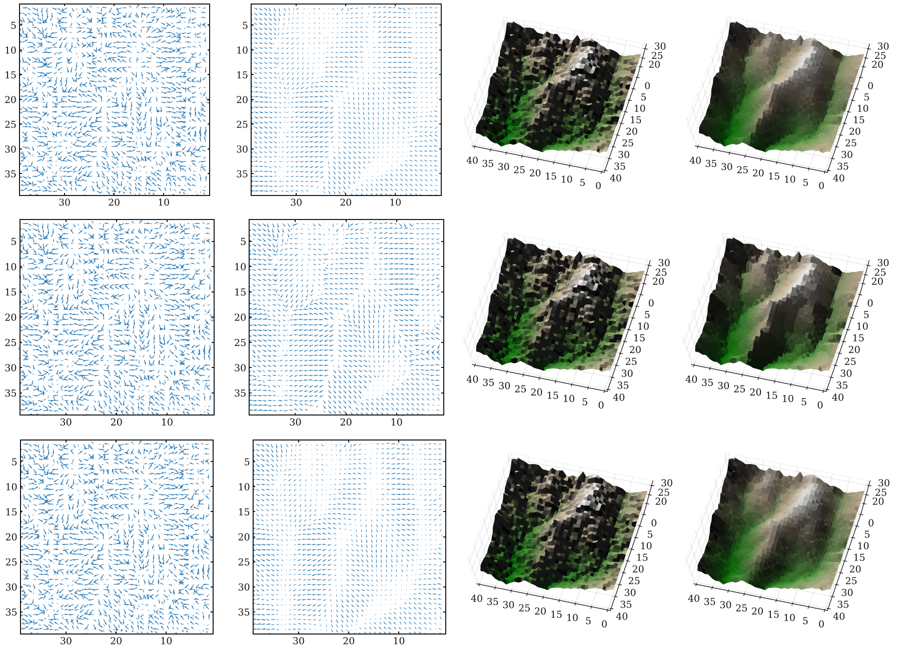


Fig. 3.12 Denoising of $\mathbb{S}^2$-valued surface normals on the digital elevation model (DEM) dataset from [28]: Noisy input (top), total variation ($\lambda = 0.4$) denoised image (second from top), Huber ($\alpha = 0.1$, $\lambda = 0.75$) denoised image (second from bottom), quadratically ($\lambda = 3.0$) denoised image (bottom). Mountain ridges are sharp while hillsides remain smooth with Huber. TV enforces flat hillsides and Tikhonov regularization smoothes out all contours

We applied our approach with TV, Huber and Tikhonov regularization. Interestingly, many of the qualitative properties known from RGB and grayscale image processing appear to transfer to the manifold-valued case: TV enforces piecewise constant areas (flat hillsides), but preserves edges (mountain ridges). Tikhonov regularization gives overall very smooth results, but tends to lose edge information. With Huber regularization, edges (Mountain ridges) remain sharp while hillsides are smooth, and flattening is avoided (Fig. 3.12).

3.3.4 Denoising of High Resolution InSAR Data

While the resolution of the DEM dataset is quite limited (40 × 40 data points), an application to high resolution (432 × 426 data points) Interferometric Synthetic Aperture Radar (InSAR) denoising shows that our model is also applicable in a more demanding scenario (Fig. 3.13).

In InSAR imaging, information about terrain is obtained from satellite or aircraft by measuring the phase difference between the outgoing signal and the incoming

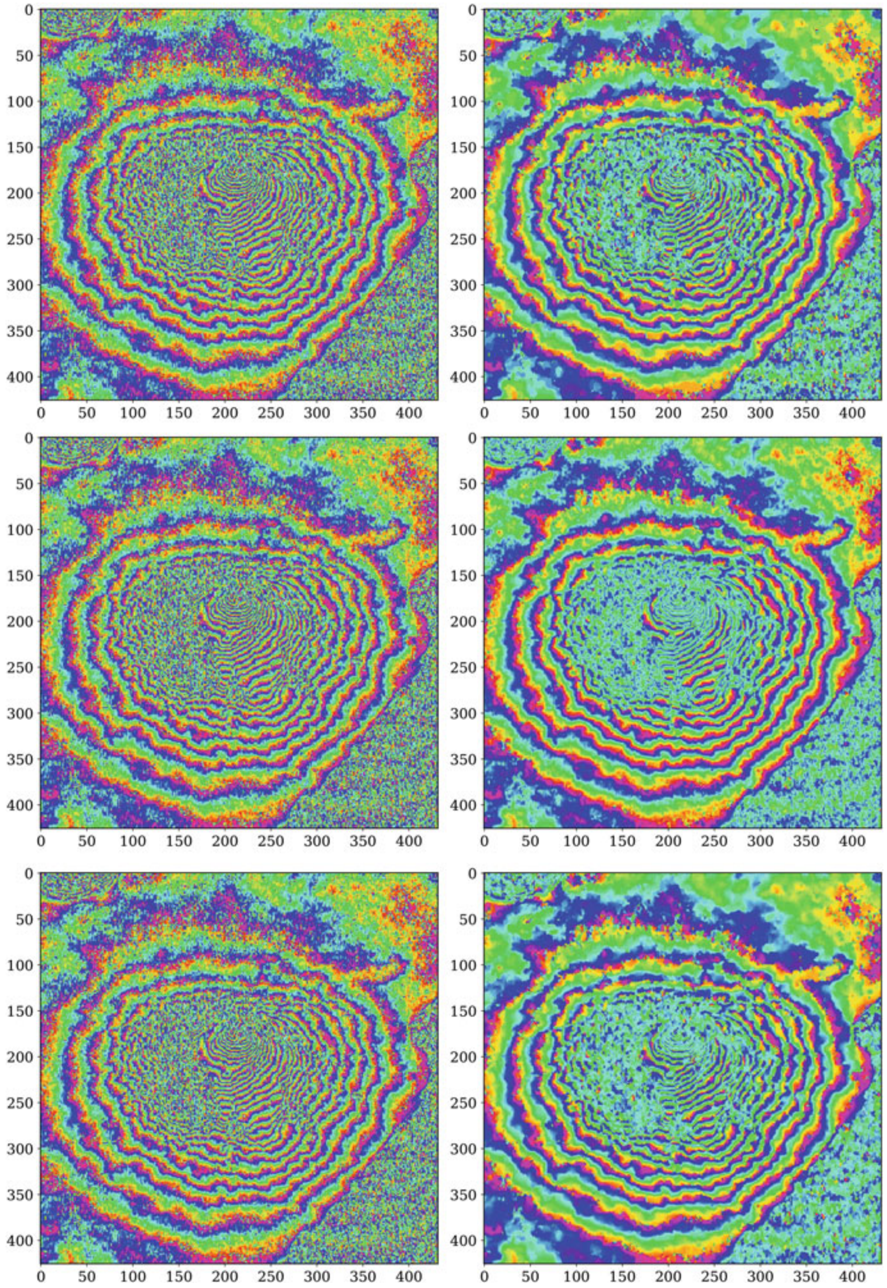


Fig. 3.13 Denoising of $\mathbb{S}^1$-valued InSAR measurements from Mt. Vesuvius, dataset from [57]: Noisy input (top left), total variation ($\lambda = 0.6$) denoised image (top right), Huber ($\alpha = 0.1$, $\lambda = 0.75$) denoised image (bottom left), quadratically ($\lambda = 1.0$) denoised image (bottom right). All regularization strategies successfully remove most of the noise. The total variation regularizer enforces clear contours, but exhibits staircasing effects. The staircasing is removed with Huber while contours are still quite distinct. Quadratic smoothing preserves some of the finer structures, but produces an overall more blurry and less contoured result

reflected signal. This allows a very high relative precision, but no immediate absolute measurements, as all distances are only recovered modulo the wavelength. After normalization to $[0, 2\pi)$, the phase data is correctly viewed as lying on the one-dimensional unit sphere $\mathbb{S}^1$. Therefore, handling the data before any phase unwrapping is performed requires a manifold-valued framework.

Again, denoising with TV, Huber, and Tikhonov regularizations demonstrates properties comparable to those known from scalar-valued image processing while all regularization approaches reduce noise substantially (Fig. 3.13).

3.4 Conclusion and Outlook

We provided an overview and framework for functional lifting techniques for the variational regularization of functions with values in arbitrary Riemannian manifolds. The framework is motivated from the theory of currents and continuous multi-label relaxations, but generalizes these from the context of scalar and vectorial ranges to geometrically more challenging manifold ranges.

Using this approach, it is possible to solve variational problems for manifold-valued images that consist of a possibly non-convex data term and an arbitrary, smooth or non-smooth, convex first-order regularizer, such as Tikhonov, total variation or Huber. A refined discretization based on manifold finite element methods achieves sublabel-accurate results, which allows to use coarser discretization of the range and reduces computational effort compared to previous lifting approaches on manifolds.

A primary limitation of functional lifting methods, which equally applies to manifold-valued models, is dimensionality: The numerical cost increases exponentially with the dimensionality of the manifold due to the required discretization of the range. Addressing this issue appears possible, but will require a significantly improved discretization strategy.

Acknowledgements The authors acknowledge support through DFG grant LE 4064/1-1 “Functional Lifting 2.0: Efficient Convexifications for Imaging and Vision” and NVIDIA Corporation.

References

1. Absil, P.A., Gousenbourger, P.Y., Striewski, P., Wirth, B.: Differentiable piecewise-Bézier surfaces on Riemannian manifolds. SIAM J. Imaging Sci. **9**, 1788–1828 (2016)
2. Absil, P.A., Mahony, R., Sepulchre, R.: Optimization algorithms on matrix manifolds. Princeton University Press, Princeton (2009)
3. Alberti, G., Bouchitté, G., Dal Maso, G.: The calibration method for the Mumford-Shah functional and free-discontinuity problems. Calc. Var. Partial Differ. Equ. **16**(3), 299–333 (2003)
4. Bačák, M.: Convex Analysis and Optimization in Hadamard Spaces. De Gruyter, Berlin (2014)

5. Bačák, M., Bergmann, R., Steidl, G., Weinmann, A.: A second order nonsmooth variational model for restoring manifold-valued images. SIAM J. Sci. Comput. **38**(1), A567–A597 (2016)
6. Bachmann, F., Hielscher, R., Schaeben, H.: Grain detection from 2d and 3d EBSD data – specification of the MTEX algorithm. Ultramicroscopy **111**(12), 1720–1733 (2011)
7. Bae, E., Yuan, J., Tai, X.C., Boykov, Y.: A fast continuous max-flow approach to non-convex multi-labeling problems. In: Bruhn, A., Pock, T., Tai, X.C. (eds.) Efficient Algorithms for Global Optimization Methods in Computer Vision, pp. 134–154. Springer, Berlin (2014)
8. Basser, P.J., Mattiello, J., LeBihan, D.: MR diffusion tensor spectroscopy and imaging. Biophys. J. **66**(1), 259–267 (1994)
9. Baust, M., Weinmann, A., Wieczorek, M., Lasser, T., Storath, M., Navab, N.: Combined tensor fitting and TV regularization in diffusion tensor imaging based on a Riemannian manifold approach. IEEE Trans. Med. Imaging **35**, 1972–1989 (2016)
10. Bergmann, R., Tenbrinck, D.: A graph framework for manifold-valued data. SIAM J. Imaging Sci. **11**, 325–360 (2018)
11. Bergmann, R., Persch, J., Steidl, G.: A parallel Douglas-Rachford algorithm for minimizing ROF-like functionals on images with values in symmetric Hadamard manifolds. SIAM J. Imaging Sci. **9**, 901–937 (2016)
12. Bergmann, R., Fitschen, J.H., Persch, J., Steidl, G.: Priors with coupled first and second order differences for manifold-valued image processing. J. Math. Imaging Vis. **60**, 1459–1481 (2018)
13. Bergmann, R., Laus, F., Persch, J., Steidl, G.: Recent advances in denoising of manifold-valued images. Technical Report (2018). arXiv:1812.08540
14. Bernard, F., Schmidt, F.R., Thunberg, J., Cremers, D.: A combinatorial solution to non-rigid 3D shape-to-image matching. In: Proceedings of ICCV 2017, pp. 1436–1445 (2017)
15. Bouchitté, G., Fragalà, I.: A duality theory for non-convex problems in the calculus of variations. Arch. Ration. Mech. Anal. **229**(1), 361–415 (2018)
16. Bredies, K., Holler, M., Storath, M., Weinmann, A.: Total generalized variation for manifold-valued data. SIAM J. Imaging Sci. **11**, 1785–1848 (2018)
17. Călinescu, G., Karloff, H., Rabani, Y.: An improved approximation algorithm for multiway cut. In: Proceedings of STOC 1998, pp. 48–52 (1998)
18. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. J Math. Imaging Vis. **40**(1), 120–145 (2011)
19. Chambolle, A., Cremers, D., Pock, T.: A convex approach to minimal partitions. SIAM J. Imaging Sci. **5**(4), 1113–1158 (2012)
20. Chan, T.F., Kang, S.H., Shen, J.: Total variation denoising and enhancement of color images based on the CB and HSV color models. J. Vis. Commun. Image Represent. **12**, 422–435 (2001)
21. Chan, T.F., Esedoglu, S., Nikolova, M.: Algorithms for finding global minimizers of image segmentation and denoising models. SIAM J. Appl. Math. **66**, 1632–1648 (2006)
22. Chefd'Hotel, C., Tschumperlé, D., Deriche, R., Faugeras, O.D.: Regularizing flows for constrained matrix-valued images. J. Math. Imaging Vis. **20**, 147–162 (2004)
23. Cremers, D., Strekalovskiy, E.: Total cyclic variation and generalizations. J. Math. Imaging Vis. **47**, 258–277 (2012)
24. Delaunoy, A., Fundana, K., Prados, E., Heyden, A.: Convex multi-region segmentation on manifolds. In: Proceedings of ICCV 2009, pp. 662–669 (2009)
25. Dziuk, G., Elliott, C.M.: Finite element methods for surface PDEs. Acta Numer. **22**, 289–396 (2013)
26. Federer, H.: Real flat chains, cochains and variational problems. Indiana Univ. Math. J. **24**, 351–407 (1974)
27. Fletcher, P.T.: Geodesic regression and the theory of least squares on Riemannian manifolds. Int. J. Comput. Vis. **105**, 171–185 (2012)
28. Gesch, D., Evans, G., Mauck, J., Hutchinson, J., Carswell Jr., W.J., et al.: The national map – elevation. US geological survey fact sheet 3053(4) (2009)
29. Giaquinta, M., Modica, G., Souček, J.: Cartesian Currents in the Calculus of Variations I and II. Springer, Berlin (1998)

30. Goldlücke, B., Cremers, D.: Convex relaxation for multilabel problems with product label spaces. In: Proceedings of ECCV 2010, pp. 225–238 (2010)
31. Goldlücke, B., Strekalovskiy, E., Cremers, D.: Tight convex relaxations for vector-valued labeling. SIAM J. Imaging Sci. **6**, 1626–1664 (2013)
32. Goldstein, T., Esser, E., Baraniuk, R.: Adaptive primal dual optimization for image processing and learning. In: Proceedings of 6th NIPS Workshop on Optimization for Machine Learning, pp. 1–5 (2013)
33. Greig, D.M., Porteous, B.T., Seheult, A.H.: Exact maximum a posteriori estimation for binary images. J. R. Stat. Soc. B Stat. Methodol. **51**(2), 271–279 (1989)
34. Ishikawa, H.: Exact optimization for Markov random fields with convex priors. IEEE Trans. Pattern Anal. Mach. Intell. **25**, 1333–1336 (2003)
35. Karcher, H.: Riemannian center of mass and mollifier smoothing. Commun. Pure Appl. Math. **30**, 509–541 (1977)
36. Kleinberg, J.M., Tardos, É.: Approximation algorithms for classification problems with pairwise relationships: metric labeling and Markov random fields. J. ACM **49**, 616–639 (2002)
37. Klodt, M., Schoenemann, T., Kolev, K., Schikora, M., Cremers, D.: An experimental comparison of discrete and continuous shape optimization methods. In: Proceedings of ECCV 2008, pp. 332–345 (2008)
38. Laude, E., Möllenhoff, T., Moeller, M., Lellmann, J., Cremers, D.: Sublabel-accurate convex relaxation of vectorial multilabel energies. In: Proceedings of ECCV 2016, pp. 614–627 (2016)
39. Laus, F., Persch, J., Steidl, G.: A nonlocal denoising algorithm for manifold-valued images using second order statistics. SIAM J. Imaging Sci. **10**, 416–448 (2017)
40. Lavenant, H.: Harmonic mappings valued in the Wasserstein space. Technical Report (2017). arXiv:1712.07528
41. Lee, J.M.: Introduction to Smooth Manifolds, vol. 218, 2nd revised edn. Springer, New York (2013)
42. Lellmann, J.: Nonsmooth convex variational approaches to image analysis. Ph.D. thesis, Ruprecht-Karls-Universität Heidelberg (2011)
43. Lellmann, J., Schnörr, C.: Continuous multiclass labeling approaches and algorithms. SIAM J. Imaging Sci. **4**(4), 1049–1096 (2011)
44. Lellmann, J., Becker, F., Schnörr, C.: Convex optimization for multi-class image labeling with a novel family of total variation based regularizers. In: Proceedings of ICCV 2009, pp. 646–653 (2009)
45. Lellmann, J., Lenzen, F., Schnörr, C.: Optimality bounds for a variational relaxation of the image partitioning problem. J. Math. Imaging Vis. **47**, 239–257 (2012)
46. Lellmann, J., Lellmann, B., Widmann, F., Schnörr, C.: Discrete and continuous models for partitioning problems. Int J. Comput. Vis. **104**(3), 241–269 (2013)
47. Lellmann, J., Strekalovskiy, E., Koetter, S., Cremers, D.: Total variation regularization for functions with values in a manifold. In: Proceedings of ICCV 2013, pp. 2944–2951 (2013)
48. Loewenhauser, B., Lellmann, J.: Functional lifting for variational problems with higher-order regularization. In: Tai, X.C., Bae, E., Lysaker, M. (eds.) Imaging, Vision and Learning Based on Optimization and PDEs, pp. 101–120. Springer International Publishing, Cham (2018)
49. Massonnet, D., Feigl, K.L.: Radar interferometry and its application to changes in the Earth's surface. Rev. Geophys. **36**(4), 441–500 (1998)
50. Möllenhoff, T., Cremers, D.: Sublabel-accurate discretization of nonconvex free-discontinuity problems. In: Proceedings of ICCV 2017, pp. 1192–1200 (2017)
51. Möllenhoff, T., Cremers, D.: Lifting vectorial variational problems: a natural formulation based on geometric measure theory and discrete exterior calculus. In: Proceedings of CVPR 2019 (2019)
52. Möllenhoff, T., Laude, E., Moeller, M., Lellmann, J., Cremers, D.: Sublabel-accurate relaxation of nonconvex energies. In: Proceedings of CVPR 2016 (2016)
53. Pock, T., Schoenemann, T., Graber, G., Bischof, H., Cremers, D.: A convex formulation of continuous multi-label problems. In: Proceedings of ECCV 2008, pp. 792–805 (2008)

54. Pock, T., Cremers, D., Bischof, H., Chambolle, A.: An algorithm for minimizing the Mumford-Shah functional. Proceedings of ICCV 2009, pp. 1133–1140 (2009)
55. Pock, T., Cremers, D., Bischof, H., Chambolle, A.: Global solutions of variational models with convex regularization. SIAM J. Imaging Sci. **3**(4), 1122–1145 (2010)
56. Ranftl, R., Pock, T., Bischof, H.: Minimizing TGV-based variational models with non-convex data terms. In: Kuijper, A., Bredies, K., Pock, T., Bischof, H. (eds.) Proceedings of SSVM 2013, pp. 282–293. Springer, Berlin (2013)
57. Rocca, F., Prati, C., Ferretti, A.: An overview of SAR interferometry. In: Proceedings of 3rd ERS Symposium on Space at the Service of Our Environment (1997). http://earth.esa.int/workshops/ers97/program-details/speeches/rocca-et-al
58. Rosman, G., Bronstein, M.M., Bronstein, A.M., Wolf, A., Kimmel, R.: Group-valued regularization framework for motion segmentation of dynamic non-rigid shapes. In: Bruckstein, A.M., ter Haar Romeny, B.M., Bronstein, A.M., Bronstein, M.M. (eds.) Proceedings of SSVM 2011, pp. 725–736. Springer, Berlin (2012)
59. Storath, M., Weinmann, A.: Wavelet sparse regularization for manifold-valued data. Technical Report (2018). arXiv:1808.00505
60. Strecke, M., Goldluecke, B.: Sublabel-accurate convex relaxation with total generalized variation regularization. In: Brox, T., Bruhn, A., Fritz, M. (eds.) Proceedings of GCPR 2018, pp. 263–277. Springer International Publishing, Cham (2019)
61. Strekalovskiy, E.: Convex relaxation of variational models with applications in image analysis. Ph.D. thesis, Technische Universität München (2015)
62. Strekalovskiy, E., Cremers, D.: Total variation for cyclic structures: convex relaxation and efficient minimization. In: Proceedings of CVPR 2011, pp. 1905–1911 (2011)
63. Strekalovskiy, E., Goldlücke, B., Cremers, D.: Tight convex relaxations for vector-valued labeling problems. In: Proceedings of ICCV 2011, pp. 2328–2335 (2011)
64. Strekalovskiy, E., Nieuwenhuis, C., Cremers, D.: Nonmetric priors for continuous multilabel optimization. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) Proceedings of ECCV 2012, pp. 208–221. Springer, Berlin (2012)
65. Vogt, T., Lellmann, J.: Measure-valued variational models with applications to diffusion-weighted imaging. J. Math. Imaging Vis. **60**, 1482–1502 (2018)
66. Vogt, T., Lellmann, J.: Functional liftings of vectorial variational problems with Laplacian regularization. In: Burger, M., Lellmann, J., Modersitzki, J. (eds.) Proceedings of SSVM 2019, pp. 559–571 (2019)
67. Weinmann, A., Demaret, L., Storath, M.: Total variation regularization for manifold-valued data. SIAM J. Imaging Sci. **7**, 2226–2257 (2014)
68. Weinmann, A., Demaret, L., Storath, M.: Mumford-Shah and Potts regularization for manifold-valued data. J. Math. Imaging Vis. **55**, 428–445 (2015)
69. Windheuser, T., Cremers, D.: A convex solution to spatially-regularized correspondence problems. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) Proceedings of ECCV 2016, pp. 853–868. Springer International Publishing, Cham (2016)
70. Zach, C., Kohli, P.: A convex discrete-continuous approach for Markov random fields. In: Proceedings of ECCV 2012, pp. 386–399 (2012)
71. Zach, C., Gallup, D., Frahm, J.M., Niethammer, M.: Fast global labeling for real-time stereo using multiple plane sweeps. In: Proceedings of VMV 2008, pp. 243–252 (2008)

Chapter 4
Geometric Subdivision and Multiscale Transforms

Johannes Wallner

Contents

Abstract Any procedure applied to data, and any quantity derived from data, is required to respect the nature and symmetries of the data. This axiom applies to refinement procedures and multiresolution transforms as well as to more basic operations like averages. This chapter discusses different kinds of geometric structures like metric spaces, Riemannian manifolds, and groups, and in what way we can make elementary operations geometrically meaningful. A nice example of this is the Riemannian metric naturally associated with the space of positive definite matrices and the intrinsic operations on positive definite matrices derived from it. We discuss averages first and then proceed to refinement operations (subdivision)

J. Wallner (✉)
TU Graz, Graz, Austria
e-mail: j.wallner@tugraz.at

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_4

and multiscale transforms. In particular, we report on the current knowledge as regards convergence and smoothness.

4.1 Computing Averages in Nonlinear Geometries

The line of research presented in this chapter was first suggested by a 2001 presentation by D. Donoho on multiscale representations of discrete data [10]. A subsequent Ph.D. thesis and accompanying publication appeared a few years later [48]. Multiscale representations are intimately connected with refinement procedures (prediction operators). These are in themselves an interesting topic with applications, e.g. in computer graphics. Iterative refinement a.k.a. subdivision in turn is based on the notion of *average*. Consequently this chapter is structured into the following parts: Firstly a discussion of averages, in particular averages in metric spaces and in manifolds. Secondly, subdivision rules and the limits generated by them. Thirdly, multiresolution representations.

We start with the *affine average* w.r.t. weights a_j of data points x_j contained in a vector space. It is defined by

$$x = \operatorname{avg}_{j\in\mathbb{Z}}(a_j, x_j) := \sum a_j x_j, \quad \text{where} \quad \sum a_j = 1. \tag{4.1}$$

In this chapter we stick to finite averages, but we allow negative coefficients. For data whose geometry is not that of a vector space, but that of a surface contained in some Euclidean space, or that of a group, or that of a Riemannian manifold, this affine average often does not make sense. In any case it is not natural. Examples of such data are, for instance, unit vectors, positions of a rigid body in space, or the 3 by 3 symmetric positive definite matrices which occur in diffusion-tensor MRI. In the following paragraphs we show how to extend the notation of affine average to nonlinear situations in a systematic way. We start by pointing out equivalent characterizations of the affine average:

$$\begin{aligned} x = \operatorname{avg}(a_j, x_j) &\iff x \text{ solves } \textstyle\sum a_j(x_j - x) = 0 && (4.2)\\ &\iff x = y + \textstyle\sum a_j(x_j - y) \text{ for any } y && (4.3)\\ &\iff x \text{ minimizes } \textstyle\sum a_j \|x - x_j\|^2. && (4.4)\end{aligned}$$

4.1.1 The Fréchet Mean

Each of (4.2)–(4.4) has been used to generalize the notion of weighted average to nonlinear geometries. Some of these generalizations are conceptually straightforward. For example, Eq. (4.4) has an analogue in any metric space $(\mathcal{M}, d_{\mathcal{M}})$, namely the weighted *Fréchet mean* defined by

$$\operatorname{avg}_F(a_j, x_j) := \arg\min_{x \in \mathcal{M}} \sum a_j \, d_{\mathcal{M}}(x, x_j)^2. \tag{4.5}$$

It is a classical result that in case of nonnegative weights, the Fréchet mean exists and is unique, if $\mathcal{M}$ is a Hadamard metric space. This property means $\mathcal{M}$ is complete, midpoints exist uniquely, and triangles are slim, cf. [1].[1]

The Fréchet Mean in Riemannian Manifolds

In a surface resp. Riemannian manifold $\mathcal{M}$, the Fréchet mean locally exists uniquely. A main reference here is the paper [39] by H. Karcher. He considered the more general situation that μ is a probability measure on $\mathcal{M}$, where the mean is defined by

$$\operatorname{avg}_F(\mu) = \arg\min_{x \in \mathcal{M}} \int d_{\mathcal{M}}(x, \cdot)^2 d\mu.$$

In this chapter we stick to the elementary case of finite averages with possibly negative weights. The Fréchet mean exists uniquely if the manifold is Hadamard—this property is usually called “Cartan-Hadamard” and is characterized by completeness, simple connectedness, and nonpositive sectional curvature. For unique existence of avg_F, we do not even have to require that weights are nonnegative [37, Th. 6].

The Fréchet Mean in the Non-unique Case

If the Cartan-Hadamard property is not fulfilled, the Fréchet mean does not have to exist at all, e.g. if the manifold is not complete (cutting a hole in $\mathcal{M}$ exactly where the mean should be makes it nonexistent). If the manifold is complete, the mean exists, but possibly is not unique.

If $\mathcal{M}$ is complete with nonpositive sectional curvature, but is not simply connected, there are situations where a unique Fréchet mean of given data points can still be defined, e.g. if the data are connected by a path $c\colon [a, b] \to \mathcal{M}$ with $c(t_j) = x_j$. This will be the case e.g. if data represent a time series. Existence or maybe even canonical existence of such a path depends on the particular application. We then consider the simply connected covering $\widetilde{\mathcal{M}}$, find a lifting $\widetilde{c}\colon I \to \widetilde{\mathcal{M}}$ of c, compute the Fréchet mean $\operatorname{avg}_F(a_j, \widetilde{c}(t_j))$, and project it back to $\mathcal{M}$. This average does not only depend on the data points and the weights, but also on the homotopy class of c. In fact instead of a path, any mapping $c\colon I \to \mathcal{M}$ can be used for such purposes as long as its domain I is simply connected [37].

[1] More precisely, for all $a, b \in \mathcal{M}$ there is a unique midpoint $x = m(a, b)$ defined by $d_{\mathcal{M}}(x, a) = d_{\mathcal{M}}(x, b) = d_{\mathcal{M}}(a, b)/2$, and for any $a, b, c \in \mathcal{M}$ and points $a', b', c' \in \mathbb{R}^2$ which have the same pairwise distances as a, b, c, the inequality $d_{\mathcal{M}}(c, m(a, b)) \le d_{\mathbb{R}^2}(c', m(a', b'))$ holds.

Finally, if $\mathcal{M}$ is complete but has positive sectional curvatures, a unique Fréchet mean is only defined locally. The size of neighbourhoods where uniqueness happens has been discussed by [15, 16, 34]. This work plays a role in investigating convergence of subdivision rules in Riemannian manifolds, see Sect. 4.2.4.

4.1.2 The Exponential Mapping

From the different expressions for the affine average, (4.2) and (4.3) seem to be specific to linear spaces, because they involve the $+$ and $-$ operations. However, it turns out that there is a big class of nonlinear geometries where natural analogues $\oplus$ and $\ominus$ of these operations exist, namely the exponential mapping and its inverse. We discuss this construction in surfaces resp. Riemannian manifolds, in groups, and in symmetric spaces.

The Exponential Mapping in Riemannian Geometry

In a Riemannian manifold $\mathcal{M}$, for any $p \in \mathcal{M}$ and tangent vector $v \in T_p\mathcal{M}$, the point $\exp_p(v)$ is the endpoint of the geodesic curve $c(t)$ which starts in p, has initial tangent vector v, and whose length equals $\|v\|$. We let

$$p \oplus v := \exp_p(v), \qquad\qquad q \ominus p := \exp_p^{-1}(q).$$

One property of the exponential mapping is the fact that curves of the form $t \mapsto p \oplus tv$ are shortest paths with initial tangent vector v. The mapping $v \mapsto p \oplus v$ is a diffeomorphism locally around $v = 0$. Its differential equals the identity.

Properties of the Riemannian Exponential Mapping

For complete Riemannian manifolds, $p \oplus v$ is always well defined. Also $q \ominus p$ exists by the Hopf-Rinow theorem, but it does not have to be unique. Uniqueness happens if $d_{\mathcal{M}}(p, q)$ does not exceed the *injectivity radius* $\rho_{\mathrm{inj}}(p)$ of p. In Cartan-Hadamard manifolds, injectivity radii are infinite and the exponential mapping does not decrease distances, i.e., $d_{\mathcal{M}}(p \oplus v, p \oplus w) \geq \|v - w\|_{T_p\mathcal{M}}$. The injectivity radius can be small for topological reasons (e.g. a cylinder of small radius which is intrinsically flat, can have arbitrarily small injectivity radius), but even in the simply connected case, one cannot expect ρ_{inj} to exceed $\pi K^{-1/2}$, if K is a positive upper bound for sectional curvatures.

Further, the $\ominus$ operation and the Riemannian distance are related by

$$\nabla d_{\mathcal{M}}(\cdot, a)(x) = -\frac{a \ominus x}{\|a \ominus x\|}, \qquad \nabla d_{\mathcal{M}}{}^2(\cdot, a)(x) = -2(a \ominus x), \tag{4.6}$$

if $v = a \ominus x$ refers to the smallest solution v of $x \oplus a = v$. For more properties of the exponential mapping we refer to [39] and to differential geometry textbooks like [8].

The Exponential Mapping in Groups

In Lie groups, which we describe only in the case of a matrix group G, a canonical exponential mapping is defined: With the notation $\mathfrak{g} = T_e G$ for the tangent space in the identity element, we let

$$v \in \mathfrak{g} \implies e \oplus v = \exp(v) = \sum\nolimits_{k\geq 0} \frac{1}{k!} v^k.$$

The curve $t \mapsto e \oplus tv$ is the unique one-parameter subgroup of G whose tangent vector at $t = 0$ is the vector $v \in \mathfrak{g}$. Again, $v \mapsto e \oplus v$ is locally a diffeomorphism whose differential is the identity mapping.

An inverse *log* of *exp* is defined locally around e. Transferring the definition of $\oplus$ to the entire group by left translation, the defining relation $g \oplus gv := g(e \oplus v)$ yields

$$p \oplus v = p \exp(p^{-1}v), \qquad\qquad q \ominus p = p \log(p^{-1}q).$$

Addition is always globally well defined, but the difference $q \ominus p$ might not exist. For example, in GL_n, the mapping $v \mapsto e \oplus v$ is not onto. The difference exists always, but not uniquely, in compact groups. See e.g. [2].

The Exponential Mapping in Symmetric Spaces

Symmetric spaces have the form G/H, where H is a Lie subgroup of G. There are several definitions which are not entirely equivalent. We use the one that the tangent spaces $\mathfrak{g} = T_e G$, $\mathfrak{h} = T_e H$ obey the condition that $\mathfrak{h}$ is the $+1$ eigenspace of an involutive Lie algebra automorphism σ of $\mathfrak{g}$.[2] The tangent space $\mathfrak{g}/\mathfrak{h}$ of G/H in the point $eH \in G/H$ is naturally identified with the -1 eigenspace $\mathfrak{s}$ of the involution, and is transported to all points of G/H by left translation. The exponential mapping in G is projected onto G/H in the canonical way and yields the exponential mapping in the symmetric space.

We do not go into more details but refer to the comprehensive classic [35] instead. Many examples of well-known manifolds fall into this category, e.g. the sphere S^n, hyperbolic space H^n, and the Grassmannians. We give an important example:

Example 4.1 (The Riemannian Symmetric Space of Positive-Definite Matrices) The space Pos_n of positive definite $n \times n$ matrices is made a metric space by letting

[2] i.e., σ obeys the law $\sigma([v, w]) = [\sigma(v), \sigma(w)]$, where in the matrix group case, the Lie bracket operation is given by $[v, w] = vw - wv$.

$$d(a,b) = \|\log(a^{-1/2}ba^{-1/2})\|_2 = \Big(\sum\nolimits_{\lambda_1,\dots,\lambda_n \in \sigma(a^{-1}b)} \log^2 \lambda_j\Big)^{1/2}. \tag{4.7}$$

Here $\|\cdot\|_2$ means the Frobenius norm, and $\sigma(m)$ means the eigenvalues of a matrix.

The metric (4.7) is actually that of a Riemannian manifold. Pos_n, as an open subset of the set Sym_n of symmetric matrices, in each point has a tangent space $T_a\mathrm{Pos}_n$ canonically isomorphic to Sym_n as a linear space. The Riemannian metric in this space is defined by $\|v\| = \|a^{-1/2}va^{-1/2}\|_2$.

Pos_n is also a symmetric space: We know that any $g \in \mathrm{GL}_n$ can be uniquely written as a product $g = au$, with $a = \sqrt{gg^T} \in \mathrm{Pos}_n$ and $u \in \mathrm{O}_n$. Thus $\mathrm{Pos}_n = G/H$, with $G = \mathrm{GL}_n$, $H = \mathrm{O}_n$, and the canonical projection $\pi(x) = \sqrt{xx^T}$.

The respective tangent spaces $\mathfrak{g}$, $\mathfrak{h}$ of G, H are given by $\mathfrak{g} = \mathbb{R}^{n\times n}$ and $\mathfrak{h} = \mathfrak{so}_n$, which is the set of skew-symmetric $n \times n$ matrices. The involution $\sigma(x) = -x^T$ in $\mathfrak{g}$ obeys $[\sigma(v), \sigma(w)] = \sigma([v,w])$, and $\mathfrak{h}$ is its $+1$ eigenspace. We have thus recognized Pos_n as a symmetric space. It turns out that $a \oplus v = a\exp(a^{-1}v)$, where *exp* is the matrix exponential function.

The previous paragraphs define two different structures on Pos_n, namely that of a Riemannian manifold, and that of a symmetric space. They are compatible in the sense that the $\oplus$, $\ominus$ operations derived from either structure coincide. For more information we refer to [24, 40, 58]. Subdivision in particular is treated by [38].

4.1.3 Averages Defined in Terms of the Exponential Mapping

If $\oplus$ and $\ominus$ are defined as discussed in the previous paragraphs, it is possible to define a weighted affine average implicitly by requiring that

$$x = \mathrm{avg}_E(a_j, x_j) :\Longleftrightarrow \sum a_j(x_j \ominus x) = 0. \tag{4.8}$$

Any Fréchet mean in a Riemannian manifold is also an average in this sense, which follows directly from (4.5) together with (4.6). Locally, avg_E is well defined and unique. As to the size of neighbourhoods where this happens, in the Riemannian case the proof given by [15, 16] for certain neighbourhoods enjoying unique existence of avg_F shows that the very same neighbourhoods also enjoy unique existence of avg_E.

Affine Averages with Respect to a Base Point

From the different expressions originally given for the affine average, $x = y + \sum a_j(x_j - y)$ is one we have not yet defined a manifold analogue for. With $\ominus$ and $\oplus$ at our disposal, this can be done by

$$\operatorname{avg}_y(a_j; x_j) := y \oplus \sum a_j(x_j \ominus y). \tag{4.9}$$

We call this the log/exp average with respect to the base point y. It has the disadvantage of a dependence on the base point, but for the applications we have in mind, there frequently is a natural choice of base point. Its advantages lie in the easier analysis compared to the Fréchet mean. One should also appreciate that the Fréchet mean is a log/exp mean w.r.t. to a basepoint, if that basepoint is the Fréchet mean itself:

$$y = \operatorname{avg}_F(a_j; x_j) \implies \operatorname{avg}_y(a_j; x_j) = y \oplus \sum a_j(x_j \ominus y) = y \oplus 0 = y, \tag{4.10}$$

because of (4.8). This may be a trivial point, but it has been essential in proving smoothness of limit curves for manifold-based subdivision processes (see Theorem 4.11 and [29]).

The possibility to define averages w.r.t. basepoints rests on the possibility of defining $\ominus$, which has been discussed above.

4.2 Subdivision

4.2.1 *Defining Stationary Subdivision*

Subdivision is a refinement process acting on input data lying in some set $\mathcal{M}$, which in the simplest case are indexed over the integers and are interpreted as samples of a function $f\colon \mathbb{R} \to \mathcal{M}$. A subdivision rule *refines* the input data, producing a sequence Sp which is thought of as denser samples of either f itself, or of a function approximating f.

One mostly considers binary rules, whose application "doubles" the number of data points. The *dilation factor* of the rule, generally denoted by the letter N, then equals 2. We require that the subdivision rule is invariant w.r.t. index shift, which by means of the left shift operator L can be formalized as

$$L^N S = SL.$$

We require that each point Sp_i depends only on finitely many data points p_j. Together with shift invariance this means that there is $s > 0$ such that p_i influences only $Sp_{Ni-s}, \dots, Sp_{Ni+s}$.

Subdivision rules are to be iterated: We create finer and finer data

$$p,\ Sp,\ S^2p,\ S^3p,\ \dots,$$

which we hope approach a continuous limit (the proper definition of which is given below).

Subdivision was invented by G. de Rham [6], who considered the process of iteratively cutting corners from a convex polygon contained in $\mathcal{M} = \mathbb{R}^2$, and asked for the limit shape. If cutting corners is done by replacing each edge $p_i p_{i+1}$ by the shorter edge with vertices $Sp_{2i} = (1-t)p_i + tp_{i+1}$, $Sp_{2i+1} = tp_i + (1-t)p_{i+1}$, this amounts to a subdivision rule. In de Rham's example, only two data points p_i contribute to any individual Sp_j.

Primal and Dual Subdivision Rules The corner-cutting rules mentioned above are invariant w.r.t. reordering indices according to $\dots, 0 \mapsto 1, 1 \mapsto 0, 2 \mapsto -1, \dots$. With inversion U defined by $(Up)_i = p_{-i}$ we can write this invariance as $(LU)S = S(LU)$. An even simpler kind of symmetry is enjoyed by subdivision rules with obey $US = SU$. The latter are called primal rules, the former dual ones. The reason why we emphasize these properties is that they give guidance for finding manifold analogues of linear subdivision rules.

Subdivision of Multivariate Data It is not difficult to generalize the concept of subdivision to multivariate data $p\colon \mathbb{Z}^s \to \mathcal{M}$ indexed over the grid $\mathbb{Z}^s$. A subdivision rule S must fulfill $L_v^N S = SL_v$, for all shifts L_v w.r.t. a vector $v \in \mathbb{Z}^s$.

Data with combinatorial singularities have to be treated separately, cf. Sect. 4.3.4. Here basically only the bivariate case is studied, but this has been done extensively, mostly because of applications in Computer Graphics [43].

Linear Subdivision Rules and Their Nonlinear Analogues

A linear subdivision rule acting on data $p\colon \mathbb{Z}^2 \to \mathbb{R}^d$ has the form

$$Sp_i = \sum\nolimits_j a_{i-Nj} p_j.$$

If the sum $\sum_j a_{i-Nj}$ of coefficients contributing to Sp_i equals 1, the application of the rule amounts to computing a weighted average:

$$Sp_i = \operatorname{avg}(a_{i-Nj}; p_j). \tag{4.11}$$

Subdivision rules not expressible in this way might occur as auxiliary tools in proofs, but are not meant to be applied to data which are *points* of an affine space. This is because if $\sum a_{i-Nj} \neq 1$, then the linear combination $\sum a_{i-Nj} p_j$ is not translation-invariant, and the rule depends on the choice of origin of the coordinate system.

Besides, the iterated application of rules not expressible as weighted averages either leads to divergent data $S^k p$, or alternatively, to data approaching zero. For this reason, one exclusively considers linear rules of the form (4.11). A common definition of *convergent* subdivision rule discounts the case of zero limits and recognizes translation invariance as a necessary condition for convergence, cf. [17].

For a thorough treatment of linear subdivision rules, conveniently done via S acting as a linear operator in $\ell^\infty(\mathbb{Z}^s, \mathbb{R})$ and using the appropriate tools of approximation theory, see, e.g. [4].

In the following we discuss some nonlinear, geometric, versions of subdivision rules. We use the various nonlinear versions of averages introduced above, starting with the Fréchet mean in metric spaces.

- *Subdivision using the Fréchet mean.* A natural analogue of (4.11) is found by replacing the affine average by the Fréchet mean. This procedure is particularly suited for Hadamard metric spaces and also in complete Riemannian manifolds.
- *Log/exp subdivision.* In a manifold equipped with an exponential mapping, an analogue of (4.11) is defined by

$$Tp_i = \mathrm{avg}_{m_i}(a_{i-Nj}; p_j),$$

  where m_i is a base point computed in a meaningful manner from the input data, e.g. $m_i = p_{\lfloor i/N \rfloor}$. In case of combinatorial symmetries of the subdivision rule, it makes sense to make the choice of m_i conform to these symmetries.
- *Subdivision using projections.* If $\mathcal{M}$ is a surface embedded in a vector space and π is a projection onto $\mathcal{M}$, we might use the subdivision rule

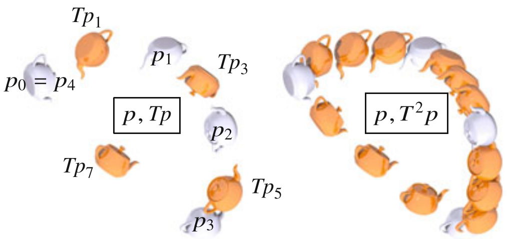
$$Tp_i = \pi(Sp_i).$$

  If the intrinsic symmetries of $\mathcal{M}$ extend to symmetries of ambient space, then this *projection analogue* of a linear subdivision rule is even intrinsic—see Example 4.2.

Example 4.2 (Subdivision in the Motion Group) The groups O_n and SO_n are $\frac{1}{2}n(n-1)$-dimensional surfaces in the linear space $\mathbb{R}^{n\times n}$. A projection onto O_n is furnished by singular value decomposition, or in an alternate way of expressing it, by the polar decomposition of Example 4.1:

$$\pi : \mathrm{GL}_n \to \mathrm{O}_n,\ \pi(g) = (gg^T)^{-1/2}g.$$

This projection is O_n-equivariant in the sense that for $u \in \mathrm{O}_n$, we have both $\pi(ug) = u\pi(g)$ and $\pi(gu) = \pi(g)u$. The same invariance applies to application of a linear subdivision rule acting in $\mathbb{R}^{n\times n}$. So for any given data in O_n, and a linear subdivision rule S, the subdivision rule $\pi \circ S$ produces data in O_n in a geometrically meaningful way, as long as we do not exceed the bounds of GL_n. Since GL_n is a rather big neighbourhood of O_n, this is in practice no restriction. Figure 4.1 shows an example.



$$Sp_{2i} = p_i$$
$$Sp_{2i+1} = \frac{9}{16}(p_i + p_{i+1}) - \frac{1}{16}(p_{i-1} + p_{i+2})$$
$$T = \pi \circ S$$

Fig. 4.1 Subdivision by projection in the motion group $\mathbb{R}^3 \rtimes O_3$. A 4-periodic sequence $p_i = (c_i, u_i)$ of positions of a rigid body is defined by the center of mass c_i, and an orientation $u_i \in O_3$. Both components undergo subdivision w.r.t. the interpolatory four-point rule S, where the matrix part is subsequently projected back onto O_3 in an invariant manner

4.2.2 Convergence of Subdivision Processes

Definition of Convergence

When discrete data p are interpreted as samples of a function, then refined data Sp, S^2p etc. are interpreted as the result of sampling which is N times, N^2 times etc. as dense as the original. We therefore define a convergent refinement rule as follows.

Definition 4.3 Discrete data $S^kp\colon \mathbb{Z}^s \to \mathcal{M}$ at the k-th iteration of refinement determine a function $f_k\colon N^{-k}\mathbb{Z}^s \to \mathcal{M}$, whose values are the given data points: For any N-adic point ξ, we have $(S^kp)_{N^k\xi} = f_k(\xi)$, provided $N^k\xi$ is an integer. For all such ξ, the sequence $(f_k(\xi))_{k\geq 0}$ is eventually defined and we let $f(\xi) = \lim_{k\to\infty} f_k(\xi)$. We say S is convergent for input data p, if the limit function f exists for all ξ and is continuous. It can be uniquely extended to a continuous function $S^\infty p\colon \mathbb{R}^s \to \mathcal{M}$.

Another way of defining the limit is possible if data $p_i, Sp_i, \dots$ lie in a vector space. We linearly interpolate them by functions $g_0, g_1, \dots$ with $g_k(N^{-k}i) = S^kp_i$. Then the limit of functions g_k agrees with the limit of Definition 4.3 (which is pointwise, but in fact convergence is usually uniform on compact sets.)

The following lemma is the basis for investigating convergence of subdivision rules in metric spaces. The terminology is that of [19, 20].

Lemma 4.4 *Let $\mathcal{M}$ be a complete metric space, and let the subdivision rule S operate with dilation N on data $p\colon \mathbb{Z}^s \to \mathcal{M}$. We measure the density of the data by*

$$\delta(p) = \sup_{|i-j|\leq 1} d_{\mathcal{M}}(p_i, p_j),$$

where we use the 1-norm on the indices. S is contractive, resp. displacement-safe, if

$$\delta(Sp) \leq \gamma\delta(p), \quad \textit{for some } \gamma < 1, \quad \textit{resp.} \quad \sup\nolimits_{i\in\mathbb{Z}^s} d_{\mathcal{M}}(Sp_{Ni}, p_i) \leq \lambda\delta(p).$$

If these two conditions are met, any input data with bounded density have a limit $S^\infty p$, which is Hölder continuous with exponent $-\frac{\log\gamma}{\log N}$.

Proof Contractivity implies $\delta(S^k p) \le \gamma^k \delta(p)$. For any N-adic rational point $\xi \in N^{-r}\mathbb{Z}^s$, the sequence $f_k(\xi) = (S^k p)_{N^k\xi}$ is defined for all $k \ge r$. It is Cauchy, since

$$d_{\mathcal{M}}(f_k(\xi), f_{k+1}(\xi)) \le \lambda\delta(S^k p) \le \lambda\gamma^k\delta(p).$$

Thus the limit function $S^\infty p \equiv f$ is defined for all N-adic points.

Consider now two N-adic points ξ, η. Choose k such that $N^{-(k+1)} \le |\xi - \eta| \le N^{-k}$. For all $j \ge k$, approximate ξ resp. η by N-adic points $a_j, b_j \in N^{-j}\mathbb{Z}^s$, such that none of $|a_j - a|$, $|b_j - b|$, $|a_j - a_{j+1}|$, $|b_j - b_{j+1}|$ exceeds sN^{-j}. One can choose $a_k = b_k$. The sequence a_j is eventually constant with limit ξ, and similarly the sequence b_j is eventually constant with limit η. Using the symbol $(*)$ for "similar terms involving b_j instead of a_j", we estimate

$$\begin{aligned} d_{\mathcal{M}}(f(\xi), f(\eta)) &\le \sum\nolimits_{j\ge k} d_{\mathcal{M}}(f_j(a_j), f_{j+1}(a_{j+1})) + (*) \\ &\le \sum d_{\mathcal{M}}(f_j(a_j), f_{j+1}(a_j)) + d_{\mathcal{M}}(f_{j+1})(a_j), f_{j+1}(a_{j+1})) + (*). \end{aligned}$$

Using the contractivity and displacement-safe condition, we further get

$$\begin{aligned} d_{\mathcal{M}}(f(\xi), f(\eta)) &\le 2\sum\nolimits_{j\ge k} \lambda\delta(S^j p) + s\delta(S^{j+1}p) \\ &\le 2(\lambda + s\gamma)\delta(p)\sum\nolimits_{j\ge k}\gamma^j \le C\delta(p)\frac{\gamma^k}{1-\gamma}. \end{aligned}$$

The index k was chosen such that $k \le -\log|\xi - \eta|/\log N$, so in particular $\gamma^k \le \gamma^{-\log|\xi-\eta|/\log N}$. We conclude that

$$d_{\mathcal{M}}(f(\xi), f(\eta)) \le C'\gamma^{-\log|\xi-\eta|/\log N} = C'|\xi - \eta|^{-\log\gamma/\log N}.$$

Thus f is continuous with Hölder exponent $-\frac{\log\gamma}{\log N}$ on the N-adic rationals, and so is the extension of f to all of $\mathbb{R}^s$. □

The scope of this lemma can be much expanded by some obvious modifications.

- *Input data with unbounded density $d(p)$.* Since points Sp_j only depend on finitely many p_i's, there is $m > 0$ such that p_i only influences Sp_{Ni+j} with $|j| < m$. By iteration, p_i influences $S^2 p_{N^2 i+j}$ with $|j| < Nm + m$, and so on. It follows that p_i influences the value $S^\infty p(i + \xi)$ of the limit function only for $|\xi| < \frac{m}{N} + \frac{m}{N^2} + \cdots = \frac{m}{N-1}$. We can therefore easily analyze the restriction of the limit function to some box by re-defining all data points away from that box in a manner which makes $d(p)$ finite.

- *Partially defined input data.* If data are defined not in all of $\mathbb{Z}^s$ but only in a subset, the limit function is defined for a certain subset of $\mathbb{R}^s$. Finding this subset goes along the same lines as the previous paragraph—we omit the details.
- *Convergence for special input data.* In order to check convergence for particular input data p, it is sufficient that the contractivity and displacement-safe conditions of Lemma 4.4 hold for all data $S^k p$ constructed by iterative refinement from p. A typical instance of this case is that contractivity can be shown only if $\delta(p)$ does not exceed a certain threshold δ_0. It follows that neither does $\delta(S^k p)$, and Lemma 4.4 applies to all p with $\delta(p) \le \delta_0$.
- *Powers of subdivision rules.* A subdivision rule S might enjoy convergence like a contractive rule without being contractive itself. This phenomenon is analogous to a linear operator A having norm $\|A\| \ge 1$ but spectral radius $\rho(A) < 1$, in which case some $\|A^m\| < 1$. In that case we consider some power S^m as a new subdivision rule with dilation factor N^m. If S^m is contractive with factor $\gamma^m < 1$, Lemma 4.4 still applies, and limits enjoy Hölder smoothness with exponent $-\frac{\log \gamma^m}{\log N^m} = -\frac{\log \gamma}{\log N}$.

Example 4.5 (Convergence of Linear Subdivision Rules) Consider a univariate subdivision rule S defined by finitely many nonzero coefficients a_j via (4.11). S acts as a linear operator on sequences $p\colon \mathbb{Z} \to \mathbb{R}^d$. The norm $\|p\| = \sup_i \|p_i\|_{\mathbb{R}^d}$ induces an operator norm $\|S\|$ which obeys $\|Sp\| \le \|S\|\|p\|$. It is an exercise to check $\|S\| = \max_i \sum_j |a_{i-Nj}|$. Equality is attained for suitable input data with values in $\{-1, 0, 1\}$.

With $(\Delta p)_i = p_{i+1} - p_i$ we express the density of the data as $\delta(p) = \sup \|\Delta p_i\|$. Contractivity means that $\sup \|\Delta S p_i\| \le \gamma \sup \|\Delta p_i\|$ for some $\gamma < 1$.

Analysis of this contractivity condition uses a trick based on the generating functions $p(z) = \sum p_j z^j$ and $a(z) = \sum a_j z^j$. Equation (4.11) translates to the relation $(Sp)(z) = a(z)p(z^N)$ between generating functions, and we also have $\Delta p(z) = (z^{-1} - 1)p(z)$. The trick consists in introducing the *derived* subdivision rule S^* with coefficients a_j^* which obeys $S^*\Delta = N\Delta S$. The corresponding relation between generating functions reads

$$a^*(z)\Delta p(z^N) = N(z^{-1} - 1)a(z)p(z^N) \iff a^*(z)(z^{-N} - 1) = N(z^{-1} - 1)a(z)$$

$$\iff a^*(z) = Na(z)z^{N-1}\frac{z-1}{z^N - 1} = Nz^{N-1}\frac{a(z)}{1 + z + \cdots + z^{N-1}}.$$

This division is possible in the ring of Laurent polynomials, because for all i, $\sum_j a_{i-Nj} = 1$. The contractivity condition now reads $\sup \|\Delta S p_i\| = \frac{1}{N}\sup \|S^* \Delta p_i\| \le \frac{1}{N}\|S^*\| \sup \|\Delta p_i\|$, i.e., the contractivity factor of the subdivision rule S is bounded from above by $\frac{1}{N}\|S^*\|$. The "displacement-safe" condition of Lemma 4.4 is fulfilled also, which we leave as an exercise (averages of points p_i are not far from the p_i's).

The above computation leads to a systematic procedure for checking convergence: we compute potential contractivity factors $\frac{1}{N}\|S^*\|$, $\frac{1}{N^2}\|S^{2*}\|$, and so on, until one of them is < 1. The multivariate case is analogous but more complicated [4, 17, 18].

Example 4.6 (Convergence of Geodesic Corner-Cutting Rules) Two points a, b of a complete Riemannian manifold $\mathcal{M}$ are joined by a shortest geodesic path $t \mapsto a \oplus tv$, $v = b \ominus a$, $t \in [0, 1]$. The difference vector v and thus the path are generically unique, but do not have to be, if the distance between a and b exceeds both injectivity radii $\rho_{\mathrm{inj}}(a)$, $\rho_{\mathrm{inj}}(b)$. The point $x = a \oplus tv$ has $d_{\mathcal{M}}(a, x) = t\, d_{\mathcal{M}}(a, b)$, $d_{\mathcal{M}}(b, x) = (1 - t)\, d_{\mathcal{M}}(a, b)$. It is a Fréchet mean of points a, b w.r.t. weights $(1 - t), t$.

With these preparations, we consider two elementary operations on sequences, namely *averaging* A_t and *corner cutting* $S_{t,s}$:

$$(A_t p)_i = p_i \oplus t(p_{i+1} \ominus p_i), \quad (S_{ts} p)_j = \begin{cases} p_i \oplus t(p_{i+1} \ominus p_i) & \text{if } j = 2i, \\ p_i \oplus s(p_{i+1} \ominus p_i) & \text{if } j = 2i + 1. \end{cases}$$

The distance of $A_t p_i$ from $A_t p_{i+1}$ is bounded by the length of the broken geodesic path which connects the first point with p_{i+1} and continues on to the second; its length is bounded by $\delta(p)$. Similarly, the distance of successive points of the sequence $S_{ts} p$, for $0 \le t < s \le 1$ is estimated by $\max(1 - (s - t), s - t)\delta(p)$. It follows immediately that a concatenation of operations of this kind is a subdivision rule where Lemma 4.4 applies, if at least one $S_{t,s}$ with $0 < s - t < 1$ is involved. Any such concatenation therefore is a convergent subdivision rule in any complete Riemannian manifold. A classical example are the rules $S^{(k)} = (A_{1/2})^k \circ S_{0,1/2}$, which insert midpoints,

$$S^{(1)} p_{2i} = p_i, \qquad\qquad S^{(1)} p_{2i+1} = p_i \oplus \frac{1}{2}(p_{i+1} \ominus p_i),$$

and then compute k rounds of averages. E.g.,

$$\begin{aligned} S^{(2)} p_{2i} &= S^{(1)} p_{2i} \quad \oplus \tfrac{1}{2}(S^{(1)} p_{2i+1} \ominus S^{(1)} p_{2i}) \quad = p_i \oplus \tfrac{1}{4}(p_{i+1} \ominus p_i), \\ S^{(2)} p_{2i+1} &= S^{(1)} p_{2i+2} \oplus \tfrac{1}{2}(S^{(1)} p_{2i+2} \ominus S^{(1)} p_{2i+1}) = p_i \oplus \tfrac{3}{4}(p_{i+1} \ominus p_i). \end{aligned}$$

The rule $S^{(2)}$ (Chaikin's rule, see [5]) is one of de Rham's corner cutting rules. In the linear case, $S^{(k)}$ has coefficients $a_j = \frac{1}{2^k}\binom{k}{j}$, apart from an index shift. Its limit curves are the B-spline curves whose control points are the initial data p_j [45].

The corner-cutting rules discussed above are well defined and convergent in any Hadamard metric space—those spaces have geodesics in much the same way as Riemannian manifolds. Subdivision rules based on geodesic averaging (not necessarily restricted to values $t, s \in [0, 1]$) have been treated by [19, 20, 51, 52]. We should also mention that adding a round $A_{1/2}$ to a subdivision increases smoothness of limit curves, which was recently confirmed in the manifold case [14].

Example 4.7 (Convergence of Interpolatory Rules) A subdivision rule S with dilation factor N is called *interpolatory* if $Sp_{Ni} = p_i$, i.e., the old data points are kept and new data points are inserted in between. In the linear case, a very well studied subdivision rule of this kind is the four-point rule proposed by Dyn et al. [21]. We let $Sp_{2i} = p_i$ and

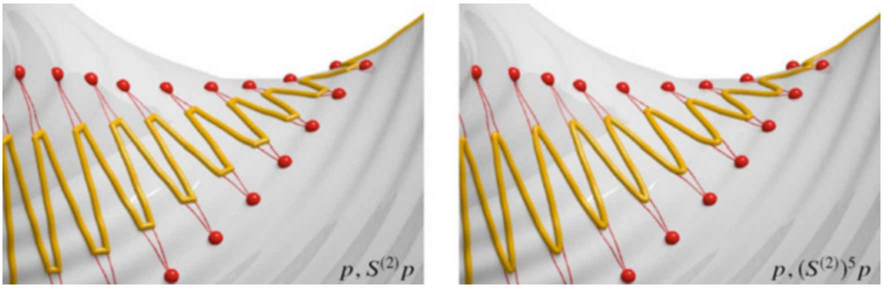


Fig. 4.2 Geodesic corner-cutting rules are among those where convergence is not difficult to show. These images show Chaikin's rule $S^{(2)}$, with the original data in red, and the result of subdivision as a yellow geodesic polygon

$$\begin{aligned} Sp_{2i+1} &= -\omega p_{i-1} + (\tfrac{1}{2}+\omega)p_i + (\tfrac{1}{2}+\omega)p_{i+1} - \omega p_{i+2} \\ &= \frac{p_i+p_{i+1}}{2} - \omega\Big(p_{i-1} - \frac{p_i+p_{i+1}}{2}\Big) - \omega\Big(p_{i+2} - \frac{p_i+p_{i+1}}{2}\Big). \end{aligned}$$

In the special case $\omega = \frac{1}{16}$, the point Sp_{2i+1} is found by evaluating the cubic Lagrange polynomial interpolating $p_{i-1}, \dots, p_{i+2}$, which accounts for the high approximation order of S. There is in fact a whole series of interpolatory rules based on the idea of evaluating Lagrange interpolation polynomials (the Dubuc-Deslauriers subdivision schemes, see [7]).

S is a binary "dual" subdivision rule with combinatorial symmetry about edges. Thus it makes sense to define a Riemannian version of S (Fig. 4.2) by means of averages w.r.t. geodesic midpoints of p_i, p_{i+1} as base points, cf. Eq. (4.9). Using $m_{p_i,p_{i+1}} = p_i \oplus \frac{1}{2}(p_{i+1} \ominus p_i)$, we let

$$Tp_{2i} = p_i, Tp_{2i+1} \quad = m_{p_i,p_{i+1}} \oplus \Big(-\omega(p_{i-1} \ominus m_{p_i,p_{i+1}}) - \omega(p_{i+2} \ominus m_{p_i,p_{i+1}})\Big).$$

The distance of successive points Tp_{2i} and Tp_{2i+1} is bounded by half the geodesic distance of p_i, p_{i+1} plus the length of the vector added to the midpoint in the previous formula. This yields the inequality $\delta(Tp) \le \frac{1}{2}\delta(p) + 2|\omega|\frac{3}{2}\delta(p) = (\frac{1}{2} + 3|\omega|)\delta(p)$. Lemma 4.4 thus shows convergence, if $|\omega| < 1/6$.

We cannot easily extend this "manifold" four-point rule to more general metric spaces. The reason is that we used the linear structure of the tangent space. A general discussion of univariate interpolatory rules is found in [50].

4.2.3 *Probabilistic Interpretation of Subdivision in Metric Spaces*

Ebner in [22, 23] gave a probabilistic interpretation of subdivision. This goes as follows. Consider a linear subdivision rule as in (4.11), namely

$$Sp_i = \sum\nolimits_j a_{i-2j} p_j = \operatorname{avg}(a_{i-2j}; p_j), \quad \text{where } a_i \ge 0, \ \sum\nolimits_j a_{i-2j} = 1, \tag{4.12}$$

acting on data $p\colon \mathbb{Z}^s \to \mathbb{R}^d$. Consider a stochastic process $J_0, J_1, \dots$ defined as the random walk on $\mathbb{Z}^s$ with transition probabilities

$$\mathbb{P}\,(J_{n+1}{=}j \mid J_n{=}i) = a_{i-2j}.$$

Then the expected value of $p_{J_{n+1}}$, conditioned on $J_n = j$ is given by

$$\mathbb{E}\,(p_{J_{n+1}} \mid J_n{=}j) = Sp_j, \tag{4.13}$$

by definition of the expected value. Now the expectation $\mathbb{E}\,(X)$ of an $\mathbb{R}^d$-valued random variable X has a characterization via distances: $\mathbb{E}\,(X)$ is that constant $c \in \mathbb{R}^d$ which is closest to X in the sense of $\mathbb{E}\,(d(X,c)^2) \to \min$. A similar characterization works for the conditional expectation $\mathbb{E}\,(X|Y)$ which is the random variable $f(Y)$ closest to X in the L^2 sense. These facts inspired a theory of random variables with values in Hadamard metric spaces developed by Sturm [46, 47]. The minimizers mentioned above can be shown to still exist if $\mathbb{R}^d$ is replaced by $\mathcal{M}$.

Since the way we compute subdivision by Fréchet means is compatible with the distance-based formula for expected values, Eq. (4.13) holds true also in the case that both the expectation and the subdivision rule are interpreted in the Hadamard space sense. On that basis, O. Ebner could show a remarkable statement on convergence of subdivision rules:

Theorem 4.8 ([23, Theorem 1]) *Consider a binary subdivision rule $Tp_i = \operatorname{avg}_F(a_{i-2j}; p_j)$ with nonnegative coefficients a_i. It produces continuous limits for any data p_j in any Hadamard space $\mathcal{M}$ if and only if it produces a continuous limit function when acting on real-valued data.*

Proof *(Sketch)* With the random walk $(J_i)_{i=0,1,\dots}$ defined above, (4.13) directly implies

$$\begin{aligned}(T^n p)_{J_0} &= \mathbb{E}\,(T^{n-1} p_{J_1} \mid J_0) = \mathbb{E}\,(\mathbb{E}\,(T^{n-2} p_{J_2} \mid J_1) \mid J_0) = \dots \\ &= \mathbb{E}\,(\dots \mathbb{E}\,(\mathbb{E}\,(p_{J_n} \mid J_{n-1}) \mid J_{n-2}) \dots \mid J_0).\end{aligned} \tag{4.14}$$

Unlike for $\mathbb{R}^d$-valued random variables, there is no tower property for iterated conditioning, so in general $(T^n p)_{J_0} \ne \mathbb{E}\,(p_{J_n}|J_0)$. That expression has a different interpretation: T is analogous to the linear rule S of (4.12), which is nothing but the restriction of the general rule T to data in Euclidean spaces. Its n-th power S^n is a linear rule of the form $(S^n q)_i = \sum a^{[n]}_{i-2^n j} q_j$, and we have

$$\mathbb{E}\,(q_{J_n} \mid J_0) = (S^n q)_{J_0}, \quad \text{if } S \text{ acts linearly on data } q\colon \mathbb{Z}^s \to \mathbb{R}^d. \tag{4.15}$$

This follows either directly (computing the coefficients of the n-th iterate S^n corresponds to computing transition probabilities for the n-iterate of the random walk), or by an appeal to the tower property in (4.14).

Sturm [46] showed a Jensen's inequality for continuous convex functions Ψ,

$$\Psi\Big(\mathbb{E}\,(\dots(\mathbb{E}\,(p_{J_n} \mid J_{n-1})\dots \mid J_0)\Big) \leq \mathbb{E}\,\Big(\Psi(p_{J_n}) \mid J_0\Big).$$

We choose $\Psi = d_{\mathcal{M}}(\cdot, x)$ and observe that $q_{J_n} = d_{\mathcal{M}}(p_{J_n}, x)$ is a real-valued random variable. Combining Jensen's inequality with (4.14) and (4.15) yields

$$d_{\mathcal{M}}(T^n p_i, x) \leq \sum\nolimits_k a^{[n]}_{i-2^n k}\, d_{\mathcal{M}}(p_k, x), \qquad \text{(for any } x\text{)}$$

$$d_{\mathcal{M}}(T^n p_i, T^n p_j) \leq \sum\nolimits_{k,l} a^{[n]}_{i-2^n k} a^{[n]}_{j-2^n l}\, d_{\mathcal{M}}(p_k, p_l) \qquad \text{(by recursion).}$$

To continue, we need some information on the coefficients $a^{[n]}_i$. For that, we use the limit function $\phi\colon \mathbb{R}^s \to [0, 1]$ generated by applying T (or rather, S), to the delta sequence. By construction (see Lemma 4.4), $|a^{[n]}_j - \phi(2^{-n} j)| \to 0$ as $n \to \infty$. These ingredients allow us to show existence of n with T^n contractive. □

As a corollary we get, for instance, that subdivision with nonnegative coefficients works in Pos_n in the same way as in linear spaces, as far as convergence is concerned. Since Pos_n is not only a Hadamard metric space, but even a smooth Riemannian manifold, also the next section will yield a corollary regarding Pos_n.

4.2.4 The Convergence Problem in Manifolds

The problem of convergence of subdivision rules in manifolds (Riemannian manifolds, groups, and symmetric spaces) was at first treated by means of so-called proximity inequalities which compare linear rules with their analogous counterparts in manifolds. This approach was successful in studying smoothness of limits (see Sect. 4.3 below), but less so for convergence. Unless subdivision rules are of a special kind (interpolatory, corner-cutting, ...) convergence can typically be shown only for "dense enough" input data, with very small bounds on the maximum allowed density. On the other hand numerical experiments demonstrate that a manifold rule analogous to a convergent linear rule usually converges. This discrepancy between theory and practice is of course unsatisfactory from the viewpoint of theory, but is not so problematic from the viewpoint of practice. The reason is the stationary nature of subdivision—if $\delta(p)$ is too big to infer existence of a continuous limit $S^\infty p$, we can check if $\delta(S^k p)$ is small enough instead. As long as S converges, this leads to an a-posteriori proof of convergence.

More recently, convergence of subdivision rules of the form $Sp_i = \mathrm{avg}_F (a_{i-Nj}; p_j)$ in Riemannian manifolds has been investigated along the lines of

Lemma 4.4. This work is mainly based on the methods of H. Karcher's seminal paper [39]. So far, only the univariate case of data $p\colon \mathbb{Z} \to \mathcal{M}$ has been treated successfully, cf. [36, 37, 54].

There are two main cases to consider. In Cartan-Hadamard manifolds (curvature ≤ 0) the Fréchet mean is well defined and unique also if weights are allowed to be negative [37, Th. 6]. Subdivision rules are therefore globally and uniquely defined. We have the following result:

Proposition 4.9 ([37, Th. 11]) *Consider a univariate subdivision rule $Sp_i = \operatorname{avg}_F(a_{i-Nj}; p_j)$ acting on sequences in a Cartan-Hadamard manifold $\mathcal{M}$. Consider also the norm $\|S^*\|$ of its linear derived subdivision rule according to Example 4.5. If*

$$\gamma = \frac{1}{N}\|S^*\| < 1,$$

then S meets the conditions of Lemma 4.4 (with contractivity factor γ) and produces continuous limits.

This result is satisfying because it allows us to infer convergence from a condition which is well known in the linear case, cf. [17]. If $\frac{1}{N}\|S^*\| \geq 1$, we can instead check if one of $\frac{1}{N^n}\|S^{*n}\|$, $n = 2, 3, \ldots$ is smaller than 1. If this is the case, then the manifold subdivision rule analogous to the linear rule S^n converges.

Subdivision in Riemannian Manifolds with Positive Curvature

Recent work [36] deals with spaces of positive curvature, and initial results have been achieved on the unit sphere, for subdivision rules of the form $Sp_i = \operatorname{avg}_F(a_{i-2j}; p_j)$. Figure 4.3 shows two examples. One aims at finding a bound δ_0 such that for all data p with $\delta(p) < \delta_0$, S acts in a contractive way so that Lemma 4.4 shows convergence.

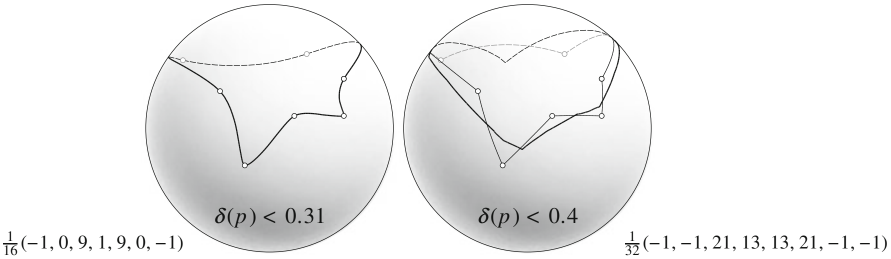

$\delta(p) < 0.31$
$\delta(p) < 0.4$
$\frac{1}{16}(-1, 0, 9, 1, 9, 0, -1)$
$\frac{1}{32}(-1, -1, 21, 13, 13, 21, -1, -1)$

Fig. 4.3 Subdivision rules $Sp_j = \operatorname{avg}_F(a_{j-2i}; p_i)$ based on the Fréchet mean operating on sequences on the unit sphere. The images visualize the interpolatory 4-point rule (left) and a rule without any special properties. We show the coefficient sequence a_j and the bound on $\delta(p)$ which ensures convergence

Rules defined in a different way are sometimes much easier to analyze. E.g. the Lane-Riesenfeld subdivision rules defined by midpoint insertion, followed by k rounds of averaging, can be transferred to any complete Riemannian manifold as a corner-cutting rule and will enjoy continuous limits, see Example 4.6. Similarly, the interpolatory four-point rule can be generalized to the manifold case in the manner described by Example 4.7, and will enjoy continuous limits. The generalization via the Fréchet mean (Fig. 4.3) on the other hand, is not so easy to analyze. The approach by [36] is to control $\delta(Sp)$ by introducing a family $S^{(t)}$, $0 \leq t \leq 1$, of rules where $S^{(0)}$ is easy to analyze, and $S^{(1)} = S$. If one manages to show $\delta(S^{(0)}p) < \gamma_1 \delta(p)$ and $\|\frac{d}{dt} S^{(t)} p_i\| \leq C\delta(p)$, then the length of each curve $t \mapsto S^{(t)} p_i$ is bounded by $C\delta(p)$, and

$$\begin{aligned}\delta(Sp) &\leq \sup_i d_{\mathcal{M}}(Sp_i, S^{(0)} p_i) + \delta(S^{(0)} p) + \sup_i d_{\mathcal{M}}(Sp_{i+1}, S^{(0)} p_{i+1}) \\ &\leq (\gamma_1 + 2C)\delta(p).\end{aligned}$$

Contractivity is established if $\gamma_1 + 2C < 1$, in which case Lemma 4.4 shows convergence. The bounds mentioned in Fig. 4.3 have been found in this way. Estimating the norm of the derivative mentioned above involves estimating the eigenvalues of the Hessian of the right hand side of (4.5).

The State of the Art Regarding Convergence of Refinement Schemes

Summing up, convergence of geometric subdivision rules is treated in a satisfactory manner for special rules (interpolatory, corner-cutting), for rules in special spaces (Hadamard spaces and Cartan-Hadamard manifolds), and in the very special case of the unit sphere and univariate rules. General manifolds with positive curvature have not been treated. Multivariate data are treated only in Hadamard metric spaces and for subdivision rules with nonnegative coefficients. In other situations, we know that convergence happens only for "dense enough" input data, where the required theoretical upper bounds on $\delta(p)$ are very small compared to those inferred from numerical evidence.

4.3 Smoothness Analysis of Subdivision Rules

For linear subdivision rules, the question of smoothness of limits can be considered as largely solved, the derived rule S^* introduced in Example 4.5 being the key to the question if limits are smooth. Manifold subdivision rules do not always enjoy the same smoothness as the linear rules they are derived from. The constructions mentioned in Sect. 4.2 basically yield manifold rules whose limits enjoy C^1 resp. C^2 smoothness if the original linear rule has this property, but this general statement is no longer true if C^3 or higher smoothness is involved. Manifold rules generated via Fréchet means or via projection [26, 59] retain the smoothness of their linear counterparts. Others, e.g. constructed by means of averages w.r.t. basepoints in general do not. This is to be expected, since the choice of basepoint introduces an element of arbitrariness into manifold subdivision rules. The following paragraphs discuss the method of *proximity inequalities* which was successfully employed in treating the smoothness of limits.

4.3.1 Derivatives of Limits

A subdivision rule S acting on a sequence p in $\mathbb{R}^d$ converges to the limit function $S^\infty p$, if the refined data $S^k p$, interpreted as samples of functions f_k at the finer grid $N^{-k}\mathbb{Z}$, approach that limit function (see Definition 4.3):

$$(S^\infty p)(\xi) \approx f_k(\xi) = (S^k p)_{N^k \xi},$$

whenever $N^k \xi$ is an integer. A similar statement holds for derivatives, which are approximated by finite differences. With $h = N^{-k}$, we get

$$\begin{aligned}(S^\infty p)'(\xi) &\approx \frac{f_k(\xi + h) - f_k(\xi)}{h} = N^k((S^k p)_{N^k\xi+1} - (S^k p)_{N^k\xi}) \\ &= (\Delta(NS)^k p)_{N^k\xi} = (S^{*k}\Delta p)_{N^k\xi}.\end{aligned}$$

Here S^* is the derived rule defined by the relation $S^*\Delta = N\Delta S$, see Example 4.5. For the r-th derivative of the limit function we get

$$(S^\infty p)^{(r)}(\xi) \approx (\Delta^r (N^r S)^k p)_{N^k\xi} = ((S^{\overbrace{**\cdots*}^{r\text{ times}}})^k \Delta^r p)_{N^k\xi}.$$

These relations, except for references to derived rules, are valid even if S does not act linearly. S could be a manifold rule expressed in a coordinate chart, or it could be acting on a surface contained in $\mathbb{R}^d$.

If S does act linearly, one proves that S has C^1 smooth limits, if S^* has continuous ones, and in that case $(S^\infty p)' = S^{*\infty}\Delta p$. To treat higher order derivatives, this statement can be iterated. For multivariate data p_i, $i \in \mathbb{Z}^s$, the situation is analogous but more complicated to write down. For the exact statements, see [4, 17].

4.3.2 Proximity Inequalities

Smoothness from Proximity

Manifold subdivision rules were first systematically analyzed with regard to derivatives by [51]. The setup is a linear rule S and a nonlinear rule T both acting on data contained in the same space $\mathbb{R}^d$. T could be a manifold version of S, with $\mathbb{R}^d$ being a coordinate chart of the manifold; or T could act on points of a surface contained in $\mathbb{R}^d$. Then S, T are in proximity, if

$$\sup_i \|Sp_i - Tp_i\| \leq C\delta(p)^2. \tag{4.16}$$

This formula is motivated by a comparison of the shortest path between two points within in a surface (which is a geodesic segment), with the shortest path in Euclidean space (which is a straight line). These two paths differ by exactly the amount stated in (4.16). Two statements were shown in [51]:

1. Certain manifold subdivision rules T derived from a convergent linear rule S obey the proximity inequality (4.16) whenever data are dense enough (i.e., $\delta(p)$ is small enough).
2. in that case, if limit curves of S enjoy C^1 smoothness, then T produces continuous limit curves for data with $d(p)$ small enough; and all continuous limit curves enjoy C^1 smoothness.

To demonstrate how proximity inequalities work, we prove a convergence statement like the ones given by [52, Th. 1] or [51, Th. 2+3] (with slightly different proofs).

Proposition 4.10 *Assume the setting of Eq.* (4.16)*, with a subdivision rule T being in proximity with a linear subdivision rule S. We also assume $\frac{1}{N}\|S^*\| < 1$.*[3] *Then T produces continuous limit curves from data p with $\delta(p)$ small enough.*

Proof Generally $\sup_i \|p_i - q_i\| \leq K \implies \delta(p) \leq \delta(q) + 2K$. Thus (4.16) implies

$$\delta(Tp) \leq \delta(Sp) + 2C\delta(Tp)^2 \leq \frac{1}{N}\|S^*\|\delta(p) + 2C\delta(p)^2.$$

Choose $\epsilon > 0$ with $\lambda := \frac{1}{N}\|S^*\| + 2C\epsilon < 1$. If $\delta(p) < \epsilon$, then T is contractive:

$$\delta(Tp) \leq (\frac{1}{N}\|S^*\| + 2C\delta(p))\delta(p) \leq \lambda\delta(p).$$

[3] Implying convergence of the linear rule S. N is the dilation factor, S^* is the derived rule, cf. Example 4.5.

By recursion, $\delta(T^{k+1}p) \leq \lambda\delta(T^k p)$. As to the displacement-safe condition of Lemma 4.4, recall from Example 4.5 that S has it. For T, observe that

$$\begin{aligned} \|Tp_{Ni} - p_i\| &\leq \|Tp_{Ni} - Sp_{Ni}\| + \|Sp_{Ni} - p_i\| \\ &\leq C\delta(p)^2 + C'\delta(p) \leq (\epsilon C + C')\delta(p). \end{aligned}$$

Now Lemma 4.4 shows convergence. □

The convergence of vectors $N^k \Delta T^k p$ to derivatives of the limit function $T^\infty p$ is proved in a way which is analogous in principle. The method was extended to treat C^2 smoothness by [49], using the proximity condition

$$\sup_i \|\Delta Sp_i - \Delta Tp_i\| \leq C(\delta(p)\delta(\Delta p) + \delta(p)^3).$$

A series of publications treated C^2 smoothness of Lie group subdivision rules based on log/exp averages [31, 53], the same in symmetric spaces [54], C^1 smoothness in the multivariate case [25], higher order smoothness of interpolatory rules in groups [27, 61], and higher order smoothness of projection-based rules [26, 59]. The proximity conditions involving higher order smoothness become rather complex, especially in the multivariate case.

Smoothness Equivalence

If a manifold subdivision rule T is created on basis of a linear rule S, it is interesting to know if the limit functions of T enjoy the same smoothness as the limits of S. For C^1 and C^2 smoothness, T can basically be constructed by any of the methods described above, and it will enjoy the same smoothness properties as S (always assuming that convergence happens, and that the manifold under consideration is itself as smooth as the intended smoothness of limits). This smoothness equivalence breaks down for C^k with $k \geq 3$.

A manifold subdivision rule based on the log/exp construction, using averages w.r.t. basepoints,

$$Tp_i = \mathrm{avg}_{m_i}(a_{i-Nj}; p_j),$$

does not enjoy C^k smoothness equivalence for $k \geq 3$ unless the base points m_i obey a technical condition which can be satisfied e.g. if they themselves are produced by certain kinds of subdivision [29, 60]. Necessary and sufficient conditions for smoothness equivalence are discussed by [13]. We pick one result whose proof is based on this method (using (4.10) for a "base point" interpretation of Fréchet means):

Theorem 4.11 ([29, Th. 4.3]) *Let S be a stable[4] subdivision rule $Sp_i = \operatorname{avg}(a_{i-Nj}; p_j)$ acting on data $p\colon \mathbb{Z}^s \to \mathbb{R}^d$, which is convergent with C^n limits. Then all continuous limits of its Riemannian version $Tp_i = \operatorname{avg}_F(a_{i-Nj}; p_j)$ likewise are C^n.*

We should also mention that proximity conditions relevant to the smoothness analysis of manifold subdivision rules can take various forms, cf. the "differential" proximity condition of [12, 13, 30].

Finally we point out a property which manifold rules share with linear ones: For any univariate linear rule S which has C^k limits, the rule $A_{1/2}^k \circ S$ has limits of smoothness C^{n+k}, where $A_{1/2}$ is midpoint-averaging as described by Example 4.6. It has been shown in [14] that an analogous statement holds true also in the manifold case, for a general class of averaging operators.

4.3.3 Subdivision of Hermite Data

Hermite subdivision is a refinement process acting not on points, but on tangent vectors, converging to a limit and its derivative simultaneously. In the linear case, data $(p, v)\colon \mathbb{Z} \to \mathbb{R}^d \times \mathbb{R}^d$ undergo subdivision by a rule S which obeys basic shift invariance $SL = L^N S$. The interpretation of p_i as points and v_i as vectors leads to

$$S\binom{p}{v}_i = \begin{pmatrix} \sum_j a_{i-Nj} p_j + \sum_j b_{i-Nj} v_j \\ \sum_j c_{i-Nj} p_j + \sum_j b_{i-Nj} v_j . \end{pmatrix}, \quad \text{where } \begin{cases} \sum_j a_{i-Nj} = 1, \\ \sum_j c_{i-Nj} = 0. \end{cases} \tag{4.17}$$

S is invariant w.r.t. translations, which act via $p \mapsto p + x$ on points, but act identically on vectors. Iterated refinement creates data $S^k\binom{p}{v}$ converging to a limit $f\colon \mathbb{R} \to \mathbb{R}^d$,

$$\begin{pmatrix} f(\xi) \\ f'(\xi) \end{pmatrix} = \lim_{k\to\infty} \begin{pmatrix} 1 & 0 \\ 0 & N^k \end{pmatrix} S^k \begin{pmatrix} p \\ v \end{pmatrix}_{N^k\xi}, \quad \text{whenever } N^k\xi \in \mathbb{Z}.$$

We say that S converges, if the limit (f, f') exists and f enjoys C^1 smoothness, with f' then being continuous. A manifold version of S, operating on data

$$\binom{p}{v}\colon \mathbb{Z} \to T\mathcal{M}, \quad \text{i.e., } v_i \in T_{p_i}\mathcal{M},$$

[4] "Stable" means existence of constants C_1, C_2 with $C_1\|p\| \le \|S^\infty p\|_\infty \le C_2\|p\|$ for all input data where $\|p\| := \sup_i \|p_i\|$ is bounded. Stable rules with C^n limits generate polynomials of degree $\le n$, which is a property used in the proof.

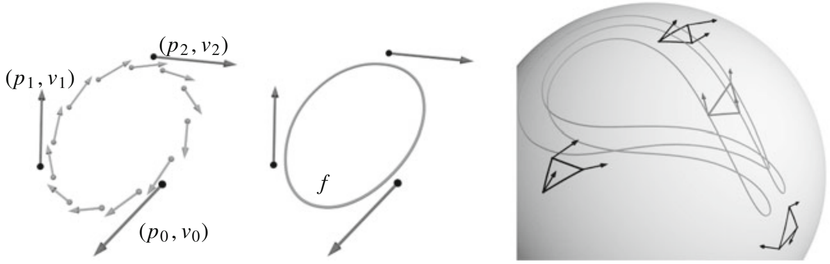


Fig. 4.4 *Left:* Hermite data (p_i, v_i) in $\mathbb{R}^2$ and the result of one round subdivision by a linear Hermite rule S. *Center:* Limit curve f (f' is not shown). *Right:* Hermite data (p_i, v_i) in the group SO_3, and the limit curve generated by a group version of S. Points $p_i \in \mathrm{SO}_3$ and tangent vectors $v_i \in T_{p_i}\mathrm{SO}_3$ are visualized by means of their action on a spherical triangle. These figures appeared in [42] (© The Authors 2017, reprinted with permission)

faces the difficulty that each v_i is contained in a different vector space. One possibility to overcome this problem is to employ parallel transport $\mathrm{pt}_p^q \colon T_p\mathcal{M} \to T_q\mathcal{M}$ between tangent spaces. In Riemannian manifolds, a natural choice for pt_p^q is parallel transport w.r.t. the canonical Levi-Civita connection along the shortest geodesic connecting p and q, cf. [8]. In groups, we can simply choose pt_p^q as left translation by qp^{-1} resp. the differential of this left translation. Then the definition

$$S\binom{p}{v} = \binom{q}{w} \text{ with } \begin{cases} q_i = m_i \oplus \big(\sum_j a_{i-Nj}(p_j \ominus m_i) + \sum_j b_{i-Nj}\, \mathrm{pt}_{p_j}^{m_i}\, v_j\big) \\ w_i = \mathrm{pt}_{m_i}^{q_i}\, \big(\sum_j c_{i-Nj}(p_j \ominus m_i) + \sum_j d_{i-Nj}\, \mathrm{pt}_{p_j}^{m_i}\, v_j\big) \end{cases}$$

is meaningful (provided the base point m_i is chosen close to $p_{\lfloor i/N \rfloor}$). In a linear space, this expression reduces to (4.17). C. Moosmüller could show C^1 smoothness of limits of such subdivision rules, by methods in the spirit of Sect. 4.3.2, see [41, 42] (Fig. 4.4).

4.3.4 Subdivision with Irregular Combinatorics

A major application of subdivision is in Computer Graphics, where it is ubiquitously used as a tool to create surfaces from a finite number of handle points whose arrangement is that of the vertices of a 2D discrete surface. That surface usually does not have the combinatorics of a regular grid.

Two well known subdivision rules acting on such data are the Catmull-Clark rule and the Doo-Sabin rule, see [3, 11]. Such subdivision rules create denser and denser discrete surfaces which are mostly regular grids but retain a constant number of combinatorial singularities. This implies that the limit surface is locally obtained via Definition 4.3, but with a nontrivial overlapping union of several such limits as one approaches a combinatorial singularity. A systematic way of analyzing convergence

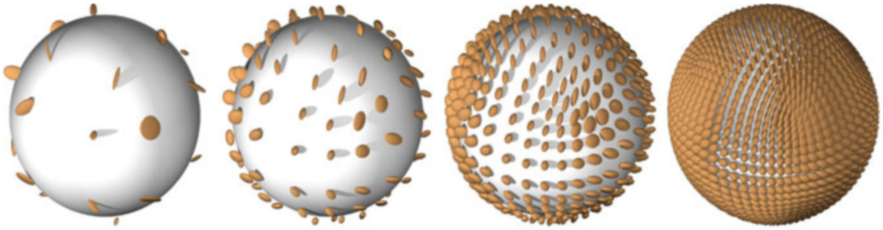


Fig. 4.5 Here data p_i in the unit sphere Σ^2 and Pos_3-valued data q_i are visualized by placing the ellipsoid with equation $x^T q_i x = 1$ in the point $p_i \in \Sigma^2$. Both data undergo iterative refinement by means of a Riemannian version S of the Doo-Sabin subdivision rule. For given initial data p, q which have the combinatorics of a cube, the four images show $S^j p$ and $S^j q$, for $q = 1, 2, 3, 4$ (from left). The correspondence $(S^k p)_i \mapsto (S^k q)_i$ converges to a C^1 immersion $f\colon \Sigma^2 \to \mathrm{Pos}_3$ as $k \to \infty$. These figures appeared in [55] (© Springer Science+Business Media, LLC 2009, reprinted with permission)

and smoothness was found by Reif [44], see also the monograph [43]. There is a wealth of contributions to this topic, mostly because of its relevance for Graphics.

Weinmann in [55–57] studied intrinsic manifold versions of such subdivision rules. They are not difficult to define, since the linear subdivision rules which serve as a model are defined in terms of averages. We do not attempt to describe the methods used for establishing convergence and smoothness of limits other than to say that a proximity condition which holds between a linear rule S and a nonlinear rule T eventually guarantees that in the limit, smoothness achieved by S carries over to T—the perturbation incurred by switching from a linear space to a manifold is not sufficient to destroy smoothness. Figure 4.5 illustrates a result obtained by [55].

4.4 Multiscale Transforms

4.4.1 Definition of Intrinsic Multiscale Transforms

A natural multiscale representation of data, which does not suffer from distortions caused by the choice of more or less arbitrary coordinate charts, is required to be based on operations which are themselves adapted to the geometry of the data. This topic is intimately connected to subdivision, since upscaling operations may be interpreted as subdivision.

A high-level introduction of certain kinds of multiscale decompositions is given by [33]. We start with an elementary example.

Example 4.12 (A Geometric Haar Decomposition and Reconstruction Procedure) Consider data $p\colon \mathbb{Z} \to \mathcal{M}$, and the upscaling rule S and downscaling rule D,

$$(\dots, p_0, p_1, \dots) \overset{S}{\longmapsto} (\dots, p_0, p_0, p_1, p_1, \dots)$$
$$(\dots, p_0, p_1, \dots) \overset{D}{\longmapsto} (\dots, m_{p_0,p_1}, m_{p_1,p_2}, \dots), \text{ where } m_{a,b} = a \oplus \frac{1}{2}(b \ominus a).$$

The use of $\oplus$ and $\ominus$ refers to the exponential mapping, as a means of computing differences of points, and adding vectors to points. D is a left inverse of S but not vice versa: $SDp \neq p$ in general. However, if we store the difference between p and SDp as *detail vectors* q:

$$(\dots, q_0, q_1, \dots) = (\dots, p_0 \ominus m_{p_0,p_1}, p_2 \ominus m_{p_2,p_3}, \dots)$$

then the reconstruction procedure

$$p_{2i} = m_{p_{2i},p_{2i+1}} \oplus q_i, \qquad p_{2i+1} = m_{p_{2i},p_{2i+1}} \ominus q_i$$

recovers the information destroyed by downsampling.

More systematically, we have employed two upscaling rules S, R and two downscaling rules D, Q which obey

$$SL = L^2 S, \quad RL = L^2 R, \quad DL^2 = LD, \quad DQ^2 = LQ$$

(L is left shift). We have data $p^{(j)}$ at level j, $j = 0, \dots, M$, where we interpret the data at the highest (finest) level as given, and the data at lower (coarser) level computed by downscaling. We also store details $q^{(j)}$ at each level (Fig. 4.6):

$$p^{(j-1)} = Dp^{(j)}, \qquad q^{(j)} = Q(p^{(j)} \ominus Sp^{(j-1)}). \tag{4.18}$$

We require that upscaled level $j-1$ data and level j details can restore level j data:

$$p^{(j)} = Sq^{(j-1)} \oplus Rq^{(j)}. \tag{4.19}$$

Generally, S, D compute points from points, so they are formulated via averages:

$$Sp_i = \operatorname{avg}(a_{i-2j}; p_j), \qquad Dp_i = \operatorname{avg}(a_{2i-j}; p_j).$$

In Example 4.12, averages are computed w.r.t. base points $p_{\lfloor i/2 \rfloor}$ for S resp. p_i for D, and coefficients a_j and b_j vanish except $a_0 = a_1 = 1$, $b_0 = b_1 = \frac{1}{2}$.

$$\begin{tikzcd}
 & p^{(j)} \arrow[dl, "Q(\mathrm{id}\ominus SD)"'] \arrow[d, "D"] \\
q^{(j)} & p^{(j-1)} \arrow[dl, "Q(\mathrm{id}\ominus SD)"'] \arrow[d, "D"] \\
q^{(j-1)} & p^{(j-2)} \arrow[d, dotted, no head] \\
 &
\end{tikzcd}
\qquad
\begin{tikzcd}
 & \arrow[d, dotted, no head] \\
q^{(j-1)} \arrow[dr, "\oplus R(\cdot)"'] & p^{(j-2)} \arrow[d, "S"] \\
q^{(j)} \arrow[dr, "\oplus R(\cdot)"'] & p^{(j-1)} \arrow[d, "S"] \\
 & p^{(j)}
\end{tikzcd}$$

Fig. 4.6 The decomposition and reconstruction chains of operations in a geometric multiscale decomposition based on upscaling and downscaling S, D for points, and upscaling and downscaling R, Q for detail vectors

The downscaling operator Q acts on tangent vectors $v_i = p_i \ominus (SDp)_i \in T_{p_i}\mathcal{M}$, so it has to deal with vectors potentially contained in different vector spaces. In our special case, Q simply forgets one half of the data:

$$(Qp)_i = p_{2i}.$$

Finally, the upscaling operator R takes the vectors stored in $q^{(j)}$ and converts them into vectors which can be added to upscaled points $Sp^{(j-1)}$. Thus R potentially has to deal with vectors contained in different tangent spaces. In our special case, the points $(Sp^{(j-1)})_{2i}$ $(Sp^{(j-1)})_{2i+1}$ both coincide with $p_i^{(j-1)}$, and that is also the point where the detail coefficient $q_i^{(j)}$ is attached to. We therefore might be tempted to write $(Rq)_{2i} = q_i$, $(Rq)_{2i+1} = -q_i$. This simple rule however does not take into account that along reconstruction, data and details could have been modified, and no longer fit together. We therefore use parallel transport to move the vector to the right tangent space, just in case:

$$(Rq)_{2i} = \mathrm{pt}^{(p^{(j-1)})_i}(q_i), \qquad (Rq)_{2i+1} = -(Rq)_{2i}.$$

The symbol $\mathrm{pt}^{(p^{(j-1)})_i}(q_i)$ refers to transporting q_i to a tangent vector attached to $(p^{(j-1)})_i$, see Sect. 4.3.3.

The operations S, R, D, Q must be compatible, in the sense that reconstruction is a left inverse of downscaling plus computing details. While in the linear case, where S, D, R, Q are linear operators on $\ell^\infty(\mathbb{R}^d)$, one usually requires $QR = \mathrm{id}$ and $QS = 0$ as well as $SD + RQ = \mathrm{id}$, in the geometric case we must be careful not to mix operations on points with operations on tangent vectors. We therefore require

$$SDp \oplus (RQ(p \ominus SDp)) = p. \tag{4.20}$$

Example 4.13 (Interpolatory Wavelets) Consider an interpolatory subdivision rule S with dilation factor 2, i.e., $Sp_{2i} = p_i$, and the forgetful downscaling operator $p_i^{(j-1)} = (Dp^{(j)})_i = p_{2i}^{(j)}$. If we store as details the difference vectors between SDp and p for odd indices, the data points p_{2i+1} can be easily reconstructed:

$$p_i^{(j-1)} = p_{2i}^{(j)}, \qquad q_i^{(j)} = p_{2i+1}^{(j)} \ominus Sp^{(j-1)} \qquad \text{(decomposition)},$$
$$p_{2i}^{(j)} = p_i^{(j-1)}, \qquad p_{2i+1}^{(j)} = (Sp^{(j-1)})_{2i+1} \oplus q_i^{(j)} \qquad \text{(reconstruction)}.$$

This procedure fits into the general scheme described above if we we let $Q = DL$ (L is left shift) and define the upscaling of details by $(Rq)_{2i} = 0$, $(Rq)_{2i+1} = q_i$. To admit the possibility that before reconstruction, data and details have been changed, we define

$$(Rq)_{2i} = 0 \in T_x\mathcal{M}, \quad (Rq)_{2i+1} = \mathrm{pt}^x(q_i^{(j)}) \in T_x\mathcal{M}, \quad \text{where } x = Sp_{2i+1}^{(j-1)},$$

in order to account for the possibility that $q_i^{(j)}$ is not yet contained in the "correct" tangent space. The decimated data $p^{(j-1)}$ together with details $q^{(j)}$ $(j \leq M)$ may be called a geometric interpolatory-wavelet decomposition of the data at the finest level $p^{(M)}$. That data itself comes e.g. from sampling a function, cf. [9].

Definability of Multiscale Transforms Without Redundancies

The previous examples use upscaling and downscaling operations which are rather simple, except that in Example 4.13 one may use any interpolatory subdivision rule. It is also possible to extend Example 4.12 to the more general case of a midpoint-interpolating subdivision rule S, which is a right inverse of the decimation operator D. In [33] it is argued that it is highly unlikely that in the setup described above, which avoids redundancies, more general upscaling and downscaling rules will manage to meet the compatibility condition (4.20) needed for perfect reconstruction. In the linear case, where all details $q_i^{(j)}$ are contained in the *same* vector space, (4.20) is merely an algebraic condition on the coefficients involved in the definition of S, D, Q, R which can be solved. In the geometric case, the usage of parallel transport makes a fundamental difference.

4.4.2 Properties of Multiscale Transforms

Characterizing Smoothness by Coefficient Decay

One purpose of a multiscale decomposition of data is to read off properties of the original data. Classically, the faster the magnitude of detail coefficients $q_i^{(j)}$ decays as $j \to \infty$, the smoother the original data. A corresponding result for the interpolatory wavelets of Example 4.13 in the linear case is given by [9, Th. 2.7]. To state a result in the multivariate geometric case, let us first introduce new notation for interpolatory wavelets, superseding Example 4.13.

We consider an interpolatory subdivision rule S acting with dilation factor N on data $p\colon \mathbb{Z}^s \to \mathcal{M}$. We define data $p^{(j)}$ at level j as samples of a function $f\colon \mathbb{R}^s \to \mathcal{M}$, and construct detail vectors similar to Example 4.13:

$$p_i^{(j)} = f(N^{-j}i), \quad q^{(j)} = p^{(j)} \ominus Sp^{(j-1)}, \quad p^{(j)} = Sp^{(j-1)} \oplus q^{(j)}. \tag{4.21}$$

This choice is consistent with the decimation operator $Dp_i := p_{Ni}$. The difference to Example 4.13 is firstly that here we allow multivariate data, and secondly that we do not "forget" redundant information such as $q_{Ni}^{(j)} = 0$.

The result below uses the notation $\operatorname{Lip}\gamma$ for functions which are C^k with $k = \lfloor\gamma\rfloor$ and whose k-th derivatives are Hölder continuous of exponent $\gamma - k$. The critical Hölder regularity of a function f is the supremum of γ such that $f \in \operatorname{Lip}\gamma$.

Theorem 4.14 ([32, Th. 8]) *Assume that the interpolatory upscaling rule S, when acting linearly on data $p\colon \mathbb{Z}^s \to \mathbb{R}$, reproduces polynomials of degree $\le d$ and has limits of critial Hölder regularity r.*

Consider a continous function $f\colon \mathbb{R}^s \to \mathcal{M}$, and construct detail vectors $q^{(j)}$ at level j for the function $x \mapsto f(\sigma \cdot s)$ for some $\sigma > 0$ (whose local existence is guaranteed for some $\sigma > 0$).

Then $f \in \operatorname{Lip}\alpha$, $\alpha < d$ implies that detail vectors decay with $\sup_i \|q_i^{(j)}\| \le C \cdot N^{-\alpha j}$ as $j \to \infty$. Conversely, that decay rate together with $\alpha < r$ implies $f \in \operatorname{Lip}\alpha$. The constant is understood to be uniform in a compact set.

The manifold $\mathcal{M}$ can be any of the cases we defined $\oplus$ and $\ominus$ operations for. Of course, smoothness of $f\colon \mathbb{R}^s \to \mathcal{M}$ is only defined up to the intrinsic smoothness of $\mathcal{M}$ as a differentiable manifold. An example of an upscaling rule S is the four-point scheme with parameter $1/16$ mentioned in Example 4.7, which reproduces cubic polynomials and has critical Hölder regularity 2, cf. [21].

The proof is conducted in a coordinate chart (it does not matter which), and uses a linear vision of the theorem as an auxiliary tool. It further deals with the extensive technicalities which surround proximity inequalities in the multivariate case.

It is worth noting that Weinmann in [56] succeeded in transferring these ideas to the combinatorially irregular setting. The results are essentially the same, with the difference that one can find upscaling rules only up to smoothness $2 - \epsilon$.

Stability

Compression of data is a main application of multiscale decompositions, and it is achieved e.g. by thresholding or quantizing detail vectors. It is therefore important to know what effect these changes have when reconstruction is performed. What we basically want to know is whether reconstruction is Lipschitz continuous. In the linear case the problem does not arise separately, since the answer is implicitly contained in norms of linear operators. For the geometric multiscale transforms defined by upscaling operations S, R and downscaling operations D, Q according to (4.20), this problem is discussed by [33]. Consider data $p^{(j)}$ at level j with $p^{(j-1)} = Dp^{(j)}$ such that $\delta(p^{(j)}) \le C\mu^j$, for some $\mu < 1$. Consider recursive reconstruction of data $p^{(j)}$ from $p^{(0)}$ and details $q^{(1)}, \dots, q^{(j)}$ according to Eq. (4.19). Then there are constants C_k such that for modified details $\tilde{q}^{(j)}$, leading to modified data $\tilde{p}^{(j)}$, we have the local Lipschitz-style estimate

$$\sup_i \|p_i^{(0)} - \tilde{p}_i^{(0)}\| \le C_1,\ \sup_i \|q_i^{(k)} - \tilde{q}_i^{(k)}\| \le C_2\mu^k$$
$$\Longrightarrow \sup_i \|p_i^{(j)} - \tilde{p}_i^{(j)}\| \le C_3\Big(\sup_i \|p_i^{(0)} - \tilde{p}_i^{(0)}\| + \sum\nolimits_{k=1}^{j} \sup_i \|q_i^{(k)} - \tilde{q}_i^{(k)}\|\Big).$$

It refers to a coordinate chart of the manifold $\mathcal{M}$ (it does not matter which).

Approximation Order

For an interpolatory upscaling operator S, and data $p_i \in \mathcal{M}$ defined by sampling, $p_i = f(h \cdot i)$, we wish to know to what extent the original function differs from the limit created by upscaling the sample. We say that S has approximation order r, if there are $C > 0$, $h_0 >$ such that for all $h < h_0$

$$\sup_x d_{\mathcal{M}}(S^\infty f(x/h), f(x)) \leq C \cdot h^r.$$

It was shown by [62] that a manifold subdivision rule has in general the same approximation order as the linear rule we get by restricting S to linear data.

This question is directly related to stability as discussed above: Both f and $S^\infty p$ can be reconstructed from samples $p^{(j)}$, if $h = N^{-j}$: Detail vectors $q^{(k)}$, $k > j$, according to (4.21) reconstruct f, whereas details $\tilde{q}^{(k)} = 0$ reconstruct $S^\infty p$. Stability of reconstruction and knowledge of the asymptotic magnitude of details $q_i^{(k)}$, $k > j$ directly corresponds to approximation order. On basis of this relationship one can again show an approximation order equivalence result, cf. [28].

Conclusion

The preceding pages give an account of averages, subdivision, and multiscale transforms defined via geometric operations which are intrinsic for various geometries (metric spaces, Riemannian manifolds, Lie groups, and symmetric spaces). We reported on complete solutions in special cases (e.g. convergence of subdivision rules in Hadmard metric spaces) and on other results with much more general scope as regards the spaces and subdivision rules involved, but with more restrictions on the data they apply to.

Acknowledgements The author gratefully acknowledges the support of the Austrian Science fund through grant No. W1230, and he also wants to thank all colleagues whose work he was able to present in this survey.

References

1. Ballmann, W.: Lectures on Spaces of Nonpositive Curvature. Birkhäuser, Basel (1995)
2. Bump, D.: Lie Groups. Graduate Texts in Mathematics, vol. 225. Springer, New York (2004)
3. Catmull, E., Clark, J.: Recursively generated b-spline surfaces on arbitrary topological meshes. Computer-Aided Des. **10**, 350–355 (1978)
4. Cavaretta, A.S., Dahmen, W., Michelli, C.A.: Stationary Subdivision. Memoirs AMS, vol. 93. American Mathematical Society, Providence (1991)
5. Chaikin, G.: An algorithm for high speed curve generation. Comput. Graph. Image Process. **3**, 346–349 (1974)

6. de Rham, G.: Sur quelques fonctions différentiables dont toutes les valeurs sont des valeurs critiques. In: Celebrazioni Archimedee del Sec. XX (Siracusa, 1961), vol. II, pp. 61–65. Edizioni “Oderisi”, Gubbio (1962)
7. Deslauriers, G., Dubuc, S.: Symmetric iterative interpolation processes. Constr. Approx. **5**, 49–68 (1986)
8. do Carmo, M.P.: Riemannian Geometry. Birkhäuser, Basel (1992)
9. Donoho, D.L.: Interpolating wavelet transforms. Technical report (1992). http://www-stat.stanford.edu/donoho/Reports/1992/interpol.pdf
10. Donoho, D.L.: Wavelet-type representation of Lie-valued data. In: Talk at the IMI “Approximation and Computation” Meeting, May 12–17, 2001, Charleston, South Carolina (2001)
11. Doo, D., Sabin, M.: Behaviour of recursive division surfaces near extraordinary points. Computer-Aided Des. **10**, 356–360 (1978)
12. Duchamp, T., Xie, G., Yu, T.: On a new proximition condition for manifold-valued subdivision schemes. In: Fasshauer, G.E., Schumaker, L.L. (eds.) Approximation Theory XIV: San Antonio 2013. Springer Proceedings in Mathematics & Statistics, vol. 83, pp. 65–79. Springer, New York (2014)
13. Duchamp, T., Xie, G., Yu, T.: A necessary and sufficient proximity condition for smoothness equivalence of nonlinear subdivision schemes. Found. Comput. Math. **16**, 1069–1114 (2016)
14. Duchamp, T., Xie, G., Yu, T.: Smoothing nonlinear subdivision schemes by averaging. Numer. Algorithms **77**, 361–379 (2018)
15. Dyer, R., Vegter, G., Wintraecken, M.: Barycentric coordinate neighbourhoods in Riemannian manifolds, 15 pp. (2016), arxiv https://arxiv.org/abs/1606.01585
16. Dyer, R., Vegter, G., Wintraecken, M.: Barycentric coordinate neighbourhoods in Riemannian manifolds. In: Extended Abstracts, SoCG Young Researcher Forum, pp. 1–2 (2016)
17. Dyn, N.: Subdivision schemes in computer-aided geometric design. In: Light, W.A. (ed.) Advances in Numerical Analysis, vol. II, pp. 36–104. Oxford University Press, Oxford (1992)
18. Dyn, N., Levin, D.: Subdivision schemes in geometric modelling. Acta Numer. **11**, 73–144 (2002)
19. Dyn, N., Sharon, N.: A global approach to the refinement of manifold data. Math. Comput. **86**, 375–395 (2017)
20. Dyn, N., Sharon, N.: Manifold-valued subdivision schemes based on geodesic inductive averaging. J. Comput. Appl. Math. **311**, 54–67 (2017)
21. Dyn, N., Gregory, J., Levin, D.: A four-point interpolatory subdivision scheme for curve design. Comput. Aided Geom. Des. **4**, 257–268 (1987)
22. Ebner, O.: Convergence of iterative schemes in metric spaces. Proc. Am. Math. Soc. **141**, 677–686 (2013)
23. Ebner, O.: Stochastic aspects of refinement schemes on metric spaces. SIAM J. Numer. Anal. **52**, 717–734 (2014)
24. Fletcher, P.T., Joshi, S.: Principal geodesic analysis on symmetric spaces: Statistics of diffusion tensors. In: Sonka, M., et al. (eds.) Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis. Number 3117 in LNCS, pp. 87–98. Springer, Berlin (2004)
25. Grohs, P.: Smoothness analysis of subdivision schemes on regular grids by proximity. SIAM J. Numer. Anal. **46**, 2169–2182 (2008)
26. Grohs, P.: Smoothness equivalence properties of univariate subdivision schemes and their projection analogues. Numer. Math. **113**, 163–180 (2009)
27. Grohs, P.: Smoothness of interpolatory multivariate subdivision in Lie groups. IMA J. Numer. Anal. **27**, 760–772 (2009)
28. Grohs, P.: Approximation order from stability of nonlinear subdivision schemes. J. Approx. Theory **162**, 1085–1094 (2010)
29. Grohs, P.: A general proximity analysis of nonlinear subdivision schemes. SIAM J. Math. Anal. **42**(2), 729–750 (2010)
30. Grohs, P.: Stability of manifold-valued subdivision and multiscale transforms. Constr. Approx. **32**, 569–596 (2010)

31. Grohs, P., Wallner, J.: Log-exponential analogues of univariate subdivision schemes in Lie groups and their smoothness properties. In: Neamtu, M., Schumaker, L.L. (eds.) Approximation Theory XII: San Antonio 2007, pp. 181–190. Nashboro Press, Brentwood (2008)
32. Grohs, P., Wallner, J.: Interpolatory wavelets for manifold-valued data. Appl. Comput. Harmon. Anal. **27**, 325–333 (2009)
33. Grohs, P., Wallner, J.: Definability and stability of multiscale decompositions for manifold-valued data. J. Franklin Inst. **349**, 1648–1664 (2012)
34. Hardering, H.: Intrinsic discretization error bounds for geodesic finite elements. PhD thesis, FU Berlin (2015)
35. Helgason, S.: Differential geometry, Lie groups, and symmetric spaces. Academic, New York (1978)
36. Hüning, S., Wallner, J.: Convergence analysis of subdivision processes on the sphere (2019), submitted
37. Hüning, S., Wallner, J.: Convergence of subdivision schemes on Riemannian manifolds with nonpositive sectional curvature. Adv. Comput. Math. **45**, 1689–1709 (2019)
38. Itai, U., Sharon, N.: Subdivision schemes for positive definite matrices. Found. Comput. Math. **13**(3), 347–369 (2013)
39. Karcher, H.: Riemannian center of mass and mollifier smoothing. Commun. Pure Appl. Math. **30**, 509–541 (1977)
40. Moakher, M.: A differential geometric approach to the geometric mean of symmetric positive-definite matrices. SIAM J. Matrix Anal. Appl. **26**(3), 735–747 (2005)
41. Moosmüller, C.: C^1 analysis of Hermite subdivision schemes on manifolds. SIAM J. Numer. Anal. **54**, 3003–3031 (2016)
42. Moosmüller, C.: Hermite subdivision on manifolds via parallel transport. Adv. Comput. Math. **43**, 1059–1074 (2017)
43. Peters, J., Reif, U.: Subdivision Surfaces. Springer, Berlin (2008)
44. Reif, U.: A unified approach to subdivision algorithms near extraordinary vertices. Comput. Aided Geom. Des. **12**(2), 153–174 (1995)
45. Riesenfeld, R.: On Chaikin's algorithm. IEEE Comput. Graph. Appl. **4**(3), 304–310 (1975)
46. Sturm, K.-T.: Nonlinear martingale theory for processes with values in metric spaces of nonpositive curvature. Ann. Probab. **30**, 1195–1222 (2002)
47. Sturm, K.-T.: Probability measures on metric spaces of nonpositive curvature. In: Heat Kernels and Analysis on Manifolds, Graphs, and Metric Spaces, pp. 357–390. American Mathematical Society, Providence (2003)
48. Ur Rahman I., Drori, I., Stodden, V.C., Donoho, D.L., Schröder, P.: Multiscale representations for manifold-valued data. Multiscale Model. Simul. **4**, 1201–1232 (2005)
49. Wallner, J.: Smoothness analysis of subdivision schemes by proximity. Constr. Approx. **24**(3), 289–318 (2006)
50. Wallner, J.: On convergent interpolatory subdivision schemes in Riemannian geometry. Constr. Approx. **40**, 473–486 (2014)
51. Wallner, J., Dyn, N.: Convergence and C^1 analysis of subdivision schemes on manifolds by proximity. Comput. Aided Geom. Des. **22**, 593–622 (2005)
52. Wallner, J., Pottmann, H.: Intrinsic subdivision with smooth limits for graphics and animation. ACM Trans. Graph. **25**(2), 356–374 (2006)
53. Wallner, J., Yazdani, E.N., Grohs, P.: Smoothness properties of Lie group subdivision schemes. Multiscale Model. Simul. **6**, 493–505 (2007)
54. Wallner, J., Yazdani, E.N., Weinmann, A.: Convergence and smoothness analysis of subdivision rules in Riemannian and symmetric spaces. Adv. Comput. Math. **34**, 201–218 (2011)
55. Weinmann, A.: Nonlinear subdivision schemes on irregular meshes. Constr. Approx. **31**, 395–415 (2010)
56. Weinmann, A.: Interpolatory multiscale representation for functions between manifolds. SIAM J. Math. Anal. **44**, 172–191 (2012)

57. Weinmann, A.: Subdivision schemes with general dilation in the geometric and nonlinear setting. J. Approx. Theory **164**, 105–137 (2012)
58. Welk, M., Weickert, J., Becker, F., Schnörr, C., Burgeth, B., Feddern, C.: Median and related local filters for tensor-valued images. Signal Process. **87**, 291–308 (2007)
59. Xie, G., Yu, T.P.-Y.: Smoothness equivalence properties of manifold-valued data subdivision schemes based on the projection approach. SIAM J. Numer. Anal. **45**, 1200–1225 (2007)
60. Xie, G., Yu, T.P.-Y.: Smoothness equivalence properties of general manifold-valued data subdivision schemes. Multiscale Model. Simul. **7**, 1073–1100 (2008)
61. Xie, G., Yu, T.P.-Y.: Smoothness equivalence properties of interpolatory Lie group subdivision schemes. IMA J. Numer. Anal. **30**, 731–750 (2010)
62. Xie, G., Yu, T.P.-Y.: Approximation order equivalence properties of manifold-valued data subdivision schemes. IMA J. Numer. Anal. **32**, 687–700 (2011)

Chapter 5
Variational Methods for Discrete Geometric Functionals

Henrik Schumacher and Max Wardetzky

Contents

Abstract While consistent discrete notions of curvatures and differential operators have been widely studied, the question of whether the resulting minimizers converge to their smooth counterparts still remains open for various geometric functionals. Building on tools from variational analysis, and in particular using the notion of Kuratowski convergence, we offer a general framework for treating convergence of minimizers of (discrete) geometric functionals. We show how to apply the resulting machinery to minimal surfaces and Euler elasticae.

5.1 Introduction

Classically, the notion of curvature relies on second derivatives (at least in a weak sense) and thus requires sufficient smoothness of the underlying space. Simplicial manifolds, such as triangle meshes, offer a natural choice for replacing smooth manifolds by discrete ones, but lack the required smoothness. A challenge, therefore, is to provide discrete notions of curvatures for simplicial manifolds that

H. Schumacher
RWTH Aachen University, Institute for Mathematics, Aachen, Germany
e-mail: schumacher@instmath.rwth-aachen.de

M. Wardetzky (✉)
Institute of Numerical and Applied Mathematics, University of Göttingen, Göttingen, Germany
e-mail: wardetzky@math.uni-goettingen.de

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_5

(1) mimic the structural properties of their smooth counterparts, such as preserving the relation between total curvature and topological properties, and (2) are consistent in the sense of converging to their smooth counterparts in the limit of mesh refinement. Both *structure-preserving* and *consistent* notions of discrete curvatures have at least been considered since the work of Alexandrov; new notions have been brought forward over the past decades, see, e.g., [4, 12, 14, 20, 23, 44, 47].

Discrete mean curvature provides a prominent example of discrete curvatures. Mean curvature of surfaces is intimately linked to Laplacians on manifolds since the mean curvature vector of a smooth hyper-surface is equal to the Laplace–Beltrami operator applied to surface positions. Requiring to maintain this relation in the discrete case therefore links consistent discrete notions of mean curvature to consistent notions of discrete Laplacians. Dziuk's seminal work [27] shows convergence of solutions to the Poisson problem for the Laplace–Beltrami operator in the limit of mesh refinement using linear Lagrange finite elements on triangle meshes. Dziuk's work treats the case of inscribed meshes, i.e., the case where mesh vertices reside on the smooth limit surface. This condition was later relaxed to meshes that are nearby a smooth limit surface in the sense that both surface positions and surface normals converge to their smooth counterparts under mesh refinement. In fact, given convergence of positions of a sequence of polyhedral surfaces to a smooth limit surface, the following are equivalent: (1) convergence of surface normals, (2) convergence of Laplace–Beltrami operators in norm, and (3) convergence of metric tensors, see [34, 54].

As pointed out by Fu, Cohen-Steiner, and Morvan, the relation between consistent discrete notions of curvatures and convergence of both positions and normals is also central when studying convergence of curvatures *in the sense of measures*, see [21, 22, 31, 43]. Indeed, convergence in the sense of measures, or (in a similar spirit) weak convergence, is the best one can hope for in general. For example, mean curvature vectors only converge as elements of the Sobolev space H^{-1} (the dual of H^1_0); even convergence in L^2 cannot be expected in general, see [34, 54]. Pointwise convergence of discrete curvatures can only be expected under special assumptions, see, e.g., [5, 15, 33, 37, 56].

Convergence of minimizers of discrete geometric functionals is significantly more involved than convergence of curvatures. Perhaps one of the most prominent examples is that of *minimal surfaces*, i.e., the problem of finding a surface of least (or critical) area among all surfaces of prescribed topology spanning a prescribed boundary curve. In the smooth setting, Radó [46] and Douglas [26] independently solved this problem for disk-like, immersed surfaces. Douglas' existence proof is based on minimizing the Dirichlet energy of conformal surface parameterizations. Douglas' ideas have been inspirational for proving convergence of discrete minimal surfaces using the finite element method (FEM) for disk-like and cylindrical surfaces [28, 29, 35, 45, 53, 55]. An entirely different approach for showing convergence of disk-like minimal surfaces to their smooth counterparts can be established using tools from discrete differential geometry (DDG) [13, 19]. Convergence of discrete minimal surfaces to smooth minimal surfaces of *arbitrary topology* has

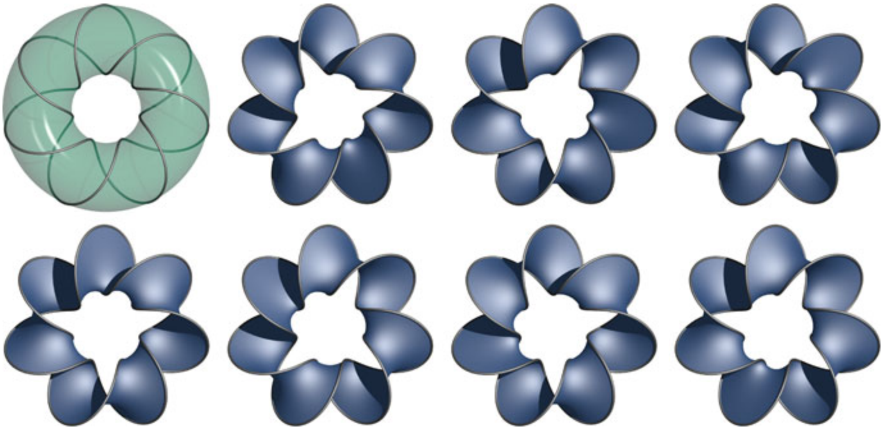


Fig. 5.1 Septfoil torus knot (top left), which exhibits a sevenfold symmetry and minimal surfaces (with the topology of Moebius strips) spanning this knot. Notice that the solution space maintains the sevenfold symmetry, while each individual minimal surface lacks this symmetry

only been established recently using the notion of Kuratowski convergence [51]. The latter work forms the basis for our survey.

Minimal surfaces offer a prime example for geometric variational problems. Apart from their relation to physics (as models of soap films spanning a given wire), they pose several mathematical challenges. The first challenge is non-uniqueness of solutions. Indeed, consider the minimal surface problem of spanning a Moebius strip into the septfoil knot (the (7, 2)-torus knot) illustrated in Fig. 5.1. Notice that the solution space maintains the sevenfold symmetry of the boundary curve, while each individual minimal surface lacks this symmetry. An even more severe example of non-uniqueness is provided by Morgan's minimal surface(s), where boundary conditions exhibit a continuous rotational symmetry, which is maintained for the solution space but lost for each individual minimizer, see Fig. 5.2.

Another peculiarity is non-existence of solutions for the minimal surface problem. Consider the example of Fig. 5.3, where the topology is prescribed by a sphere with six holes. Within this topological class, there exist minimizing sequences, which, however, do not converge in C^0, but leave the topological class (resulting in six disjoint disks).

An additional complication arises from the fact that the area functional is *non-convex*. Consider the example in Fig. 5.4, where a given boundary curve gets spanned by Enneper's classical minimal surface, see Fig. 5.4, top. There exist two other minimal surfaces (Fig. 5.4, left and right), supported on the same boundary, which are congruent to one another, with a smaller area than that of Enneper's surface. When the positions of these two surfaces are linearly interpolated in 3-space, one obtains a surface of larger surface area than that of Enneper's surface, see Fig. 5.4, bottom. Clearly, this could not occur if surface area were a convex functional.

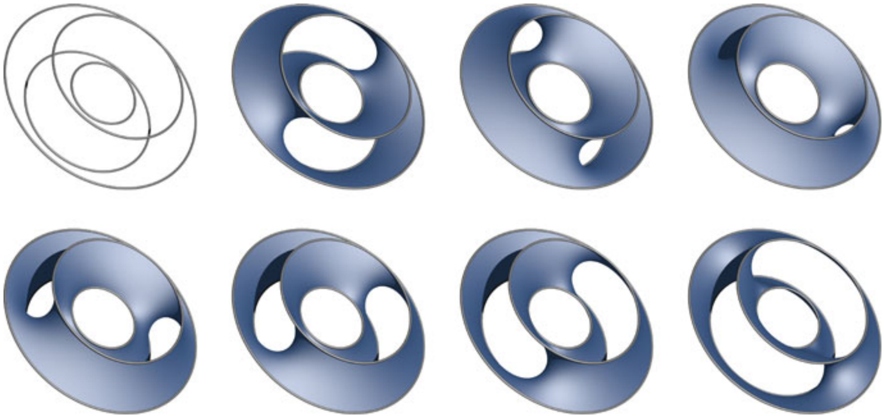


Fig. 5.2 Morgan's minimal surface(s): The boundary curves (top left) exhibit continuous rotational symmetry, which is maintained for the solution space but lost for each individual minimizer

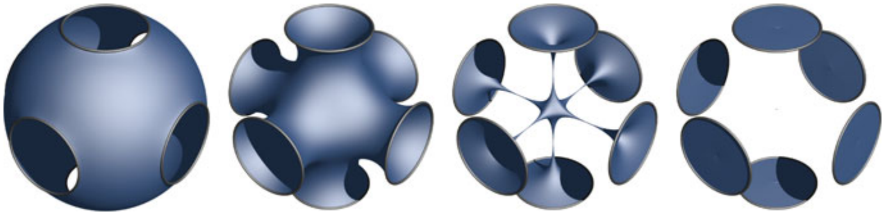


Fig. 5.3 Within the class of surfaces homeomorphic to a sphere with six holes, there exist minimizing sequences; however, the minimizer leaves the topological class resulting in six disjoint disks

Finally, working with triangle meshes when approximating classical minimal surfaces leads to complications with boundary conditions since mesh boundaries are polygonal curves instead of smooth ones, leading to so-called *non-conforming* methods or *variational crimes*, see Fig. 5.5. Of course, the issue of approximating a smooth boundary curve by a polygonal curve disappears in the limit of refinement; nonetheless, one needs to attend to the underlying issue that the space of smooth manifolds and the space of discrete manifolds need to be embedded in a common *shape space* $\mathcal{X}$ in which (discrete and smooth) minimizers can be compared. One of the subtleties is to choose the *topology of shape space*. One likes to choose the topology of shape space as fine as possible, since finer topologies lead to stronger convergence results. However, choosing the topology of $\mathcal{X}$ too fine might prohibit the option to embed both smooth and discrete manifolds into the common space $\mathcal{X}$.

Shape spaces naturally arise when studying minimizers of geometric functionals. Within the context of shape spaces, it is natural to view manifolds (or other shapes) as geometric objects, i.e., independently of any particular choice of parameterization. Thus, one considers the (often infinite-dimensional) configuration space $\mathcal{C}$ of parameterized manifolds modulo the (then also infinite-dimensional) group $\mathcal{G}$

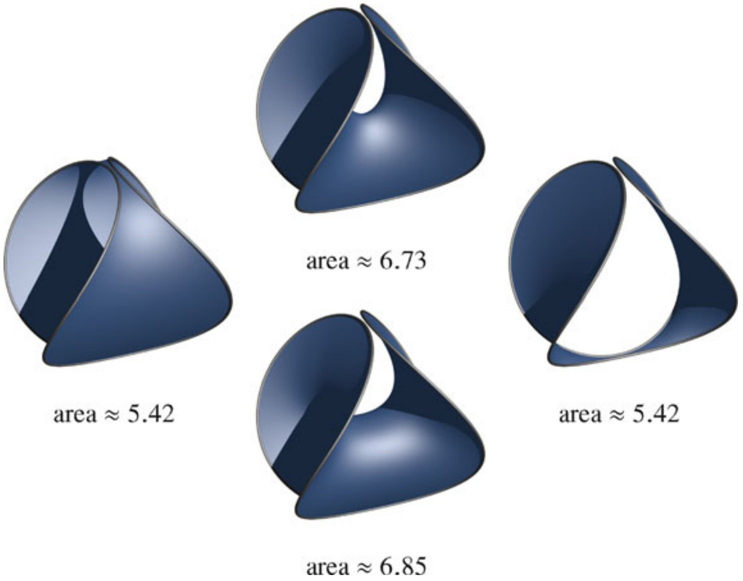


Fig. 5.4 Non-convexity of the area functional: A space curve gets spanned by Enneper's surface (top), and two other surfaces (left and right) with the same boundary but strictly lower area. Interpolating the positions of these two surfaces in 3-space results in a surface of larger area than that of Enneper's surface (bottom)

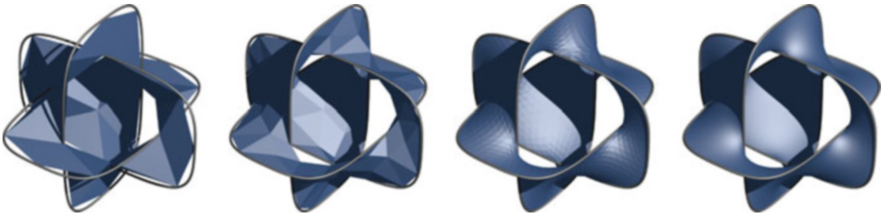


Fig. 5.5 Discrete minimal surfaces lead to non-conforming methods since smooth boundaries (here: Borromean rings) get approximated by polygonal curves

of reparameterizations. Suppose $\mathcal{C}$ is equipped with a metric such that $\mathcal{G}$ acts by isometries—a natural assumption in view of the parameterization independence of geometric functionals. With $\mathcal{G}$ acting by isometries on $\mathcal{C}$, the metric structure on $\mathcal{C}$ can be factored to the (quotient) *shape space* $\mathcal{X} := \mathcal{C}/\mathcal{G}$; however, it is not clear (and often indeed false) that this quotient "metric" remains to be definite. For shape spaces, the resulting problem of degenerate metrics has been observed in the seminal work by Michor and Mumford when working with L^2-type metrics on shape space, see [39, 40]. This issue can be overcome when considering curvature-weighted L^2-metrics, see [6–9, 41], or Sobolev metrics, see [10, 11, 17, 18, 38]. We follow an approach similar to the latter. Indeed, in order to embed both discrete and smooth manifolds into a common space, we work with the configuration space $\mathcal{C}$ of

Lipschitz immersions of an abstract, compact, k-dimensional Riemannian manifold (Σ, g) into $\mathbb{R}^m$, and we work with the group $\mathcal{G}$ of Lipschitz diffeomorphisms acting on Σ. The resulting quotient metric on the shape space $\mathcal{X} = \mathcal{C}/\mathcal{G}$ is then indeed definite and thus *not* degenerate. For details, see Sect. 5.2.

In order to study *convergence of discrete minimizers* to their smooth counterparts, we require tools for reconstructing smooth manifolds from discrete ones and—vice versa—tools for approximating (or sampling) smooth manifolds by discrete ones. These tools are provided by respective *reconstruction* and *sampling operators* that take discrete manifolds to smooth ones nearby and vice versa. The notion of "nearby" needs to be understood in the sense of the metric on configuration space $\mathcal{C}$ that gives rise to the quotient metric on shape space discussed above. Providing the requisite reconstruction and sampling operators and studying their properties is technically somewhat involved, with details heavily depending on the choice of metric on configuration space. We choose to omit these technicalities in our exposition and rather discuss the overall utility of these operators in the setting of geometrically nonlinear variational problems.

For showing convergence, it is natural to study *consistency* of both reconstruction and sampling operators. Additionally, the fact that smooth and discrete manifolds have different regularity (or smoothness) properties also requires the notion of *proximity*, which measures the "nearness" of smooth and discrete manifolds when mapped to the common shape space $\mathcal{X}$. Together, consistency and proximity imply "nearness" of lower level sets of (smooth and discrete) energy functionals. In order to obtain convergence of (almost) minimizers, one additionally requires certain growth conditions of the energy functionals near minimizers. Such growth conditions can be subsumed within the notion of *stability*. The concepts of consistency, proximity, and stability are outlined in Sect. 5.4.

The notions of consistency, proximity, and stability enable the study of *convergence in the sense of Kuratowski*. In a nutshell, Kuratowski convergence implies:

(A) Every smooth minimizer is a limit point of discrete minimizers.
(B) Every accumulation point of discrete minimizers is a smooth minimizer.

The concept of Kuratowski convergence is related, but not identical, to the concept of Γ-convergence, see Sect. 5.3. Kuratowski convergence is a concept for studying convergence of *sets* and thus naturally lends itself to incorporating a priori information about (regularity of) minimizers. Indeed, Kuratowski convergence allows for focusing on *restricted* subsets of configuration space and thus avoids the need for constructing so-called recovery sequences for *every* element in configuration space.

The concept of Kuratowski convergence can be applied to a variety of geometric energy functionals. Minimal surfaces provide one such example—with the result of obtaining properties (A) and (B) above, see Sect. 5.5. Another prominent example is provided by Euler's elasticae, i.e., the problem of finding a curve of *prescribed* total length and *prescribed* positions and tangents at the end points that minimizes *bending energy*, see Fig. 5.6. Since Euler elasticae posses rather

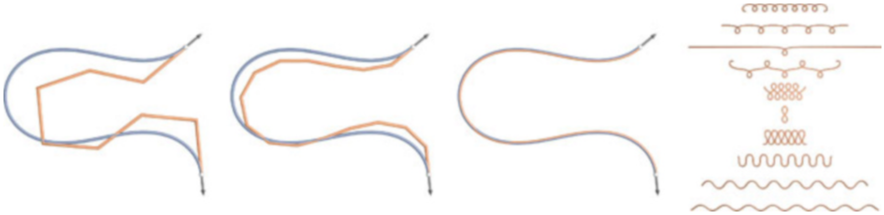


Fig. 5.6 Discrete Euler elasticae converging to a smooth Euler elastica under refinement (first three images) and a one-parameter family of resulting elasticae considered by Euler (right)

strong regularity properties (both in the smooth and discrete setting), we are able to apply our machinery for even deducing *Hausdorff convergence* of discrete (almost) minimizers to their smooth counterparts, see Sect. 5.5.

5.2 Shape Space of Lipschitz Immersions

In order to treat both discrete (simplicial) and smooth manifolds as elements of a common space, we work with the configuration space C of Lipschitz immersions of an abstract, compact, k-dimensional Riemannian manifold (Σ, g) into $\mathbb{R}^m$, and we work with the group $\mathcal{G}$ of Lipschitz diffeomorphisms acting on Σ. More precisely, define the fiber bundle $\pi : P_\Sigma \to \Sigma$ by $P_\Sigma|_x := P(T_x\Sigma)$ for all $x \in \Sigma$, where $P(T_x\Sigma)$ denotes the manifold of positive-definite, symmetric bilinear forms on the tangent space $T_x\Sigma$. Notice that for a k-dimensional real vector space V, the set $P(V)$ of positive-definite, symmetric bilinear forms on V is an open subset of the vector space $\operatorname{Sym}(V)$ of symmetric bilinear forms on V. Thus, $P(V)$ is a smooth manifold with tangent bundle $T_bP(V) = \operatorname{Sym}(V)$. One defines a Riemannian metric g_P on $P(V)$ via

$$g_P|_b(X, Y) := \langle X, Y\rangle_b \quad \text{for all } X,\ Y \in T_bP(V) = \operatorname{Sym}(V),$$

where $\langle\cdot,\cdot\rangle_b$ denotes the inner product on $\operatorname{Sym}(V)$ that is induced by b. This Riemannian structure is indeed one of the commonly used metrics on the manifold of inner products (see [42] and references therein). It gives rise to a distance function d_P on $P(V)$ that can be computed explicitly as

$$d_P(b, c) = \left\| \log\left(\mathbf{B}^{-1/2}\mathbf{C}\mathbf{B}^{-1/2}\right)\right\|_F,$$

where $\mathbf{B}$ and $\mathbf{C}$ denote matrix representations (Gram matrices) of the inner products b and c, respectively, $\mathbf{B}^{\frac{1}{2}}$ is any symmetric square root of $\mathbf{B}$, and $\|\cdot\|_F$ denotes the Frobenius norm. This distance function, in turn, can be used for defining the metric space $(\mathcal{P}(\Sigma), d_P)$ of Riemannian metrics that have bounded distortion with respect to the given Riemannian metric g on Σ:

$$\mathcal{P}(\Sigma) := \{b\colon \Sigma \to P_\Sigma \mid b \text{ is measurable}, d_{\mathcal{P}}(b, g) < \infty\} \quad \text{and}$$
$$d_{\mathcal{P}}(b_1, b_2) := \operatorname*{ess\,sup}_{x\in\Sigma} d_P\big(b_1|_x, b_2|_x\big).$$

Then, following [48], we define the *space of Lipschitz immersions* as

$$\mathrm{Imm}(\Sigma; \mathbb{R}^m) := \{f \in W^{1,\infty}(\Sigma; \mathbb{R}^m) \mid f^\# g_0 \in \mathcal{P}(\Sigma)\}$$

and equip it with the distance

$$\begin{aligned} d_{\mathrm{Imm}}(f_1, f_2) := {} & \|f_1 - f_2\|_{L^\infty} + d_{\mathcal{P}}(f_1^\# g_0, f_2^\# g_0) \\ & + \operatorname*{ess\,sup}_{x\in\Sigma, u\in T_x\Sigma\setminus\{0\}} \measuredangle(\mathrm{d}f_1|_x(u), \mathrm{d}f_2|_x(u)) \end{aligned}$$

for $f_1, f_2 \in \mathrm{Imm}(\Sigma; \mathbb{R}^m)$. Here $f^\# g_0$ denotes the pullback of the induced metric on $f(\Sigma)$ to Σ, and $\measuredangle(\mathrm{d}f_1|_x(u), \mathrm{d}f_2|_x(u))$ denotes the unsigned angle between the lines spanned by $\mathrm{d}f_1|_x(u)$ and $\mathrm{d}f_2|_x(u)$ (considered as lines through $0 \in \mathbb{R}^m$). The idea behind this distance is that it measures deviation of surface positions (first term), deviation of tangent spaces (which is related to deviation of normals in codimension one, third term), and distortion of metric tensors (second term). In order to treat boundary conditions, let Σ be a compact, smooth manifold with boundary. We define the configuration space of *strong Lipschitz immersions* by

$$\mathcal{C} := \mathrm{Imm}^\times(\Sigma; \mathbb{R}^m) := \{f \in \mathrm{Imm}(\Sigma; \mathbb{R}^m) \mid f|_{\partial\Sigma} \in \mathrm{Imm}(\partial\Sigma; \mathbb{R}^m)\}$$

and equip it with the graph metric

$$d^\times_{\mathrm{Imm}}(f_1, f_2) := d_{\mathrm{Imm}}(f_1, f_2) + d_{\mathrm{Imm}}(f_1|_{\partial\Sigma}, f_2|_{\partial\Sigma}), \quad \text{for } f_1, f_2 \in \mathrm{Imm}^\times(\Sigma; \mathbb{R}^m).$$

Finally, in order to mod out parameterizations, we define the group of *Lipschitz diffeomorphisms* by

$$\mathcal{G} := \mathrm{Diff}(\Sigma) := \{\varphi \in W^{1,\infty}_g(\Sigma; \Sigma) \mid \varphi \text{ is a bi-Lipschitz homeomorphism}\}.$$

This group acts by isometries on $\mathcal{C} = \mathrm{Imm}^\times(\Sigma; \mathbb{R}^m)$, thus giving rise to the *shape space of Lipschitz immersions*, defined as the quotient

$$\mathcal{X} := \mathrm{Shape}(\Sigma; \mathbb{R}^m) := \mathrm{Imm}^\times(\Sigma; \mathbb{R}^m)/\mathrm{Diff}(\Sigma).$$

One can show that the resulting quotient metric on $\mathcal{X} = \mathcal{C}/\mathcal{G}$ is then indeed definite and does *not* exhibit the problem of degeneration. For details we refer to [51].

5.3 Notions of Convergence for Variational Problems

As illustrated by the examples given in Figs. 5.1 and 5.2, it does not suffice to consider convergence of isolated minimizers. Rather, for geometric variational problems, one is sometimes required to consider *sets* of minimizers. Therefore, we review notions of convergence of sets, for which the notions of *Hausdorff convergence* and *Kuratowski convergence* are most natural.

Let $(\mathcal{X}, d_{\mathcal{X}})$ be a metric space. For a subset $\Omega \subset \mathcal{X}$ and a radius $r > 0$, we define the *r-thickening* of Ω by

$$\bar{B}(\Omega, r) := \bigcup_{x \in \Omega} \{y \in \mathcal{X} \mid d_{\mathcal{X}}(x, y) \leq r\}.$$

The *Hausdorff distance* between two sets Ω_1 and $\Omega_2 \subset \mathcal{X}$ is given by

$$\operatorname{dist}_{\mathcal{X}}(\Omega_1, \Omega_2) := \inf\bigl\{r > 0 \mid \Omega_1 \subset \bar{B}(\Omega_2, r) \text{ and } \Omega_2 \subset \bar{B}(\Omega_1, r)\bigr\},$$

and a sequence of sets $\Omega_n \subset \mathcal{X}$ *Hausdorff converges* to $\Omega \subset \mathcal{X}$ if and only if $\operatorname{dist}_{\mathcal{X}}(\Omega_n, \Omega_0) \overset{n\to\infty}{\longrightarrow} 0$.

Hausdorff convergence is perhaps one of the most elementary and expressive notions of set convergence. In the following, we also require another, somewhat weaker notion of set convergence.

Definition 5.1 Let $\mathcal{X}$ be a topological space and denote by $\mathfrak{U}(x)$ the set of all open neighborhoods of $x \in \mathcal{X}$. For a sequence of sets $(\Omega_n)_{n\in\mathbb{N}}$ in $\mathcal{X}$ one defines the *upper limit* $\operatorname{Ls}_{n\to\infty} \Omega_n$ and the *lower limit* $\operatorname{Li}_{n\to\infty} \Omega_n$, respectively, as follows:

$$\operatorname*{Ls}_{n\to\infty} \Omega_n := \{x \in X \mid \forall U \in \mathfrak{U}(x)\, \forall n \in \mathbb{N}\, \exists k \geq n\colon\ U \cap \Omega_k \neq \emptyset\} \quad \text{and}$$

$$\operatorname*{Li}_{n\to\infty} \Omega_n := \{x \in X \mid \forall U \in \mathfrak{U}(x)\, \exists n \in \mathbb{N}\, \forall k \geq n\colon\ U \cap \Omega_k \neq \emptyset\}.$$

If $\Omega := \operatorname{Ls}_{n\to\infty} \Omega_n = \operatorname{Li}_{n\to\infty} \Omega_n$ agree, one says that Ω_n *Kuratowski converges to* Ω and writes $\operatorname{Lt}_{n\to\infty} \Omega_n = \Omega$.

Both lower and upper limit are closed sets, and one has $\operatorname{Li}_{n\to\infty} \Omega_n \subset \operatorname{Ls}_{n\to\infty} \Omega_n$. One often refers to $\operatorname{Ls}_{n\to\infty} \Omega_n$ as the *set of cluster points* since x is an element of the upper limit if and only if there is a sequence of elements $x_n \in \Omega_n$ that has x as a cluster point, i.e., there is a subsequence $(x_{n_k})_{k\in\mathbb{N}}$ that converges to x as $k \to \infty$. A point x is contained in the lower limit if and only if there is a sequence of elements $x_n \in \Omega_n$ that has x as its limit.

Hausdorff convergence is stronger than Kuratowski convergence: If Ω_n Hausdorff converges to Ω, then it also Kuratowski converges to Ω. The converse is in general not true. For example, let $(e_n)_{n\in\mathbb{N}}$ be a countable orthonormal system in an infinite-dimensional Hilbert space. Then $\Omega_n := \{e_n\} \cup \{0\}$ Kuratowski converges to $\{0\}$. But Ω_n and Ω_m have Hausdorff distance equal to 1 whenever $m \neq n$, which prohibits Hausdorff convergence.

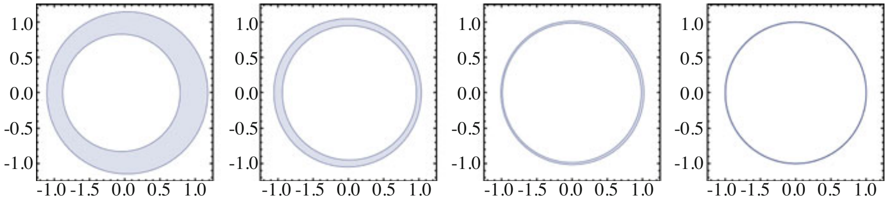


Fig. 5.7 The *almost minimizing sets* $\arg\min^{\frac{3}{n}}(\mathcal{F}_n)$ of the tilted Mexican hat potentials $\mathcal{F}_n(x) = (1-|x|^2)^2-(-1)^n\frac{1}{n}\frac{x_1}{1+|x|^2}$ Hausdorff converge to the set of minimizers of the Mexican hat potential $\mathcal{F}(x) = (1-|x|^2)^2$ (the unit circle). In contrast, while $\mathcal{F}_n$ Γ-converges to $\mathcal{F}$, the respective minimizers ($\{(1,0)\}$ for even n and $\{(-1,0)\}$ for odd n) *do not* converge to the minimizers of $\mathcal{F}$

The notion of Γ-convergence of *functions* is related to the notion of Kuratowski convergence of sets. A sequence of functions $\mathcal{F}_n\colon \mathcal{X} \to\ ]-\infty,\infty]$ is said to *Γ-converge* to $\mathcal{F}\colon \mathcal{X} \to\ ]-\infty,\infty]$ if and only if the sequence of epigraphs $\operatorname{epi}(\mathcal{F}_n)$ Kuratowski converges to $\operatorname{epi}(\mathcal{F})$. Here, the epigraph of a function $\mathcal{F}$ is defined as $\operatorname{epi}(\mathcal{F}) := \{(x,t) \in \mathcal{X} \times\ ]-\infty,\infty] \mid \mathcal{F}(x) \leq t\}$. By denoting the set of minimizers of $\mathcal{F}$ by

$$\arg\min(\mathcal{F}) := \{x \in \mathcal{X} \mid \forall y \in \mathcal{X}\colon\ \mathcal{F}(x) \leq \mathcal{F}(y)\},$$

we may state the principal property of Γ-convergence as follows:

$$\text{If } \mathcal{F}_n\ \Gamma\text{-converges to } \mathcal{F}, \text{ then } \operatorname*{Ls}_{n\to\infty} \arg\min(\mathcal{F}_n) \subset \arg\min(\mathcal{F}). \tag{5.1}$$

One often rephrases this by saying that cluster points of minimizers of $\mathcal{F}_n$ are minimizers of $\mathcal{F}$. But neither does Γ-convergence imply equality in (5.1), nor does it imply that $\arg\min(\mathcal{F}_n)$ Kuratowski converges to $\arg\min(\mathcal{F})$. As an example, consider the functions $\mathcal{F}_n(x) = (1-|x|^2)^2 - (-1)^n\frac{1}{n}\frac{x_1}{1+|x|^2}$ and $\mathcal{F}(x) = (1-|x|^2)^2$ (see also Fig. 5.7). It is not hard to show that $\mathcal{F}_n$ Γ-converges to $\mathcal{F}$. Notice, however, that $\arg\min(\mathcal{F})$ is the unit circle while the minimizers of $\mathcal{F}_n$ are given by $\arg\min(\mathcal{F}_n) = \{((-1)^n, 0)\}$. We have $\arg\min(\mathcal{F}_n) \subset \arg\min(\mathcal{F})$, but the upper and lower limits do not coincide:

$$\operatorname*{Ls}_{n\to\infty} \arg\min(\mathcal{F}_n) = \{(1,0),\ (-1,0)\} \quad \text{while} \quad \operatorname*{Li}_{n\to\infty} \arg\min(\mathcal{F}_n) = \emptyset.$$

So $\arg\min(\mathcal{F}_n)$ does neither Kuratowski converge nor Hausdorff converge to any set, let alone to $\arg\min(\mathcal{F})$. We point out that this lack of convergence of $\arg\min(\mathcal{F}_n)$ to $\arg\min(\mathcal{F})$ is basically caused by symmetry breaking of $\mathcal{F}_n$ vs. $\mathcal{F}$. This is actually very similar to what happens when one discretizes parameterization invariant optimization problems for immersed manifolds.

Despite these issues, the functions $\mathcal{F}_n$ actually do carry quite a lot of information about $\arg\min(\mathcal{F})$. In order to make this precise, we introduce the notion of *δ-minimizers*:

$$\arg\min{}^{\delta}(\mathcal{F}) := \{x \in \mathcal{X} \mid \forall y \in \mathcal{X}\colon\ \mathcal{F}(x) \leq \mathcal{F}(y) + \delta\}.$$

Utilizing the notion of δ-minimizers, Eq. (5.1) can be strengthened to the following statement (see [24, Theorem 7.19]):

Theorem 5.2 *Suppose that $\mathcal{F}_n$ Γ-converges to $\mathcal{F}$. Moreover, suppose that $\mathcal{F}$ is not constantly equal to ∞, that* $\arg\min(\mathcal{F})$ *is nonempty, and that* $\inf(\mathcal{F}) = \lim_{n\to\infty} \inf(\mathcal{F}_n)$. *Then one has*

$$\arg\min(\mathcal{F}) = \bigcap_{\delta>0} \operatorname*{Lt}_{n\to\infty} \arg\min{}^{\delta}(\mathcal{F}_n) = \operatorname*{Lt}_{\delta\searrow 0} \operatorname*{Lt}_{n\to\infty} \arg\min{}^{\delta}(\mathcal{F}_n).$$

The fact that the limit $\mathrm{Lt}_{\delta\searrow 0}$ is applied *after* $\mathrm{Lt}_{n\to\infty}$ is somewhat unfortunate. In in the example of Fig. 5.7, we actually have a much stronger convergence statement:

$$\arg\min(\mathcal{F}) = \operatorname*{Lt}_{n\to\infty} \arg\min{}^{\delta_n}(\mathcal{F}_n) \quad \text{with} \quad \delta_n := \tfrac{3}{n},$$

and the convergence is even in Hausdorff distance. From the point of view of numerical analysis, this last convergence result is very desirable, since numerical algorithms can usually only compute δ-minimizers. In particular, a rough knowledge on the size of δ_n provides a guide on how to choose the stopping criterion for numerical optimization algorithms.

This concludes our brief review of convergence concepts for variational problems. For more comprehensive treatments of the relationship between Kuratowski or K-convergence and epi- or Γ-convergence, we refer the reader to [49] and [24].

5.4 Practitioner's Guide to Kuratowski Convergence of Minimizers

Let $\mathcal{F}\colon C \to \mathbb{R}$ be a function that we seek to minimize. Abbreviate the set of minimizers by

$$\mathcal{M} := \arg\min(\mathcal{F}).$$

We aim at approximating $\mathcal{M}$ by solving optimization problems for a sequence of functions $\mathcal{F}_n\colon C_n \to \mathbb{R}$ with minimizers $\mathcal{M}_n := \arg\min(\mathcal{F}_n)$. Therefore, we have to compare subsets of C and C_n in some way. Suppose that we are given certain "measurements" or "test mappings" $\Psi\colon C \to \mathcal{X}$ and $\Psi_n\colon C_n \to \mathcal{X}$ with values in a common metric space $(\mathcal{X}, d_{\mathcal{X}})$, e.g., the shape space from Sect. 5.2. Then the Hausdorff distance between $\Psi(\mathcal{M})$ and $\Psi_n(\mathcal{M}_n)$ is a canonical measure of approximation quality.

When analyzing the relationship between the optimization problems for $\mathcal{F}$ and $\mathcal{F}_n$, one often takes advantage of a priori information. Indeed, it is quite typical for shape optimization problems that minimizers (or at least certain representatives of each orbit of minimizers under the reparameterization group) have significantly higher regularity than generic elements of the energy space. Minimal surfaces lend themselves once more to good examples; see, e.g., [25, 32] for their regularity theory. Usually, regularity of minimizers of the smooth problem is measured in terms of uniform bounds on Sobolev or Hölder norms. Often, these norms are stronger than the norm of the energy space. This implies that, compared to generic elements of the energy space, minimizers of the smooth problem can be more accurately approximated by discrete entities. We are going to incorporate such *a priori information* in the form of subsets $\mathcal{A} \subset C$, $\mathcal{A}_n \subset C_n$ that, on the one hand, may be assumed to be "small" (e.g., relatively compact), but, on the other hand, are sufficiently representative for the minimizers in the sense that $\mathcal{A}$ and $\mathcal{A}_n$ contain minimizing sequences and such that the following inclusions hold:

$$\Psi(\mathcal{M}) \subset \Psi(\mathcal{A} \cap \mathcal{M}) \quad \text{and} \quad \Psi_n(\mathcal{M}_n) \subset \Psi_n(\mathcal{A}_n \cap \mathcal{M}_n).$$

The advantage of a priori information is that it allows for quantifying various types of discretization errors in a uniform way. In order to make this precise, suppose that we have at our disposal *sampling operators* $\mathcal{S}_n \colon C \to C_n$, transferring smooth objects to discrete ones, and *reconstruction operators* $\mathcal{R}_n \colon C_n \to C$, transferring discrete objects to smooth ones. Sufficiently detailed a priori information then allows for controlling the *sampling consistency error*

$$\delta_n^{\mathcal{S}} := \sup_{a \in \mathcal{A}} \max \left\{ 0, \, \mathcal{F}_n \circ \mathcal{S}_n(a) - \mathcal{F}(a) \right\},$$

the *reconstruction consistency error*

$$\delta_n^{\mathcal{R}} := \sup_{a \in \mathcal{A}_n} \max \left\{ 0, \, \mathcal{F} \circ \mathcal{R}_n(a) - \mathcal{F}_n(a) \right\},$$

and thus the *total consistency error*

$$\delta_n := \delta_n^{\mathcal{S}} + \delta_n^{\mathcal{R}}.$$

From the very definition of these errors and from the presence of minimizing sequences of $\mathcal{F}$ in $\mathcal{A}$ and of $\mathcal{F}_n$ in $\mathcal{A}_n$, it follows that

$$\inf(\mathcal{F}_n) \leq \inf(\mathcal{F}) + \delta_n^{\mathcal{S}} \quad \text{and} \quad \inf(\mathcal{F}) \leq \inf(\mathcal{F}_n) + \delta_n^{\mathcal{R}}.$$

Hence, if one of $\inf(\mathcal{F})$ or $\inf(\mathcal{F}_n)$ is finite, the other one is also finite and we have

$$|\inf(\mathcal{F}_n) - \inf(\mathcal{F})| \leq \max\,(\delta_n^{\mathcal{S}}, \delta_n^{\mathcal{R}}).$$

This in turn implies the following inclusions of lower level sets for all $\varrho \geq 0$:

$$\mathcal{S}_n(\mathcal{A} \cap \mathcal{M}^{\varrho}) \subset \mathcal{S}_n(\mathcal{A}) \cap \mathcal{M}_n^{\varrho+2\delta_n} \quad \text{and} \quad \mathcal{R}_n(\mathcal{A}_n \cap \mathcal{M}_n^{\varrho}) \subset \mathcal{R}_n(\mathcal{A}_n) \cap \mathcal{M}^{\varrho+2\delta_n}.$$

Combined with knowledge on the *sampling proximity error* $\varepsilon_n^{\mathcal{S}}$ and the *reconstruction proximity error* $\varepsilon_n^{\mathcal{R}}$

$$\varepsilon_n^{\mathcal{S}} := \sup_{a \in \mathcal{A}} d_{\mathcal{X}}\left(\Psi_n \circ \mathcal{S}_n(a), \Psi(a)\right) \quad \text{and} \quad \varepsilon_n^{\mathcal{R}} := \sup_{a \in \mathcal{A}_n} d_{\mathcal{X}}\left(\Psi \circ \mathcal{R}_n(a), \Psi_n(a)\right),$$

this leads to

$$\Psi(\mathcal{A} \cap \mathcal{M}^{\varrho}) \subset \bar{B}\left(\Psi_n\left(\mathcal{S}_n(\mathcal{A}) \cap \mathcal{M}_n^{\varrho+2\delta_n}\right), \varepsilon_n^{\mathcal{S}}\right) \quad \text{and} \tag{5.2}$$

$$\Psi_n(\mathcal{A}_n \cap \mathcal{M}_n^{\varrho}) \subset \bar{B}\left(\Psi\left(\mathcal{R}_n(\mathcal{A}_n) \cap \mathcal{M}^{\varrho+2\delta_n}\right), \varepsilon_n^{\mathcal{R}}\right) \quad \text{for all } \varrho \geq 0. \tag{5.3}$$

Provided that $\varepsilon_n^{\mathcal{S}} \overset{n\to\infty}{\longrightarrow} 0$ and $\varepsilon_n^{\mathcal{R}} \overset{n\to\infty}{\longrightarrow} 0$ (and under the mild requirements $\mathcal{S}_n(\mathcal{A}) \subset \mathcal{A}_n$ and $\mathcal{R}_n(\mathcal{A}_n) \subset \mathcal{K}$ with some closed set $\mathcal{K} \subset \mathcal{C}$ satisfying $\Psi(\mathcal{M}) \subset \Psi(\mathcal{K})$), this implies

$$\begin{aligned}\Psi(\mathcal{M}) \subset \Psi(\mathcal{A} \cap \mathcal{M}) &\subset \operatorname*{Li}_{n\to\infty} \Psi_n(\mathcal{A}_n \cap \mathcal{M}_n^{2\delta_n}) \\ &\subset \operatorname*{Ls}_{n\to\infty} \Psi_n(\mathcal{A}_n \cap \mathcal{M}_n^{2\delta_n}) \subset \operatorname*{Ls}_{n\to\infty} \Psi(\mathcal{K} \cap \mathcal{M}^{4\delta_n}).\end{aligned}$$

Finally, we require some notion of stability, i.e., a condition on the convergence behavior of those lower level sets whose level is close to $\inf(\mathcal{F})$. For simplicity, we say that $\mathcal{F}$ is *stable along* Ψ *over* $\mathcal{K}$ if $\mathrm{Ls}_{\varrho\searrow 0}\, \Psi(\mathcal{K} \cap \mathcal{M}^{\varrho}) = \Psi(\mathcal{M})$. This leads to the following result; for a more detailed derivation, see [51]:

Theorem 5.3 *Suppose (1)* consistency *(*$\delta_n \overset{n\to\infty}{\longrightarrow} 0$*), (2)* proximity *(*$\varepsilon_n^{\mathcal{R}} \to 0$ *and* $\varepsilon_n^{\mathcal{S}} \to 0$*), and (3)* stability *of* $\mathcal{F}$ *along* Ψ *over* $\mathcal{K}$*. Then*

$$\Psi_n(\mathcal{A}_n \cap \mathcal{M}_n^{2\delta_n}) \overset{n\to\infty}{\longrightarrow} \Psi(\mathcal{M}) \quad \textit{in the sense of Kuratowski.}$$

In particular, this implies $\mathrm{Ls}_{n\to\infty}\, \Psi_n(\mathcal{M}_n) \subset \Psi(\mathcal{M})$ (compare to Eq. (5.1)). The steps for establishing this result (constructing sampling and reconstruction operators, estimating consistency and proximity errors, showing stability) are very similar to the work flow of showing Γ-convergence. So, essentially without investing additional effort and by incorporating a priori information (which makes it *easier* to estimate consistency and proximity errors), we may derive a considerably stronger statement on convergence than Theorem 5.2.

Finally, if $\mathcal{K}$ can be chosen to be *compact* (which tends to require quite detailed a priori knowledge in the *discrete* setting), then Theorem 5.3 can be strengthened as follows:

Theorem 5.4 *In addition to the conditions of Theorem 5.3, suppose that $\mathcal{K}$ is compact. Then*

$$\Psi_n(\mathcal{A}_n \cap \mathcal{M}_n^{2\delta_n}) \overset{n\to\infty}{\longrightarrow} \Psi(\mathcal{M}) \quad \textit{in the sense of Hausdorff.}$$

This is mostly due to the fact that Kuratowski and Hausdorff convergence are equivalent in compact metric spaces (see [3, Proposition 4.4.14]). One can put this result to great use in the convergence analysis for discrete Euler elastica (see Theorem 5.6 below and [50]).

5.5 Convergence of Discrete Minimal Surfaces and Euler Elasticae

The machinery of Kuratowski convergence can in particular be applied to discrete minimal surfaces and discrete Euler elasticae. Here we summarize the respective results. For details, we refer to [50, 51].

For the case of *minimal surfaces*, let (Σ, g) be a compact, k-dimensional smooth Riemannian manifold with boundary, and let $\gamma \in \mathrm{Imm}(\partial\Sigma; \mathbb{R}^m) \cap W^{2,\infty}(\partial\Sigma; \mathbb{R}^m)$ be an embedding and hence a bi-Lipschitz homeomorphism onto its image. In order to fix boundary conditions and using the notation of Sect. 5.2, we restrict to the subset of configuration space

$$\mathrm{Imm}_\gamma(\Sigma; \mathbb{R}^m) := \{f \in \mathrm{Imm}^\times(\Sigma; \mathbb{R}^m) \mid f|_{\partial\Sigma} = \gamma\}$$

that respects the prescribed boundary. Indeed, one can show that the trace mapping $f \mapsto f|_{\partial\Sigma}$ is well defined and Lipschitz continuous on $\mathrm{Imm}^\times(\Sigma; \mathbb{R}^m)$. By slight abuse of notation, we work with the configuration space

$$\mathcal{C} := \mathrm{Imm}_\gamma(\Sigma; \mathbb{R}^m)$$

for the case of minimal surfaces.

Let $(\mathcal{T}_n)_{n\in\mathbb{N}}$ be a uniformly shape regular sequence of smooth triangulations of (Σ, g) with mesh size tending to zero as $n \to \infty$. For every $n \in \mathbb{N}$, let $f_n \in \mathcal{C}_n$ be an embedding of the vertex set of $\mathcal{T}_n$ into $\mathbb{R}^m$, where $\mathcal{C}_n$ denotes the discrete configuration space that respects an appropriate discretization of the boundary curve γ and consists of those mappings that map vertices of simplices to points in general position. The reconstruction operator $\mathcal{R}_n$ is then defined such that $\mathcal{R}_n(f_n)\colon \Sigma \to \mathbb{R}^m$ is a Lipschitz continuous mapping given by barycentric interpolation. Likewise, the sampling operator $\mathcal{S}_n$ is defined by restricting $f \in \mathcal{C}$ to $\mathcal{T}_n$. Let $\mathcal{M}_n \subset \mathcal{C}_n$ be the set of discrete minimizers. Likewise, let $\mathcal{M} \subset \mathcal{C}$ denote the minimizing set of (smooth) minimal surfaces spanning γ. We let $\Psi : \mathcal{C} \to \mathcal{X}$ denote the attendant quotient map into shape space. Likewise, we denote by $\Psi_n := \Psi \circ \mathcal{R}_n : \mathcal{C}_n \to \mathcal{X}$ the mapping from discrete configuration space to shape space.

We additionally require subsets of immersions that have certain regularity properties, both in the smooth and discrete setting, resembling regularity of minimizers. We encode this regularity as *a priori information*. In the smooth setting, we let

$$\mathcal{A}^s := \{f \in C \cap W_g^{2,\infty}(\Sigma; \mathbb{R}^m) \mid d_{\mathcal{P}}(g, f^\# g_0) \le s, \|f\|_{W_g^{2,\infty}} \le s\} \quad \text{for } s \ge 0.$$

In particular, this means that every element of $\mathcal{A}^s$ yields a "nice" parameterization with injective differentials, controlled metric distortion, and controlled $W_g^{2,\infty}$-norm.

Likewise, in the discrete setting we let

$$\mathcal{A}_n^r := \{f_n \in C_n \mid d_{\mathcal{P}}(g, \mathcal{R}_n(f)^\# g_0) \le r\} \quad \text{for } r \ge 0.$$

This implies that all simplices resulting from a mapping in $\mathcal{A}_n^r$ are uniformly non-degenerate in the sense that the aspect ratios of the embedded simplices are uniformly bounded. For the case of discrete minimal surfaces, our main result is:

Theorem 5.5 *Suppose that* $\emptyset \neq \Psi(\mathcal{M}) \subset \Psi(\mathcal{A}^s \cap \mathcal{M})$ *for some* $s \in]0, \infty[$ *and that* $\emptyset \neq \Psi_n(\mathcal{M}_n) \subset \Psi_n(\mathcal{A}_n^r \cap \mathcal{M}_n)$ *for some* $r \in]s, \infty[$ *and all* $n \in \mathbb{N}$. *Then one has Kuratowski convergence of discrete almost minimizers, i.e.,*

$$\Psi_n(\mathcal{A}_n^r \cap \mathcal{M}_n^{2\delta_n}) \overset{n\to\infty}{\longrightarrow} \Psi(\mathcal{M}) \quad \textit{in the sense of Kuratowski,}$$

where δ_n *decreases on the order of decreasing mesh size. The convergence is with respect to the topology generated by the quotient metric on shape space.*

For the case of *Euler elasticae*, let $\Sigma \subset \mathbb{R}$ be a compact interval, and let $\gamma \in W^{2,2}$ be a curve. The *Euler–Bernoulli bending energy* is defined as the integral of squared curvature with respect to the curve's line element ω_γ , i.e.,

$$\mathcal{E}(\gamma) := \frac{1}{2} \int_\Sigma |\kappa_\gamma|^2 \, \omega_\gamma .$$

The Euler–Bernoulli bending energy is frequently used as a model for the bending part of the stored elastic energy of a thin, flexible and inextensible piece of material that has a straight cylindrical rest state.

The classical *Euler elastica problem* is to find minimizers of $\mathcal{E}$ in the feasible set C of all curves of given *fixed* curve length L subject to *fixed* first order boundary conditions that pin down positions and tangent directions at both ends of the curve. Together, these constraints—fixed curve length and fixed boundary conditions—constitute the main difficulty of the problem. Indeed, in dimension two, dropping the positional constraints (while keeping the tangent and length constraints) would yield rather trivial minimizers in the form of circular arcs. Likewise, dropping the length constraint (while keeping endpoint and end tangent constraints) would prevent existence of solutions: In dimension two, consider two straight line segments that respectively meet the two boundary conditions; connect these line segments by a circular arc at their free ends. Then the energy of such a curve is reciprocal to the length of the circular arc and thus arbitrarily small, yielding a minimizing sequence that does not converge.

In the discrete setting, we represent curves as polygonal lines. On a finite partition $\mathcal{T}$ of the interval Σ with vertex set $V(\mathcal{T})$, consider the set of *discrete immersions*. This space consists of all polygons $P\colon V(\mathcal{T}) \to \mathbb{R}^m$ whose successive vertices are mapped to distinct points. On this set, define the *discrete Euler–Bernoulli energy* by

$$\mathcal{E}_n(P) := \frac{1}{2} \sum_{v \in V(\mathcal{T})} \left(\frac{\alpha_P(v)}{\bar{l}_P(v))} \right)^2 \bar{l}_P(v) = \frac{1}{2} \sum_{v \in V(\mathcal{T})} \frac{\alpha_P^2(v)}{\bar{l}_P(v)}, \tag{5.4}$$

where $\alpha_P(v)$ is the *turning angle* at an interior vertex v and $\bar{l}_P(v)$ is the *dual edge length*, i.e., the arithmetic mean of the lengths of the two adjacent (embedded) edges. This energy is motivated by the observation that turning angles are in many ways a reasonable surrogate for integrated absolute curvature on dual edges (see, e.g., [23, 52]).

We define smooth and discrete *configuration spaces*, respectively, as

$$C := \{\gamma \in W^{2,p} \mid \gamma \text{ satisfies the boundary conditions}\},$$
$$C_n := \{P\colon V(\mathcal{T}) \to \mathbb{R}^m \mid P \text{ satisfies the boundary conditions}\},$$

where for C_n we restrict to those polygons whose successive vertices are mapped to distinct points. The *boundary conditions* consist of prescribing positions and tangent directions at both endpoints and fixing total curve length L. We call boundary conditions *commensurable* if C and C_n are not empty.

As in the case of minimal surfaces, we consider certain *a priori assumptions* that encode regularity properties of minimizers. Accordingly, we define the two sets

$$\mathcal{A} := \{\gamma \in C \mid [\gamma]_{W^{2,\infty}}\,,\ [\gamma]_{W^{3,\infty}} \leq K_1\},$$
$$\mathcal{A}_n := \{P \in C_n \mid [P]_{w^{2,\infty}},\ [P]_{tv^3} \leq K_2\},$$

where K_1 and $K_2 \geq 0$ are suitable constants. The norms $[P]_{w^{2,\infty}}$ and $[P]_{tv^3}$ are certain discrete versions of Sobolev and total variation norms, see [50]. These a priori assumptions are justified by regularity properties of smooth and discrete minimizers. In the smooth setting, regularity of minimizers can be verified in various ways, e.g., by invoking elliptic integrals. We use a functional analytic approach to prove these regularity properties, since this approach can be closely mimicked in the discrete case. The above sets satisfy the following inclusions:

$$\mathcal{M} \subset \mathcal{A} \subset C \quad \text{and} \quad \mathcal{M}_n \subset \mathcal{A}_n \subset C_n.$$

As in the case of minimal surfaces, we rely on a *reconstruction operator* $\mathcal{R}_n \colon \mathcal{A}_n \to C$ and a *sampling operator* $\mathcal{S}_n \colon \mathcal{A} \to C_n$, taking polygons to smooth curves and vice versa. In order to construct these operator, we first construct *approximate* reconstruction and sampling operators $\widetilde{\mathcal{R}}_n$ and $\widetilde{\mathcal{S}}_n$ that map $\mathcal{A}_n$ and $\mathcal{A}$ into sufficiently small vicinities of C and C_n, respectively. The main idea for these *approximate* operators is that they only satisfy the boundary conditions and the length constraint *approximately* but not necessarily exactly. We then apply a Newton–Kantorovich-type theorem in order to show that *exact* reconstruction and sampling operators $\mathcal{R}_n$ and $\mathcal{S}_n$ (i.e., those that satisfy the requisite constraints exactly) can be obtained from $\widetilde{\mathcal{R}}_n$ and $\widetilde{\mathcal{S}}_n$ by sufficiently small perturbations. For the case of discrete Euler elasticae, our main result is:

Theorem 5.6 *Fix a prescribed curve length L and commensurable boundary conditions. Denote by $\rho(\mathcal{T}_n)$ the maximum edge length of a partition $\mathcal{T}_n$ of the domain Σ, and let $\Psi_n \colon C_n \to W^{1,\infty}$ denote the interpolation of vertices of discrete curves by continuous, piecewise affine curves. Then for each $p \in [2, \infty[$ there is a constant $C \geq 0$ such that one has the following convergence in Hausdorff distances:*

$$\lim_{\rho(\mathcal{T}_n)\to 0} \operatorname{dist}_{W^{2,p}} \left(\mathcal{M}, \mathcal{R}_n\left(\mathcal{A}_n \cap \mathcal{M}_n^{C\rho(\mathcal{T}_n)}\right)\right) = 0 \quad \textit{and}$$

$$\lim_{\rho(\mathcal{T}_n)\to 0} \operatorname{dist}_{W^{1,\infty}} \left(\mathcal{M}, \Psi_n\left(\mathcal{A}_n \cap \mathcal{M}_n^{C\rho(\mathcal{T}_n)}\right)\right) = 0.$$

Notice that although sampling operators do not appear explicitly in this result, they play a prominent role in the proof since they guarantee existence of discrete almost minimizers in the vicinity of every smooth minimizer.

By relying on a priori assumptions, our result is different from the Γ-convergence results from [1, 2, 16, 30, 36]. Indeed, by restricting to the sets $\mathcal{A}$ and $\mathcal{A}_n$, we avoid the need for recovery sequences for every element in configuration space. We thus obtain a stronger convergence result in the sense that *all* discrete minimizers are uniformly close to the set of smooth minimizers $\mathcal{M}$ with respect to $W^{2,p}$-norm, i.e., there exists a function $f\colon [0,\infty] \to [0,\infty]$, continuous at 0 and with $f(0) = 0$, such that

$$\sup_{P\in\mathcal{M}_n} \inf_{\gamma\in\mathcal{M}} \|\gamma - \mathcal{R}_n(P)\|_{W^{2,p}} \leq f(\rho(\mathcal{T}_n)).$$

Since $p > 2$ is allowed, *we obtain convergence in a topology that is finer than the one of the energy space.*

Acknowledgements The authors gratefully acknowledge support from DFG-project 282535003: *Geometric curvature functionals: energy landscape and discrete methods.*

References

1. Alibert, J.J., Della Corte, A., Giorgio, I., Battista, A.: Extensional Elastica in large deformation as Γ-limit of a discrete 1D mechanical system. Z. Angew. Math. Phys. **68**(2), Art. 42, 19 (2017)
2. Alibert, J.J., Della Corte, A., Seppecher, P.: Convergence of Hencky-type discrete beam model to Euler inextensible elastica in large deformation: rigorous proof. In: Mathematical Modelling in Solid Mechanics. Advanced Structured Materials, vol. 69, pp. 1–12. Springer, Singapore (2017)
3. Ambrosio, L., Tilli, P.: Topics on Analysis in Metric Spaces. Oxford Lecture Series in Mathematics and its Applications, vol. 25. Oxford University Press, Oxford (2004)
4. Banchoff, T.: Critical points and curvature for embedded polyhedra. J. Differ. Geom. **1**, 257–268 (1967)
5. Bauer, U., Polthier, K., Wardetzky, M.: Uniform convergence of discrete curvatures from nets of curvature lines. Discrete Comput. Geom. **43**(4), 798–823 (2010)
6. Bauer, M., Harms, P., Michor, P.W.: Sobolev metrics on shape space of surfaces. J. Geom. Mech. **3**(4), 389–438 (2011)
7. Bauer, M., Harms, P., Michor, P.W.: Almost local metrics on shape space of hypersurfaces in n-space. SIAM J. Imaging Sci. **5**, 244–310 (2012)
8. Bauer, M., Harms, P., Michor, P.W.: Curvature weighted metrics on shape space of hypersurfaces in n-space. Differ. Geom. Appl. **30**(1), 33–41 (2012)
9. Bauer, M., Harms, P., Michor, P.W.: Sobolev metrics on shape space, II: weighted Sobolev metrics and almost local metrics. J. Geom. Mech. **4**(4), 365–383 (2012)
10. Bauer, M., Bruveris, M., Marsland, S., Michor, P.W.: Constructing reparametrization invariant metrics on spaces of plane curves. Differ. Geom. Appl. **34**, 139–165 (2014)
11. Bauer, M., Bruveris, M., Michor, P.W.: Why use Sobolev metrics on the space of curves. In: Turaga, P., Srivastava, A. (eds.) Riemannian Computing in Computer Vision, pp. 223–255. Springer, Cham (2016)
12. Bobenko, A.I., Suris, Y.B.: Discrete Differential Geometry: Integrable Structure. Graduate Studies in Mathematics, vol. 98. American Mathematical Society, Providence (2008)
13. Bobenko, A.I., Hoffmann, T., Springborn, B.A.: Minimal surfaces from circle patterns: geometry from combinatorics. Ann. Math. **164**(1), 231–264 (2006)
14. Bobenko, A.I., Sullivan, J.M., Schröder, P., Ziegler, G.: Discrete Differential Geometry. Oberwolfach Seminars. Birkhäuser, Basel (2008)
15. Borrelli, V., Cazals, F., Morvan, J.M.: On the angular defect of triangulations and the pointwise approximation of curvatures. Comput. Aided Geom. Des. **20**(6), 319–341 (2003)
16. Bruckstein, A.M., Netravali, A.N., Richardson, T.J.: Epi-convergence of discrete elastica. Appl. Anal. **79**(1–2), 137–171 (2001)
17. Bruveris, M.: Completeness properties of Sobolev metrics on the space of curves. J. Geom. Mech. **7**(2), 125–150 (2015)
18. Bruveris, M., Michor, P.W., Mumford, D.: Geodesic completeness for Sobolev metrics on the space of immersed plane curves. Forum Math. Sigma **2**, e19 (2014)
19. Bücking, U.: Minimal surfaces from circle patterns: boundary value problems, examples. In: Bobenko, A.I., Sullivan, J.M., Schröder, P., Ziegler, G. (eds.) Discrete Differential Geometry, pp. 37–56. Birkhäuser, Basel (2008)
20. Cheeger, J., Müller, W., Schrader, R.: Curvature of piecewise flat metrics. Commun. Math. Phys. **92**, 405–454 (1984)
21. Cohen-Steiner, D., Morvan, J.M.: Restricted Delaunay triangulations and normal cycle. In: Symposium on Computational Geometry, pp. 312–321 (2003)
22. Cohen-Steiner, D., Morvan, J.M.: Second fundamental measure of geometric sets and local approximation of curvatures. J. Differ. Geom. **73**(3), 363–394 (2006)
23. Crane, K., Wardetzky, M.: A glimpse into discrete differential geometry. Not. Am. Math. Soc. **64**(10), 1153–1159 (2017)

24. Dal Maso, G.: An introduction to Γ-convergence. In: Progress in Nonlinear Differential Equations and their Applications, vol. 8. Birkhäuser, Boston (1993)
25. Dierkes, U., Hildebrandt, S., Tromba, A.J.: Regularity of minimal surfaces. In: Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences], vol. 340, 2nd edn. Springer, Heidelberg (2010)
26. Douglas, J.: Solution of the problem of plateau. Trans. Am. Math. Soc. **33**(1), 263–321 (1931)
27. Dziuk, G.: Finite elements for the Beltrami operator on arbitrary surfaces. In: Hildebrandt, S., Leis, R. (eds.) Partial Differential Equations and Calculus of Variations. Lecture Notes in Mathematics, vol. 1357, pp. 142–155. Springer, Berlin (1988)
28. Dziuk, G., Hutchinson, J.E.: The discrete plateau problem: algorithm and numerics. Math. Comput. **68**, 1–23 (1999)
29. Dziuk, G., Hutchinson, J.E.: The discrete plateau problem: convergence results. Math. Comput. **68**, 519–546 (1999)
30. Español, M.I., Golovaty, D., Wilber, J.P.: Euler elastica as a γ-limit of discrete bending energies of one-dimensional chains of atoms. Math. Mech. Solids (2017)
31. Fu, J.H.: Convergence of curvatures in secant approximations. J. Differ. Geom. **37**, 177–190 (1993)
32. Hardt, R., Simon, L.: Boundary regularity and embedded solutions for the oriented plateau problem. Ann. Math. (2) **110**(3), 439–486 (1979)
33. Hildebrandt, K., Polthier, K.: On approximation of the Laplace–Beltrami operator and the Willmore energy of surfaces. Comput. Graph. Forum **30**(5), 1513–1520 (2011)
34. Hildebrandt, K., Polthier, K., Wardetzky, M.: On the convergence of metric and geometric properties of polyhedral surfaces. Geom. Dedicata **123**, 89–112 (2006)
35. Hinze, M.: On the numerical approximation of unstable minimal surfaces with polygonal boundaries. Numer. Math. **73**(1), 95–118 (1996)
36. Iglesias, J.A., Bruckstein, A.M.: On the Gamma-convergence of some polygonal curvature functionals. Appl. Anal. **94**(5), 957–979 (2015)
37. Meek, D.S., Walton, D.J.: On surface normal and Gaussian curvature approximations given data sampled from a smooth surface. Comput. Aided Geom. Des. **17**(6), 521–543 (2000)
38. Mennucci, A., Yezzi, A., Sundaramoorthi, G.: Properties of Sobolev-type metrics in the space of curves. Interfaces Free Bound. **10**(4), 423–445 (2008)
39. Michor, P.W., Mumford, D.: Vanishing geodesic distance on spaces of submanifolds and diffeomorphisms. Doc. Math. **10**, 217–245 (2005)
40. Michor, P.W., Mumford, D.: Riemannian geometries on spaces of plane curves. J. Eur. Math. Soc. **8**(1), 1–48 (2006)
41. Michor, P.W., Mumford, D.: An overview of the Riemannian metrics on spaces of curves using the Hamiltonian approach. Appl. Comput. Harmon. Anal. **23**(1), 74–113 (2007)
42. Moakher, M., Zéraï, M.: The Riemannian geometry of the space of positive-definite matrices and its application to the regularization of positive-definite matrix-valued data. J. Math. Imaging Vis. **40**(2), 171–187 (2011)
43. Morvan, J.M.: Generalized Curvatures. Springer Publishing Company, Berlin (2008)
44. Pinkall, U., Polthier, K.: Computing discrete minimal surfaces and their conjugates. Exp. Math. **2**, 15–36 (1993)
45. Pozzi, P.: The discrete Douglas problem: convergence results. IMA J. Numer. Anal. **25**(2), 337–378 (2005)
46. Radó, T.: On the Problem of Plateau. Ergebn. d. Math. u. ihrer Grenzgebiete, vol. 2. Springer, Berlin (1933)
47. Regge, T.: General relativity without coordinates. Nuovo Cimento **19**(10), 558–571 (1961)
48. Rivière, T.: Lipschitz conformal immersions from degenerating Riemann surfaces with L^2-bounded second fundamental forms. Adv. Calc. Var. **6**(1), 1–31 (2013)
49. Rockafellar, R.T., Wets, R.J.B.: Variational Analysis. Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences], vol. 317. Springer, Berlin (1998)

50. Scholtes, S., Schumacher, H., Wardetzky, M.: Variational convergence of discrete elasticae (2019). https://arxiv.org/abs/1901.02228
51. Schumacher, H., Wardetzky, M.: Variational convergence of discrete minimal surfaces. Numer. Math. **141**(1), 173–213 (2019)
52. Sullivan, J.M.: Curves of finite total curvature. In: Discrete Differential Geometry. Oberwolfach Seminar, vol. 38, pp. 137–161. Birkhäuser, Basel (2008)
53. Tsuchiya, T.: Discrete solution of the plateau problem and its convergence. Math. Comput. **49**(179), 157–165 (1987)
54. Wardetzky, M.: Discrete differential operators on polyhedral surfaces–convergence and approximation. Ph.D. thesis, Freie Universität Berlin (2006)
55. Wilson Jr., W.L.: On discrete Dirichlet and plateau problems. Numer. Math. **3**, 359–373 (1961)
56. Xu, Z., Xu, G., Sun, J.G.: Convergence analysis of discrete differential geometry operators over surfaces. In: IMA Conference on the Mathematics of Surfaces, pp. 448–457 (2005)

Part II
Geometry as a Tool

Chapter 6
Variational Methods for Fluid-Structure Interactions

François Gay-Balmaz and Vakhtang Putkaradze

Contents

Abstract This chapter gives an introduction to the variational methods recently developed in fluid-structure interaction, by focusing on the dynamics of flexible tubes conveying fluid. This is a topic of high importance for biomedical and industrial applications, such as arterial or lung flows, and problems involving high-speed motion of gas in flexible pipes. Our goal is to derive a variational approach to fluid-structure interaction with the aim of developing the corresponding variational numerical methods. Variational approaches and corresponding discretizations are advantageous for fluid-structure interactions, since they possess excellent long term

F. Gay-Balmaz (✉)
CNRS & Ecole Normale Supérieure, Paris, France
e-mail: francois.gay-balmaz@lmd.ens.fr

V. Putkaradze
University of Alberta, Edmonton, AB, Canada

ATCO SpaceLab, SW Calgary, AB, Canada
e-mail: putkarad@ualberta.ca

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_6

energy behavior, exact conservation of momenta, and yield consistent models for complex mechanical systems. We present a model for the three-dimensional evolution of tubes with expandable walls conveying fluid, that can accommodate arbitrary deformations of the tube, arbitrary elasticity of the walls, and both compressible and incompressible flows inside the tube. We show how this model reduces to previously derived models under specific assumptions. We present particular solutions of the model, such as propagation of a shock wave in an elastic tubes via the Rankine–Hugoniot conditions. Finally, we develop the variational discretization of the model, based on the discretization of the back-to-labels map, for the cases of spatial and spatio-temporal discretizations.

6.1 Introduction

Tubes with flexible walls conveying fluid are frequently encountered in nature and exhibit complex behavior due to the interaction between the fluid and elastic walls. Important examples naturally appear in physiological applications, such as flow of blood in arteries. There has been a great number of studies in this area, especially with the focus on arterial flows, e.g. [20, 35, 54–56, 59, 66, 67] and lung flows, e.g. [16, 17, 40]. For more informations about the application of collapsible tubes to biological flows and a summary of the literature in the field, we refer the reader to the reviews [31, 32, 61]. Analytical studies for such flows are usually limited to cases when the centerline of the tube is straight, which restricts the utility of the models for practical applications, but is important for theoretical understanding. While substantial progress in the analysis of the flow has been achieved so far, it was difficult to describe analytically the general dynamics of 3D deformations of the tube.

On the other hand, studies involving non-trivial dynamics of the centerline have a long history in the context of engineering applications also loosely called the "garden hose instabilities". While one of the earliest works on the subject was [4], Benjamin [6, 7] was perhaps the first to formulate a quantitative theory for the 2D dynamics of *initially straight tubes* by considering a linked chain of tubes conveying fluids and using an augmented Hamilton principle of critical action that takes into account the momentum of the jet leaving the tube. A continuum equation for the linear disturbances was then derived as the limit of the discrete system. It is interesting to note that Benjamin's approach was, in essence, the first discrete variational method applied to this problem. This linearized equation for the initially straight tubes was further studied by Gregory and Païdoussis [29]. These initial developments formed the basis for further stability analysis for finite, initially straight tubes [1–3, 15, 49–53, 63], which showed a reasonable agreement with experimentally observed onset of the instability [11, 19, 30, 37, 50]. Models treating nonlinear deflection of the centerline were also considered in [28, 47, 51, 62], and the compressible (acoustic) effects in the flowing fluid in [72]. Alternatively, a more detailed 3D theory of motion for the flow was developed in [5] and extended in [60]. That theory was based on a modification of the Cosserat rod treatment for the description of elastic dynamics of the tube, while keeping the cross-section of the tube constant and orthogonal to the centerline. In particular [60], analyzes several non-straight configurations, such as a tube deformed from its straight state by gravity, both from the point of view of linear stability and nonlinear behavior. Unfortunately, this approach cannot easily incorporate the effects of the cross-sectional changes in the dynamics, either prescribed (e.g. a tube with variable cross-section in the initial state) or dynamically occurring. The history of the parallel development of this problem in the Soviet/Russian literature is summarized in [26].

The goal of this chapter is to describe the variational approach to the problem developed by the authors in [21, 22, 24]. Based on this approach, the linear stability of helical tubes was studied in [26]. The theory derived in [21, 22, 24], being a truly variational theory of the 3D motion of the tube, opened the opportunity for developing variational approximation for the equations, both from the point of view of deriving simplified reduced models and developing structure preserving numerical schemes, as initiated in [23]. The description in this paper summarizes and extends this approach. We base our method on the recent works [13, 14] on multisymplectic discretization of an elastic beam in $\mathbb{R}^3$, which is in turn based on the geometric variational spacetime discretization of Lagrangian field theories developed in [45].

6.2 Preliminaries on Variational Methods

Variational numerical schemes, also known as variational integrators, are powerful discretization methods as they allow exact conservation of appropriately defined momenta and excellent long-term energy behavior and are thus very useful for constructing efficient description of long-term dynamics, especially in the case when friction effects are small. We refer the reader to the extensive review [43] of variational integrators in Lagrangian mechanics, and [39] for the application of variational integrators to constrained systems, important for this work. There are two major difficulties which we will need to address here, namely, the appropriate coupling of the fluid to the elastic tube, and the treatment of the constraint of fluid volume conservation. Before we describe these concepts in more details, we give a review of variational methods in mechanics in order to keep the exposition self-contained.

Hamilton's Principle

One of the most fundamental statement in classical mechanics is the principle of critical action or Hamilton's principle, according to which the motion of a mechanical system between two given positions follows a curve that makes the integral of the Lagrangian of the system critical (see, for instance [38]).

Consider a mechanical system with configuration manifold Q and Lagrangian $L : TQ \to \mathbb{R}$ defined on the tangent bundle of Q. The Lagrangian L is usually given by the kinetic minus the potential energy of the system as $L(q, v) = K(q, v) - U(q)$. The Hamilton principle reads

$$\delta \int_0^T L(q, \dot{q}) \mathrm{d}t = 0\,, \tag{6.1}$$

for arbitrary variations δq with $\delta q(0) = \delta q(T) = 0$, and yields the Euler–Lagrange equations, given in coordinates as

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}^i} - \frac{\partial L}{\partial q^i} = 0, \quad i = 1, \dots, n.$$

External forces, given by a fiber preserving maps $F^{\mathrm{ext}} : TQ \to T^*Q$, can be included in the Hamilton principle, by considering the principle

$$\delta \int_0^T L(q, \dot{q}) \mathrm{d}t + \int_0^T \langle F^{\mathrm{ext}}, \delta q \rangle \mathrm{d}t = 0\,. \tag{6.2}$$

Requiring that $F^{\mathrm{ext}} : TQ \to T^*Q$ is a fiber preserving map means that for each fixed $q \in Q$, it restricts to a map from the fiber T_qQ to the fiber T_q^*Q, at the same point q.

The Hamilton principle has a natural extension to continuum systems, for which the configuration manifold becomes infinite dimensional, typically a manifold of maps, and which will be of crucial use in this chapter. For instance, let us assume that the motion of the continuum system is described by a curve of embeddings $\varphi_t : \mathcal{B} \to \mathbb{R}^3$, where $\mathcal{B}$ is the reference configuration of the continuum. The current position at time t of the particle with label $X \in \mathcal{B}$ is $x = \varphi_t(X) \in \mathbb{R}^3$. In this case, the configuration manifold Q of the system is infinite dimensional and given by all smooth embeddings of $\mathcal{B}$ into $\mathbb{R}^3$, i.e., $Q = \mathrm{Emb}(\mathcal{B}, \mathbb{R}^3)$. Given a Lagrangian $L : TQ \to \mathbb{R}$, Hamilton's principle formally takes the same form as Eq. (6.1), namely

$$\delta \int_0^T L(\varphi, \dot{\varphi}) \mathrm{d}t = 0\,, \tag{6.3}$$

for variations $\delta\varphi$ such that $\delta\varphi(0) = \delta\varphi(T) = 0$. We refer to [27] for a detailed account on Hamilton's principle and its symmetry reduced versions in continuum mechanics. An extension of Hamilton's principle to include irreversible processes in continuum systems was presented in [25]. Note that since L is defined on the tangent bundle of $Q = \mathrm{Emb}(\mathcal{B}, \mathbb{R}^3)$, it can a priory depend in an arbitrary way on the spatial derivatives of φ and $\dot{\varphi}$. Typically L only depends on the first spatial derivative of φ and is thus expressed with the help of a Lagrangian density as

$$L(\varphi, \dot{\varphi}) = \int_{\mathcal{B}} \mathfrak{L}\big(\varphi(X), \dot{\varphi}(X), \nabla\varphi(X)\big) \mathrm{d}^3 X.$$

Holonomic Constraints

Hamilton's principle can be naturally extended to include constraints, should they be holonomic or not. In the holonomic case, which is the case that we will need, the constraint defines a submanifold $N \subset Q$ of the configuration manifold. Assuming that $N = \Phi^{-1}(0)$, for a submersion $\Phi : Q \to \mathbb{R}^r$, the equations of motion follow from the Hamilton principle with Lagrange multipliers

$$\delta \int_0^T \big[L(q, \dot{q}) + \lambda_\alpha \Phi^\alpha(q)\big] \mathrm{d}t = 0\,, \tag{6.4}$$

in which one considers also arbitrary variations $\delta\lambda_\alpha$. In this holonomic case the equations of motion can be also directly obtained by applying the Hamilton principle to the Lagrangian L restricted to TN, but in most examples in practice, as it will be the case for us, the constraint submanifold N takes such a complicated expression that it is impossible to avoid the use of (6.4).

Lagrangian Reduction by Symmetry

When a symmetry is available in a mechanical system, it is often possible to exploit it in order to reduce the dimension of the system and thereby facilitating its study. This process, called *reduction by symmetry*, is well developed both on the Lagrangian and Hamiltonian sides, see [44] for an introduction and references.

While on the Hamiltonian side, this process is based on the reduction of symplectic or Poisson structures, on the Lagrangian side it is usually based on the reduction of variational principles, see [9, 41, 42]. Consider a mechanical system with configuration manifold Q and Lagrangian $L : TQ \to \mathbb{R}$ and consider also the action of a Lie group G on Q, denoted here simply as $q \mapsto g \cdot q$, for $g \in G$, $q \in Q$. This action naturally induces an action on the tangent bundle TQ, denoted here simply as $(q, v) \mapsto (g \cdot q, g \cdot v)$, called the *tangent lifted action*. We say that the action is a symmetry for the mechanical system if the Lagrangian L is invariant under this tangent lifted action. In this case, L induces a *symmetry reduced Lagrangian* $\ell : (TQ)/G \to \mathbb{R}$ defined on the quotient space $(TQ)/G$ of the tangent bundle with respect to the action. The goal of the Lagrangian reduction process is to derive the equations of motion directly on the reduced space $(TQ)/G$. Under standard hypotheses on the action, this quotient space is a manifold and one obtains the *reduced Euler–Lagrange equations* by computing the *reduced variational principle* for the action integral $\int_{t_1}^{t_2} \ell \, \mathrm{d}t$ induced by Hamilton's principle (6.1) for the action integral $\int_{t_1}^{t_2} L \, \mathrm{dt}$. The main difference between the reduced variational principle and Hamilton's principle is the occurrence of constraints on the variations to be considered when computing the critical curves for $\int_{t_1}^{t_2} \ell \, \mathrm{d}t$. These constraints are uniquely associated to the reduced character of the variational principle and not to physical constraints.

Passing from the Hamilton principle and Euler–Lagrange equations to their symmetry reduced versions corresponds in practical examples to pass from the *material* (or *Lagrangian*) description to either the *spatial* (or *Eulerian*) description (in case of symmetries associated to actions on the right), or to the *convective* (or *body*) description (in case of symmetries associated to actions on the left), see [27]. A mixing of the two descriptions also arises, as it will be the case for the flexible tube conveying fluid.

Variational Discretization

Another useful side of the Hamilton principle is that it admits discrete versions that are useful to derive structure preserving numerical schemes. Such schemes, called *variational integrators*, see [43, 70], are originally based on Moser–Veselov discretizations [48, 68, 69]. Very briefly and in the simplest case, one fixes a time step h, and considers a discrete Lagrangian $L_d : Q \times Q \to \mathbb{R}$ that approximates the time integral of the continuous Lagrangian between two consecutive configurations q_k and q_{k+1}

$$L_d(q_k, q_{k+1}) \approx \int_{t_k}^{t_{k+1}} L(q(t), \dot{q}(t)) \mathrm{d}t \,, \tag{6.5}$$

where $q_k = q(t_k)$ and $q_{k+1} = q(t_{k+1})$, with $t_{k+1} = t_k + h$. Equipped with such a discrete Lagrangian, one can formulate a discrete version of Hamilton's principle (6.1) according to

$$\delta \sum_{k=0}^{N-1} L_d(q_k, q_{k+1}) = 0 \,,$$

for variations δq_k vanishing at the endpoints. Thus, if we denote D_i the partial derivative with respect to the ith variable, three consecutive configuration variables q_{k-1}, q_k, q_{k+1} must verify the discrete analogue of the Euler–Lagrange equations:

$$D_2 L_d(q_{k-1}, q_k) + D_1 L_d(q_k, q_{k+1}) = 0 \,. \tag{6.6}$$

These *discrete Euler–Lagrange equations* define, under appropriate conditions, a symplectic integration scheme which solves for q_{k+1}, knowing the two previous configuration variables q_{k-1} and q_k. Discrete versions of the Hamilton principle with holonomic constraints (6.4) can be derived in a similar way, see [43]. Variational integrators have been extended to spacetime discretization in [45], leading to multisymplectic integrators, which will be greatly exploited in our development.

6.2.1 Exact Geometric Rod Theory via Variational Principles

The theory of geometrically exact rod that we recall here has been developed in [64] and [65] and is based on the original work of [10]. We shall derive below the equations of geometrically exact rods by using a symmetry reduced version of the classical Hamilton principle (6.1) written on the infinite dimensional configuration manifold of the rod. We follow the variational formulation given in [33] and [18].

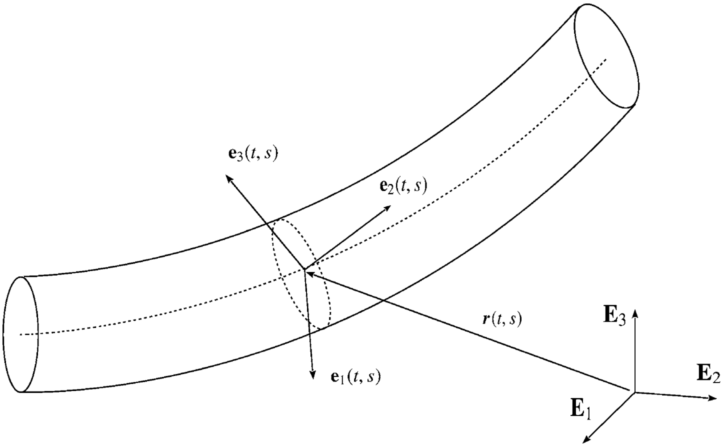


Fig. 6.1 Illustration of the geometrically exact rod. Are represented the fixed frame $\{\mathbf{E}_i \mid i = 1, 2, 3\}$, the moving frame $\{\mathbf{e}_i(t, s) = \Lambda(t, s)\mathbf{E}_i \mid i = 1, 2, 3\}$, and the position of the line of centroids $\boldsymbol{r}(t, s)$

Configuration Manifold and Hamilton's Principle The configuration of the rod deforming in the ambient space $\mathbb{R}^3$ is defined by specifying the position of its line of centroids $\boldsymbol{r}(t, s) \in \mathbb{R}^3$, and by giving the orientation of the cross-section at each point $\boldsymbol{r}(t, s)$. This orientation can be defined by using a moving basis $\{\mathbf{e}_i(t, s) \mid i = 1, 2, 3\}$ attached to the cross section relative to a fixed frame $\{\mathbf{E}_i \mid i = 1, 2, 3\}$. The moving basis is described by means of an orthogonal transformation $\Lambda(t, s) \in SO(3)$ such that $\mathbf{e}_i(t, s) = \Lambda(t, s)\mathbf{E}_i$. Here t is the time and $s \in [0, L]$ is a parameter along the rod that does not need to be arclength. The cross-section is not required to be orthogonal to line of centroids. More generally, there is no constraint relating the frame $\{\mathbf{e}_i(t, s) \mid i = 1, 2, 3\}$ and the vector $\partial_s \boldsymbol{r}(t, s)$. See Fig. 6.1 for an illustration of the geometrically exact rod.

The configuration manifold of a geometrically exact rod is thus the infinite dimensional manifold $Q_{\rm rod} = \mathcal{F}\big([0, L], SO(3) \times \mathbb{R}^3\big)$ of $SO(3) \times \mathbb{R}^3$-valued smooth maps defined on the interval $[0, L]$. Given the Lagrangian function of the rod

$$L : TQ_{\rm rod} \to \mathbb{R}, \quad (\Lambda, \dot{\Lambda}, \boldsymbol{r}, \dot{\boldsymbol{r}}) \mapsto L\big(\Lambda, \dot{\Lambda}, \boldsymbol{r}, \dot{\boldsymbol{r}}\big) \tag{6.7}$$

defined on the tangent bundle $TQ_{\rm rod}$ of the configuration manifold, one obtains the equations of motion by using the Hamilton principle (6.1), more precisely its continuum extension (6.3), which here reads

$$\delta \int_0^T L\big(\Lambda, \dot{\Lambda}, \boldsymbol{r}, \dot{\boldsymbol{r}}\big)\, {\rm d}t = 0\,, \tag{6.8}$$

for arbitrary variations $\delta\Lambda$, $\delta\boldsymbol{r}$ vanishing at $t = 0, T$. In general, a Lagrangian L defined on $TQ_{\rm rod}$ can depend in an arbitrary way on the spatial derivatives of Λ, $\dot{\Lambda}$,

$\boldsymbol{r}$, and $\dot{\boldsymbol{r}}$. For geometrically exact rods, L only depends on the first spatial derivatives of Λ and $\boldsymbol{r}$, i.e., it is expressed with the help of a Lagrangian density as

$$L\big(\Lambda, \dot{\Lambda}, \boldsymbol{r}, \dot{\boldsymbol{r}}\big) = \int_0^L \mathfrak{L}\left(\Lambda(s), \Lambda'(s), \dot{\Lambda}(s), \boldsymbol{r}(s), \boldsymbol{r}'(s), \dot{\boldsymbol{r}}(s)\right) \mathrm{d}s.$$

Moreover, it turns out that the Lagrangian of geometrically exact rods can be exclusively expressed in terms of the convective variables

$$\boldsymbol{\gamma} = \Lambda^{-1}\dot{\boldsymbol{r}}\,, \qquad \omega = \Lambda^{-1}\dot{\Lambda}\,, \qquad \boldsymbol{\Gamma} = \Lambda^{-1}\boldsymbol{r}'\,, \qquad \Omega = \Lambda^{-1}\Lambda'\,, \tag{6.9}$$

see [64, 65]. Here $\boldsymbol{\gamma}, \boldsymbol{\omega} \in \mathcal{F}\big([0, L], \mathbb{R}^3\big)$ are the *linear and angular convective velocities* and $\boldsymbol{\Gamma}, \boldsymbol{\Omega} \in \mathcal{F}\big([0, L], \mathbb{R}^3\big)$ are the *linear and angular convective strains*, with $(\ )' = \partial_s$ and $\dot{(\ )} = \partial_t$. We use the hat map isomorphism to obtain $\boldsymbol{\omega}$ and $\boldsymbol{\Omega}$ from ω and Ω in (6.9):

$$\boldsymbol{\omega} \in \mathbb{R}^3 \mapsto \omega = \widehat{\boldsymbol{\omega}} \in \mathfrak{so}(3)\,, \qquad \omega_{ij} = -\epsilon_{ijk}\boldsymbol{\omega}_k\,, \tag{6.10}$$

where ϵ_{ijk} is the completely antisymmetric tensor with $\epsilon_{123} = 1$. Note that we have $\widehat{\boldsymbol{\omega}}\mathbf{a} = \boldsymbol{\omega}\times\mathbf{a}$, for all $\mathbf{a} \in \mathbb{R}^3$. Since $[\,\widehat{\mathbf{a}}\,, \widehat{\mathbf{b}}\,] = \widehat{\mathbf{a}\times\mathbf{b}}$, the correspondence (6.10) is a Lie algebra isomorphism. Writing L in terms of those variables, we get the *convective Lagrangian* that we denote

$$\ell : \mathcal{F}\big([0, L], \mathbb{R}^3\big)^4 \to \mathbb{R}\,, \qquad (\boldsymbol{\omega}, \boldsymbol{\gamma}, \boldsymbol{\Omega}, \boldsymbol{\Gamma}) \mapsto \ell(\boldsymbol{\omega}, \boldsymbol{\gamma}, \boldsymbol{\Omega}, \boldsymbol{\Gamma})\,. \tag{6.11}$$

For the moment, we leave the Lagrangian function unspecified, we will give its explicit expression later in the case of fluid-conveying tubes.

The equations of motion in convective description are obtained by writing the critical action principle (6.8) in terms of the Lagrangian ℓ. This is accomplished by computing the constrained variations of $\boldsymbol{\omega}$, $\boldsymbol{\gamma}$, $\boldsymbol{\Omega}$, $\boldsymbol{\Gamma}$ induced by the free variations $\delta\Lambda$, $\delta\boldsymbol{r}$ via the definition (6.9). We find the constrained variations

$$\delta\boldsymbol{\omega} = \partial_t\boldsymbol{\Sigma} + \boldsymbol{\omega}\times\boldsymbol{\Sigma}\,, \qquad \delta\boldsymbol{\gamma} = \partial_t\boldsymbol{\eta} + \boldsymbol{\gamma}\times\boldsymbol{\Sigma} + \boldsymbol{\omega}\times\boldsymbol{\eta}\,, \tag{6.12}$$

$$\delta\boldsymbol{\Omega} = \partial_s\boldsymbol{\Sigma} + \boldsymbol{\Omega}\times\boldsymbol{\Sigma}\,, \qquad \delta\boldsymbol{\Gamma} = \partial_s\boldsymbol{\eta} + \boldsymbol{\Gamma}\times\boldsymbol{\Sigma} + \boldsymbol{\Omega}\times\boldsymbol{\eta}\,, \tag{6.13}$$

where $\widehat{\boldsymbol{\Sigma}}(t,s) = \Sigma(t,s) = \Lambda(t,s)^{-1}\delta\Lambda(t,s) \in \mathfrak{so}(3)$ and $\boldsymbol{\eta}(t,s) = \Lambda(t,s)^{-1}\delta\boldsymbol{r}(t,s) \in \mathbb{R}^3$ are arbitrary functions vanishing at $t = 0, T$. We show explicitly how the expression of $\delta\boldsymbol{\omega}$ is obtained. Let $\delta\Lambda = \frac{d}{d\epsilon}\big|_{\epsilon=0}\Lambda_\epsilon$ be an arbitrary variation of Λ in (6.8), $\Lambda_{\epsilon=0} = \Lambda$. Using the definition of ω in (6.9), we have

$$\delta\omega = \left.\frac{d}{d\epsilon}\right|_{\epsilon=0}\omega_\epsilon = \left.\frac{d}{d\epsilon}\right|_{\epsilon=0}\Lambda_\epsilon^{-1}\dot{\Lambda}_\epsilon = -\Lambda^{-1}\delta\Lambda\Lambda^{-1}\dot{\Lambda} + \Lambda^{-1}\delta\dot{\Lambda}$$

$$
\begin{aligned}
&= -\Lambda^{-1}\delta\Lambda\Lambda^{-1}\dot{\Lambda} + \frac{d}{dt}\big(\Lambda^{-1}\delta\Lambda\big) - \Big(\frac{d}{dt}\Lambda^{-1}\Big)\delta\Lambda \\
&= -\Lambda^{-1}\delta\Lambda\Lambda^{-1}\dot{\Lambda} + \Lambda^{-1}\dot{\Lambda}\Lambda^{-1}\delta\Lambda + \frac{d}{dt}\big(\Lambda^{-1}\delta\Lambda\big) \\
&= -\Sigma\omega + \omega\Sigma + \partial_t\Sigma = \partial_t\Sigma + [\omega, \Sigma].
\end{aligned}
$$

Applying the inverse of the hat map (6.10) to this equality yields the desired expression for $\delta\boldsymbol{\omega}$ in (6.12). The other expressions $\delta\boldsymbol{\gamma}$, $\delta\boldsymbol{\Omega}$, and $\delta\boldsymbol{\Gamma}$ are derived in a similar way.

Hamilton's principle (6.8) yields the principle

$$
\delta \int_0^T \ell(\boldsymbol{\omega}, \boldsymbol{\gamma}, \boldsymbol{\Omega}, \boldsymbol{\Gamma})\, \mathrm{d}t = 0\,, \tag{6.14}
$$

for constrained variations $\delta\boldsymbol{\omega}$, $\delta\boldsymbol{\gamma}$, $\delta\boldsymbol{\Omega}$, $\delta\boldsymbol{\Gamma}$ given in (6.12) and (6.13), from which a direct computation yields the convective Euler–Lagrange equations

$$
\frac{D}{Dt}\frac{\delta\ell}{\delta\boldsymbol{\omega}} + \boldsymbol{\gamma}\times\frac{\delta\ell}{\delta\boldsymbol{\gamma}} + \frac{D}{Ds}\frac{\delta\ell}{\delta\boldsymbol{\Omega}} + \boldsymbol{\Gamma}\times\frac{\delta\ell}{\delta\boldsymbol{\Gamma}} = 0\,, \qquad \frac{D}{Dt}\frac{\delta\ell}{\delta\boldsymbol{\gamma}} + \frac{D}{Ds}\frac{\delta\ell}{\delta\boldsymbol{\Gamma}} = 0\,, \tag{6.15}
$$

together with the boundary conditions

$$
\left.\frac{\delta\ell}{\delta\boldsymbol{\Omega}}\right|_{s=0,L} = 0 \quad \text{and} \quad \left.\frac{\delta\ell}{\delta\boldsymbol{\Gamma}}\right|_{s=0,L} = 0. \tag{6.16}
$$

In (6.15) the symbols $\delta\ell/\delta\boldsymbol{\omega}, \delta\ell/\delta\boldsymbol{\gamma}, \ldots$ denote the functional derivatives of ℓ relative to the L^2 pairing, defined as

$$
\left.\frac{d}{d\epsilon}\right|_{\epsilon=0} \ell(\boldsymbol{\omega}+\epsilon\delta\boldsymbol{\omega}, \boldsymbol{\gamma}, \boldsymbol{\Omega}, \boldsymbol{\Gamma}) = \int_0^L \frac{\delta\ell}{\delta\boldsymbol{\omega}} \cdot \delta\boldsymbol{\omega}\, \mathrm{d}s\,. \tag{6.17}
$$

We also introduced the notations

$$
\frac{D}{Dt} = \partial_t + \boldsymbol{\omega}\times \quad \text{and} \quad \frac{D}{Ds} = \partial_s + \boldsymbol{\Omega}\times\,. \tag{6.18}
$$

If one of the extremities (say $s = 0$) of the rod is kept fixed, i.e., $\boldsymbol{r}(t, 0) = \boldsymbol{r}_0$, $\Lambda(t, 0) = \Lambda_0$ for all t, then only the boundary condition at $s = L$ arises above in (6.16). From their definition (6.9), the convective variables verify the compatibility conditions

$$
\partial_t\boldsymbol{\Omega} = \boldsymbol{\omega}\times\boldsymbol{\Omega} + \partial_s\boldsymbol{\omega} \quad \text{and} \quad \partial_t\boldsymbol{\Gamma} + \boldsymbol{\omega}\times\boldsymbol{\Gamma} = \partial_s\boldsymbol{\gamma} + \boldsymbol{\Omega}\times\boldsymbol{\gamma}. \tag{6.19}
$$

Lagrangian Reduction by Symmetry The process of passing from the Hamilton principle (6.8) to its convective version (6.14) with appropriate constrained variations is rigorously justified by the process of Lagrangian reduction that we briefly recalled earlier in Sect. 6.2. In view of the symmetries of the problem, it is natural to endow $Q_{\rm rod}$ wit the Lie group multiplication of the special Euclidean group $SE(3) = SO(3)\,\mathbb{R}^3\,\circledS$[1] (rather than, for example, the direct Lie group multiplication of $SO(3) \times \mathbb{R}^3$). We shall thus write the configuration manifold as the infinite dimensional Lie group $Q_{\rm rod} = \mathcal{F}\big([0, L], SE(3)\big)$ and observe that the Lagrangian L defined on $TQ_{\rm rod}$ is invariant under the action on left by the subgroup $G = SE(3) \subset Q_{\rm rod}$ of special Euclidean transformations that are constant on $[0, L]$. Indeed, an equivalence class $[\Lambda, \boldsymbol{r}, \dot{\Lambda}, \dot{\boldsymbol{r}}]$ in the quotient space $(TQ_{\rm rod})/G$ can be identified with the element $(\boldsymbol{\omega}, \boldsymbol{\gamma}, \boldsymbol{\Omega}, \boldsymbol{\Gamma})$, via the relations given by (6.9). From this point of view, (6.14) is the symmetry reduced variational principle induced from the Hamilton principle (6.8) and Eq. (6.15) are the symmetry reduced Euler–Lagrange equations.

This approach via Lagrangian reduction not only gives an elegant and efficient way to derive the equations for geometrically exact rods, but also turns out to be a powerful tool for the derivation of new models, as we show in the next section.

6.3 Variational Modeling for Flexible Tubes Conveying Fluids

In this section we derive a geometrically exact model for a flexible tube with expandable walls conveying a compressible fluid. We shall derive the model from the Hamilton principle, reformulated in convective variables for the tube and in spatial variables for the fluid, by making use of the symmetries of the Lagrangian.

To achieve this goal we first need to identify the infinite dimensional configuration manifold of the system as well as the convective and spatial variables, together with their relation with the Lagrangian variables.

6.3.1 Configuration Manifold for Flexible Tubes Conveying Fluid

In addition to the rod variables $(\Lambda, \boldsymbol{r}) \in \mathcal{F}\big([0, L], SO(3) \times \mathbb{R}^3\big)$ considered above, the configuration manifold for the fluid-conveying tube also contains the description of the wall and fluid motion. For the fluid, it is easier to start by defining the *back-to-labels* map, which is an embedding $\psi : [0, L] \to \mathbb{R}$, assigning to a current fluid label particle $s \in [0, L]$ located at $\boldsymbol{r}(s)$ in the tube, its Lagrangian label $s_0 \in \mathbb{R}$. Its inverse $\varphi := \psi^{-1} : \psi([0, L]) \subset \mathbb{R} \to [0, L]$ gives the current configuration of

[1] We recall that $SE(3) = SO(3)\,\mathbb{R}^3\,\circledS$ is the semidirect product of the Lie group $SO(3)$ and the vector space $\mathbb{R}^3$, with group multiplication given by $(\Lambda_1, \boldsymbol{r}_1)(\Lambda_2, \boldsymbol{r}_2) = (\Lambda_1\Lambda_2, \Lambda_1\boldsymbol{r}_2 + \boldsymbol{r}_1)$.

the fluid in the tube. A time dependent curve of such maps thus describes the fluid motion in the tube, i.e., $s = \varphi(t, s_0)$, with $s \in [0, L]$.

We include the description of the motion of the wall in the simplest possible way by considering the tube radius $R(t, s)$ to be a free variable. In this case, the Lagrangian depends on R, as well as on its time and space derivatives. If we assume that R can lie on an interval I_R, for example, $I_R = \mathbb{R}_+$ (the set of positive numbers), then the configuration manifold for the fluid-conveying tube is given by the infinite dimensional manifold

$$Q := \mathcal{F}\big([0, L], SO(3) \times \mathbb{R}^3 \times I_R\big) \times \Big\{\varphi : \varphi^{-1}[0, L] \to [0, L] \mid \varphi \text{ diffeom.}\Big\}. \tag{6.20}$$

Note that as the system evolves in time, the domain of definition of the fluid motion $s = \varphi(t, s_0)$ also changes in time since we have $\varphi(t) : [a(t), b(t)] \to [0, L]$, for $\varphi(t, a(t)) = 0$ and $\varphi(t, b(t)) = L$. The time dependent interval $[a(t), b(t)]$ contains the labels of all the fluid particles that are present in the tube at time t. See Fig. 6.2 for an illustration of the geometrically exact and expandable flexible tube conveying fluid.

In principle, we could have considered, e.g., an ellipsoidal shape with the semi-axes $\big(R_1(t, s), R_2(t, s)\big)$, or, more generally, chosen a shape that is parameterized by N variables (shape parameters). In that case, the Lagrangian depends on these parameters and there are N additional Euler–Lagrange equations for these parameters. Such generalizations of our system are relatively straightforward from the

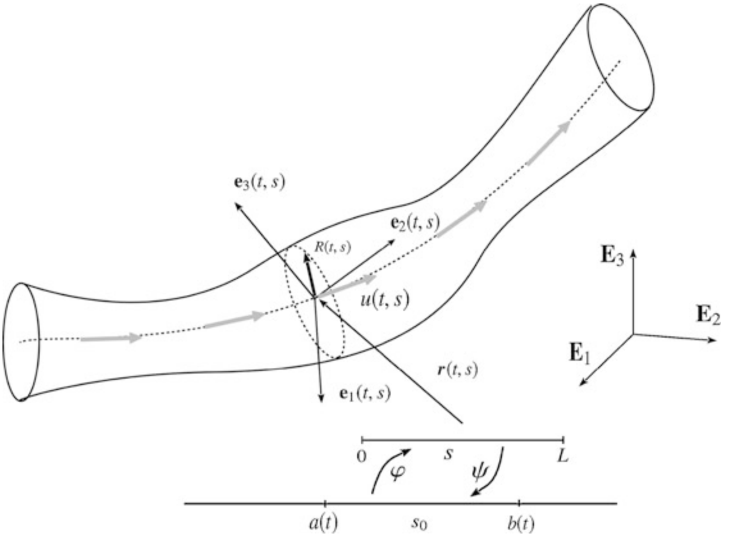


Fig. 6.2 Illustration of the geometrically exact and expandable flexible tube conveying fluid. Are represented the two frames $\{\mathbf{E}_i\}$ and $\{\mathbf{e}_i(t, s)\}$, $i = 1, 2, 3$, the radius of the cross section $R(t, s)$, the back-to-labels map ψ and its inverse φ, and the Eulerian velocity $u(t, s)$

abstract point of view, but very cumbersome from the practical point of view, since one will need to compute the dependence of moments of inertia, potential energy etc., on the shape parameters. While these considerations are useful to explain certain regimes breaking the local radial symmetry of the tube, for simplicity, we shall only consider the radially symmetric cross sections here.

6.3.2 Definition of the Lagrangian

In this section we derive the Lagrangian $L : TQ \to \mathbb{R}$ of the system, defined on the tangent bundle TQ of the infinite dimensional configuration manifold Q. It has the standard form

$$L = K_{\mathrm{rod}} + K_{\mathrm{fluid}} - E_{\mathrm{rod}} - E_{\mathrm{int}}\,, \tag{6.21}$$

with K_{rod} the kinetic energy of the rod, K_{fluid} the kinetic energy of the fluid, E_{rod} the elastic energy of the rod, and E_{int} the internal energy of the fluid.

Kinetic Energy The kinetic energy of the rod has the standard expression

$$K_{\mathrm{rod}} = \frac{1}{2}\int_0^L \Big(\alpha|\boldsymbol{\gamma}|^2 + a\dot{R}^2 + \mathbb{I}(R)\boldsymbol{\omega}\cdot\boldsymbol{\omega}\Big)|\boldsymbol{\Gamma}|\mathrm{d}s\,,$$

where α is the linear density of the tube and $\mathbb{I}(R)$ is its local moment of inertia. The term $\frac{1}{2}a\dot{R}^2$ describes the kinetic energy of the radial motion of the tube.

We now derive the total kinetic energy of the fluid K_{fluid}. In material description, the total velocity of a fluid particle with label s_0 is given by

$$\begin{aligned}\frac{d}{dt}\boldsymbol{r}(t,\varphi(t,s_0)) &= \partial_t\boldsymbol{r}(t,\varphi(t,s_0)) + \partial_s\boldsymbol{r}(t,\varphi(t,s_0))\partial_t\varphi(t,s_0)\\ &= \partial_t\boldsymbol{r}(t,\varphi(t,s_0)) + \partial_s\boldsymbol{r}(t,\varphi(t,s_0))u(t,\varphi(t,s_0))\,,\end{aligned} \tag{6.22}$$

where the Eulerian velocity is defined as usual by

$$u(t,s) = \big(\partial_t\varphi\circ\varphi^{-1}\big)(t,s)\,, \quad s\in[0,L]\,. \tag{6.23}$$

Let us recall from Sect. 6.3.1 that for each t fixed, the map $s_0 \mapsto s = \varphi(t,s_0)$ is a diffeomorphism from some interval $[a(t), b(t)]$ to the interval $[0, L]$. The map $s \mapsto s_0 = \varphi^{-1}(t,s)$ denotes its inverse, from $[0, L]$ to $[a(t), b(t)]$, for each fixed t. The total kinetic energy of the fluid reads

$$K_{\text{fluid}} = \frac{1}{2}\int_{\varphi^{-1}(t,0)}^{\varphi^{-1}(t,L)} \xi_0(s_0)\left|\frac{d}{dt}\boldsymbol{r}(t,\varphi(t,s_0))\right|^2 \mathrm{d}s_0 = \frac{1}{2}\int_0^L (\xi_0\circ\varphi^{-1})\partial_s\varphi^{-1}\,|\boldsymbol{\gamma}+\boldsymbol{\Gamma}u|^2\,\mathrm{d}s\,,$$

where

$$\xi(t,s) = \left[(\xi_0\circ\varphi^{-1})\partial_s\varphi^{-1}\right](t,s) \tag{6.24}$$

is the mass density per unit length in the Eulerian description. Equation (6.24) assumes that the fluid fills the tube completely. This assumption is clearly satisfied for tubes filled with gas, and is also presumed to be satisfied for tubes filled with fluid throughout the literature on this subject, as spontaneous creation of internal voids in the tube seems unlikely. Note the relation $\xi(t,s) = \rho(t,s)Q(t,s)$, where $\rho(t,s)$ is the mass density of the fluid per unit volume, in units Mass/Length^3, and $Q(t,s)$ is the area of the tube's cross section, in units Length^2. It is important to note that, while ξ, ξ_0 are related as in (6.24), such a relation does not hold for ρ, ρ_0 and Q, Q_0, e.g., $Q(s,t) \neq (Q_0\circ\varphi^{-1})\partial_s\varphi^{-1}$. That relationship between Q and Q_0 is only valid when the fluid inside the tube is incompressible, as we shall see below.

It is interesting to note that mathematically, the Lagrangian mapping defines the physical movement of particles from one cross-section to another, parameterized by the Lagrangian label s_0. All particles for one cross-section have the same Lagrangian label s_0 and move to the same cross-section s after time t. This corresponds to the assumption of a 'plug flow'. To generalize this notion for movement of fluid with friction, one could consider (formally) the velocity $u(t,s)$ given by (6.23) to be an 'effective' Eulerian velocity (e.g., averaged over a given cross-section). Then, one could derive an effective kinetic energy by adding experimentally relevant coefficients to the kinetic energy of the fluid presented above. We shall avoid this approach and all discussion of friction in this paper, as these are difficult subjects beyond the scope of our considerations here. Instead, we assume an inviscid flow and consider the plug flow assumption to be valid, consistent with the previous literature on the subject.

Internal Energy We assume that the internal energy of the gas is described by the specific energy function $e(\rho,S)$, with ρ the mass density and S the specific entropy. The total internal energy of the fluid is thus $E_{\text{int}} = \int_0^L \xi e(\rho,S)\mathrm{d}s$.

Elastic Energy The potential energy due to elastic deformation is a function of $\boldsymbol{\Omega}$, $\boldsymbol{\Gamma}$ and R. While the equations will be derived for an arbitrary potential energy, we shall assume the simplest possible quadratic expression for the calculations, namely,

$$E_{\rm rod} = \frac{1}{2}\int_0^L \Big(\mathbb{J}\boldsymbol{\Omega}\cdot\boldsymbol{\Omega} + \lambda(R)|\boldsymbol{\Gamma} - \boldsymbol{\chi}|^2 + 2F(R, R', R'')\Big)|\boldsymbol{\Gamma}|{\rm d}s\,, \tag{6.25}$$

where $\boldsymbol{\chi} \in \mathbb{R}^3$ is a fixed vector denoting the axis of the tube in the reference configuration, $\mathbb{J}$ is a symmetric positive definite 3×3 matrix, which may depend on R, R' and R'', and $\lambda(R)$ is the stretching rigidity of the tube. The stretching term proportional to $\lambda(R)$ can take the more general form $\mathbb{K}(\boldsymbol{\Gamma} - \boldsymbol{\chi}) \cdot (\boldsymbol{\Gamma} - \boldsymbol{\chi})$, where $\mathbb{K}$ is a 3×3 tensor. The part of this expression for the elastic energy containing the first two terms in (6.25) is commonly used for a Cosserat elastic rod, but more general functions of deformations $\boldsymbol{\Gamma}$ are possible. A particular case is a quadratic function of $\boldsymbol{\Gamma}$ leading to a linear dependence between stresses and strains. We have also introduced the elastic energy of wall $F(R, R', R'')$ which can be explicitly computed for simple elastic tubes.

Mass Conservation We shall assume that the fluid fills the tube completely, and the fluid velocity at each given cross-section is aligned with the axis of the tube. Since we are assuming a one-dimensional approximation for the fluid motion inside the tube, the mass density per unit length $\xi(t, s)$ has to verify (6.24), from which we deduce the conservation law

$$\partial_t \xi + \partial_s(\xi u) = 0. \tag{6.26}$$

The physical meaning of Q can be understood through the fact that $Q{\rm d}s$ is the volume filled by the fluid in the tube for the parameter interval $[s, s + {\rm d}s]$. Then, the infinitesimal volume is $A|{\rm d}\boldsymbol{r}|$, where A is the area of the cross section. Since s is not necessarily the arc length, it is useful to define the variable $Q = A|\boldsymbol{r}'| = A|\boldsymbol{\Gamma}|$, so the infinitesimal volume at a given point is then written as $A|{\rm d}\boldsymbol{r}| = Q{\rm d}s$.

Expression of the Cross Section Area In general Q is a given function of the tube's variables, i.e.,

$$Q = Q(R, \boldsymbol{\Omega}, \boldsymbol{\Gamma}) = A(R, \boldsymbol{\Omega}, \boldsymbol{\Gamma})|\boldsymbol{\Gamma}|\,. \tag{6.27}$$

A dependence of the form $A = A(\boldsymbol{\Omega}, \boldsymbol{\Gamma})$ was taken in [22, 23, 26]. Such choice prevents the independent dynamics of the tube's wall and states that the cross-sectional area only depends on the deformation of the tube as an elastic rod. For the physical explanation of possible particular expressions of $A(\boldsymbol{\Omega}, \boldsymbol{\Gamma})$ we refer the reader to [26]. The simplest choice allowing for independent dynamics of the wall is $A(R) = \pi R^2$, in which case the tube preserves its circular cross-section under deformations, see [24].

Lagrangian From all the expressions given above and assuming there is a uniform external pressure $p_{\rm ext}$ acting on the tube, we get the Lagrangian

$$
\begin{aligned}
&\ell(\boldsymbol{\omega}, \boldsymbol{\gamma}, \boldsymbol{\Omega}, \boldsymbol{\Gamma}, u, \xi, S, R, \dot{R})\\
&\quad = \int_0^L \Big[\Big(\frac{1}{2}\alpha|\boldsymbol{\gamma}|^2 + \frac{1}{2}\mathbb{I}(R)\boldsymbol{\omega}\cdot\boldsymbol{\omega} + \frac{1}{2}a\dot{R}^2 - F(R, R', R'') - \frac{1}{2}\mathbb{J}\boldsymbol{\Omega}\cdot\boldsymbol{\Omega}\\
&\qquad\qquad - \frac{1}{2}\lambda(R)|\boldsymbol{\Gamma} - \boldsymbol{\chi}|^2\Big)|\boldsymbol{\Gamma}| + \frac{1}{2}\xi\,|\boldsymbol{\gamma} + \boldsymbol{\Gamma}u|^2 - \xi e(\rho, S) - p_{\text{ext}}Q\Big]\mathrm{d}s\\
&\quad =: \int_0^L \Big[\mathcal{L}_0\big(\boldsymbol{\omega}, \boldsymbol{\gamma}, \boldsymbol{\Omega}, \boldsymbol{\Gamma}, u, \xi, R, \dot{R}, R', R''\big) - \xi e(\rho, S) - p_{\text{ext}}Q\Big]\mathrm{d}s\,,
\end{aligned}
\tag{6.28}
$$

where ρ, in the term $\xi e(\rho, S)$ in the above formula, is defined in terms of the independent variables ξ, $\boldsymbol{\Omega}$, $\boldsymbol{\Gamma}$, R as

$$
\rho := \frac{\xi}{Q(\boldsymbol{\Omega}, \boldsymbol{\Gamma}, R)}\,. \tag{6.29}
$$

We have denoted $\mathcal{L}_0$ the part of the integrand of the Lagrangian related to just the tube dynamics, without the incorporation of the internal energy.

6.3.3 Variational Principle and Equations of Motion

Recall that the Lagrangian of the system is defined on the tangent bundle TQ and is therefore a function of the form $L\big(\Lambda, \dot{\Lambda}, \boldsymbol{r}, \dot{\boldsymbol{r}}, \varphi, \dot{\varphi}, R, \dot{R}\big)$. The equations of motion are directly obtained from the Hamilton principle with boundary forces

$$
\delta\int_0^T L\big(\Lambda, \dot{\Lambda}, \boldsymbol{r}, \dot{\boldsymbol{r}}, \varphi, \dot{\varphi}, R, \dot{R}\big)\mathrm{d}t = \int_0^T \big(F_\Lambda\cdot\delta\Lambda + F_{\boldsymbol{r}}\cdot\delta\boldsymbol{r} + F_\varphi\delta\varphi + F_R\delta R\big)\Big|_{s=L}\mathrm{d}t\,, \tag{6.30}
$$

for arbitrary variations $\delta\Lambda, \delta\boldsymbol{r}, \delta\varphi, \delta R$ vanishing at $t = 0, T$. This is a special instance of the variational principle (6.2) with the boundary terms in (6.30) playing the role of external forces in (6.2). We chose here the free boundary extremity to be at $s = L$ and denoted by F_Λ, $F_{\boldsymbol{r}}$, F_φ the generalized forces exerted by the boundary effects on the linear momentum equation for the tube, on the angular momentum equation for the tube and on the fluid momentum equation. We will explain later how these forces are chosen. We also assume that all variables are known at $s = 0$ hence all variations vanish at $s = 0$.

As we have seen in Sect. 6.3.2, the Lagrangian is naturally expressed in terms of the *convective variables* $\boldsymbol{\omega}$, $\boldsymbol{\gamma}$, $\boldsymbol{\Omega}$, $\boldsymbol{\Gamma}$ for the rod and the *spatial variables* u, ξ, S for the fluid, see (6.28). Here, the term 'spatial' is used to the motion with respect to the rod's coordinate frame, and not the corresponding quantities in the laboratory frame. In a similar way with the case of the geometrically exact rod above, ℓ can be interpreted as the symmetry reduced Lagrangian associated to L defined on TQ.

By the process of Lagrangian reduction by symmetries, (6.30) induces the reduced Hamilton principle

$$\delta \int_0^T \ell\big(\boldsymbol{\omega}, \boldsymbol{\gamma}, \boldsymbol{\Omega}, \boldsymbol{\Gamma}, u, \xi, S, R, \dot{R}\big) \mathrm{d}t = \int_0^T \left(f_{\boldsymbol{\Omega}} \cdot \boldsymbol{\Sigma} + f_{\boldsymbol{\Gamma}} \cdot \boldsymbol{\eta} + f_u \eta + F_R \delta R\right)\Big|_{s=L} \mathrm{d}t, \tag{6.31}$$

for variations $\delta\boldsymbol{\omega}, \delta\boldsymbol{\gamma}, \delta\boldsymbol{\Omega}, \delta\boldsymbol{\Gamma}$ given in (6.12) and (6.13), and δu, $\delta\xi$, and δS computed as

$$\delta u = \partial_t \eta + u \partial_s \eta - \eta \partial_s u\,, \qquad \delta\xi = -\partial_s(\xi\eta)\,, \qquad \delta S = -\eta\partial_s S\,, \tag{6.32}$$

where $\eta = \delta\varphi \circ \varphi^{-1}$. Note that $\eta(t,s)$ is an arbitrary function vanishing at $t = 0, T$. We have assumed that the forces in (6.30) have the same symmetry with the Lagrangian and denoted by $f_{\boldsymbol{\Omega}}$, $f_{\boldsymbol{\Gamma}}$, f_u their reduced expressions. A lengthy computation yields the system

$$\begin{cases} \dfrac{D}{Dt}\dfrac{\delta\ell}{\delta\boldsymbol{\omega}} + \boldsymbol{\gamma} \times \dfrac{\delta\ell}{\delta\boldsymbol{\gamma}} + \dfrac{D}{Ds}\dfrac{\delta\ell}{\delta\boldsymbol{\Omega}} + \boldsymbol{\Gamma} \times \dfrac{\delta\ell}{\delta\boldsymbol{\Gamma}} = 0\,, \quad \dfrac{D}{Dt}\dfrac{\delta\ell}{\delta\boldsymbol{\gamma}} + \dfrac{D}{Ds}\dfrac{\delta\ell}{\delta\boldsymbol{\Gamma}} = 0 \\ \partial_t \dfrac{\delta\ell}{\delta u} + u\partial_s \dfrac{\delta\ell}{\delta u} + 2\dfrac{\delta\ell}{\delta u}\partial_s u = \xi\partial_s \dfrac{\delta\ell}{\delta\xi} - \dfrac{\delta\ell}{\delta S}\partial_s S \\ \partial_t \dfrac{\delta\ell}{\delta\dot{R}} - \dfrac{\delta\ell}{\delta R} = 0\,, \qquad \partial_t\xi + \partial_s(\xi u) = 0\,, \qquad \partial_t S + u\partial_s S = 0\,, \\ \partial_t\boldsymbol{\Omega} = \boldsymbol{\Omega} \times \boldsymbol{\omega} + \partial_s\boldsymbol{\omega}\,, \qquad \partial_t\boldsymbol{\Gamma} + \boldsymbol{\omega} \times \boldsymbol{\Gamma} = \partial_s\boldsymbol{\gamma} + \boldsymbol{\Omega} \times \boldsymbol{\gamma}\,, \end{cases} \tag{6.33}$$

where we used the notations D/Dt, D/Ds introduced in (6.18) and the functional derivatives as defined in (6.17). The principle also yields boundary conditions that will be computed explicitly below.

Note that the first equation arises from the terms proportional to $\boldsymbol{\Sigma}$ in the variation of the action functional and thus describes the conservation of angular momentum. The second equation arises from the terms proportional to $\boldsymbol{\psi}$ and describes the conservation of linear momentum. The third equation is obtained by collecting the terms proportional to η and describes the conservation of fluid momentum. The fourth equation comes from collecting the terms proportional to δR and describes the elastic deformation of the walls due to the pressure. Finally, the last four equations arise from the four definitions $\Omega = \Lambda^{-1}\Lambda'$, $\boldsymbol{\Gamma} = \Lambda^{-1}\boldsymbol{r}'$, $\xi = (\xi_0 \circ \varphi^{-1})\partial_s\varphi^{-1}$, and $S = S_0 \circ \varphi^{-1}$.

For the Lagrangian given in (6.28), the functional derivatives are computed as

$$\begin{aligned} &\frac{\delta\ell}{\delta\boldsymbol{\Omega}} = \frac{\partial\mathcal{L}_0}{\partial\boldsymbol{\Omega}} + (p - p_{\text{ext}})\frac{\partial Q}{\partial\boldsymbol{\Omega}}\,, \qquad \frac{\delta\ell}{\delta\boldsymbol{\Gamma}} = \frac{\partial\mathcal{L}_0}{\partial\boldsymbol{\Gamma}} + (p - p_{\text{ext}})\frac{\partial Q}{\partial\boldsymbol{\Gamma}} \\ &\frac{\delta\ell}{\delta R} = \frac{\partial\mathcal{L}_0}{\partial R} - \partial_s\frac{\partial\mathcal{L}_0}{\partial R'} + \partial_s^2\frac{\partial\mathcal{L}_0}{\partial R''} + (p - p_{\text{ext}})\frac{\partial Q}{\partial R} \\ &\frac{\delta\ell}{\delta\xi} = \frac{\partial\mathcal{L}_0}{\partial\xi} - e - \rho\frac{\partial e}{\partial\rho}\,, \end{aligned} \tag{6.34}$$

where $p(\rho, S) = \rho^2 \frac{\partial e}{\partial \rho}(\rho, S)$ is the pressure and $\partial \mathcal{L}_0/\partial \boldsymbol{\Omega}$, $\partial \mathcal{L}_0/\partial \boldsymbol{\Gamma}, \ldots$ denote the ordinary partial derivatives of $\mathcal{L}_0$, whose explicit form can be directly computed from the expression of $\mathcal{L}_0$ in (6.28).

Theorem 6.1 *For the Lagrangian ℓ in (6.28), the variational principle (6.31) with constrained variations (6.12), (6.13), and (6.32) yields the equations of motion*

$$\begin{cases} \dfrac{D}{Dt}\dfrac{\partial \mathcal{L}_0}{\partial \boldsymbol{\omega}} + \boldsymbol{\gamma} \times \dfrac{\partial \mathcal{L}_0}{\partial \boldsymbol{\gamma}} + \dfrac{D}{Ds}\left(\dfrac{\partial \mathcal{L}_0}{\partial \boldsymbol{\Omega}} + (p - p_{\text{ext}})\dfrac{\partial Q}{\partial \boldsymbol{\Omega}}\right) \\ \qquad\qquad\qquad + \boldsymbol{\Gamma} \times \left(\dfrac{\partial \ell_0}{\partial \boldsymbol{\Gamma}} + (p - p_{\text{ext}})\dfrac{\partial Q}{\partial \boldsymbol{\Gamma}}\right) = 0 \\ \dfrac{D}{Dt}\dfrac{\partial \mathcal{L}_0}{\partial \boldsymbol{\gamma}} + \dfrac{D}{Ds}\left(\dfrac{\partial \mathcal{L}_0}{\partial \boldsymbol{\Gamma}} + (p - p_{\text{ext}})\dfrac{\partial Q}{\partial \boldsymbol{\Gamma}}\right) = 0 \\ \partial_t \dfrac{\partial \mathcal{L}_0}{\partial u} + u \partial_s \dfrac{\partial \mathcal{L}_0}{\partial u} + 2\dfrac{\partial \mathcal{L}_0}{\partial u}\partial_s u = \xi \partial_s \dfrac{\partial \mathcal{L}_0}{\partial \xi} - Q \partial_s p \\ \partial_t \dfrac{\partial \mathcal{L}_0}{\partial \dot{R}} - \partial_s^2 \dfrac{\partial \mathcal{L}_0}{\partial R''} + \partial_s \dfrac{\partial \mathcal{L}_0}{\partial R'} - \dfrac{\partial \mathcal{L}_0}{\partial R} - (p - p_{\text{ext}})\dfrac{\partial Q}{\partial R} = 0 \\ \partial_t \boldsymbol{\Omega} = \boldsymbol{\Omega} \times \boldsymbol{\omega} + \partial_s \boldsymbol{\omega}, \qquad \partial_t \boldsymbol{\Gamma} + \boldsymbol{\omega} \times \boldsymbol{\Gamma} = \partial_s \boldsymbol{\gamma} + \boldsymbol{\Omega} \times \boldsymbol{\gamma} \\ \partial_t \xi + \partial_s(\xi u) = 0, \qquad \partial_t S + u \partial_s S = 0 \end{cases} \tag{6.35}$$

together with the boundary conditions

$$\left.\frac{\partial \mathcal{L}_0}{\delta \boldsymbol{\Omega}} + (p - p_{\text{ext}})\frac{\partial Q}{\partial \boldsymbol{\Omega}} - f_{\boldsymbol{\Omega}}\right|_{s=L} = 0\,, \quad \left.\frac{\partial \mathcal{L}_0}{\delta \boldsymbol{\Gamma}} - (p - p_{\text{ext}})\frac{\partial Q}{\partial \boldsymbol{\Gamma}} - f_{\boldsymbol{\Gamma}}\right|_{s=L} = 0\,,$$
$$\left.\frac{\partial \mathcal{L}_0}{\partial u}u - \frac{\partial \mathcal{L}_0}{\partial \xi}\xi + h\xi - f_u\right|_{s=0,L} = 0\,, \quad \left.\frac{\partial \mathcal{L}_0}{\partial R'} - \partial_s \frac{\partial \mathcal{L}_0}{\partial R''}\right|_{s=L} = F_R\,, \quad \left.\frac{\partial \mathcal{L}_0}{\partial R''}\right|_{s=L} = 0\,, \tag{6.36}$$

with $h = e + p/\rho$ the enthalpy of the fluid.

Let us assume that there are no external forces acting on the system, and all the extra work is provided by the nonconservative boundary conditions. In this case, the boundary forces $f_{\boldsymbol{\Omega}}$, $f_{\boldsymbol{\gamma}}$, f_u are determined from the relations (6.36), by imposing standard boundary conditions for the tube variables at $s = L$, such as vanishing stresses of the elastic part of the Lagrangian, and a assuming specified velocity at $s = L$ (due, for example, to a special nozzle), see [22].

Note that by defining the variable $m := \frac{1}{\rho Q}\frac{\partial \mathcal{L}_0}{\partial u} = \boldsymbol{\Gamma} \cdot (\boldsymbol{\gamma} + u\boldsymbol{\Gamma})$ the third equation in (6.35) can be simply written as

$$\partial_t m + \partial_s\left(mu - \frac{\partial \mathcal{L}_0}{\partial \xi}\right) = -\frac{1}{\rho}\partial_s p\,, \tag{6.37}$$

which is strongly reminiscent of the 1D gas dynamics. For $Q(R, \boldsymbol{\Gamma}) = A(R)|\boldsymbol{\Gamma}|$, Eq. (6.35) can be further simplified since $\boldsymbol{\Gamma} \times \frac{\partial Q}{\partial \boldsymbol{\Gamma}} = 0$.

6.3.4 Incompressible Fluids

The incompressibility of the fluid motion is imposed by requiring that the mass density per unit volume is a constant number:

$$\rho(t,s)=\rho_0\,. \tag{6.38}$$

For a given expression $Q=Q(\boldsymbol{\Omega},\boldsymbol{\Gamma},R)$ of the area in terms of the tube variables, the relation (6.29) still holds with $\rho=\rho_0$. The constraint (6.38) can thus be written as

$$Q(\boldsymbol{\Omega},\boldsymbol{\Gamma},R)=\frac{\xi}{\rho_0}. \tag{6.39}$$

By recalling the relations $\xi=(\xi_0\circ\varphi^{-1})\partial_s\varphi^{-1}$, $\Omega=\Lambda^{-1}\Lambda'$, $\boldsymbol{\Gamma}=\Lambda^{-1}\boldsymbol{r}'$, we see that condition (6.39) defines a constraint of the abstract form $\Phi(\Lambda,\boldsymbol{r},\varphi)=0$ which is therefore holonomic on the infinite dimensional manifold Q. Following (6.4) recalled in Sect. 6.2, the holonomic constraint is included in the Hamilton principle (6.30) as

$$\begin{aligned}\delta\int_0^T\left[L\big(\Lambda,\dot\Lambda,\boldsymbol{r},\dot{\boldsymbol{r}},\varphi,\dot\varphi,R,\dot R\big)+\int_0^L\mu\,\Phi\big(\Lambda,\boldsymbol{r},\varphi\big)\mathrm{d}s\right]\mathrm{d}t\\ =\int_0^T\big(F_\Lambda\cdot\delta\Lambda+F_{\boldsymbol{r}}\cdot\delta\boldsymbol{r}+F_\varphi\delta\varphi+F_R\delta R\big)\Big|_{s=L}\mathrm{d}t\,,\end{aligned} \tag{6.40}$$

for arbitrary variations $\delta\Lambda$, $\delta\boldsymbol{r}$, $\delta\varphi$, δR, $\delta\mu$. In the incompressible case, the variables ξ and S are not present, as well as the terms associated to the internal energy, Therefore the reduced Lagrangian takes the form

$$\ell(\boldsymbol{\omega},\boldsymbol{\gamma},\boldsymbol{\Omega},\boldsymbol{\Gamma},u,R,\dot R)=\int_0^L\left[\mathcal{L}_0\big(\boldsymbol{\omega},\boldsymbol{\gamma},\boldsymbol{\Omega},\boldsymbol{\Gamma},u,R,\dot R,R',R''\big)-p_{\rm ext}Q\right]\mathrm{d}s\,,$$

and (6.40) yields the reduced Hamilton's principle with holonomic constraint and forces

$$\begin{aligned}\delta\int_0^T\!\!\Big[\ell\big(\boldsymbol{\omega},\boldsymbol{\gamma},\boldsymbol{\Omega},\boldsymbol{\Gamma},u,R,\dot R\big)+\int_0^L\mu\Big(Q(\boldsymbol{\Omega},\boldsymbol{\Gamma},R)-(Q_0\circ\varphi^{-1})\partial_s\varphi^{-1}\Big)\mathrm{d}s\Big]\,\mathrm{d}t\\ =\int_0^T\big(f_{\boldsymbol{\Omega}}\cdot\boldsymbol{\Sigma}+f_{\boldsymbol{\Gamma}}\cdot\boldsymbol{\eta}+f_u\eta\big)\Big|_{s=L}\mathrm{d}t\end{aligned} \tag{6.41}$$

from which the following system is obtained

$$\begin{cases} \dfrac{D}{Dt}\dfrac{\partial \mathcal{L}_0}{\partial \boldsymbol{\omega}} + \boldsymbol{\gamma} \times \dfrac{\partial \mathcal{L}_0}{\partial \boldsymbol{\gamma}} + \dfrac{D}{Ds}\left(\dfrac{\partial \mathcal{L}_0}{\partial \boldsymbol{\Omega}} + (\mu - p_{\text{ext}})\dfrac{\partial Q}{\partial \boldsymbol{\Omega}}\right) \\ \qquad\qquad + \boldsymbol{\Gamma} \times \left(\dfrac{\partial \mathcal{L}_0}{\partial \boldsymbol{\Gamma}} + (\mu - p_{\text{ext}})\dfrac{\partial Q}{\partial \boldsymbol{\Gamma}}\right) = 0 \\ \dfrac{D}{Dt}\dfrac{\partial \mathcal{L}_0}{\partial \boldsymbol{\gamma}} + \dfrac{D}{Ds}\left(\dfrac{\partial \mathcal{L}_0}{\partial \boldsymbol{\Gamma}} + (\mu - p_{\text{ext}})\dfrac{\partial Q}{\partial \boldsymbol{\Gamma}}\right) = 0 \\ \partial_t \dfrac{\partial \mathcal{L}_0}{\partial u} + u\partial_s \dfrac{\partial \mathcal{L}_0}{\partial u} + 2\dfrac{\partial \mathcal{L}_0}{\partial u}\partial_s u = -Q\partial_s \mu \\ \partial_t \dfrac{\partial \mathcal{L}_0}{\partial \dot{R}} - \partial_s^2 \dfrac{\partial \mathcal{L}_0}{\partial R''} + \partial_s \dfrac{\partial \mathcal{L}_0}{\partial R'} - \dfrac{\partial \mathcal{L}_0}{\partial R} - (\mu - p_{\text{ext}})\dfrac{\partial Q}{\partial R} = 0 \\ \partial_t \boldsymbol{\Omega} = \boldsymbol{\omega} \times \boldsymbol{\Omega} + \partial_s \boldsymbol{\omega}\,, \qquad \partial_t \boldsymbol{\Gamma} + \boldsymbol{\omega} \times \boldsymbol{\Gamma} = \partial_s \boldsymbol{\gamma} + \boldsymbol{\Omega} \times \boldsymbol{\gamma} \\ \partial_t Q + \partial_s (Qu) = 0\,, \end{cases} \tag{6.42}$$

together with the boundary conditions

$$\left.\frac{\partial \mathcal{L}_0}{\delta \boldsymbol{\Omega}} + (\mu - p_{\text{ext}})\frac{\partial Q}{\partial \boldsymbol{\Omega}} - f_{\boldsymbol{\Omega}}\right|_{s=0,L} = 0\,, \qquad \left.\frac{\partial \mathcal{L}_0}{\delta \boldsymbol{\Gamma}} - (\mu - p_{\text{ext}})\frac{\partial Q}{\partial \boldsymbol{\Gamma}} - f_{\boldsymbol{\Gamma}}\right|_{s=0,L} = 0\,,$$
$$\left.\frac{\partial \mathcal{L}_0}{\partial u}u + \mu Q - f_u\right|_{s=0,L} = 0\,, \qquad \left.\frac{\partial \mathcal{L}_0}{\partial R'} - \partial_s \frac{\partial \mathcal{L}_0}{\partial R''}\right|_{s=L} = F_R\,, \qquad \left.\frac{\partial \mathcal{L}_0}{\partial R''}\right|_{s=L} = 0\,. \tag{6.43}$$

A direct comparison with the compressible system (6.35) shows that the Lagrange multiplier μ plays the role of the fluid pressure p.

Note that the fluid momentum equation in (6.42) takes the following simple expression in terms of $m := \frac{1}{Q}\frac{\partial \mathcal{L}_0}{\partial u}$

$$\partial_t m + \partial_s (mu + \mu) = 0\,. \tag{6.44}$$

6.3.5 *Comparison with Previous Models*

The case of an inextensible unshearable tube can be easily obtained from the above variational formulation by imposing the constraint $\boldsymbol{\Gamma}(t,s) = \boldsymbol{\chi}$, for all t, s, via a Lagrange multiplier approach, see [24]. If we further assume a straight tube without rotational motion, the incompressible version (6.42) yields

$$\begin{cases} \partial_t(\rho_0 A u) + \partial_s(\rho_0 A u^2) = -A\partial_s \mu \\ a\ddot{R} - \partial_s \dfrac{\partial F}{\partial R'} + \dfrac{\partial F}{\partial R} = 2\pi R(\mu - p_{\text{ext}}) \\ \partial_t A + \partial_s(Au) = 0\,, \end{cases} \tag{6.45}$$

where we chose $A(R) = \pi R^2$. In the incompressible system (6.45), we have $\xi = \rho_0 A$, with $\rho_0 = const$. As is demonstrated in [24], system (6.45) reduces to previous models in the literature on the dynamics of tubes with expandable walls [32, 34, 36, 56, 61, 66, 67], as well as more complex models involving wall inertia [20, 58, 59], as long as necessary friction terms are added as external forces.

6.3.6 Conservation Laws for Gas Motion and Rankine–Hugoniot Conditions

Since we are concerned with the flow of compressible fluids, it is natural to ask about the existence of shock waves and the conditions the shock solutions must satisfy at the discontinuity. In the one-dimensional motion of a compressible fluid, the constraints on jumps of quantities across the shock are known as the Rankine–Hugoniot conditions. Let us for shortness denote by $[a]$ the jump of the quantity a across the shock, and c the velocity of the shock. The classical Rankine–Hugoniot conditions for the one-dimensional motion of a compressible fluid gives the continuity of the corresponding quantities as

$$c[\rho] = [\rho u] \quad \text{(mass)}\,, \tag{6.46}$$

$$c[\rho u] = [\rho u^2 + p] \quad \text{(momentum)}\,, \tag{6.47}$$

$$c[E] = \left[\left(\frac{1}{2}\rho u^2 + \rho e + p\right) u\right], \quad E = \frac{1}{2}\rho u^2 + \rho e \quad \text{(energy)}\,, \tag{6.48}$$

see, e.g. [71], where we have defined E to be the total energy density of the gas. As far as we are aware, the analogue of the Rankine–Hugoniot conditions for the moving and expandable tube conveying gas have been first derived in [24], to which we refer the reader for details.

The mass conservation (6.26) is already written in a conservation law form. We rewrite the balance of fluid momentum in the following form

$$\partial_t\big(\xi\boldsymbol{\Gamma}\cdot(\boldsymbol{\gamma} + u\boldsymbol{\Gamma})\big) + \partial_s\big(u\xi\boldsymbol{\Gamma}\cdot(\boldsymbol{\gamma} + u\boldsymbol{\Gamma}) + pQ\big) - \xi(\boldsymbol{\gamma} + u\boldsymbol{\Gamma})\cdot(\partial_s\boldsymbol{\gamma} + u\partial_s\boldsymbol{\Gamma}) = p\partial_s Q\,, \tag{6.49}$$

obtained by inserting the actual expressions of the functional derivatives in the third equation of (6.35).

The derivation of the corresponding energy equation is rather tedious and we will only sketch it, presenting the final solution. For simplicity, we set $p_{\text{ext}} = 0$. We define the total energy $\mathbb{E}$, including the thermal and mechanical terms, and the energy density E as

$$\mathbb{E} = \int_0^L E\,\mathrm{d}s\,, \qquad E := \xi e + \dot{R}\frac{\partial\mathcal{L}_0}{\partial\dot{R}} + \boldsymbol{\omega}\cdot\frac{\partial\mathcal{L}_0}{\partial\boldsymbol{\omega}} + \boldsymbol{\gamma}\cdot\frac{\partial\mathcal{L}_0}{\partial\boldsymbol{\gamma}} + u\frac{\partial\mathcal{L}_0}{\partial u} - \mathcal{L}_0\,. \tag{6.50}$$

Then, performing appropriate substitution for time derivatives of the terms in (6.50) using equations of motion (6.35), we obtain the conservation laws for the energy density E as

$$\partial_t E + \partial_s J = 0$$

for the energy flux J given by

$$J := \boldsymbol{\omega} \cdot \frac{\partial \ell_0}{\partial \boldsymbol{\Omega}} + \boldsymbol{\gamma} \cdot \frac{\partial \ell_0}{\partial \boldsymbol{\Gamma}} + \dot{R} \frac{\partial \ell_0}{\partial R'} + u^2 \frac{\partial \ell_0}{\partial u} - \xi u \frac{\partial \ell_0}{\partial \xi} + p \boldsymbol{\gamma} \cdot \frac{\partial Q}{\partial \boldsymbol{\Gamma}} + \left(\frac{p}{\rho} + e \right) \xi u. \tag{6.51}$$

Notice an interesting symmetry between time derivatives and spatial derivatives in the expression for the energy flux J in (6.51). Taking only the jumps at the discontinuous terms, and assuming the continuity of the tube, i.e., continuity of $\boldsymbol{\gamma}$, $\boldsymbol{\Gamma}$, Q, as well as R and R', we arrive at the following conservation laws for the shock wave moving with velocity c [24]:

$$c[\rho] = [\rho u] \tag{6.52}$$

$$c\,[\rho]\,\boldsymbol{\Gamma} \cdot \boldsymbol{\gamma} + c\,[\rho u]\,|\boldsymbol{\Gamma}|^2 = [\rho u]\,\boldsymbol{\Gamma} \cdot \boldsymbol{\gamma} + \left[\rho u^2\right] |\boldsymbol{\Gamma}|^2 + [p] \tag{6.53}$$

$$c \left[\rho \left(e + \frac{1}{2} |\boldsymbol{\gamma} + \boldsymbol{\Gamma} u|^2 \right) \right] = \left[\frac{1}{2} \rho u \, |\boldsymbol{\gamma} + \boldsymbol{\Gamma} u|^2 + \frac{p}{|\boldsymbol{\Gamma}|^2} \boldsymbol{\Gamma} \cdot (\boldsymbol{\gamma} + \boldsymbol{\Gamma} u) + \rho u e \right]. \tag{6.54}$$

For comparison with the classical Rankine–Hugoniot condition we set the tube to be circular, so $R' = 0$ and $\dot{R} = 0$, and static with a straight centerline, so $\boldsymbol{\gamma} = \mathbf{0}$, $\boldsymbol{\omega} = \mathbf{0}$, $\boldsymbol{\Omega} = \mathbf{0}$, $\boldsymbol{\Gamma} = \mathbf{E}_1$, hence $Q = A_0$. Then, the mass conservation law (6.52) reduces to (6.46) and (6.53) reduces to (6.47), and (6.54) to (6.48). We note that the extensions of the Rankine–Hugoniot conditions we have derived here are valid for all configurations of the tube in our framework, and they account for motion of the fluid, the motion of the tube in space and its deformations, and also the expansion/contraction of its cross-section coming from the dynamics of the radius $R(t, s)$.

We also present numerical simulations of a traveling wave with a shock in the gas propagating inside an elastic tube on Fig. 6.3, which is also the solution computed in [24]. We are looking for a physical configuration coming from a shock tube experiment, when the motion of the gas is driven by the initial jump in pressure on one side of the tube, and consider the case when the motion has stabilized as a traveling wave. Such a traveling wave solution will have all the variables depend on the combination of $s - ct$, for some constant c depending on the parameters of the problem. See the description presented in [24] for the exact values of parameters of the tube and pressures used in simulations. We are looking for a solution $R(x)$ that is smooth for $x > 0$ and $x < 0$ and tends to steady states $R = R_\pm$ as $R \to \pm\infty$, which is a configuration expected for an experimental realization in

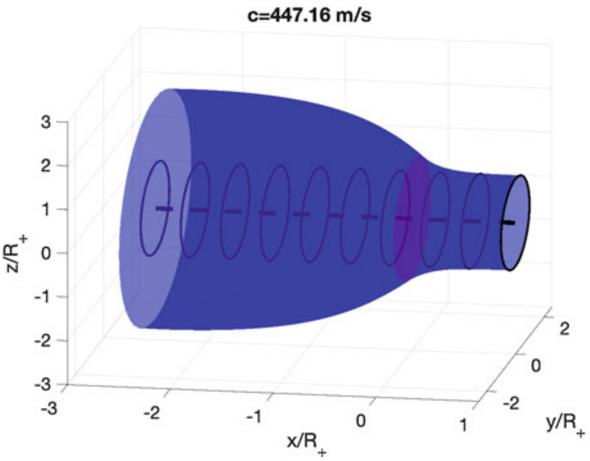


Fig. 6.3 A shock propagating along a tube. Propagating shock is shown in red, moving in the direction of positive x with velocity c, i.e. $s = x - ct$. The position of the shock is chosen to be at $x = 0$. The speed of the shock is computed to be $c \sim 447$ m/s. The centerline of the tube is shown with a solid black line, and the undisturbed position of the cross-section of the tube are shown by solid circles

a shock tube. The solution $R(x)$ and $R'(x)$ is expected to be continuous at the shock, whereas the variable $u(x)$, $\rho(x)$, and $S(x)$ have a jump at the shock satisfying Rankine–Hugoniot conditions (6.52)–(6.54). For a chosen value of $R = R_s$ at the shock, such solution will only exist for a particular value of c, also yielding the limiting value $R = R_-$ that is dependent on the choice of R_s. we present a solution computed at $c \simeq 447$ m/s, with the limiting pressure behind the shock wave being $p_+ \simeq 1.826$ atm. The solution is presented in the dimensionless coordinates $(x, y, z)/R_+$.

6.4 Variational Discretization for Flexible Tubes Conveying Fluids

6.4.1 Spatial Discretization

We first consider the spatial discretization of the variational approach outlined in the previous section, keeping the time continuous. These results yield simplified, but geometrically consistent, low-component models for further analytical and numerical analysis of the system. This approach can be viewed as the development of ideas put forward by Benjamin [6, 7] for the case of nonlinear dynamics in three dimensions and with cross-sectional dependence.

Discrete Setting For simplicity, we consider a spatial discretization with equal space steps, $s_{i+1} - s_i = \Delta s$. The linear and angular deformation gradients are discretized by the elements

$$\lambda_i := \Lambda_i^{-1}\Lambda_{i+1} \in SO(3) \quad \text{and} \quad \boldsymbol{\kappa}_i = \Lambda_i^{-1}(\boldsymbol{r}_{i+1} - \boldsymbol{r}_i) \in \mathbb{R}^3, \tag{6.55}$$

which is a standard discretization on Lie groups, see [8, 46, 48], adapted here to the spatial, rather than temporal, discretization, and for the special Euclidean group $SE(3)$. The discrete angular and linear velocities are given as before by

$$\omega_i := \Lambda_i^{-1}\dot{\Lambda}_i \in \mathfrak{so}(3) \quad \text{and} \quad \gamma_i := \Lambda_i^{-1}\dot{\boldsymbol{r}}_i \in \mathbb{R}^3. \tag{6.56}$$

We also discretize the radius of the cross-section as R_i. This extends the method of [23] to a cross-section with a variable radius having its own dynamics, rather than being only a prescribed function of deformations.

As shown in [23], the main mathematical difficulty lies in the appropriate discretization of the fluid velocity $u_i \simeq u(t, s_i)$. It turns out that the key to the solution lies in the discretization of the *inverse* of the Lagrangian mapping $\varphi(t, s)$, namely, the back-to-labels map $\psi = \varphi^{-1} : [0, L] \to \mathbb{R}$ recalled in Sect. 6.3.1. We discretize $\psi(t, s)$ by its values at the points s_i by introducing the vector $\overline{\psi} = (\psi_1, \psi_2, \ldots, \psi_N)$, with ψ_i being functions of time t.

Consider a discretization of the spatial derivative $\partial_s\psi(t, s_i)$ given by $D_i\overline{\psi}(t) := \sum_{k\in K} a_k\psi_{i+k}(t)$, where K is a finite set of integers in a neighborhood of $m = 0$ and D_i is the linear operator of differentiation acting on the vector $\overline{\psi} = (\psi_1, \ldots \psi_n)$. In the more general case of unequal space steps, the discretization of the derivative may depend explicitly on the index i and is described by the formula $D_i\overline{\psi}(t) := \sum_{k\in K} A_{ik}\psi_{i+k}(t)$. We shall only consider, for simplicity, the case of uniform space steps. In order to approximate the fluid velocity $u(t, s) = (\partial_t\varphi \circ \varphi^{-1})(t, s)$, we rewrite this relation in terms of the back-to-labels map ψ as

$$u(t, s) = (\partial_t\varphi \circ \psi)(t, s) = -\frac{\partial_t\psi(t, s)}{\partial_s\psi(t, s)}\,. \tag{6.57}$$

This relation can be discretized as

$$u_i = -\dot{\psi}_i / D_i\overline{\psi}. \tag{6.58}$$

Let us now show how to discretize the conservation law

$$\begin{aligned} \xi(t, s) &= \xi_0\big(\psi(t, s)\big)\partial_s\psi(t, s) \quad \text{(compressible fluid)},\\ Q(\boldsymbol{\Omega}, \boldsymbol{\Gamma}, R) &= Q_0\big(\psi(t, s)\big)\partial_s\psi(t, s) \quad \text{(incompressible fluid)}. \end{aligned} \tag{6.59}$$

We shall focus on the incompressible fluid case, with the compressible fluid case computed analogously by discretizing $\xi(t, s)$. We assume that at rest, the cross-section of the tube is constant and the tube is straight, which gives $Q_0 = const$ in (6.59). This assumption is not essential for the derivation, but it does simplify the final expressions, otherwise the resulting equations explicitly contain ψ_i. For a constant Q_0, the conservation law is discretized as $Q_0 D_i\overline{\psi} = Q_i(\lambda_i, \boldsymbol{\kappa}_i, R_i) := Q_i$. The appropriate discretization of the fluid conservation law, i.e. the last equation in (6.42), is found by differentiating this discrete conservation with respect to time and using the definition of Eulerian velocity $u_i = -\dot{\psi}_i / D_i\overline{\psi}$, leading to

$$\dot{Q}_i + D_i\big(\overline{uQ}\big) = 0\,. \tag{6.60}$$

Discrete Variational Principle The Lagrangian density is spatially discretized as

$$\int_{s_i}^{s_{i+1}} \mathcal{L}_0\big(\boldsymbol{\omega}, \boldsymbol{\gamma}, \boldsymbol{\Omega}, \boldsymbol{\Gamma}, u, R, \dot{R}, R', R''\big)\mathrm{d}s \simeq \ell_d\big(\boldsymbol{\omega}_i, \boldsymbol{\gamma}_i, \lambda_i, \boldsymbol{\kappa}_i, u_i, R_i, \dot{R}_i, D_i\overline{R}, D_i^2\overline{R}\big)\,, \tag{6.61}$$

where on the right-hand side, we have used the notation $D_i\overline{R}$ and $D_i^2\overline{R}$ for the first and second discrete derivatives of R. The expression (6.61) involves the value of R at point s_i, and the discrete derivatives computed from the values $\overline{R} = (R_1, \dots R_N)$.

The discrete equations are obtained by a spatial discretization of the variational principles developed in Sect. 6.3. In a similar way with the continuous case, we have to start with the standard Hamilton principle with holonomic constraints in the material (or Lagrangian) description, i.e., the discrete analogue of (6.40). This principle is expressed in terms of the variables Λ_i, $\dot{\Lambda}_i$, $\boldsymbol{r}_i$, $\dot{\boldsymbol{r}}_i$, ψ_i, $\dot{\psi}_i$, R_i, $\dot{R}_i$, with free variations $\delta\Lambda_i$, $\delta\boldsymbol{r}_i$, $\delta\psi_i$, δR_i, vanishing at $t = 0, T$.

From this, we deduce the variational principle in terms of the variables appearing in (6.61) by computing the constrained variations $\delta\boldsymbol{\omega}_i$, $\delta\boldsymbol{\gamma}_i$, $\delta\lambda_i$, $\delta\boldsymbol{\kappa}_i$, δu_i induced by the free variations $\delta\Lambda_i$, $\delta\boldsymbol{r}_i$, $\delta\psi_i$. From the definitions (6.55), (6.56), and (6.58), we compute

$$\begin{aligned}
&\delta\boldsymbol{\omega}_i = \dot{\boldsymbol{\Sigma}}_i + \boldsymbol{\omega}_i \times \boldsymbol{\Sigma}_i\,, && \delta\boldsymbol{\gamma}_i = \dot{\boldsymbol{\eta}}_i + \boldsymbol{\gamma}_i \times \boldsymbol{\Sigma}_i + \boldsymbol{\omega}_i \times \boldsymbol{\eta}_i\,,\\
&\delta\lambda_i = -\Sigma_i\lambda_i + \lambda_i\Sigma_{i+1}\,, && \delta\boldsymbol{\kappa}_i = -\boldsymbol{\Sigma}_i \times \boldsymbol{\kappa}_i + \lambda_i\boldsymbol{\eta}_{i+1} - \boldsymbol{\eta}_i\,,\\
&\delta u_i = -\frac{\delta\dot{\psi}_i}{D_i\overline{\psi}} + \frac{\dot{\psi}_i}{(D_i\overline{\psi})^2}\sum_{k\in K} a_k\delta\psi_{i+k} = -\frac{1}{D_i\overline{\psi}}\big(\delta\dot{\psi}_i + u_i D_i\overline{\delta\psi}\big)\,,
\end{aligned} \tag{6.62}$$

where $\Sigma_i := \Lambda_i^{-1}\delta\Lambda_i \in \mathfrak{so}(3)$ and $\boldsymbol{\eta}_i := \Lambda_i^{-1}\delta\boldsymbol{r}_i \in \mathbb{R}^3$. Therefore $\Sigma_i(t) \in \mathfrak{so}(3)$ and $\boldsymbol{\eta}_i(t) \in \mathbb{R}^3$ are arbitrary curves vanishing at $t = 0, T$. As before $\omega_i = \widehat{\boldsymbol{\omega}}_i$ and $\Sigma_i = \widehat{\boldsymbol{\Sigma}}_i$. From the definition of the variables λ_i and $\boldsymbol{\kappa}_i$ we get the discrete compatibility equations

$$\dot{\lambda}_i = -\omega_i\lambda_i + \lambda_i\omega_{i+1}, \quad \dot{\boldsymbol{\kappa}}_i = -\boldsymbol{\omega}_i \times \boldsymbol{\kappa}_i + \lambda_i\boldsymbol{\gamma}_{i+1} - \boldsymbol{\gamma}_i. \tag{6.63}$$

These are the discretization of the $\boldsymbol{\Omega}$- and $\boldsymbol{\Gamma}$-equations in (6.42).

The next result is the discrete analogue of (6.41) and (6.42). For simplicity, we do not consider the boundary effects and assume that all variations vanish at the boundary. They can be included in a similar way with the continuous case.

Theorem 6.2 *The critical action principle for the spatially discretized flexible collapsible tube conveying incompressible fluid reads*

$$\begin{aligned}\delta\int_0^T \sum_i\Big[&\ell_d\big(\boldsymbol{\omega}_i, \boldsymbol{\gamma}_i, \lambda_i, \boldsymbol{\kappa}_i, u_i, R_i, \dot{R}_i, D_i R, D_i^2 R\big) - p_{\rm ext}\, Q_i(\lambda_i, \boldsymbol{\kappa}_i, R_i)\\ &+ \mu_i\left(Q_i(\lambda_i, \boldsymbol{\kappa}_i, R_i) - Q_0 D_i \overline{\psi}\right)\Big]\mathrm{d}t = 0\,,\end{aligned} \tag{6.64}$$

with respect to the variations (6.62). *It yields the equations of motion*

$$\begin{cases} \dfrac{D}{Dt}\dfrac{\partial \ell_d}{\partial \boldsymbol{\omega}_i} + \boldsymbol{\gamma}_i \times \dfrac{\partial \ell_d}{\partial \boldsymbol{\gamma}_i} + \mathbf{Z}_i + \boldsymbol{\kappa}_i \times \left(\dfrac{\partial \ell_d}{\partial \boldsymbol{\kappa}_i} + (\mu_i - p_{\rm ext})\dfrac{\partial Q_i}{\partial \boldsymbol{\kappa}_i}\right) = 0 \\ \dfrac{D}{Dt}\dfrac{\partial \ell_d}{\partial \boldsymbol{\gamma}_i} + \dfrac{\partial \ell_d}{\partial \boldsymbol{\kappa}_i} + (\mu_i - p_{\rm ext})\dfrac{\partial Q_i}{\partial \boldsymbol{\kappa}_i} - \lambda_{i-1}^{\mathsf{T}}\left(\dfrac{\partial \ell_d}{\partial \boldsymbol{\kappa}_{i-1}} + (\mu_{i-1} - p_{\rm ext})\dfrac{\partial Q_i}{\partial \boldsymbol{\kappa}_{i-1}}\right) = 0 \\ \dfrac{d}{dt}m_i + D_i^+\big(\overline{um} - \overline{\mu}\big) = 0\,, \quad m_i := \dfrac{1}{Q_i}\dfrac{\partial \ell_d}{\partial u_i} \\ \dfrac{d}{dt}\dfrac{\partial \ell_d}{\partial \dot{R}_i} - D_i^{2,+}\dfrac{\partial \ell_d}{\partial D_i^2 R} + D_i^+\dfrac{\partial \ell_d}{\partial D_i R} - \dfrac{\partial \ell}{\partial R_i} - (\mu_i - p_{\rm ext})\dfrac{\partial Q_i}{\partial R_i} = 0\,, \end{cases} \tag{6.65}$$

together with conservation law (6.60).

In (6.65) we have defined for shortness the following quantities

$$\begin{aligned} &\mathbf{Z}_i := \left[\left(\frac{\partial \ell_d}{\partial \lambda_i} + (\mu_i - p_{\rm ext})\frac{\partial Q}{\partial \lambda_i}\right)\lambda_i^{\mathsf{T}} - \lambda_{i-1}^{\mathsf{T}}\left(\frac{\partial \ell_d}{\partial \lambda_{i-1}} + (\mu_{i-1} - p_{\rm ext})\frac{\partial Q}{\partial \lambda_{i-1}}\right)\right]^{\vee} \\ &D_i^+\overline{X} := -\sum_{k\in K} a_k X_{i-k}\,, \qquad \sum_i (D_i^{2+}X)Y_i = \sum_i X_i D_i^2 Y \\ &(m^{\vee})_c := -\frac{1}{2}\sum_{ab}\epsilon_{abc} m_{ab}\,. \end{aligned} \tag{6.66}$$

The term D_i^+ is the discrete analogue of the space derivative obtained using the integration by parts; D_i^{2+} is the discrete analogue of the second space derivative. Equation (6.65) are the discrete equivalents of conservation laws for the angular, linear, fluid and wall momenta in (6.42).

6.4.2 Variational Integrator in Space and Time

Let us now turn our attention to the derivation of a variational integrator for this problem. The important novel step here is to provide a discretization of the fluid part, as was done in [23]. The rest of the analysis can be obtained according to the methods of spacetime multisymplectic variational discretization [45]. In particular, the recent work [14] derived a variational discretization based on the multisymplectic nature of the exact geometric rods, which we shall use as the foundation of our approach.

Discrete Setting Let us select a rectangular lattice of points in space and time (s_i, t_j) and define the discretization of the back-to-labels map $\psi(t,s)$ as $\overline{\psi}$ to have discrete values $\psi_{i,j}$ at the spacetime points (s_i, t_j). Assume that the spatial and time derivatives are given by the discrete operators

$$D^s_{i,j}\overline{\psi} := \sum_{k\in K} a_k \psi_{i+k,j}\,, \quad D^t_{i,j}\overline{\psi} := \sum_{m\in M} b_m \psi_{i,j+m}\,, \tag{6.67}$$

where M and K are discrete finite sets of indices in a neighborhood of 0. Using (6.57), we obtain the following relation between the velocity and back-to-labels map

$$u_{i,j} = -\frac{D^t_{i,j}\overline{\psi}}{D^s_{i,j}\overline{\psi}}\,. \tag{6.68}$$

The discretization of the conservation law $Q_0 \partial_s \psi = Q$, where we again assume $Q_0 = const$ for simplicity, is then given as $Q_0 D^s_{i,j}\overline{\psi} = Q_{i,j}$. Applying D^t_{ij} to both sides of this conservation law, noticing that the operators D^t_{ij} and D^s_{ij} defined by (6.67) commute on a rectangular lattice in (s_i, t_j), and using (6.68) to eliminate $D^t_{i,j}\overline{\psi}$ from the equations, we obtain the discrete conservation law

$$D^t_{i,j}\overline{Q} + D^s_{i,j}\left(\overline{uQ}\right) = 0\,. \tag{6.69}$$

We assume that the discretization of the spatial and temporal derivatives of R are also given by (6.67).

Let us now define the spacetime discrete versions of the continuous deformations $(\boldsymbol{\Omega}, \boldsymbol{\Gamma})$ and velocities $(\boldsymbol{\omega}, \boldsymbol{\gamma})$. If $(\Lambda_{i,j}, \boldsymbol{r}_{i,j}) \in SE(3)$ are the orientation and position at (t_i, s_j), we define

$$\begin{aligned} \lambda_{i,j} &:= \Lambda^{-1}_{i,j}\Lambda_{i+1,j} \in SO(3) \qquad & \boldsymbol{\kappa}_{i,j} &:= \Lambda^{-1}_{i,j}\left(\boldsymbol{r}_{i+1,j} - \boldsymbol{r}_{i,j}\right) \in \mathbb{R}^3\,, \\ q_{i,j} &:= \Lambda^{-1}_{i,j}\Lambda_{i,j+1} \in SO(3) \qquad & \boldsymbol{\gamma}_{i,j} &:= \Lambda^{-1}_{i,j}\left(\boldsymbol{r}_{i,j+1} - \boldsymbol{r}_{i,j}\right) \in \mathbb{R}^3\,. \end{aligned} \tag{6.70}$$

Discrete Variational Principle We assume that the Lagrangian density is discretized in space and time as[2]

[2] Here, $\overline{R}$ denotes the matrix of space-time discretized variables $R_{i,j}$. Note that in the previous section describing the spatial discretization and continuous time, $\overline{R}$ denoted a vector of functions $R_1(t), \ldots R_N(t)$. We hope no confusion arises from this clash of notation.

$$\int_{t_j}^{t_{j+1}}\int_{s_i}^{s_{i+1}} \mathcal{L}_0\big(\boldsymbol{\omega}, \boldsymbol{\gamma}, \boldsymbol{\Omega}, \boldsymbol{\Gamma}, u, R, \dot{R}, R', R''\big)\, \mathrm{d}s\, \mathrm{d}t$$
$$\simeq \mathcal{L}_d\big(\lambda_{i,j}, \boldsymbol{\kappa}_{i,j}, q_{i,j}, \boldsymbol{\gamma}_{i,j}, u_{i,j}, R_{ij}, D^t_{ij}\overline{R}, D^s_{ij}\overline{R}, D^{s,2}_{ij}\overline{R}\big).$$

The discrete equations are obtained by a spacetime discretization of the variational principles developed in Sect. 6.3. In a similar way with the continuous case in Sect. 6.3.4 and the spatially discretized case in Sect. 6.4.1, we have to start with the standard Hamilton principle with holonomic constraints in the material (or Lagrangian) description, i.e., the spacetime discrete analogue of (6.40). This principle is expressed in terms of the variables $\Lambda_{i,j}$, $\boldsymbol{r}_{i,j}$, $\psi_{i,j}$, $R_{i,j}$, with free variations $\delta\Lambda_{i,j}$, $\delta\boldsymbol{r}_{i,j}$, $\delta\psi_{i,j}$, $\delta R_{i,j}$, vanishing at temporal boundary.

From this, we deduce the variational principle in terms of the variables appearing in (6.61) by computing the constrained variations $\delta\lambda_{i,j}$, $\delta\boldsymbol{\kappa}_{i,j}$, $\delta q_{i,j}$, $\delta\boldsymbol{\gamma}_{i,j}$, $\delta u_{i,j}$ induced by the free variations $\delta\Lambda_{i,j}$, $\delta\boldsymbol{r}_{i,j}$, $\delta\psi_{i,j}$. From the definitions (6.68) and (6.70), we compute the variations $\delta\lambda_{i,j}$, $\delta\boldsymbol{\kappa}_{i,j}$, $\delta\boldsymbol{\gamma}_{i,j}$, $\delta u_{i,j}$ similar to (6.62), not shown here for brevity, to prove the following

Theorem 6.3 *The critical action principle for the fully discretized flexible collapsible tube conveying incompressible fluid reads*

$$\delta\Bigg[\sum_{i,j}\mathcal{L}_d\Big(\lambda_{i,j}, \boldsymbol{\kappa}_{i,j}, q_{i,j}, \boldsymbol{\gamma}_{i,j}, u_{i,j}, R_{ij}, D^t_{i,j}R, D^s_{i,j}R, D^{s,2}_{i,j}R\Big)$$
$$\sum_{i,j} - p_{\mathrm{ext}}\, Q_{i,j}\big(\lambda_{i,j}, \boldsymbol{\kappa}_{i,j}, R_{ij}\big) + \mu_{i,j}\,\Big(Q_{i,j}\big(\lambda_{i,j}, \boldsymbol{\kappa}_{i,j}, R_{ij}\big) - Q_0 D^s_{i,j}\overline{\psi}\Big)\Bigg] = 0, \tag{6.71}$$

with respect to the reduced variations. It yields the discrete equations of motion

$$\begin{cases} \mathbf{Y}_{i,j} + \mathbf{Z}_{i,j} + \boldsymbol{\gamma}_{i,j} \times \dfrac{\partial\mathcal{L}_d}{\partial\boldsymbol{\gamma}_{i,j}} + \boldsymbol{\kappa}_{i,j} \times \left(\dfrac{\partial\mathcal{L}_d}{\partial\boldsymbol{\kappa}_{i,j}} + (\mu_{i,j} - p_{\mathrm{ext}})\dfrac{\partial Q_{i,j}}{\partial\boldsymbol{\kappa}_{i,j}}\right) = 0 \\ \dfrac{\partial\mathcal{L}_d}{\partial\boldsymbol{\gamma}_{i,j}} - q^{\mathsf{T}}_{i,j-1}\dfrac{\partial\mathcal{L}_d}{\partial\boldsymbol{\gamma}_{i,j-1}} + \left(\dfrac{\partial\mathcal{L}_d}{\partial\boldsymbol{\kappa}_{i,j}} + (\mu_{i,j} - p_{\mathrm{ext}})\dfrac{\partial Q_{i,j}}{\partial\boldsymbol{\kappa}_{i,j}}\right) \\ \qquad\qquad -\lambda^{\mathsf{T}}_{i-1,j}\left(\dfrac{\partial\mathcal{L}_d}{\partial\boldsymbol{\kappa}_{i-1,j}} + (\mu_{i-1,j} - p_{\mathrm{ext}})\dfrac{\partial Q_{i-1,j}}{\partial\boldsymbol{\kappa}_{i-1,j}}\right) = 0 \\ D^{t,+}_{i,j}\overline{m} + D^{s,+}_{i,j}\left(\overline{um} + \overline{\mu}\right) = 0\,, \qquad m := \dfrac{1}{Q_{ij}}\dfrac{\partial\mathcal{L}_d}{\partial u_{ij}} \\ D^t_{i,j}\dfrac{\partial\mathcal{L}_d}{\partial D^t_{i,j}R} - D^{s,2+}_{i,j}\dfrac{\partial\mathcal{L}_d}{\partial D^{2,s}_{i,j}R} + D^{s,+}_{i,j}\dfrac{\partial\mathcal{L}_d}{\partial D^s_{i,j}R} - \dfrac{\partial\mathcal{L}_d}{\partial R_{ij}} - \big(\mu_{ij} - p_{\mathrm{ext}}\big)\dfrac{\partial Q_{i,j}}{\partial R_{ij}} = 0\,. \end{cases} \tag{6.72}$$

In (6.72), we have defined the following quantities

$$
\begin{aligned}
\mathbf{Y}_{i,j} &= \left[\frac{\partial \mathcal{L}_d}{\partial q_{i,j}} q_{i,j}^{\mathsf{T}} - q_{i,j-1}^{\mathsf{T}} \frac{\partial \mathcal{L}_d}{\partial q_{i,j-1}} \right]^{\vee} , \\
\mathbf{Z}_{i,j} &= \left[\left(\frac{\partial \mathcal{L}_d}{\partial \lambda_{i,j}} + (\mu_{i,j} - p_{\text{ext}}) \frac{\partial Q}{\partial \lambda_{i,j}} \right) \lambda_{i,j}^{\mathsf{T}} - \lambda_{i-1,j}^{\mathsf{T}} \right. \\
&\qquad \left. \times \left(\frac{\partial \mathcal{L}_d}{\partial \lambda_{i-1,j}} + (\mu_{i-1,j} - p_{\text{ext}}) \frac{\partial Q}{\partial \lambda_{i-1,j}} \right) \right]^{\vee} \qquad (6.73) \\
D_{i,j}^{s,+} \overline{X} &:= - \sum_{k \in K} a_k X_{i-k,j} , \qquad D_{i,j}^{t,+} \overline{X} := - \sum_{m \in M} b_m X_{i,j-m} , \\
\sum_j \left(D_{i,j}^{2,s+} X \right) Y_{ij} &= \sum_i X_{ij} \left(D_{i,j}^{2,s} Y \right) .
\end{aligned}
$$

The terms $D_{i,j}^{s,+}$ and $D_{i,j}^{t,+}$ are the discrete analogues of the derivatives obtained by integration by parts in the s- and t- directions, respectively; $D_{i,j}^{2,s+}$ is the discrete analogue of the dual to second derivative obtained by two integration by parts.

The system (6.72) is solved by taking into account the relations (6.70) which play the role of the compatibility conditions (6.63) in the spacetime discretized case. Note that the third equation in (6.72) is the discrete version of the balance of fluid momentum as expressed in (6.44). For simplicity of the exposition, we have assumed that all the boundary terms arising from discrete integration by parts vanish at the spatial extremities in the variational principles.

6.5 Further Developments

In this chapter, we presented a variational approach for a particular example of fluid-structure interaction, namely, the elastic tube conveying fluid. By discretizing this variational approach, we derived a structure preserving numerical scheme. Other applications of the variational methods are possible, such as the linear stability analysis of initially curved pipes which can be treated very efficiently in this framework [26], whereas it is very difficult via the traditional approach. Thus, variational methods are useful for both the analytical understanding of the mechanics of complex problems, and the systematic derivation of structure preserving numerical methods for these cases. Further interesting directions of studies in the field are, to name a few examples, the variational methods and the corresponding variational integrators for moving porous media (poromechanics) [12] and waves and instabilities of fluid under elastic sheet (hydroelastic waves) [57].

Acknowledgements We are grateful to Profs. D. Georgievskii, D. D. Holm, T. S. Ratiu, P. Vorobieff and many others for their suggestions and comments, as well as experimental realizability of the models. The work of FGB was partially supported by the ANR project GEOMFLUID, ANR-

14-CE23-0002-01. The work of VP was partially supported by the NSERC Discovery grant and the University of Alberta. This work has also been made possible by the awarding of a James M Flaherty Visiting Professorship from the Ireland Canada University Foundation, with the assistance of the Government of Canada/avec l'appui du gouvernement du Canada.

References

1. Akulenko, L.D., Georgievskii, D.V., Nesterov, S.V.: Transverse vibration spectrum of a part of a moving rod under a longitudinal load. Mech. Solids **50**, 227–231 (2015)
2. Akulenko, L.D., Georgievskii, D.V., Nesterov, S.V.: Spectrum of transverse vibrations of a pipeline element under longitudinal load. Dokl. Akad. Nauk **467**, 36–39 (2016)
3. Akulenko, L.D., Ivanov, M.I., Korovina, L.I., Nesterov, S.V.: Basic properties of natural vibrations of an extended segment of a pipeline. Mech. Solids **48**, 458–472 (2013)
4. Ashley, H., Haviland, G.: Bending vibrations of a pipe line containing flowing fluid. J. Appl. Mech. **17**, 229–232 (1950)
5. Beauregard, M.A., Goriely, A., Tabor, M.: The nonlinear dynamics of elastic tubes conveying a fluid. Int. J. Solids Struct. **47**, 161–168 (2010)
6. Benjamin, B.T.: Dynamics of a system of articulated pipes conveying fluid I. Theory. Proc. R. Soc. A **261**, 457–486 (1961)
7. Benjamin, B.T.: Dynamics of a system of articulated pipes conveying fluid II. Experiments. Proc. R. Soc. A **261**, 487–499 (1961)
8. Bobenko, A.I., Suris, Y.B.: Discrete Lagrangian reduction, discrete Euler-Poincaré equations, and semidirect products. Lett. Math. Phys. **49**, 79–93 (1999)
9. Cendra, H., Marsden, J.E., Ratiu, T.S.: Lagrangian Reduction by Stages, vol. 152. Memoirs American Mathematical Society, Providence (2001)
10. Cosserat, E., Cosserat, F.: Théorie des corps déformables. Hermann, Paris (1909)
11. Cros, A., Romero, J.A.R., Flores, F.C.: Sky Dancer: A complex fluid-structure interaction. Experimental and Theoretical Advances in Fluid Dynamics: Environmental Science and Engineering, pp. 15–24. Springer, Berlin (2012)
12. dell'Isola, F., Madeo, A., Seppecher, P.: Boundary conditions at fluid-permeable interfaces in porous media: a variational approach. Int. J. Solids Struct. **46**(17), 3150–3164 (2009)
13. Demoures, F., Gay-Balmaz, F., Kobilarov, M., Ratiu, T.S.: Multisymplectic lie group variational integrator for a geometrically exact beam in R^3. Commun. Nonlinear Sci. Numer. Simul. **19**, 3492–3512 (2014)
14. Demoures, F., Gay-Balmaz, F., Ratiu, T.S.: Multisymplectic variational integrators and space/time symplecticity. Anal. Appl. **14**(3), 341–391 (2016)
15. Doaré, O., de Langre, E.: The flow-induced instability of long hanging pipes. Eur. J. Mech. A Solids **21**, 857–867 (2002)
16. Donovan, G.: Systems-level airway models of bronchoconstriction. Wiley Interdiscip. Rev. Syst. Biol. Med. **8**, 459 (2016)
17. Elad, D., Kamm, R.D., Shapiro, A.H.: Steady compressible flow in collapsible tubes: application to forced expiration. J. Appl. Physiol. **203**, 401–418 (1989)
18. Ellis, D., Holm, D.D., Gay-Balmaz, F., Putkaradze, V., Ratiu, T.: Symmetry reduced dynamics of charged molecular strands. Arch. Ration. Mech. Anal. **197**, 811–902 (2010)
19. Flores, F.C., Cros, A.: Transition to chaos of a vertical collapsible tube conveying air flow. J. Phys. Conf. Ser. **166**, 012017 (2009)
20. Formaggia, L., Lamponi, D., Quarteroni, A.: One-dimensional models for blood flow in arteries. J. Eng. Math. **47**, 251–276 (2003)
21. Gay-Balmaz, F., Putkaradze, V.: Exact geometric theory for flexible, fluid-conducting tubes. C.R. Acad. Sci. Paris, Sé. Méc. **342**, 79–84 (2014)

22. Gay-Balmaz, F., Putkaradze, V.: On flexible tubes conducting fluid: geometric nonlinear theory, stability and dynamics. J. Nonlinear. Sci. **25**, 889–936 (2015)
23. Gay-Balmaz, F., Putkaradze, V.: Variational discretizations for the dynamics of flexible tubes conveying fluid. C. R. Méc. **344**, 769–775 (2016)
24. Gay-Balmaz, F., Putkaradze, V.: Geometric theory of flexible and expandable tubes conveying fluid: Equations, solutions and shock waves. J. Nonlinear Sci. **29**(2), 377–414 (2019)
25. Gay-Balmaz, F., Yoshimura, H.: A lagrangian variational formulation for nonequilibrium thermodynamics. Part II: continuous systems. J. Geom. Phys. **111**, 194–212 (2017)
26. Gay-Balmaz, F., Georgievskii, D., Putkaradze, V.: Stability of helical tubes conveying fluid. J. Fluids Struct. **78**, 146–174 (2018)
27. Gay-Balmaz, F., Marsden, J.E., Ratiu, T.S.: Reduced variational formulations in free boundary continuum mechanics. J. Nonlinear Sci. **22**(4), 463–497 (2012)
28. Ghayesh, M., Païdoussis, M.P., Amabili, M.: Nonlinear dynamics of cantilevered extensible pipes conveying fluid. J. Sound Vib. **332**, 6405–6418 (2013)
29. Gregory, R.W., Païdoussis, M.P.: Unstable oscillation of tubular cantilevers conveying fluid I. Theory. Proc. R. Soc. A **293**, 512–527 (1966)
30. Gregory, R.W., Païdoussis, M.P.: Unstable oscillation of tubular cantilevers conveying fluid II. Experiments. Proc. R. Soc. A **293**, 528–542 (1966)
31. Grotberg, J.B., Jensen, O.E.: Biofluid mechanics in flexible tubes. Ann. Rev. Fluid Mech. **36**, 121–147 (2004)
32. Heil, M., Hazel, A.L.: Fluid-structure interaction in internal physiological flows. Ann. Rev. Fluid Mech. **43**, 141–62 (2011)
33. Holm, D.D., Putkaradze, V.: Nonlocal orientation-dependent dynamics of charged strands and ribbons. C. R. Acad. Sci. Paris Sér. I: Math. **347**, 1093–1098 (2009)
34. Juel, A., Heap, A.: The reopening of a collapsed fluid-filled elastic tube. J. Fluid Mech. **572**, 287–310 (2007)
35. Kounanis, K., Mathioulakis, D.S.: Experimental flow study within a self oscillating collapsible tube. J. Fluids Struct. **13**, 61–73 (1999)
36. Kounanis, K., Mathioulakis, D.S.: Experimental flow study within a self-oscillating collapsible tube. J. Fluids Struct. **13**, 61–73 (1999)
37. Kuronuma, S., Sato, M.: Stability and bifurcations of tube conveying flow. J. Phys. Soc. Jpn. **72**, 3106–3112 (2003)
38. Landau, L.D., Lifshitz, E.M.: Mechanics, Volume 1 of A Course of Theoretical Physics. Pergamon Press, Oxford (1969)
39. Leyendecker, S., Marsden, J.E., Ortiz, M.: Variational integrators for constrained dynamical systems. Z. Angew. Math. Mech. **88**, 677–708 (2008)
40. Marchandise, E., Flaud, P.: Accurate modelling of unsteady flows in collapsible tubes. Comput. Methods Biomech. Biomed. Eng. **13**, 279 (2010)
41. Marsden, J.E., Scheurle, J.: Lagrangian reduction and the double spherical pendulum. ZAMP **44**, 17–43 (1993)
42. Marsden, J.E., Scheurle, J.: The reduced Euler–Lagrange equations. Fields Inst. Commun. **1**, 139–164 (1993)
43. Marsden, J., West, M.: Discrete mechanics and variational integrators. Acta Numer. **10**, 357–514 (2001)
44. Marsden, J.E., Ratiu, T.S.: Introduction to Mechanics and Symmetry: A Basic Exposition of Classical Mechanical Systems, 2nd edn. Texts in Applied Mathematics. Springer, Berlin (2002)
45. Marsden, J.E., Patrick, G.W., Shkoller, S.: Multisymplectic geometry, variational integrators and nonlinear PDEs. Commun. Math. Phys. **199**, 351–395 (1998)
46. Marsden, J., Pekarsky, S., Shkoller, S.: Symmetry reduction of discrete Lagrangian mechanics on Lie groups. J. Geom. Phys. **36**, 140–151 (1999)
47. Modarres-Sadeghi, Y., Païdoussis, M.P.: Nonlinear dynamics of extensible fluid-conveying pipes supported at both ends. J. Fluids Struct. **25**, 535–543 (2009)
48. Moser, J., Veselov, A.: Discrete versions of some classical integrable systems and factorization of matrix polynomials. Commun. Math. Phys. **139**, 217–243 (1991)

49. Païdoussis, M.P.: Dynamics of tubular cantilevers conveying fluid. Int. J. Mech. Eng. Sci. **12**, 85–103 (1970)
50. Païdoussis, M.P.: Fluid-Structure interactions. Slender structures and axial flow, vol. 1. Academic Press, London (1998)
51. Païdoussis, M.P.: Fluid-Structure interactions. Slender structures and axial flow, vol. 2. Academic Press, London (2004)
52. Païdoussis, M.P., Issid, N.T.: Dynamic stability of pipes conveying fluid. J. Sound Vib. **33**, **267**–294 (1974)
53. Païdoussis, M.P., Li, G.X.: Pipes conveying fluid: a model dynamical problem. J. Fluids Struct. **7**, 137–204 (1993)
54. Pedley, T.: The Fluid Mechanics of Large Blood Vessels. Cambridge University Press, Cambridge (1980)
55. Pedley, T.J.: Longitudinal tension variation in collapsible channels: a new mechanism for the breakdown of steady flow. J. Biomech. Eng. **114**, 60–67 (1992)
56. Pedley, T.J., Luo, X.Y.: The effects of wall inertia on flow in a two-dimensional collapsible channel. J. Fluid Mech. **363**, 253–280 (1998)
57. Plotnikov, P.I., Toland, J.F.: Modelling nonlinear hydroelastic waves. Philos. Trans. R. Soc. A Math. Phys. Eng. Sci. **369**(1947), 2942–2956 (2011)
58. Quarteroni, A., Formaggia, L.: Mathematical modelling and numerical simulation of the cardiovascular system. In: Handbook of Numerical Analysis Series. Modelling of Living Systems. Elsevier, Amsterdam (2002)
59. Quarteroni, A., Tuveri, M., Veneziani, A.: Computational vascular fluid dynamics: problems, models and methods. Comput Visual Sci. **2**, 163–197 (2000)
60. Rivero-Rodriguez, J., Perez-Saborid, M.: Numerical investigation of the influence of gravity on flutter of cantilevered pipes conveying fluid. J. Fluids Struct. **55**, 106–121 (2015)
61. Secomb, T.W.: Hemodynamics. Comp. Physiol. **6**, 975–1003 (2016)
62. Semler, C., Li, G.X., Païdoussis, M.P.: The non-linear equations of motion of pipes conveying fluid. J. Sound Vib. **169**, 577–599 (1994)
63. Shima, S., Mizuguchi, T.: Dynamics of a tube conveying fluid (2001). arXiv preprint: nlin.CD/0105038
64. Simo, J.C.: A finite strain beam formulation. the three-dimensional dynamic problem. Part I. Comput. Methods Appl. Mech. Eng. **49**, 79–116 (1985)
65. Simó, J.C., Marsden, J.E., Krishnaprasad, P.S.: The Hamiltonian structure of nonlinear elasticity: the material and convective representations of solids, rods, and plates. Arch. Ration. Mech. Anal. **104**, 125–183 (1988)
66. Stewart, P.S., Waters, S.L., Jensen, O.E.: Local and global instabilities of flow in a flexible-walled channel. Eur. J. Mech. B Fluids **28**, 541–557 (2009)
67. Tang, D., Yang, Y., Yang, C., Ku, D.N.: A nonlinear axisymmetric model with fluid-wall interactions for steady viscous flow in stenotic elastic tubes. Trans. ASME **121**, 494–501 (2009)
68. Veselov, A.P.: Integrable discrete-time systems and difference operators (Russian). Funktsional. Anal. i Prilozhen. **22**(2), 1–26 (1988)
69. Veselov, A.P.: Integrable lagrangian correspondences and the factorization of matrix polynomials (Russian). Funkts. Anal. Prilozhen **25**(2), 38–49 (1991)
70. Wendlandt, J.M., Marsden, J.: Mechanical integrators derived from a discrete variational principle. Phys. D **106**, 223–246 (1997)
71. Whitham, G.B.: Linear and Nonlinear Waves. Wiley, London (1974)
72. Zhermolenko, V.N.: Application of the method of extremal deviations to the study of forced parametric bend oscillations of a pipeline (in Russian). Autom. Telemech. **9**, 10–32 (2008)

Chapter 7
Convex Lifting-Type Methods for Curvature Regularization

Ulrich Böttcher and Benedikt Wirth

Contents

Abstract Human visual perception is able to complete contours of objects even if they are disrupted or occluded in images. A possible mathematical imitation of this property is to represent object contours in the higher-dimensional space of positions and directions, the so-called roto-translation space, and to use this representation to promote contours with small curvature and separate overlapping objects. Interpreting image level lines as contours then leads to curvature-penalizing regularization functionals for image processing, which become convex through the

U. Böttcher (✉) · B. Wirth
Westfälische Wilhelms-Universität Münster, Applied Mathematics Münster, Münster, Germany
e-mail: ulrich.boettcher@wwu.de; benedikt.wirth@wwu.de

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_7

additional dimension. We present the basic concept, some of its properties, as well as numerical discretization approaches for those functionals.

7.1 Introduction

In the early twentieth century, psychologists such as Wertheimer, Köhler, and Koffka investigated human perception experimentally and theoretically, which resulted in the psychological branch of Gestalt theory. It describes how humans connect different visual stimuli to a meaningful picture, typically via a number of basic, experimentally found rules (e.g., the collection in [45]). A particular instance, the "law of continuity", states that humans tend to connect image elements that are arranged along straight or slightly bent lines, which allows them to complete partially occluded contours (cf. Fig. 7.1). In the 1950s, Hubel and Wiesel identified a first physiological counterpart of these purely empirical rules [26], which won them a Nobel Prize in 1981. They showed that certain neurons in cats are active whenever the cats see lines of a certain orientation. Thus, orientation information is explicitly represented in the visual cortex. This information could be aggregated to curvature and ultimately be used for a promotion of straight lines or contours with little curvature. Correspondingly, curvature regularization has been of interest to mathematical image processing. Note that here we primarily speak about extrinsic curvature.

7.1.1 Curvature-Dependent Functionals and Regularization

Promoting lines or contours of low curvature can be interpreted as an energy minimization principle: Human perception preferentially reconstructs those (smooth) lines $\gamma \subset \mathbb{R}^2$ in an image that have small energy

$$R(\gamma) = \int_\gamma \alpha + \beta g(\kappa_\gamma)\, \mathrm{d}\mathcal{H}^1 \,, \tag{7.1}$$

where κ_γ is the curvature of γ, $\alpha \geq 0$ and $\beta \geq 0$ are weights given to the curve length and to the curvature term, $g : \mathbb{R} \to [0, \infty)$, and $\mathcal{H}^m$ denotes the

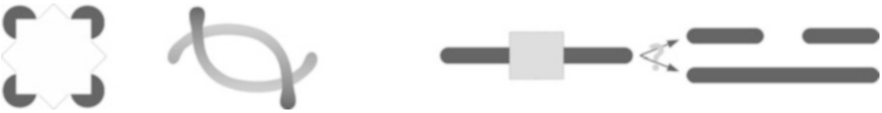


Fig. 7.1 By the "law of continuity" of Gestalt theory, partially occluded contours can be completed by human vision if they do not bend too strongly, as is illustrated by Kanizsa type illusions [28]

m-dimensional Hausdorff measure. The choice $g(\kappa) = \kappa^2$ yields Euler's elastica functional which describes the bending energy of an elastically deformed rod [25]. Nitzberg et al. proposed its use in image processing for the completion of missing lines in 1993 [34]. Until then first-order derivative-based regularization methods prevailed, e.g., total variation [37]. In three-dimensional images, where an object contour γ would be a surface and κ_γ its mean curvature (or more generally the principal curvatures), this energy (with $\alpha = 0$) is known as the Willmore energy [46] and has been used for geometry processing [8, 15, 23, 40].

For binary images, i.e., characteristic functions of sets $E \subset \Omega$ for some domain $\Omega \subset \mathbb{R}^2$, a curvature regularization functional can simply be defined as $R(\partial E)$ if ∂E is sufficiently smooth. The relaxation (i.e., the lower semi-continuous envelope) of this energy with respect to L^1-convergence of the characteristic functions was characterized in [3, 9, 10]. Such a functional can be extended to nonbinary images by considering its convex relaxation (the convex lower semi-continuous envelope or at least approximations thereof, which will be the approach considered here) or by applying the curvature regularization to all individual level lines of the image $u : \Omega \to \mathbb{R}$ [30], in which case the regularization functional becomes

$$R(u) = \int_\Omega |\nabla u| \left(\alpha + \beta g \left(\nabla \cdot \frac{\nabla u}{|\nabla u|} \right) \right) \,\mathrm{d}x \tag{7.2}$$

for $u \in C^2(\Omega)$. Its L^1-relaxation, Euler–Lagrange equation, and application to image inpainting in the setting of functions of bounded variation is considered in [3, 42], a variant giving a more even emphasis to all level lines is proposed and analyzed in [6, 7]. The strong nonlinearity and nonconvexity require efficient optimization algorithms for which various augmented Lagrangian type methods have been proposed [5, 43, 47, 48, 50]. In the next section we will discuss techniques that embed functionals (7.1) or (7.2) into a higher-dimensional space to study their variational properties (such as lower semi-continuity, coercivity, existence of minimizers or convex relaxations) and to perform numerics. Those form the focus of this article.

7.1.2 Convex Relaxation of Curvature Regularization Functionals

Energies like (7.2) which penalize the curvature of image level lines are highly nonconvex so that their use as a regularizer in image processing leads to many local minima (for instance, a partially occluded object could either be reconstructed as one piece or as two separate pieces, Fig. 7.1 right). As a remedy, one can replace any energy with its convex relaxation, the largest convex lower semi-continuous functional below the original energy, thereby removing local minima without changing the global minimizers. The lifting approaches presented in this article all yield approximations to the convex relaxation of (7.2). In this respect it is interesting to note that any functional penalizing *solely* curvature has trivial convex relaxation.

Theorem 7.1 (Convex Hull of Images with Flat Level Sets, [14, Thm. 2.4]) *Let $\Omega \subset \mathbb{R}^d$ be open. Any $u \in C_0^\infty(\Omega)$ can be approximated in $C^k(\overline{\Omega})$ for any $k \geq 0$ by finite convex combinations of smooth images with flat level sets.*

Proof Let S^{d-1} be the unit sphere in $\mathbb{R}^d$ and $\mathcal{S}(\mathbb{R}^d)$ and $\mathcal{S}(S^{d-1} \times \mathbb{R})$ be the space of Schwartz functions on the respective domains. The Radon transform $\mathfrak{R} : \mathcal{S}(\mathbb{R}^d) \to \mathcal{S}(S^{d-1} \times \mathbb{R})$ and its dual $\mathfrak{R}^* : \mathcal{S}(S^{d-1} \times \mathbb{R}) \to \mathcal{S}(\mathbb{R}^d)$ are defined by [33, Sec. II.1]

$$\mathfrak{R}w(\theta, r) = \int_{\theta^\perp} w(r\theta + y)\,\mathrm{d}y\,, \qquad \mathfrak{R}^*g(x) = \int_{S^{d-1}} g(\theta, \theta \cdot x)\,\mathrm{d}\theta\,.$$

Obviously, $g = \mathfrak{R}u \in C_0^\infty(S^{d-1} \times \mathbb{R})$, and by a classical inversion formula for the Radon transform [33, Thm. 2.1], $u = \mathfrak{R}^*(J^{n-1}g)$, in which the operator J^k is defined via $(J^k g)\hat{}(\theta, \xi) = \frac{1}{2}|\frac{\xi}{2\pi}|^k \hat{g}(\theta, \xi)$ with $\hat{}$ denoting the Fourier transform in the second argument. Since $\hat{g}$ is a Schwartz function and thus $(J^{n-1}g)\hat{}$ decays rapidly, $J^{n-1}g \in C^\infty(S^{d-1} \times \mathbb{R})$. Now for $\theta \in S^{d-1}$ define $u_\theta \in C^\infty(\overline{\Omega})$ by $u_\theta(x) = J^{n-1}g(\theta, \theta \cdot x)$, then u_θ has flat level sets with normal θ, and $u = \mathfrak{R}^* J^{n-1} g = \int_{S^{d-1}} u_\theta \,\mathrm{d}\theta$. Approximating the latter integral by a Riemann sum, which can be shown to converge in $C^k(\overline{\Omega})$, yields an approximation of u by finite convex combinations of u_θ, $\theta \in S^{d-1}$. □

Consequently, for any energy which is zero for smooth images with flat level sets (lines in 2D and surfaces in 3D), its convex relaxation is zero in any topology in which $C_0^\infty(\Omega)$ or $C_0^\infty(\tilde{\Omega})$ with $\Omega \subset\subset \tilde{\Omega}$ is dense, in particular in all Sobolev spaces $W^{n,p}(\Omega)$ or Hölder spaces $C^{n,\alpha}(\overline{\Omega})$ with $n \geq 0$, $p \in [1, \infty)$, and $\alpha \in [0, 1)$.

In Sect. 7.2 we will present the mathematical concepts of lifting based curvature regularization for 2D images, followed by corresponding discretization methods in Sect. 7.3 and a lifting variant for 3D images in Sect. 7.4. Throughout, $\Omega \subset \mathbb{R}^d$ is an open Lipschitz domain (on which the images are defined), $S^{d-1} \subset \mathbb{R}^d$ is the $(d-1)$-dimensional sphere, and $\mathrm{Gr}(m, \mathbb{R}^d)$ is the Grassmannian, the space of m-dimensional subspaces of $\mathbb{R}^d$ ($\widetilde{\mathrm{Gr}}(m, \mathbb{R}^d)$ is used for oriented subspaces). The sets of signed and unsigned Radon measures on some Borel space A are denoted by $\mathrm{rca}(A)$ and $\mathrm{rca}_+(A)$, respectively, and the pushforward of a measure $\mu \in \mathrm{rca}(A \times B)$ under the projection of $A \times B$ onto A is written as $\mu(\cdot, B)$. The m-dimensional Hausdorff measure is $\mathcal{H}^m$, and the restriction of a measure μ to a set S is denoted $\mu \llcorner S$. The set of functions of bounded variation on Ω is written as $\mathrm{BV}(\Omega)$ with total variation seminorm $|\cdot|_{\mathrm{TV}}$. Finally, we assume $\alpha > 0$, $\beta \geq 0$, and the function g in (7.1) to be nonnegative.

7.2 Lifting-Type Methods for Curvature Regularization

We first set the discussed methods (which we will term the approaches of hyper-varifolds, of curvature varifolds, of Gauss graph currents, and of jump set calibrations) into context by providing a historical overview over lifting methods, after which we present the methods' details and their relations.

7.2.1 Concepts for Curve- (and Surface-) Lifting

The idea of lifting embedded curves and surfaces into a higher-dimensional space containing also directional information is at least as old as L. C. Young's work on generalized curves and surfaces [49], which in modern mathematics correspond to varifolds as introduced by Almgren [1, 2]. An m-dimensional varifold in $\mathbb{R}^d$ is a Radon measure on $\mathbb{R}^d \times \mathrm{Gr}(m, \mathbb{R}^d)$. A classical curve or surface γ (i.e., $m = 1$ or $m = 2$, respectively) is represented by the varifold μ_γ with

$$\mu_\gamma(\varphi) = \int_\gamma \varphi(x, \tau_\gamma(x))\,\mathrm{d}\mathcal{H}^m(x) \qquad \text{for all } \varphi \in C_0^\infty(\mathbb{R}^d \times \mathrm{Gr}(m, \mathbb{R}^d))\,,$$

where $\tau_\gamma(x)$ is the tangent space to γ in x. Replacing $\mathrm{Gr}(m, \mathbb{R}^d)$ by $\widetilde{\mathrm{Gr}}(m, \mathbb{R}^d)$ yields oriented varifolds that can describe directed curves and oriented surfaces.

Curvature Varifolds Varifolds were used to study compactness and lower semi-continuity of the (surface) area functional. Only much later their weak second fundamental form was defined based on the (weak) first derivative of their tangent or Grassmannian component [27]. Curvature varifolds (those for which this weak second fundamental form exists) enjoy nice compactness properties and were used to study lower semi-continuity and existence of minimizers for integral functionals (7.1) whose integrand is convex in the curvature [27, 29]. We come back to this approach in Sect. 7.2.2.

Hyper-Varifolds In a lower semi-continuous integral functional, the integrand needs to be (poly-)convex in the weak derivatives of the argument (such as the weak second fundamental form of the curvature varifolds). Thus, to study the relaxation of functionals (7.1) with nonconvex g, one needs to add yet more dimensions in which the curvature is encoded explicitly (rather than as a derivative). Just like the Grassmannian dimensions of a varifold explicitly encode the tangent space of the *underlying manifold*, one can enlarge $\mathbb{R}^d \times \widetilde{\mathrm{Gr}}(m, \mathbb{R}^d)$ by the tangent space of the *Grassmannian*, in which then the curvature will be encoded (one could speak of a hyper-varifold). This is the approach taken in [14] (discussed in Sect. 7.2.3) for curves: An oriented curve $\gamma \subset \mathbb{R}^2$ is represented by a Radon measure μ_γ on $\mathbb{R}^2 \times \widetilde{\mathrm{Gr}}(1, \mathbb{R}^2) \times \mathbb{R}$ (where factor $\mathbb{R}$ is viewed as the tangent space to $\widetilde{\mathrm{Gr}}(1, \mathbb{R}^2)$) with

$$\mu_\gamma(\varphi) = \int_\gamma \varphi(x, \tau_\gamma(x), \kappa_\gamma(x))\, \mathrm{d}\mathcal{H}^1(x) \qquad \text{for all } \varphi \in C_0^\infty(\mathbb{R}^2 \times \widetilde{\mathrm{Gr}}(1, \mathbb{R}^2) \times \mathbb{R})\,.$$

A discrete version of this approach is obtained if $\mathbb{R}^2$ is replaced by a discrete planar graph in $\mathbb{R}^2$ and only curves γ on the graph are considered. Each pair of adjacent edges can then be interpreted as a (discrete version of) a point in $\mathbb{R}^2 \times \widetilde{\mathrm{Gr}}(1, \mathbb{R}^2) \times \mathbb{R}$, leading to essentially the same curvature regularization functionals [24, 39]. A discrete version for surfaces γ in $\mathbb{R}^3$ is described in [40].

Gauss Graph Currents So-called currents (or flat chains) can be viewed as a slightly dimension-reduced version of oriented varifolds. Formally, the current $\bar{\mu}$ associated with a varifold μ is obtained as

$$\bar{\mu}(\varphi) = \int_{\mathbb{R}^d \times \widetilde{\mathrm{Gr}}(m, \mathbb{R}^d)} \langle \varphi(x), \tau \rangle\, \mathrm{d}\mu(x, \tau) \qquad \text{for all } \varphi \in C_0^\infty(\mathbb{R}^d; \textstyle\bigwedge^m \mathbb{R}^d)\,,$$

where $C_0^\infty(\mathbb{R}^d; \bigwedge^m \mathbb{R}^d)$ are smooth m-forms, $\bigwedge^m \mathbb{R}^d$ are the real-valued alternating m-linear functionals on $(\mathbb{R}^d)^m$, and $\langle \cdot, \cdot \rangle$ is the bilinear pairing between $\bigwedge^m \mathbb{R}^d$ and the exterior products $\bigwedge_m \mathbb{R}^d$. Roughly speaking, we reduce a measure on $\mathbb{R}^d \times \widetilde{\mathrm{Gr}}(m, \mathbb{R}^d)$ to a $\widetilde{\mathrm{Gr}}(m, \mathbb{R}^d)$-valued measure on $\mathbb{R}^d$. To provide an example, for curves in $\mathbb{R}^2$ we have $\bar{\mu}_\gamma = \tau_\gamma \mathcal{H}^1 \llcorner \gamma$, where $\tau_\gamma \in S^1$ is the tangent vector to γ. The same dimension-reduction trick can be applied to the hyper-varifolds from above: Instead of the Radon measure on $\mathbb{R}^d \times \widetilde{\mathrm{Gr}}(m, \mathbb{R}^d) \times \mathbb{R}^{(d-m)\times m}$ one considers a current on $\mathbb{R}^d \times \widetilde{\mathrm{Gr}}(m, \mathbb{R}^d)$. Identifying $\widetilde{\mathrm{Gr}}(d-1, \mathbb{R}^d)$ with S^{d-1}, a $(d-1)$-dimensional surface γ in $\mathbb{R}^d$ is thus represented by the current associated to the graph $\{(x, \nu_\gamma(x)) \mid x \in \gamma\}$ of its Gauss map $x \mapsto \nu_\gamma(x) \in S^{d-1}$ (where ν_γ is the normal to γ in x). Expressing curvature dependent functionals (7.1) as functionals of these currents, Anzelotti et al. and Delladio et al. examined their relaxation and coercivity properties as well as regularity properties of the minimizers [4, 21, 22]. The exact same representation is used in [18] for curves in $\mathbb{R}^2$ (discussed in Sect. 7.2.4).

Sub-Riemannian Interpretation In mathematical image processing, the same lifting of curves and image level lines into $\mathbb{R}^2 \times \widetilde{\mathrm{Gr}}(1, \mathbb{R}^2)$ has been inspired by the structure of the visual cortex, where to each point in the retina there belong neurons for different orientations [19]. In this context, $\mathbb{R}^2 \times \widetilde{\mathrm{Gr}}(1, \mathbb{R}^2) \equiv \mathbb{R}^2 \times S^1$ is often identified with the roto-translation space $SE(2)$, the special Euclidean group, since (7.1) is naturally invariant under the action of $SE(2)$. $SE(2) \equiv \mathbb{R}^2 \times S^1$ is endowed with a sub-Riemannian structure whose horizontal curves $t \mapsto (x(t), \tau(t))$ satisfy $\dot{x}(t) \in \mathrm{span}(\tau(t))$. Thus, the position (x, τ) of a horizontal curve explicitly encodes the tangent direction just like varifolds do. An image $u : \mathbb{R}^2 \to \mathbb{R}$ is then represented by the measure μ on $\mathbb{R}^2 \times S^1$ with

$$\mu(\varphi) = \int_{\mathbb{R}^2} \varphi\left(x, \tfrac{\nabla u(x)}{|\nabla u(x)|}\right) |\nabla u(x)|\, \mathrm{d}x \qquad \text{for all } \varphi \in C_0^\infty(\mathbb{R}^d \times S^1)\,,$$

and inpainting or denoising is performed by alternating between a step of sub-Riemannian diffusion on μ and a projection of the smoothed measure $\hat{\mu}$ onto a two-dimensional image by choosing the level line in each point x parallel to $\mathrm{argmax}\hat{\mu}(x,\cdot)$ [20]. Modified versions only use few discrete orientations [12, 35] or an additional scale dimension [38, 41] and diffusion in the wavelet rather than real domain.

Calibration for Mumford–Shah Type Functionals In (7.1) we think of a convex regularization g of curvature along *smooth* curves γ. To allow *piecewise smooth* curves one could add a concave term of the curvature singularities, just like the well-known Mumford–Shah functional has both a convex term in the Lebesgue-continuous part of the image gradient and a concave term in the singular part. A common technique to convexify the Mumford–Shah problem is to represent the sought image as a measure on its graph (typically as the gradient of the characteristic function of its subgraph) [16]. The reformulation in this new variable is a convex, generalized mass minimization problem whose dual solutions are known as calibrations. Applying the same technique to curvature singularities, one has to lift a curve γ to a measure on the graph of its tangent map $x \mapsto \tau_\gamma(x)$, thus to an oriented varifold. This approach is detailed in Sect. 7.2.5.

7.2.2 The Curvature Varifold Approach

Even though convex curvature regularization functionals for image processing have nowhere been formulated with curvature varifolds, this approach still deserves being presented as it forms the first level of the lifting hierarchy from the previous section. As explained before, (sufficiently) smooth oriented curves $\gamma \subset \Omega \subset \mathbb{R}^2$ are here represented as nonnegative Radon measures $\mu_\gamma \in \mathrm{rca}_+(\Omega \times \widetilde{\mathrm{Gr}}(1,\mathbb{R}^2)) \equiv \mathrm{rca}_+(\Omega \times S^1)$ with $\int_{\Omega\times S^1} \varphi(x,\tau)\,\mathrm{d}\mu_\gamma(x,\tau) = \int_\gamma \varphi(x,\tau_\gamma(x))\,\mathrm{d}\mathcal{H}^1(x)$ for all smooth test functions φ, where τ_γ is the unit tangent vector to γ. A curvature dependent functional (7.1) can then be expressed as

$$R(\gamma) = \int_{\Omega\times S^1} \alpha + \beta g\left(\tfrac{\kappa(\mu_\gamma)}{\mu_\gamma}\right)\,\mathrm{d}\mu_\gamma\,,$$

where $\frac{\kappa(\mu_\gamma)}{\mu_\gamma}$ is the Radon–Nikodym derivative of the measure $\kappa(\mu_\gamma)$ with respect to μ_γ, and $\kappa(\mu_\gamma)$ represents the weak curvature of the varifold μ_γ as defined below.

Definition 7.2 (Generalized Curvature, [27, Def. 5.2.1], [14, Def. 2.1]) Given an oriented varifold $\mu_\gamma \in \mathrm{rca}_+(\Omega \times S^1)$, its *generalized curvature*, if it exists, is the measure $\kappa = \kappa(\mu_\gamma) \in \mathrm{rca}(\Omega \times S^1)$ defined via

$$\int_{\Omega\times S^1} \tfrac{\partial\psi(x,\tau)}{\partial\tau}\,\mathrm{d}\kappa(x,\tau) = -\int_{\Omega\times S^1} \nabla_x\psi(x,\tau)\cdot\tau\,\mathrm{d}\mu_\gamma(x,\tau) \quad \text{for all } \psi \in C_0^\infty(\Omega\times S^1).$$

This definition just generalizes the fact that along a closed curve γ and for a smooth test function ψ we have $0 = \int_\gamma \frac{\mathrm{d}}{\mathrm{d}\gamma}\psi(x,\tau_\gamma(x))\,\mathrm{d}\mathcal{H}^1(x) = \int_\gamma \nabla_x\psi(x,\tau_\gamma(x))\cdot\tau_\gamma(x) + \frac{\partial\psi(x,\tau_\gamma(x))}{\partial\tau}\kappa_\gamma(x)\,\mathrm{d}\mathcal{H}^1(x)$. One can show that the generalized curvature is unique, if it exists.

Next, we extend the approach to image level lines. If γ is the boundary ∂E of some open set $E \subset \Omega$, then this representation can be interpreted as a lifting of the gradient $D\chi_E = \tau_\gamma^\perp \mathcal{H}^1 \llcorner \partial E$ of the corresponding characteristic function χ_E, where $(\cdot)^\perp$ denotes counterclockwise rotation by $\frac{\pi}{2}$. Indeed, $D\chi_E$ is obtained from μ_γ by a projection,

$$D\chi_E = [\tau^\perp\mu_\gamma](\cdot, S^1)\,,$$

where $\tau^\perp\mu_\gamma$ is the measure μ_γ multiplied with the continuous function $(x,\tau)\mapsto\tau^\perp$. A general image $u:\Omega\to[0,1]$ can now be represented as a convex combination $\int_0^1 \chi_{E(s)}\,\mathrm{d}s$ of such characteristic functions, and the corresponding lifting of the image gradient Du is

$$\mu = \int_0^1 \mu_{\partial E(s)}\,\mathrm{d}s\,.$$

A corresponding cost could thus be defined as $\int_0^1 R(\partial E(s))\,\mathrm{d}s$. Representing an image via different convex combinations of characteristic functions leads to different lifted representations of its gradient, and picking the one with lowest cost will produce the convex relaxation of (7.2). In fact, since convex combinations of characteristic functions of smooth sets are difficult to characterize, we admit even more liftings of the image gradient Du, which will be the reason for our functional being actually smaller than the convex relaxation.

Definition 7.3 (Image Gradient Lifting) Given an image $u\in\mathrm{BV}(\Omega)$, a measure $\mu_1\in\mathrm{rca}_+(\Omega\times S^1)$ is called a *lifting* of Du if

$$Du = [\tau^\perp\mu_1](\cdot, S^1)\,.$$

In summary, this leads to the following regularization functional.

Definition 7.4 (Curvature Varifold Regularization Functional) The *curvature varifold regularization functional* for an image $u\in\mathrm{BV}(\Omega)$ is given by

$$R_1(u) = \inf\left\{\int_{\Omega\times S^1}\alpha+\beta g\left(\frac{\kappa(\mu_1)}{\mu_1}\right)\mathrm{d}\mu\;\middle|\;\mu_1 \text{ is a lifting of } Du,\ \kappa(\mu_1)\text{ exists}\right\}.$$

Above, if $\frac{\kappa(\mu_1)(x)}{\mu_1(x)}=\infty$, then we shall tacitly interpret $g(\kappa(\mu_1)(x)/\mu_1(x))\,\mathrm{d}\mu_1(x) = g^\infty(\kappa(\mu_1)(x)/|\kappa(\mu_1)(x)|)\mathrm{d}\kappa(\mu_1)(x)$ with $g^\infty(t)=\lim_{s\to\infty}g(st)/s$ the recession function. The following properties, important to apply the direct method in the calculus of variations, are straightforward to check and show that R_1 is as nice a curvature regularizing functional as one can think of.

Theorem 7.5 (Coercivity and Lower Semi-continuity) *We have $R_1(u) \geq \alpha|u|_{\mathrm{TV}}$. If g is convex and lower semi-continuous, then $\int_{\Omega\times S^1} \alpha + \beta g\big(\frac{\kappa(\mu_1)}{\mu_1}\big)\,\mathrm{d}\mu_1$ is jointly convex in $\kappa(\mu_1)$ and μ_1, and R_1 is convex and lower semi-continuous on $L^1(\Omega)$.*

7.2.3 The Hyper-Varifold Approach

Here we detail the approach of [14]. As already mentioned, (sufficiently) smooth curves $\gamma \subset \mathbb{R}^2$ are here represented by nonnegative measures $\mu_\gamma \in \mathrm{rca}_+(\Omega \times S^1 \times \mathbb{R})$ with $\mu_\gamma(\varphi) = \int_\gamma \varphi(x, \tau_\gamma(x), \kappa_\gamma(x))\,\mathrm{d}\mathcal{H}^1(x)$ for all smooth test functions φ, where from any such hyper-varifold μ_γ one can retrieve the classical oriented varifold of γ from the previous section as

$$\mathrm{red}\mu_\gamma = \mu_\gamma(\cdot, \cdot, \mathbb{R})\,.$$

Consequently, liftings of image gradients now have one more space dimension.

Definition 7.6 (Image Gradient Lifting, [14, (3.11)–(3.12)]) Given an image $u \in \mathrm{BV}(\Omega)$, a measure $\mu_2 \in \mathrm{rca}_+(\Omega \times S^1 \times \mathbb{R})$ is called a *lifting* of Du if

$$Du = [\tau^\perp \mathrm{red}\mu_2](\cdot, S^1) \quad \text{and} \quad \kappa(\mathrm{red}\mu_2) = [\kappa\mu_2](\cdot, \cdot, \mathbb{R})$$

or equivalently

$$\int_{\Omega\times S^1\times\mathbb{R}} \phi(x)\cdot\tau^\perp\,\mathrm{d}\mu_2(x,\tau,\kappa) = -\int_\Omega u(x)\mathrm{div}\phi(x)\,\mathrm{d}x \quad \forall\phi \in C_0^\infty(\Omega;\mathbb{R}^2)\,,$$

$$0 = \int_{\Omega\times S^1\times\mathbb{R}} \nabla_x\psi(x,\tau)\cdot\tau + \frac{\partial\psi(x,\tau)}{\partial\tau}\kappa\,\mathrm{d}\mu_2(x,\tau,\kappa) \quad \forall\psi \in C_0^\infty(\Omega\times S^1)\,.$$

Again, the first condition just ensures that μ_2 fits to the given image u, and the second condition ensures that the support of μ_2 in the curvature dimension is consistent with the orientation encoded in the Grassmannian dimension. Essentially, the second condition means that the hyper-varifold μ_2 represents the varifold $\mathrm{red}\mu_2$, while the first condition means that the varifold $\mathrm{red}\mu_2$ represents Du. The curvature regularizing functional then turns into the following.

Definition 7.7 (Hyper-Varifold Regularization Functional) The *hyper-varifold regularization functional* for an image $u \in \mathrm{BV}(\Omega)$ is given by

$$R_2(u) = \inf\left\{\int_{\Omega\times S^1\times\mathbb{R}} \alpha + \beta g(\kappa)\,\mathrm{d}\mu_2(x,\tau,\kappa)\,\middle|\,\mu_2 \text{ is a lifting of } Du\right\}\,.$$

Again, this regularization enjoys nice properties.

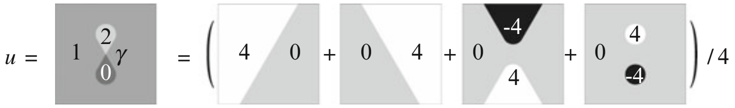


Fig. 7.2 The hyper-varifold belonging to the 8-shaped discontinuity curve γ is a lifting of the image gradient Du, even though it does not correspond to a convex combination of hyper-varifolds $\mu_{\partial E}$. Consequently, R_2 is strictly smaller than the convex relaxation of (7.2) (which is finite nevertheless, since u admits the shown decomposition into images with smooth level lines)

Theorem 7.8 (Coercivity and Lower Semi-continuity, [14, Thm. 3.3–3.6]) *R_2 is convex with $R_2(u) \geq \alpha|u|_{\mathrm{TV}}$. If g is lower semi-continuous, then R_2 is so on $L^1(\Omega)$. If additionally g grows superlinearly, the minimum in the definition of R_2 is attained.*

As in the previous section, we admit more hyper-varifolds as liftings of an image gradient Du than just those that are convex combinations of hyper-varifolds $\mu_{\partial E}$ corresponding to boundaries of (sufficiently) smooth open sets $E \subset \Omega$. In particular, the hyper-varifolds corresponding to self-intersecting curves γ are admissible liftings that cannot be represented this way, see Fig. 7.2. As a consequence, R_2 lies below the convex relaxation of (7.2).

Theorem 7.9 (Convex Relaxation Gap, [14, Sec. 3.4]) *R_2 lies below the convex relaxation of* (7.2) *in $L^1(\Omega)$, and strictly so for some images $u \in \mathrm{BV}(\Omega)$.*

7.2.4 The Gauss Graph Current Approach

We now detail the ansatz in [18]. A closed curve $\gamma \subset \mathbb{R}^2$ is here represented by the corresponding 1-current μ_γ on the graph of its Gauss map $x \mapsto \nu_\gamma(x)$ or equivalently on the graph of its tangent space map $x \mapsto \tau_\gamma$ (which we shall consider for consistency with the other models). This 1-current is nothing else than a divergence-free vector-valued measure in $\mathbb{R}^2 \times S^1$ (more general 1-currents can be more complicated), given by $\mu_\gamma(\varphi) = \int_\gamma \varphi(x, \tau_\gamma(x)) \cdot (\tau_\gamma(x), \kappa_\gamma(x))\, \mathrm{d}\mathcal{H}^1(x)$ for all $\mathbb{R}^3$-valued smooth test functions φ. Replacing φ with the gradient $\nabla_{(x,\tau)}\psi$ of a smooth function ψ, the integrand becomes the directional derivative of ψ along the curve (γ, τ_γ), which integrates to zero since the curve is closed. Thus, $\mu_\gamma = (\mu_\gamma^x, \mu_\gamma^\tau)$ is divergence-free in the distributional sense. Furthermore, $\mu_\gamma^x/|\mu_\gamma^x|$ describes the tangent vector to γ and $\mu_\gamma^\tau/|\mu_\gamma^x|$ its curvature. The transfer to lifted image gradients and the corresponding curvature regularization functional are as follows.

Definition 7.10 (Image Gradient Lifting) Given an image $u \in \mathrm{BV}(\Omega)$, a measure $\mu_3 = (\mu_3^x, \mu_3^\tau) \in \mathrm{rca}(\Omega \times S^1; \mathbb{R}^3)$ is called a *lifting* of Du if

$$Du = [\mu_3^x]^\perp(\cdot, S^1) \quad \text{and} \quad \tfrac{\mu_3^x}{|\mu_3^x|}(x,\tau)=\tau \text{ for } |\mu_3^x|\text{-almost all } (x,\tau) \quad \text{and} \quad \mathrm{div}\mu_3=0.$$

Definition 7.11 (Gauss Graph Current Regularization Functional) The *Gauss graph current regularization functional* for an image $u \in \mathrm{BV}(\Omega)$ is given by

$$R_3(u) = \inf \left\{ \int_{\Omega \times S^1} \alpha + \beta g\left(\frac{\mu_3^\tau}{|\mu_3^x|}\right) \mathrm{d}|\mu_3^x| \;\middle|\; \mu_3 \text{ is a lifting of } Du \right\}.$$

It turns out that for each $u \in \mathrm{BV}(\Omega)$ there exists a smooth recovery sequence $u_n \to u$ in $L^1(\Omega)$ along which the functional converges. Also, just like the previous models, this one lies below the convex relaxation of (7.2), however, in [18] it is shown that the convexification is tight for images u that are characteristic functions of a set $E \subset \Omega$ with C^2-boundary.

Theorem 7.12 (Approximation and Tightness, [18, Prop. 3.1, Thm. 1]) *For any $u \in \mathrm{BV}(\Omega)$ there is a sequence $u_n \in C^\infty(\overline{\Omega})$ with $u_n \to u$ in $L^1(\Omega)$ so that $R_3(u_n) \to R_3(u)$. Furthermore, $R_3(\chi_E) = \int_{\partial E} \alpha + \beta g(\kappa_{\partial E})\, \mathrm{d}\mathcal{H}^1$ for all $E \subset \Omega$ with C^2-boundary.*

The latter result depends on the Smirnov-decomposition of divergence-free vector-valued measures (or 1-currents of finite mass without boundary) into 1-currents belonging to curves. Thus it implicitly makes use of the assumption $\alpha > 0$ (since otherwise the mass might be unbounded), as is necessary by Theorem 7.1. The following relation between the different image gradient liftings is straightforward to check.

Theorem 7.13 (Image Gradient Liftings) *Let $u \in \mathrm{BV}(\Omega)$. From an image gradient lifting μ_i in one model one can construct an image gradient lifting μ_j in any other model:*

If $\frac{\kappa(\mu_1)}{\mu_1}$ exists, set $\mu_2(\varphi) = \int_{\mathbb{R}^2 \times S^1} \varphi\left(x, \tau, \frac{\kappa(\mu_1)(x,\tau)}{\mu_1(x,\tau)}\right) \mathrm{d}\mu_1(x,\tau)$ $\quad \forall \varphi \in C_0^\infty(\mathbb{R}^2 \times S^1 \times \mathbb{R})$.

Given μ_2, set $\mu_3(\varphi) = \int_{\mathbb{R}^2 \times S^1 \times \mathbb{R}} \varphi(x,\tau) \cdot \binom{\tau}{\kappa} \mathrm{d}\mu_2(x,\tau,\kappa)$ $\quad \forall \varphi \in C_0^\infty(\mathbb{R}^2 \times S^1; \mathbb{R}^3)$.

Given μ_3, set $\mu_1(\varphi) = \int_{\mathbb{R}^2 \times S^1} \varphi(x,\tau)\tau \cdot \mathrm{d}\mu_3^x(x,\tau)$ $\quad \forall \varphi \in C_0^\infty(\mathbb{R}^2 \times S^1)$,

and $\kappa(\mu_1) = \mu_3^\tau$.

A direct corollary is the equivalence of all three models presented so far. Note that for nonconvex g, R_2 still makes sense and is the same as if g is replaced by its convex relaxation.

Corollary 7.14 (Model Equivalence, [18, Cor. 3.4]) *If g is convex, $R_1 = R_2 = R_3$.*

Proof Let $u \in \mathrm{BV}$ and μ_1, μ_2, μ_3 denote liftings of Du according to the three models.

$R_2(u) \leq R_1(u)$ Given μ_1 with $\kappa(\mu_1) \ll \mu_1$, construct μ_2 as in Theorem 7.13. Then by construction $\int_{\Omega \times S^1 \times \mathbb{R}} \alpha + \beta g(\kappa)\, \mathrm{d}\mu_2(x,\tau,\kappa) = \int_{\Omega \times S^1} \alpha + \beta g\left(\frac{\kappa(\mu_1)}{\mu_1}\right) \mathrm{d}\mu_1$. If not $\kappa(\mu_1) \ll \mu_1$ (e.g., if g only has linear growth so that points with infinite curvature are allowed), one needs to argue via duality as done in [18, Cor. 3.4].

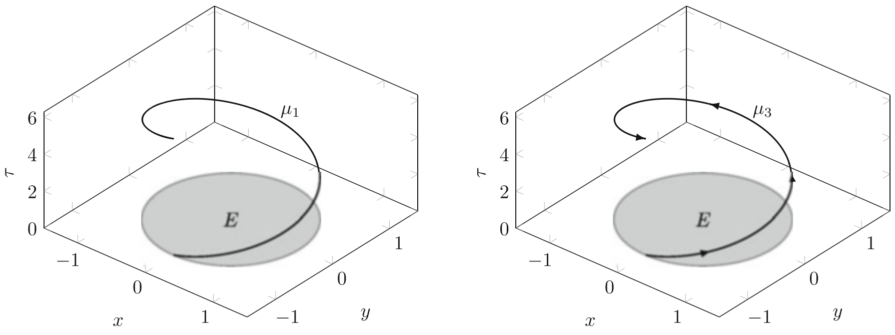


Fig. 7.3 Image gradient lifting for the characteristic function of a disk with radius 1. Left: Scalar lifting μ_1 from Sects. 7.2.2 and 7.2.5 (the black line shows the support). The same graph illustrates μ_2 from Sect. 7.2.3 by depicting its $\kappa = 1$ slice (all other slices are empty). Right: Vector-valued lifting μ_3 from Sect. 7.2.4 (the arrows indicate the direction of the vector-valued measure)

$R_3(u) \leq R_2(u)$ Given μ_2, by [14, Thm.3.11] we may assume the existence of a function $\bar{\kappa}(x, \tau)$ such that μ_2 has support on $\{(x, \tau, \bar{\kappa}(x, \tau)) \,|\, (x, \tau) \in \Omega \times S^1\}$. Construct μ_3 as in Theorem 7.13, then $(|\mu_3^x|, \mu_3^\tau) = (\mu_2, \bar{\kappa}\mu_2)(\cdot, \cdot, \mathbb{R})$ so that $\mu_3^\tau/|\mu_3^x| = \hat{\kappa}$ and thus $\int_{\Omega\times S^1} \alpha + \beta g(\mu_3^\tau/|\mu_3^x|)\,\mathrm{d}|\mu_3^x| = \int_{\Omega\times S^1} \alpha + \beta g(\hat{\kappa})\,\mathrm{d}\mu_2(\cdot, \cdot, \mathbb{R}) = \int_{\Omega\times S^1\times\mathbb{R}} \alpha + \beta g(\kappa)\,\mathrm{d}\mu_2(x, \tau, \kappa)$.

$R_1(u) \leq R_3(u)$ Given μ_3, construct μ_1 as in Theorem 7.13, then using $\mu_1 = \tau \cdot \mu_3^x = |\mu_3^x|$ and $\kappa(\mu_1) = \mu_3^\tau$ one has $\int_{\Omega\times S^1} \alpha + \beta g\left(\frac{\kappa(\mu_1)}{\mu_1}\right) \mathrm{d}\mu_1 = \int_{\Omega\times S^1} \alpha + \beta g\left(\frac{\mu_3^\tau}{|\mu_3^x|}\right) \mathrm{d}|\mu_3^x|$. □

Thus, all model properties previously cited actually hold for all three models. A disadvantage of the hyper-varifold model is that it requires an additional dimension and that an optimal lifting μ_2 may not exist if g does not grow superlinearly. On the other hand, it can deal with nonconvex g (though then R_2 will be the same as if g were replaced by its convex relaxation). The different liftings are illustrated in Fig. 7.3.

7.2.5 The Jump Set Calibration Approach

This section presents the method from [13], where the same lifting μ_1 of curves and image gradients is used as in Sect. 7.2.2. In Sect. 7.2.2 the regularizer R_1 was a convex penalization of the μ_1-continuous part of the generalized curvature $\kappa(\mu_1)$. Here instead we aim for a regularization of the μ_1-singular part, which corresponds to discontinuities in the tangent vector of curves or level lines. A corresponding Mumford–Shah type regularization functional reads

$$R(\gamma)=\int_\gamma \alpha+\beta g(\kappa_\gamma)\,\mathrm{d}\mathcal{H}^1+\int_{\{\text{discontinuity points of }\tau_\gamma\}}\rho(\tau_\gamma^-,\tau_\gamma^+)\,\mathrm{d}\mathcal{H}^0\,,$$

where κ_γ is the $\mathcal{H}^1$-continuous part of the curvature, $\tau_\gamma^\pm$ represents the tangent on either side of a kink, and $\rho : S^1 \times S^1 \to [0,\infty)$ is lower semi-continuous and satisfies the triangle inequality (both the classical requirements for well-posedness of Mumford–Shah type energies). As explained in Sect. 7.2.1, the natural lifting in which this problem becomes convex is the one of oriented varifolds as in Sect. 7.2.2. Since the previous sections have already dealt with the nonsingular part, below we simply assume $g(0) = 0$ and $g \equiv \infty$ otherwise (the more general case can be obtained by combining the models from the previous sections and this section).

To derive the form of the lifted regularization functional, consider a closed polygonal curve $\gamma \subset \Omega$ which is smooth except for a finite number of kinks $\Gamma \subset \gamma$. Testing its lifting μ_1 with $\tau \cdot \nabla_x\psi(x,\tau)$ for $\psi \in C_0^\infty(\Omega \times S^1)$ yields

$$\int_{\Omega\times S^1}\tau\cdot\nabla_x\psi(x,\tau)\,\mathrm{d}\mu_1(x,\tau)=\sum_{x\in\Gamma}\psi(x,\tau_\gamma^-(x))-\psi(x,\tau_\gamma^+(x))\,.$$

Taking the supremum over all ψ with $\psi(x,\tau_1)-\psi(x,\tau_2) \le \rho(\tau_1,\tau_2)$ for all x,τ_1,τ_2 yields the desired $R(\gamma)=\int_\Gamma \rho(\tau_\gamma^-(x),\tau_\gamma^+(x))\,\mathrm{d}\mathcal{H}^0$.

Definition 7.15 (Jump Set Calibration Regularization Functional) The *jump set calibration regularization functional* for an image $u \in \mathrm{BV}(\Omega)$ is given by

$$R_\rho(u)=\inf\left\{\int_{\Omega\times S^1}\alpha\,\mathrm{d}\mu_1+T_\rho(\mu_1)\,\middle|\,\mu_1\text{ is a lifting of }Du\right\}\quad\text{with}$$

$$T_\rho(\mu_1)=\sup\left\{\int_{\Omega\times S^1}\tau\cdot\nabla_x\psi(x,\tau)\,\mathrm{d}\mu_1(x,\tau)\,\middle|\,\psi\in M_\rho\right\}$$

and $M_\rho=\left\{\psi\in C_0^\infty(\Omega\times S^1)\,\middle|\,\psi(x,\tau_1)-\psi(x,\tau_2)\le\rho(\tau_1,\tau_2)\,\forall x\in\Omega,\ \tau_1,\tau_2\in S^1\right\}$.

Theorem 7.16 (Regularization Properties, [13, Prop. 4.4]) *R_ρ is convex, positively one-homogeneous, and lower semi-continuous on $L^1(\Omega)$. Further, $R_\rho(u) \ge \alpha|u|_{\mathrm{TV}}$.*

Above, ψ plays the role of a calibration. Taking ρ as the discrete metric ρ_0 (as in the original Mumford–Shah functional) or the geodesic metric ρ_1 on S^1 yields two interesting, extremal cases, the former measuring the number of kinks (only allowing straight line segments) and the latter the total absolute curvature.

Theorem 7.17 (Singular Curvature Penalization, [13, Prop. 3.16]) *Let $\gamma \subset \Omega$ be a piecewise C^2 curve and μ_1 the corresponding oriented varifold. Then*

$$T_{\rho_0}(\mu_1)=\mathcal{H}^0(\Gamma)\ \textit{if}\ \kappa_\gamma=0\ \textit{and}\ T_{\rho_0}(\mu_1)=\infty\ \textit{else,}$$

$$T_{\rho_1}(\mu_1) = \int_\gamma |\kappa_\gamma| \, \mathrm{d}\mathcal{H}^1 + \sum_{x\in\Gamma} |\theta(x)| \,,$$

where $\Gamma \subset \gamma$ is the set of kinks and $\theta(x)$ the exterior angle of γ at $x \in \Gamma$.

For numerical purposes note that the corresponding M_ρ can be reduced to

$$M_{\rho_0} = \left\{\psi \in C_0^\infty(\Omega \times S^1) \,\middle|\, |\psi(x,\tau)| \le \tfrac{1}{2} \,\forall x \in \Omega,\ \tau \in S^1\right\} ,$$
$$M_{\rho_1} = \left\{\psi \in C_0^\infty(\Omega \times S^1) \,\middle|\, \left|\tfrac{\partial\psi(x,\tau)}{\partial\tau}\right| \le 1 \,\forall x \in \Omega,\ \tau \in S^1\right\} .$$

7.3 Discretization Strategies

In order to make use of the models from the previous section numerically, their discretization is required, in particular a discretization of the lifted image gradients μ_1, μ_2, μ_3. This discretization poses two major challenges, the high problem dimensionality (due to the additional lifting dimensions) and the discretization of very singular objects (varifolds and currents). Both are mutually dependent—indeed, it is only due to the addition of the lifting dimensions that the lifted image gradients μ_1, μ_2, μ_3 are singular, concentrating essentially on two-dimensional surfaces (or even one-dimensional lines) in a higher-dimensional space. A parameterized representation of these two-dimensional surfaces would thus eliminate both challenges at the same time, but come at the expense of loosing convexity.

The choice of the discretization has a direct impact on the computational effort or memory consumption and the quality of the computational results, and the approaches discussed below try to balance both aspects. In essence, computational effort has to be balanced with the discrete resolution of the additional orientation (and curvature) dimension. Once this resolution is chosen, there is still a tradeoff between the number of orientations that actually *occur* in the regularized images and the blurriness of these images: Tying the level line orientations only weakly to the discrete grid on S^1 (in other words, using a discretization that is not localized but rather a little diffuse in S^1 direction) makes roughly all level line orientations available, but results in blurry images, while a strict consistency constraint between level line orientation and varifold support will result in sharp images, which however only contain the small number of allowed discrete orientations. Section 7.3.4 will present an adaptive discretization approach to address this dilemma. In any case, the lifted image gradients can only be approximated in a weak sense.

We will not discuss corresponding optimization methods; typically, primal-dual algorithms such as [17] work efficiently on these non-smooth convex optimization problems, and the models of Sects. 7.2.3 and 7.2.5 can even be tackled with linear program solvers such as [32].

7.3.1 *Finite Differences*

A common discretization is the representation of functions by their values on the nodes of a regular Cartesian grid (which is particularly natural in image processing since image data typically is of this format) and the implementation of derivatives via finite differences. In [13] such an approach is used to discretize the model of Sect. 7.2.5: The image gradient lifting μ_1 is replaced by a function on $\Omega \times S^1 \equiv \Omega \times [0, 2\pi)$ (identifying 0 with 2π), which is then represented discretely by its values at the voxels of an evenly spaced voxel grid on $\Omega \times [0, 2\pi)$. The image u is analogously represented on the corresponding pixel grid on Ω, and the same discretizations are used for the dual variables (ψ from Definition 7.15 and the Lagrange multiplier for the constraint in Definition 7.3). The difficulty of this approach lies in the approximation of the directional derivative $\tau \cdot \nabla_x$ in Definition 7.15. In a primal-dual optimization, it is applied to ψ, and its adjoint, which is of the same form, is applied to μ_1. Since μ_1 is expected to concentrate along lines (think of the image gradient lifting of a characteristic function) and the directional derivative is used to accurately detect the corresponding line orientation, it turns out insufficient to simply implement $\tau \cdot \nabla_x$ with nearest neighbor finite differences (as would for instance be done in the upwind scheme for the transport equation; just like the upwind discretization causes numerical dissipation when solving the transport equation, it results in too blurry images in our context). Instead, $\tau \cdot \nabla_x$ is computed by finite differences along the stencil in Fig. 7.4 left, where function values at the stencil endpoints are obtained by interpolation. Still, results remain a little blurry (cf. Figs. 7.6 and 7.7), but as a great advantage this model is easily parallelizable on a GPU. Note that the above discussion would not change if a different model and image gradient lifting were considered.

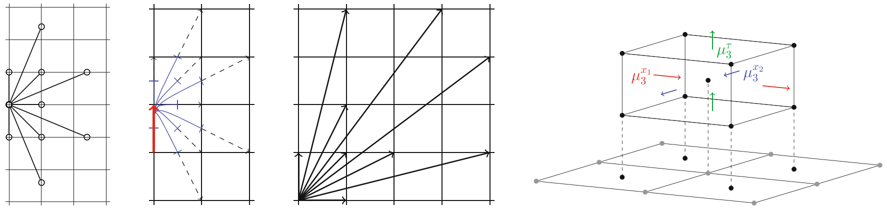


Fig. 7.4 Stencil for finite difference discretization (left, showing only half the orientations); basis of curved line measure discretization (second, for initial upwards orientation, indicated by red arrow, and showing only half the final orientations); adaptively refinable basis of straight line measures (third, showing only line measures into one quadrant); Raviart–Thomas discretization with a voxel in $\Omega \times S^1$ shifted relative to the pixels in Ω by half a grid width (right)

7.3.2 Line Measure Segments

Measures are frequently approximated (weakly-$*$) by linear combinations of Dirac masses. Since image level lines are closed, the natural extension of this idea to image gradient liftings would be an approximation by linear combinations of line measure segments μ_1^i or μ_2^i belonging to curve segments $\gamma_i \subset \Omega$, $i = 1, \dots, N$, i.e., $\mu_1^i(\varphi) = \int_{\gamma_i} \varphi(x, \tau_{\gamma_i}(x)) \, \mathrm{d}\mathcal{H}^1(x)$ or $\mu_2^i(\varphi) = \int_{\gamma_i} \varphi(x, \tau_{\gamma_i}(x), \kappa_{\gamma_i}(x)) \, \mathrm{d}\mathcal{H}^1(x)$ for smooth test functions φ. All image level lines will then be composed of curve segments γ_i, and the regularization functionals will turn into simple functionals of the linear coefficients of the line measure segments. This discretization was employed for the model of Sect. 7.2.3 in [14] and for the model of Sect. 7.2.5 in [31, 44]. In principle, any collection of curves γ_i can be chosen, however, it is most convenient to build these curves from the underlying 2D pixel grid. For the model of Sect. 7.2.5 it suffices to use all straight lines γ_i that connect a pixel with one of its nth-nearest neighbors (e.g., for $n = 4$ as in Fig. 7.4 middle right; if γ_i passes exactly through another pixel, it can be shortened accordingly). For the model of Sect. 7.2.3 on the other hand one is especially interested in curves with minimal energy (7.1), so one can precompute the optimal curves γ_i for each incoming and outgoing direction at a pixel (where the same discrete directions are used as before, see the example curves in Fig. 7.4 middle left). The discretization of the dual variables now follows rather naturally: The discretization of ψ from Definition 7.6 or from Definition 7.15 lives at all points $(x, \tau) \in \Omega \times S^1$ such that there is a curve γ_i which starts or ends in x with direction τ. The Lagrange multiplier ϕ for Definition 7.3 or 7.6 lies in the span of one-dimensional hat functions on all pairs of neighboring pixels, which turns out to most naturally couple curves γ_i with piecewise constant pixel values. This approach yields sharp images, but only uses a limited number of level line directions, see Figs. 7.6 and 7.7.

7.3.3 Raviart–Thomas Finite Elements on a Staggered Grid

A Finite Element discretization is proposed in [18], where it is applied to the model of Sect. 7.2.4. Essentially all models have an underlying divergence constraint, which is most explicit in Definition 7.10, but translates to Definition 7.2 or the second condition in Definition 7.6 in the other models. Therefore, a divergence conforming method seems advantageous, for which the authors choose a first-order Raviart–Thomas Finite Element discretization [36] of μ_3 on a regular rectilinear voxel grid over $\Omega \times S^1 \equiv \Omega \times [0, 2\pi)$ (identifying 2π with 0). The degrees of freedom within each voxel element are the average (normal) fluxes through each element face, and $\mu_3^{x_1}, \mu_3^{x_2}, \mu_3^{\tau}$ within each element are obtained as the linear interpolation between the normal fluxes through the x_1-, x_2-, and τ-faces, respectively (i.e., the faces orthogonal to these coordinate directions). The objective functional as well as the directional constraint that $\mu_3^x(x, \tau)$ must be parallel to

τ are both evaluated in each element at the center. In contrast, the conditions $\mu_3^{x_1}(\cdot, S^1) = \partial_{x_2} u$ and $\mu_3^{x_2}(\cdot, S^1) = -\partial_{x_1} u$ (i.e., the requirement that μ_3 is a lifting of the image gradient Du) use the values of $\mu_3^{x_1}$ on the x_1-faces and of $\mu_3^{x_2}$ on the x_2-faces. Since the image gradient is here defined via finite differences between the pixels, the voxel grid is staggered with respect to the 2D image pixel grid (i.e., the voxels lie above the pixel corners, Fig. 7.4 right). Results seem to provide a good compromise between allowing many level line directions and introduction of blurriness, see Figs. 7.6 and 7.7.

7.3.4 Adaptive Line Measure Segments

As already mentioned before, the additional dimensions of the lifting models lead to a substantial size increase of the discretized optimization problems and an associated high computational effort. However, it is expected that the lifted image gradients concentrate on two-dimensional surfaces in the higher-dimensional lifting space (in the case of binary images as in segmentation applications the lifted image gradient even concentrates on one-dimensional lines). Thus, a uniform discretization of all of $\Omega \times S^1$ or $\Omega \times S^1 \times \mathbb{R}$ is a waste of computational resources that could be reduced by an adaptive discretization. We consider here an adaptive discretization of the varifold lifting for the model from Sect. 7.2.5 using the line measure segments from Sect. 7.3.2. As before, the image level lines will be composed of straight line segments γ_i that connect different pixels. In contrast to Sect. 7.3.2, however, any pixel can be (iteratively) refined by dividing it into four pixels of half the original width, thereby introducing new and finer line segments γ_i. Furthermore, while the line segment orientations were before chosen to be the ones associated with the nth-nearest neighbors of a pixel for n fixed, the number n can now also be chosen adaptively as a power of 2. Of course, not only the lifted image gradient μ_1 is discretized adaptively in this way, but also the image u itself, see Fig. 7.5.

The idea is to first solve the optimization problem on a very coarse discretization and then to refine the image and lifting grid at those locations where a finer

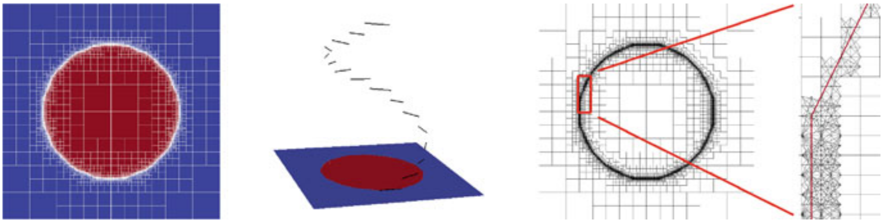


Fig. 7.5 The image u (left) and the image gradient lifting μ_1 (second) can be discretized adaptively. For μ_1 not only the spatial coordinates are adaptively refined, but also the directional τ-coordinate, resulting in a locally high number of available line segments (right, the zoom shows the active line measures of μ_1 in red)

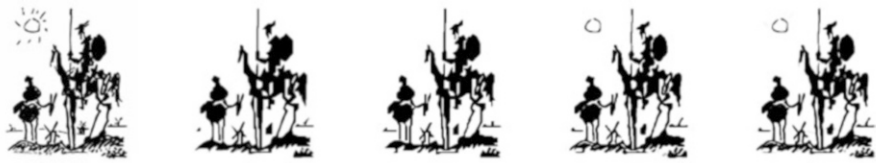


Fig. 7.6 Binary segmentation of an image $f : \Omega \to [0, 1]$ (left) by minimizing $\lambda \int_\Omega (\frac{1}{2} - f)u\,\underset{\cdot}{x} + R(u)$ among images $u : \Omega \to [0, 1]$ with regularizer R chosen as R_{ρ_0} (second, using finite differences, and third, using adaptive line measure segments), as R_2 with $g(\kappa) = \kappa^2$ (third, using curved line measure segments), and as R_3 with $g(\kappa) = \kappa^2$ (right, using Raviart–Thomas Finite Elements). In all experiments $\Omega = [0, 1]^2$, $\beta = 1$, $\alpha = 25$, $\lambda = 256^2$

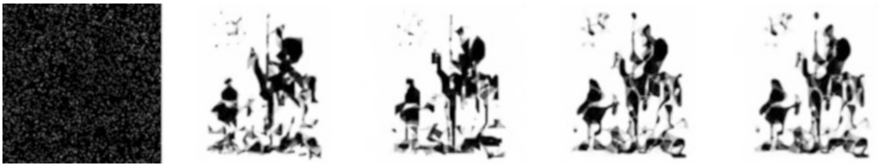


Fig. 7.7 Image inpainting from an image f with 95% missing pixels (left) by minimizing $R(u)$ among images $u : \Omega \to [0, 1]$ coinciding with f at the given pixels. Same regularizers, discretizations and parameters as in 7.6

resolution is needed (one may also coarsen the grid away from the support of μ_1). This process is repeated until a satisfactory resolution is obtained. As a local refinement criterion, a combination of two indicators turned out to be useful. One indicator for necessary refinement is the localized duality gap between the optimization problem and its dual (as used for mere length regularization in [11]). The other indicator marks a pixel for refinement if it is intersected by the support of μ_1 projected onto the image grid; furthermore, new line segments γ_i will be added in the refined pixels with discrete directions neighboring the discrete directions of μ_1. Of course, unused line segments can be removed from the discretization. The results are sharp and can be computed to a much higher resolution than with nonadaptive methods, see Figs. 7.6 and 7.7.

7.4 The Jump Set Calibration Approach in 3D

Any lifting brings with it the curse of dimensionality. Thus, for models to be feasible numerically one should try to add as few dimensions as possible. For curvature regularization of surfaces in three space dimensions this requirement becomes even more crucial than in 2D. Therefore, below we only discuss an approach based on a lifting to a varifold (a discrete version of the hyper-varifold approach in 3D is described in [40]), where we restrict ourselves to the jump set calibration approach.

7.4.1 Regularization Model

Here again, we are interested in a regularization of the singular part of the curvature. We concentrate on curvature singularities along curves (i.e., on creases of surfaces; a model regularizing cone singularities is also conceivable). A corresponding regularization functional for a surface $\gamma \subset \mathbb{R}^3$ reads

$$R(\gamma) = \int_\gamma \alpha + \beta g(\kappa_\gamma)\,\mathrm{d}\mathcal{H}^2 + \int_{\{\text{discontinuity lines of } \nu_\gamma\}} \rho(\nu_\gamma^-, \nu_\gamma^+)\,\mathrm{d}\mathcal{H}^1 ,$$

where $\nu_\gamma^\pm$ represents the surface normal on either side of the discontinuity, κ_γ represents the $\mathcal{H}^2$-continuous part of the curvatures (say, the second fundamental form or the principal curvatures), and ρ is a lower semi-continuous metric on S^2. Again we restrict ourselves to the special case $g(0) = 0$ and $g \equiv \infty$ otherwise.

The varifold lifting μ_1 of a (sufficiently) smooth surface $\gamma \subset \Omega \subset \mathbb{R}^3$ is a measure on $\Omega \times \widetilde{\mathrm{Gr}}(2, \mathbb{R}^3)$. Identifying (for notational simplicity) the surface tangent space $\tau_\gamma \in \widetilde{\mathrm{Gr}}(2, \mathbb{R}^3)$ with the surface normal $\nu_\gamma \in S^2$, μ_1 is given by

$$\mu_1(\varphi) = \int_\gamma \varphi(x, \nu_\gamma(x))\,\mathrm{d}\mathcal{H}^2(x) \qquad \text{for all } \varphi \in C_0^\infty(\Omega \times S^2) ,$$

and the corresponding lifting of an image gradient is defined as follows, simply writing τ for the mapping $(x, \tau) \mapsto \tau$.

Definition 7.18 (Image Gradient Lifting) Given an image $u \in \mathrm{BV}(\Omega)$, a measure $\mu_1 \in \mathrm{rca}_+(\Omega \times S^2)$ is called a *lifting* of Du if

$$Du = [\nu\mu_1](\cdot, S^2) .$$

To derive the regularization functional, we consider a closed oriented piecewise C^2 surface $\gamma \subset \Omega$ with oriented (potentially nonplanar) faces $F_1, \dots, F_N$, corresponding normals $\nu_{F_i} = \nu_\gamma$, and edge set Γ. Further, on each F_i consider the (spatially varying) principal curvatures κ_1, κ_2 and associated orthonormal principal tangent vectors τ_1, τ_2 to γ. Note that for any $\hat{\psi} \in C^1(\Omega; \mathbb{R}^3)$ and $x \in \gamma \setminus \Gamma$ we can write

$$\mathrm{curl}\hat{\psi}(x)\cdot\nu_\gamma(x) = \mathrm{curl}\hat{\psi}(x)\cdot(\tau_1(x)\times\tau_2(x)) = \tau_2(x)\cdot\partial_{\tau_1(x)}\hat{\psi}(x) - \tau_1(x)\cdot\partial_{\tau_2(x)}\hat{\psi}(x) ,$$

where ∂_τ denotes the directional derivative in direction τ. Thus, for the choice $\hat{\psi}(x) = \psi(x, \nu_\gamma(x))$ with $\psi \in C^1(\Omega \times S^2; \mathbb{R}^3)$ we obtain

$$\begin{aligned}\mathrm{curl}\hat{\psi} \cdot \nu_\gamma &= \tau_2 \cdot (\tau_1 \cdot \nabla_x \psi) - \tau_1 \cdot (\tau_2 \cdot \nabla_x \psi) + \tau_2 \cdot \tfrac{\partial\psi}{\partial\nu}(\partial_{\tau_1}\nu_\gamma) - \tau_1 \cdot \tfrac{\partial\psi}{\partial\nu}(\partial_{\tau_2}\nu_\gamma) \\ &= \mathrm{curl}_x\psi\cdot\nu_\gamma + \tau_2\cdot\tfrac{\partial\psi}{\partial\nu}(\partial_{\tau_1}\nu_\gamma) - \tau_1\cdot\tfrac{\partial\psi}{\partial\nu}(\partial_{\tau_2}\nu_\gamma) = \mathrm{curl}_x\psi\cdot\nu_\gamma + \kappa_1\tau_2\cdot\tfrac{\partial\psi}{\partial\nu}(\tau_1) - \kappa_2\tau_1\cdot\tfrac{\partial\psi}{\partial\nu}(\tau_2),\end{aligned}$$

where we used $\partial_{\tau_i} \nu_\gamma = \kappa_i \tau_i$ (above, ψ and its derivatives are always evaluated at points $(x, \nu_\gamma(x))$). We now test the surface lifting μ_1 with $\nu \cdot \mathrm{curl}_x \psi(x, \nu)$ for $\psi \in C_0^\infty(\Omega \times S^2; \mathbb{R}^3)$, the ν-component of the curl in the x-variables. Via Stokes' theorem, this yields

$$\begin{aligned}
\int_{\Omega\times S^2} \nu\cdot\mathrm{curl}_x\psi(x,\nu)\,\mathrm{d}\mu_1(x,\nu) =& \sum_{i=1}^N \int_{F_i} \nu_{F_i}\cdot\mathrm{curl}\psi(\cdot,\nu_{F_i}) + \kappa_2\tau_1\cdot\tfrac{\partial\psi}{\partial\nu}(\tau_2) \\
&\qquad -\kappa_1\tau_2\cdot\tfrac{\partial\psi}{\partial\nu}(\tau_1)\,\mathrm{d}\mathcal{H}^2 \\
= \sum_{i=1}^N \int_{\partial F_i} \tau_{\partial F_i}\cdot\psi(\cdot,\nu_{F_i})\,\mathrm{d}\mathcal{H}^1 &+ \sum_{i=1}^N \int_{F_i} \kappa_2\tau_1\cdot\tfrac{\partial\psi}{\partial\nu}(\tau_2) - \kappa_1\tau_2\cdot\tfrac{\partial\psi}{\partial\nu}(\tau_1)\,\mathrm{d}\mathcal{H}^2 \\
= \sum_{i<j} \int_{F_i\cap F_j} \tau_{\partial F_i}\cdot[\psi(\cdot,\nu_{F_i}) - \psi(\cdot,\nu_{F_j})]\,\mathrm{d}\mathcal{H}^1 &+ \sum_{i=1}^N \int_{F_i} \kappa_2\tau_1\cdot\tfrac{\partial\psi}{\partial\nu}(\tau_2) - \kappa_1\tau_2\cdot\tfrac{\partial\psi}{\partial\nu}(\tau_1)\,\mathrm{d}\mathcal{H}^2,
\end{aligned} \tag{7.3}$$

where $\tau_{\partial F_i}$ is the unit tangent to ∂F_i consistent with the orientation of F_i. For planar faces, the second sum vanishes, and taking the supremum over all ψ with $-\rho(\nu_1, \nu_2) \leq [\psi(x, \nu_1) - \psi(x, \nu_2)]\cdot(\nu_1 \times \nu_2) \leq \rho(\nu_1, \nu_2)$ for all x, ν_1, ν_2 will yield the desired $R(\gamma) = \int_\Gamma \rho(\nu_\gamma^-(x), \nu_\gamma^+(x))\,\mathrm{d}\mathcal{H}^1(x)$, hence the following definition.

Definition 7.19 (Jump Set Calibration Regularization Functional) The *jump set calibration regularization functional* for a 3D image $u \in \mathrm{BV}(\Omega)$ is given by

$$R_\rho(u) = \inf\left\{ \int_{\Omega\times S^2} \alpha\,\mathrm{d}\mu_1 + T_\rho(\mu_1) \,\middle|\, \mu_1 \text{ is a lifting of } Du \right\} \quad \text{with}$$

$$T_\rho(\mu_1) = \sup\left\{ \int_{\Omega\times S^2} \nu\cdot\mathrm{curl}_x\psi(x,\nu)\,\mathrm{d}\mu_1(x,\nu) \,\middle|\, \psi \in M_\rho \right\} \quad \text{and}$$

$M_\rho = \big\{\psi \in C_0^\infty(\Omega \times S^2; \mathbb{R}^3) \,\big|\, -\rho(\nu_1, \nu_2) \leq [\psi(x, \nu_1) - \psi(x, \nu_2)]\cdot(\nu_1 \times \nu_2) \leq \rho(\nu_1, \nu_2)\,\forall x \in \Omega,\ \nu_1, \nu_2 \in S^2\big\}$.

In the same way as in two space-dimensions, the following regularization properties are obtained.

Theorem 7.20 (Regularization Properties) *R_ρ is convex, positively one-homogeneous, and lower semi-continuous on $L^1(\Omega)$. Further, $R_\rho(u) \geq \alpha|u|_{\mathrm{TV}}$.*

Again, taking ρ as the discrete metric ρ_0 or the geodesic metric ρ_1 on S^2 with corresponding (slightly simplified) sets

$$M_{\rho_0} = \left\{ \psi \in C_0^\infty(\Omega\times S^2; \mathbb{R}^3) \,\middle|\, \psi(x,\nu_1)\cdot\nu_2 \leq \tfrac{1}{2}\,\forall x \in \Omega,\ \nu_1, \nu_2 \in S^2,\ \nu_1 \perp \nu_2 \right\},$$

$$M_{\rho_1} = \Big\{\psi \in C_0^\infty(\Omega \times S^2; \mathbb{R}^3) \,\Big|\, -1 \le \tfrac{\nu_2\cdot\nabla_\nu(\psi(x,\nu_1)\cdot(\nu_1\times\nu_2))}{\partial\nu} \le 1 \;\forall x \in \Omega,\; \nu_1, \nu_2 \in S^2,\; \nu_1 \perp \nu_2\Big\}$$

yields two extremal cases, one measuring the $\mathcal{H}^1$-measure of all creases, the other measuring the total absolute curvature.

Theorem 7.21 (Singular Curvature Penalization for Surfaces) *Let $\gamma \subset \Omega$ be a piecewise C^2 surface and μ_1 the corresponding oriented varifold. Then*

$$T_{\rho_0}(\mu_1) = \mathcal{H}^1(\Gamma) \textit{ if } \kappa_\gamma = 0 \textit{ and } T_{\rho_0}(\mu_1) = \infty \textit{ else,}$$

$$T_{\rho_1}(\mu_1) = \int_\gamma |\kappa_\gamma|_1 \,\mathrm{d}\mathcal{H}^2 + \int_\Gamma |\theta(x)| \,\mathrm{d}\mathcal{H}^1(x)\,,$$

where $\Gamma \subset \gamma$ is the set of creases (where ν_γ is discontinuous), θ is the change of the surface normal across Γ, and $|\kappa_\gamma|_1$ is the nuclear norm of the second fundamental form of γ, i.e., the sum of the unsigned principal curvatures.

7.4.2 Derivation of Theorem 7.21

Theorem 7.21 directly follows from the below small Lemmas. As (7.3) tells us, $\int_{\Omega\times S^2} \nu \cdot \mathrm{curl}_x\psi(x,\nu)\,\mathrm{d}\mu_1(x,\nu)$ even makes sense for (and is continuous in) $\psi \in C(\Omega; C^1(S^2; \mathbb{R}^3))$, i.e., continuous functions that are once differentiable in the second argument. By density we may replace $C_0^\infty(\Omega \times S^2; \mathbb{R}^3)$ with $C_0(\Omega; C^1(S^2; \mathbb{R}^3))$ in the definition of M_ρ, which below we will do without explicit mention. Further, we will use two test functions, one for the edges and one for the faces.

Test Function for Edges Let $\delta, \epsilon > 0$. We use the notation of (7.3) and abbreviate $E_{ij} = F_i \cap F_j$. Since γ is piecewise C^2 we may assume the E_{ij} to be C^2 as well (if the E_{ij} contain kink or cusp singularities, we just split them up). We set $E_{ij}^\delta = E_{ij} \setminus B_\delta(\partial E_{ij})$ with $B_\delta(\partial E_{ij})$ the δ-neighborhood of the endpoints of E_{ij} (see Fig. 7.8 left). Further, we let $\eta_{ij}^\delta \in C_0^\infty(\Omega; [0,1])$ be a smooth cutoff function with support on $B_{\delta^p}(E_{ij}^\delta)$ and $\eta_{ij}^\delta = 1$ on E_{ij}^δ with $p > 2$. For δ small enough and for sufficiently high p, all η_{ij}^δ have disjoint support. Finally, let $\psi_i^\epsilon \in M_\rho$ be a mollification of the function $F_i \times S^2 \ni (x,\nu) \mapsto \rho(\nu_{F_i}(x), \nu)$, smoothly extended to $\Omega \times S^2$. The mollification shall be such that $|\psi_i^\epsilon(x, \nu_{F_k}(x)) - \rho(\nu_{F_i}(x), \nu_{F_k}(x))| < \epsilon$ for $x \in E_{ij}$ and $k = i, j$ (which can for instance be achieved by a sufficiently short time step of heat flow). Then $\psi^{\delta,\epsilon}(x,\nu) = \sum_{E_{ij}} \eta_{ij}^\delta(x)\psi_i^\epsilon(x,\nu)\tau_{E_{ij}}(x)$ lies in M_ρ.

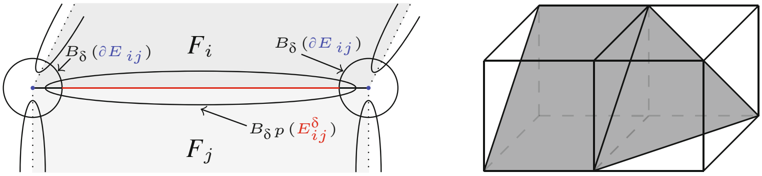

$B_\delta(\partial E_{ij})$
F_i
$B_\delta(\partial E_{ij})$
$B_{\delta p}(E^\delta_{ij})$
F_j

Fig. 7.8 Left: Neighborhoods of the joint edge $E_{ij} = F_i \cap F_j$ of the surface patches F_i and F_j. Right: Two exemplary surface patches connecting to each other

Test Function for Faces Without loss of generality we may assume $\kappa_1, \kappa_2, \tau_1, \tau_2$ to depend continuously on $x \in F_i$ for all i (otherwise we simply introduce artificial edges at umbilics). For $\lambda > 0$, $\eta \in C_0(\Omega; [0, 1])$ and $\zeta \in C^\infty(\mathbb{R}^3; \mathbb{R}^3)$ with $\zeta(0) = 0$ and $\nabla\zeta(0) = I$ we set

$$\psi_{\lambda,\eta,\zeta}(x, \nu) = \eta(x)\zeta\,(\lambda M(x)\nu)\ , \quad M = \operatorname{sgn}(\kappa_2)\tau_1 \otimes \tau_2 - \operatorname{sgn}(\kappa_1)\tau_2 \otimes \tau_1\ ,$$

where for $x \notin \gamma$ the quantities $\tau_1(x), \tau_2(x), \kappa_1(x), \kappa_2(x)$ shall be the respective quantities after projecting x orthogonally onto γ. The latter is well-defined in a sufficiently small neighborhood of γ and away from the edge set Γ so that $\psi_{\lambda,\eta,\zeta}$ is well-defined if the support of η is correspondingly chosen. We have $\psi_{\lambda,\eta,\zeta} \in C_0(\Omega; C^1(S^2; \mathbb{R}^3))$. Using $M(x)\nu_\gamma(x) = 0$ and thus $\psi(x, \nu_\gamma(x)) = 0$ and $\frac{\partial\psi(x,\nu_\gamma(x))}{\partial\nu} = \eta(x)\lambda M(x)$ for $x \in \gamma$, (7.3) simplifies to

$$\int_{\Omega\times S^2} \nu\cdot\operatorname{curl}_x \psi_{\lambda,\eta,\zeta}(x, \nu)\,\mathrm{d}\mu_1(x, \nu) = \lambda\sum_{i=1}^{N}\int_{F_i} \eta(x)[|\kappa_2(x)| + |\kappa_1(x)|]\,\mathrm{d}\mathcal{H}^2(x)\,. \tag{7.4}$$

Lemma 7.22 *Let ρ be a lower semi-continuous metric on S^2 and μ_1 the lifting of a closed oriented polyhedron $\gamma \subset \Omega$. Then*

$$T_\rho(\mu_1) = \sum_{\textit{edges } E \textit{ of } \gamma} \rho(\nu_\gamma^-(E), \nu_\gamma^+(E)) \cdot \mathcal{H}^1(E)\,.$$

Proof For planar faces, the second sum in (7.3) vanishes, thus implying

$$T_\rho(\mu_1) \le \sum_{1\le i<j\le N} \int_{E_{ij}} \rho(\nu_{F_i}, \nu_{F_j})\,\mathrm{d}\mathcal{H}^1 = \sum_{\text{edges } E \text{ of } \gamma} \rho(\nu_\gamma^-(E), \nu_\gamma^+(E)) \cdot \mathcal{H}^1(E)\,.$$

The opposite inequality follows from (7.3) with the test functions for edges,

$$T_\rho(\mu_1) \ge \int_{\Omega\times S^2} \operatorname{curl}_x \psi^{\delta,\epsilon}(x, \nu)\,\mathrm{d}\mu_1(x, \nu)$$

$$=\sum_{E_{ij}}\int_{E_{ij}}\tau_{E_{ij}}\cdot[\psi^{\delta,\epsilon}(x,\nu_{F_i})-\psi^{\delta,\epsilon}(x,\nu_{F_j})]\,\mathrm{d}\mathcal{H}^1(x)\geq\sum_{E_{ij}}\mathcal{H}^1(E_{ij}^\delta)[\rho(\nu_{F_i},\nu_{F_j})-2\epsilon],$$

which converges to the desired expression for $\epsilon,\delta\to 0$. □

Lemma 7.23 *Let μ_1 be the lifting of a nonpolyhedral piecewise C^2 surface $\gamma\subset\Omega$, then $T_{\rho_0}(\mu_1)=\infty$.*

Proof Pick a nonplanar face $F_i\subset\gamma$ and a small open ball $U\subset F_i$ with nonzero second fundamental form κ_γ. Choosing $\zeta(w)=\frac{\arctan|w|}{\pi|w|}w$ and η to be a nonzero cutoff function with support in U, we have $\psi_{\lambda,\eta,\zeta}\in M_{\rho_0}$. Thus, (7.4) implies $T_{\rho_0}(\mu_1)\geq\lambda\int_U\eta[|\kappa_1|+|\kappa_2|]\,\mathrm{d}\mathcal{H}^2$, which diverges for $\lambda\to\infty$. □

Lemma 7.24 *Under the assumptions of Theorem 7.21, $T_{\rho_1}(\mu_1)=\int_\gamma|\kappa_\gamma|_1\,\mathrm{d}\mathcal{H}^2+\int_\Gamma|\theta(x)|\,\mathrm{d}\mathcal{H}^1(x)$.*

Proof The definition of M_{ρ_1} together with (7.3) immediately implies that the right-hand side is an upper bound on $T_{\rho_1}(\mu_1)$. To show that it is attained, set $\psi=\psi_{1,\eta^\delta,\mathrm{id}}+\psi^{\delta,\epsilon}$ for $\eta^\delta:\Omega\to[0,1]$ a smooth cutoff function with $\eta=1$ on the δ^2-neighborhood $B_{\delta^2}(\gamma\setminus B_{2\delta}(\Gamma))$ and support in $B_{2\delta^2}(\gamma\setminus B_{2\delta}(\Gamma))$. Then the two summands have disjoint support, and it is readily checked that $\psi\in M_{\rho_1}$ (if δ is small enough). Now (7.4) implies

$$\int_{\Omega\times S^2}\nu\cdot\mathrm{curl}_x\psi_{1,\eta^\delta,\mathrm{id}}(x,\nu)\,\mathrm{d}\mu_1(x,\nu)=\sum_{i=1}^N\int_{F_i}\eta^\delta[|\kappa_2|+|\kappa_1|]\,\mathrm{d}\mathcal{H}^2\,,$$

while (7.3) can be used to estimate

$$\int_{\Omega\times S^2}\nu\cdot\mathrm{curl}_x\psi^{\delta,\epsilon}(x,\nu)\,\mathrm{d}\mu_1(x,\nu)$$
$$\geq\sum_{i<j}\int_{F_i\cap F_j}\rho(\nu_{F_i},\nu_{F_j})-2\epsilon\,\mathrm{d}\mathcal{H}^1-\sum_{i=1}^N\int_{F_i\cap B_{\delta p}(\partial F_i)}|\kappa_2|+|\kappa_1|\,\mathrm{d}\mathcal{H}^2\,.$$

Now $T_{\rho_1}(\mu_1)\geq\int_{\Omega\times S^2}\nu\cdot\mathrm{curl}_x\psi(x,\nu)\,\mathrm{d}\mu_1(x,\nu)$, which is no smaller than the sum of the above right-hand sides and for $\epsilon,\delta\to 0$ converges to the desired expression. □

7.4.3 Adaptive Discretization with Surface Measures

Since the image gradient liftings will be concentrated on three- or even two-dimensional hypersurfaces in the five-dimensional space $\Omega\times S^2$, an adaptive discretization approach seems indispensable. We shall thus extend the approach of Sect. 7.3.4. We discretize μ_1 as linear combination of liftings μ_1^i of surface patches

$\gamma_i \subset \Omega$, $i = 1, \ldots, N$, i.e., $\mu_1^i(\varphi) = \int_{\gamma_i} \varphi(x, \nu_{\gamma_i}(x))\,\mathrm{d}\mathcal{H}^2(x)$ for smooth test functions φ. Considering a grid of cuboid voxels on Ω, for the γ_i we choose the intersections of these voxels with planes of different normals $\nu_i \in S^2$. The choice of appropriate discrete normals ν_i is much more complicated than in Sect. 7.3.4: the surface patch sides have to be compatible so as to be able to connect the patches to closed surfaces (cf. Figs. 7.8 right and 7.9 left). We choose the normals such that the corresponding planes contain at least three grid points of the voxel grid. Even then, a large class of surfaces cannot be built from these patches, for instance, a cylinder is not allowed since it would require half a voxel side as a surface patch (Fig. 7.9 right). To avoid introducing such additional surface patches, we only introduce them virtually. This leaves the discretization of the dual variables invariant and only changes the discrete bilinear operator in the definition of T_ρ. As a consequence, μ_1 can now describe a discontinuous, non-closed surface in which the missing pieces are the virtual surface patches, thereby increasing the class of representable surfaces.

As in Sect. 7.3.4, the underlying voxel grid and the number of considered discrete normals ν_i in a voxel can be adaptively refined, using the same refinement criteria. Corresponding segmentation results are shown in Fig. 7.10.

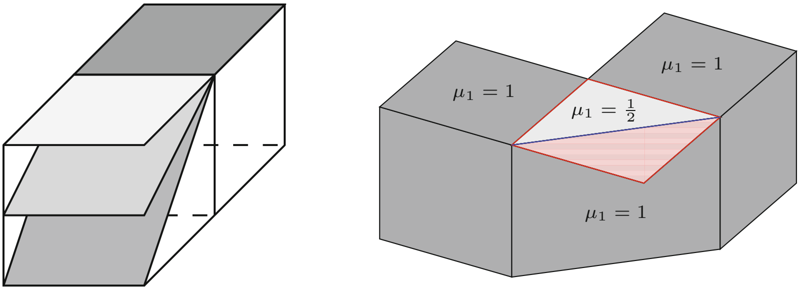


Fig. 7.9 Challenges when discretizing surfaces by intersections of planes with voxels. Left: Surface patches with an edge aligned with the voxel grid are rectangular and consequently can only connect to surface patches of the same class. Right: The top of a cylinder would require half the red surface patch

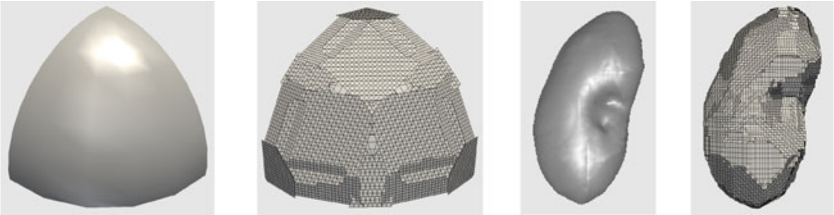


Fig. 7.10 Binary segmentation of a 3D image f by minimizing $\lambda \int_\Omega (\frac{1}{2} - f) u \,\mathrm{d}x + R_{\rho_0}(u)$ with $\alpha = \beta = 1$ among images $u : \Omega \to [0, 1]$, left for f the characteristic function of an octant of a ball, right for the characteristic function of a kidney (data courtesy Werner Bautz, radiology department at the university hospital Erlangen, Germany)

7.5 Summary

We provided an overview over convex lifting based models for curvature regularization in 2D image processing, focusing on four different modeling approaches (three of which turned out to be equivalent). One of the models was extended to 3D image processing. We also discussed different corresponding discretization schemes, which need to balance computational effort, sharpness of the results, and availability of a sufficiently large number of level line (or level surface) directions. A novel adaptive line (or surface) measure discretization allowed to push the limits of this balance to higher spatial and directional resolution.

Acknowledgements This work has been supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), Cells-in-Motion Cluster of Excellence (EXC 1003—CiM), University of Münster, Germany, by Germany's Excellence Strategy—EXC 2044—, Mathematics Münster: Dynamics—Geometry—Structure, and by the Alfried Krupp Prize for Young University Teachers awarded by the Alfried Krupp von Bohlen und Halbach-Stiftung.

References

1. Allard, W.K.: On the first variation of a varifold. Ann. Math. (2) **95**, 417–491 (1972)
2. Almgren Jr., F.J.: Existence and regularity almost everywhere of solutions to elliptic variational problems among surfaces of varying topological type and singularity structure. Ann. Math. (2) **87**, 321–391 (1968)
3. Ambrosio, L., Masnou, S.: A direct variational approach to a problem arising in image reconstruction. Interfaces Free Bound. **5**, 63–81 (2003)
4. Anzellotti, G., Serapioni, R., Tamanini, I.: Curvatures, functionals, currents. Indiana Univ. Math. J. **39**(3), 617–669 (1990)
5. Bae, E., Tai, X.C., Zhu, W.: Augmented Lagrangian method for an Euler's elastica based segmentation model that promotes convex contours. Inverse Prob. Imaging **11**(1), 1–23 (2017)
6. Ballester, C., Bertalmio, M., Caselles, V., Sapiro, G., Verdera, J.: Filling-in by joint interpolation of vector fields and gray levels. IEEE Trans. Image Process. **10**(8), 1200–1211 (2001)
7. Ballester, C., Caselles, V., Verdera, J.: Disocclusion by joint interpolation of vector fields and gray levels. Multiscale Model. Simul. **2**(1), 80–123 (2003)
8. Balzani, N., Rumpf, M.: A nested variational time discretization for parametric Willmore flow. Interfaces Free Bound. **14**(4), 431–454 (2012)
9. Bellettini, G., Mugnai, L.: Characterization and representation of the lower semicontinuous envelope of the elastica functional. Ann. Inst. Henri Poincare (C) Non Linear Anal. **21**(6), 839–880 (2004)
10. Bellettini, G., Dal Maso, G., Paolini, M.: Semicontinuity and relaxation properties of a curvature depending functional in 2d. Ann. Sc. Norm. Super. Pisa Cl. Sci. (4) **20**(2), 247–297 (1993)
11. Berkels, B., Effland, A., Rumpf, M.: A posteriori error control for the binary Mumford-Shah model. Math. Comput. **86**(306), 1769–1791 (2016)
12. Boscain, U., Chertovskih, R.A., Gauthier, J.P., Remizov, A.O.: Hypoelliptic diffusion and human vision: a semidiscrete new twist. SIAM J. Imaging Sci. **7**(2), 669–695 (2014)
13. Bredies, K., Pock, T., Wirth, B.: Convex relaxation of a class of vertex penalizing functionals. J. Math. Imaging Vis. **47**(3), 278–302 (2012)

14. Bredies, K., Pock, T., Wirth, B.: A convex, lower semicontinuous approximation of Euler's elastica energy. SIAM J. Math. Anal. **47**(1), 566–613 (2015)
15. Bretin, E., Masnou, S., Oudet, É.: Phase-field approximations of the Willmore functional and flow. Numer. Math. **131**(1), 115–171 (2014)
16. Chambolle, A.: Convex representation for lower semicontinuous envelopes of functionals in L1. J. Convex Anal. **8**(1), 149–170 (2001)
17. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vis. **40**(1), 120–145 (2011)
18. Chambolle, A., Pock, T.: Total roto-translational variation. Numer. Math. **142**, 611–666 (2019)
19. Citti, G., Sarti, A.: A cortical based model of perceptual completion in the roto-translation space. J. Math. Imaging Vis. **24**(3), 307–326 (2006)
20. Citti, G., Franceschiello, B., Sanguinetti, G., Sarti, A.: Sub-Riemannian mean curvature flow for image processing. SIAM J. Imaging Sci. **9**(1), 212–237 (2016)
21. Delladio, S.: Minimizing functionals depending on surfaces and their curvatures: a class of variational problems in the setting of generalized Gauss graphs. Pac. J. Math. **179**(2), 301–323 (1997)
22. Delladio, S.: Special generalized Gauss graphs and their application to minimization of functional involving curvatures. J. Reine Angew. Math. **1997**(486), 17–44 (1997)
23. Droske, M., Rumpf, M.: A level set formulation for Willmore flow. Interfaces Free Bound. **6**, 361–378 (2004)
24. El-Zehiry, N.Y., Grady, L.: Contrast driven elastica for image segmentation. IEEE Trans. Image Process. **25**(6), 2508–2518 (2016)
25. Euler, L.: Methodus inveniendi lineas curvas maximi minimive proprietate gaudentes. Lausanne (1744)
26. Hubel, D.H., Wiesel, T.N.: Receptive fields of single neurones in the cat's striate cortex. J. Physiol. **148**(3), 574–591 (1959)
27. Hutchinson, J.: Second fundamental form for varifolds and the existence of surfaces minimising curvature. Indiana Univ. Math. J. **35**(1), 45 (1986)
28. Kanizsa, G.: Organization in Vision: Essays on Gestalt Perception. Praeger, New York (1979)
29. Mantegazza, C.: Curvature varifolds with boundary. J. Differ. Geom. **43**(4), 807–843 (1996)
30. Masnou, S.: Disocclusion: a variational approach using level lines. IEEE Trans. Image Process. **11**(2), 68–76 (2002)
31. Mekes, B.: Konvexe Krümmungsregularisierung für Linienmaße in drei Dimensionen. Master's thesis, University of Münster, 681–682 (2016). https://www.uni-muenster.de/AMM/wirth/theses/
32. MOSEK ApS: The MOSEK optimization toolbox for MATLAB manual. Version 9.0. (2019). http://docs.mosek.com/9.0/toolbox/index.html
33. Natterer, F.: The Mathematics of Computerized Tomography. B. G. Teubner/Wiley, Stuttgart/Chichester (1986)
34. Nitzberg, M., Mumford, D., Shiota, T.: Filtering, Segmentation and Depth. Springer, Berlin (1993)
35. Prandi, D., Boscain, U., Gauthier, J.P.: Image processing in the semidiscrete group of rototranslations. In: Lecture Notes in Computer Science, pp. 627–634. Springer International Publishing, Cham (2015)
36. Raviart, P.A., Thomas, J.M.: A mixed finite element method for 2-nd order elliptic problems. In: Lecture Notes in Mathematics, pp. 292–315. Springer, Berlin (1977)
37. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D Nonlinear Phenom. **60**(1–4), 259–268 (1992)
38. Sarti, A., Citti, G., Petitot, J.: The symplectic structure of the primary visual cortex. Biol. Cybern. **98**(1), 33–48 (2008)
39. Schoenemann, T., Kahl, F., Masnou, S., Cremers, D.: A linear framework for region-based image segmentation and inpainting involving curvature penalization. Int. J. Comput. Vis. **99**(1), 53–68 (2012)

40. Schoenemann, T., Masnou, S., Cremers, D.: On a linear programming approach to the discrete Willmore boundary value problem and generalizations. In: Curves and Surfaces, pp. 629–646. Springer, Berlin (2012)
41. Sharma, U., Duits, R.: Left-invariant evolutions of wavelet transforms on the similitude group. Appl. Comput. Harmon. Anal. **39**(1), 110–137 (2015)
42. Shen, J., Kang, S.H., Chan, T.F.: Euler's elastica and curvature-based inpainting. SIAM J. Appl. Math. **63**(2), 564–592 (2003)
43. Tai, X.C., Hahn, J., Chung, G.J.: A fast algorithm for Euler's elastica model using augmented Lagrangian method. SIAM J. Imaging Sci. **4**(1), 313–344 (2011)
44. Tinius, D.: Discretization via line measures for a curvature regularization framework. Master's thesis, University of Münster, 711–712 (2015). https://www.uni-muenster.de/AMM/wirth/theses/
45. Wertheimer, M.: Untersuchungen zur Lehre von der Gestalt. Psychol. Forsch. **4**, 301–350 (1923)
46. Willmore, T.J.: Riemannian Geometry. Oxford University Press, Oxford (1996)
47. Yashtini, M., Kang, S.H.: Alternating direction method of multiplier for Euler's elastica-based denoising. In: Lecture Notes in Computer Science, pp. 690–701. Springer International Publishing, Cham (2015)
48. Yashtini, M., Kang, S.H.: A fast relaxed normal two split method and an effective weighted TV approach for Euler's elastica image inpainting. SIAM J. Imaging Sci. **9**(4), 1552–1581 (2016)
49. Young, L.C.: Generalized surfaces in the calculus of variations. Ann. Math. (2) **43**, 84–103 (1942)
50. Zhu, W., Tai, X.C., Chan, T.: Augmented Lagrangian method for a mean curvature based image denoising model. Inverse Prob. Imaging **7**(4), 1409–1432 (2013)

Chapter 8
Assignment Flows

Christoph Schnörr

Contents

Abstract Assignment flows comprise basic dynamical systems for modeling data labeling and related machine learning tasks in supervised and unsupervised scenarios. They provide adaptive time-variant extensions of established discrete graphical models and a basis for the design and better mathematical understanding of hierarchical networks, using methods from information (differential) geometry, geometric numerical integration, statistical inference, optimal transport and control. This chapter introduces the framework by means of the image labeling problem and outlines directions of current and further research.

C. Schnörr (✉)
Institute of Applied Mathematics, Heidelberg University, Heidelberg, Germany
e-mail: schnoerr@math.uni-heidelberg.de

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_8

8.1 Introduction

Let $(\mathcal{F}, d_{\mathcal{F}})$ be a metric space and $\mathcal{F}_n = \{f_i \in \mathcal{F} \colon i \in \mathcal{I}\}$ given data with $|\mathcal{I}| = n$. Assume that a predefined set of *prototypes* $\mathcal{F}_* = \{f_j^* \in \mathcal{F} \colon j \in \mathcal{J}\}$ is given with $|\mathcal{J}| = c$. *Data labeling* denotes the *assignment*

$$j \to i, \qquad f_j^* \to f_i \tag{8.1}$$

of a single *prototype* $f_j^* \in \mathcal{F}_*$ to each data point $f_i \in \mathcal{F}_n$. Adopting the common model assumption that $\mathcal{F}_n$ is a *finite sample set* generated by an unknown underlying probability distribution $\mu_{\mathcal{F}}$, the quality of assignments may be defined via the *quantization* of $\mu_{\mathcal{F}}$ in terms of the selected (assigned) prototypes and by corresponding optimality criteria of information theory [14, 16, 18]. The assignment of indices $j \to i$ induces a *partition* (*classification*) of $\mathcal{F}_n$. Accordingly, depending on the research area, prototypes $f_j^* \in \mathcal{F}_*$ are also called *class representatives*, *feature dictionary*, *codebook* or simply *labels*, and we will use interchangeably these terms throughout this chapter.

What makes the data labeling problem challenging is that *context-sensitive* label assignments are required: $\mathcal{I}$ forms the vertex set of a given undirected graph $\mathcal{G} = (\mathcal{I}, \mathcal{E})$ which defines a relation $\mathcal{E} \subset \mathcal{I} \times \mathcal{I}$ and neighborhoods

$$\mathcal{N}_i = \{k \in \mathcal{I} \colon ik \in \mathcal{E}\} \cup \{i\}, \tag{8.2}$$

where ik is a shorthand for the unordered pair (edge) $(i, k) = (k, i)$. Indices $i \in \mathcal{I}$ frequently index positions $x_i \in \Omega \subset \mathbb{R}^d$ in a Euclidean domain,[1] and $k \in \mathcal{N}_i$ indicates a small Euclidean distance $\|x_i - x_k\|$. A basic example are image features $\mathcal{F}_n$ extracted from raw pixel data on a image grid graph $\mathcal{G}$ and the corresponding *image labeling* problem.

In such situations, it is plausible to assume that $k \in \mathcal{N}_i$ implies that the *same* label is assigned to both i and k more frequently than different labels, which explains the success of the total variation measure of 'piecewise image homogeneity' for image denoising [45]. Yet, this assumption falls short of the enormous complexity of *assignment relations* that define *natural real* image structure across the scales up to a semantic level. While information theory clearly says that *joint* assignments are more appropriate than individual assignments for the quantization of complex data sources $\mu_{\mathcal{F}}$ [14], how to accomplish this task in a mathematically and statistically satisfying way using algorithms that are computationally feasible, has remained an unsolved problem.

The aforementioned data encoding-decoding tasks are nowadays mainly performed using deep networks, due to their striking empirical performance in benchmark tests across many disciplines like, e.g., in image labeling [31]. However,

[1] This includes spatio-temporal data—like e.g. videos—observed at points $(t_i, x_i) \in [0, T] \times \Omega \subset \mathbb{R} \times \mathbb{R}^d$ in time and space.

this rapid development during recent years has not improved our mathematical understanding in the same way, so far [15]. The 'black box' behavior of deep networks and systematic failures [2] are worrying not only researchers from mathematics and scientific computing, but also industrial partners in connection with safety-critical applications.

In this context, *assignment flows* are introduced in this chapter as an attempt to extend discrete *graphical models* in a systematic way, which defined the prevailing framework for data modeling, inference and learning during the last three decades [17, 30, 33, 36, 51]. Regarding inference *algorithms* using discrete graphical models, we refer to [28] for an assessment of the state of the art.

Assignment flows are *smooth dynamical systems* defined using *information geometry* [1, 4]. Elementary statistical manifolds [32] provide both a target space for *data embedding* and a *state space* on which the assignment flow evolves in order to determine a data labeling. Corresponding vector fields are parametrized and thus enable to learn the *adaptivity* of regularized label assignments within neighborhoods (8.2), rather than parameters of a *fixed* regularizer as with graphical models or traditional variational approaches to inverse problems. *Smoothness* and *modular compositional design* yield efficient algorithms based on numerical *geometric integration* and enable to switch seamlessly between supervised and unsupervised scenarios within a single framework.

The assignment flow for *supervised* data labeling is introduced in Sect. 8.2. *Unsupervised* scenarios involving *label evolution* and *learning labels from data* are discussed in Sect. 8.3. Section 8.4 reports first steps towards *learning* (*estimating*) adaptivity parameters of the assignment flow via optimal control. Section 8.5 outlines current and future work that will be undertaken along this research direction, in order to contribute to a better mathematical understanding of the representation and inference of natural image structure.

This chapter focuses on the basic mathematical ingredients and the discussion of corresponding modeling aspects. We refer to [3, 22, 23, 49, 56–59] for more detailed expositions of the respective topics, numerical experiments and a discussion of related work. Regarding the latter, we include few comments on historical developments as Remarks 8.1 and 8.2 on page 241.

Basic Notation We set $n = |\mathcal{I}|$ (number of vertices), $c = |\mathcal{J}|$ (number of classes resp. labels) and $[m] = \{1, 2, \dots, m\}$ for $m \in \mathbb{N}$. $\mathbb{1} = (1, 1, \dots, 1)^{\top}$ denotes the one-vector whose dimension depends on the context. $\langle \cdot, \cdot \rangle$ denotes the Euclidean inner product. The probability simplex of dimension $c - 1$ is

$$\Delta_c = \Big\{ p \in \mathbb{R}^c_+ \colon \langle \mathbb{1}, p \rangle = \sum_{j \in [c]} p^j = 1 \Big\}. \tag{8.3}$$

It is the convex hull of its vertices (extreme points) which are the unit vectors $e_1 = (1, 0, \dots, 0)^{\top}, \dots, e_c = (0, 0, \dots, 0, 1)^{\top}$. The expectation with respect to a distribution $p \in \Delta_c$ is denoted by

$$\mathbb{E}_p[q] = \langle p, q\rangle, \qquad q \in \mathbb{R}^c. \tag{8.4}$$

$I = \mathrm{Diag}(\mathbb{1})$ denotes the identity matrix.

Inequalities between vectors $\mathbb{R}^c \ni p > 0$ hold for each component, $p^1 > 0, \dots, p^c > 0$. Likewise, the exponential function and the logarithm apply to each component of the argument vector,

$$e^p = (e^{p^1}, \dots, e^{p^c})^\top, \qquad \log p = (\log p^1, \dots, \log p^c)^\top, \tag{8.5}$$

and componentwise multiplication and subdivision are simply written as

$$uv = (u^1 v^2, \dots, u^c v^c)^\top, \qquad \frac{v}{p} = \Big(\frac{v^1}{p^1}, \dots, \frac{v^c}{p^c}\Big)^\top, \qquad u, v \in \mathbb{R}^c, \quad p > 0. \tag{8.6}$$

It will be convenient to write the exponential function with large expressions as argument in the form $e^p = \exp(p)$. The latter expression should not be confused with the exponential map $\exp_p$ defined by (8.23) that always involves a subscript. Likewise, log always means the logarithm function and should not be confused with the inverse exponential maps defined by (8.23).

We write $E(\cdot)$ for specifying various objective functions in this chapter. The context disambiguates this notation.

8.2 The Assignment Flow for Supervised Data Labeling

We collect in Sect. 8.2.1 basic notions of information geometry [1, 4, 10, 32] that are required for introducing the assignment flow for supervised data labeling in Sect. 8.2.2. See e.g. [27, 34] regarding general differential geometry and background reading.

8.2.1 Elements of Information Geometry

We sketch a basic framework of information geometry and then consider the specific instance on which the assignment flow is based.

8.2.1.1 Dually Flat Statistical Manifolds

Information geometry is generally concerned with smoothly parametrized families of densities on some sample space $\mathcal{X}$ with open parameter set Ξ in a Euclidean space,

$$\Xi \ni \xi \mapsto p(x;\xi), \qquad x \in \mathcal{X}, \tag{8.7}$$

that are regarded as immersions into the space of all densities. Equipped with the Fisher-Rao metric g which is a unique choice due to its invariance against reparametrization,

$$(\mathcal{M}, g) \quad \text{with} \quad \mathcal{M} = \{p(\,\cdot\,;\xi) \colon \xi \in \Xi\} \tag{8.8}$$

becomes a Riemannian manifold. Let $\mathcal{X}(\mathcal{M})$ denote the space of all smooth vector fields on $\mathcal{M}$. The Riemannian (Levi-Civita) connection ∇^g is the unique affine connection being torsion-free (or symmetric) and compatible with the metric, i.e. the covariant derivative of the metric tensor

$$(\nabla^g_Z g)(X,Y) = 0 \qquad \Leftrightarrow \qquad Zg(X,Y) = g(\nabla^g_Z X, Y) + g(X, \nabla^g_Z Y), \tag{8.9}$$

vanishes for all $X, Y, Z \in \mathcal{X}(\mathcal{M})$. A key idea of information geometry is to replace ∇^g by two affine connections ∇, ∇^* that are *dual* to each other, which means that they *jointly* satisfy (8.9),

$$Zg(X,Y) = g(\nabla_Z X, Y) + g(X, \nabla^*_Z Y), \qquad \forall X, Y, Z \in \mathcal{M}(\mathcal{X}). \tag{8.10}$$

In particular, computations simplify if in addition both ∇ and ∇^* can be chosen *flat*, i.e. for either connection and every point $p_\xi \in \mathcal{M}$ there exists a chart $\mathcal{U} \subset \mathcal{M}$ and local coordinates, called *affine coordinates*, such that the coordinate vector fields are *parallel* in $\mathcal{U}$. $(\mathcal{M}, g, \nabla, \nabla^*)$ is then called a *dually flat statistical manifold.*

8.2.1.2 The Assignment Manifold

Adopting the framework above, the specific instance of $\mathcal{M}$ relevant to data labeling (classification) is

$$(\mathcal{S}, g), \qquad \mathcal{S} = \{p \in \Delta_c \colon p > 0\} \tag{8.11}$$

with sample space $\mathcal{J} = [c]$,

$$\mathbb{1}_{\mathcal{S}} = \frac{1}{c}\mathbb{1} \in \mathcal{S}, \qquad \text{(barycenter)} \tag{8.12}$$

tangent bundle $T\mathcal{S} = \mathcal{S} \times T_0$,

$$T_0 = \{v \in \mathbb{R}^c \colon \langle \mathbb{1}, v\rangle = 0\}, \tag{8.13}$$

orthogonal projection

$$\Pi_0 \colon \mathbb{R}^c \to T_0, \qquad \Pi_0 = I - \mathbb{1}_{\mathcal{S}}\mathbb{1}^\top, \tag{8.14}$$

and Fisher-Rao metric

$$g_p(u, v) = \sum_{j \in \mathcal{J}} \frac{u^j v^j}{p^j}, \quad p \in \mathcal{S}, \quad u, v \in T_0. \tag{8.15}$$

Given a smooth function $f : \mathbb{R}^c \to \mathbb{R}$ and its restriction to $\mathcal{S}$, also denoted by f, with Euclidean gradient ∂f_p,

$$\partial f_p = (\partial_1 f_p, \ldots, \partial_c f_p)^\top, \tag{8.16}$$

the Riemannian gradient reads

$$\operatorname{grad}_p f = R_p \partial f_p = p\big(\partial f_p - \mathbb{E}_p[\partial f_p]\mathbb{1}\big) \tag{8.17}$$

with the linear map

$$R_p : \mathbb{R}^c \to T_0, \qquad R_p = \operatorname{Diag}(p) - pp^\top, \qquad p \in \mathcal{S} \tag{8.18}$$

satisfying

$$R_p = R_p \Pi_0 = \Pi_0 R_p. \tag{8.19}$$

The affine connections ∇, ∇^* are flat and given by the *e-connection* and *m-connection*, respectively, where 'e' and 'm' stand for the exponential and mixture representation of distributions $p \in \mathcal{S}$ [1]. The corresponding affine coordinates are given by $\theta \in \mathbb{R}^{c-1}$ and $0 < \mu \in \mathbb{R}^{c-1}$ with $\langle \mathbb{1}, \mu \rangle < 1$ such that

$$p = p_\theta = \frac{1}{1 + \langle \mathbb{1}, e^\theta \rangle} (e^{\theta^1}, \ldots, e^{\theta^{c-1}}, 1)^\top \in \mathcal{S}, \tag{8.20a}$$

$$p = p_\mu = (\mu^1, \ldots, \mu^{c-1}, 1 - \langle \mathbb{1}, \mu \rangle)^\top \in \mathcal{S}. \tag{8.20b}$$

Choosing affine geodesics in the parameter spaces

$$\theta(t) = \theta + t\dot{\theta}, \qquad \mu(t) = \mu + t\dot{\mu}, \tag{8.21}$$

the affine e- and m-geodesics in $\mathcal{S}$ read with $p = p_\theta = p_\mu \in \mathcal{S}$

$$p_{\theta(t)} = \frac{p e^{t \frac{v}{p}}}{\langle p, e^{t \frac{v}{p}} \rangle}, \qquad v = \begin{pmatrix} \dot{\mu} \\ -\langle \mathbb{1}, \dot{\mu} \rangle \end{pmatrix} \in T_0, \tag{8.22a}$$

$$p_{\mu(t)} = p_\mu + tv, \qquad t \in [t_{\min}, t_{\max}], \tag{8.22b}$$

where t in (8.22b) has to be restricted to an interval around $0 \in [t_{\min}, t_{\max}]$, depending on μ and v, such that $p_{\mu(t)} \in \mathcal{S}$. Therefore, regarding numerical computations, it is more convenient to work with the *unconstrained* e-representation.

Accordingly, with $v \in T_0,\ p \in \mathcal{S}$, we define the exponential maps and their inverses

$$\mathrm{Exp}\colon \mathcal{S} \times T_0 \to \mathcal{S}, \qquad (p, v) \mapsto \mathrm{Exp}_p(v) = \frac{pe^{\frac{v}{p}}}{\langle p, e^{\frac{v}{p}} \rangle}, \tag{8.23a}$$

$$\mathrm{Exp}_p^{-1}\colon \mathcal{S} \to T_0, \qquad q \mapsto \mathrm{Exp}_p^{-1}(q) = R_p \log \frac{q}{p}, \tag{8.23b}$$

$$\exp_p\colon T_0 \to \mathcal{S}, \qquad \exp_p = \mathrm{Exp}_p \circ R_p, \tag{8.23c}$$

$$\exp_p^{-1}\colon \mathcal{S} \to T_0, \qquad \exp_p^{-1}(q) = \Pi_0 \log \frac{q}{p}, \tag{8.23d}$$

Applying the map $\exp_p$ to a vector in $\mathbb{R}^c = T_0 \oplus \mathbb{R}\mathbb{1}$ does not depend on the constant component of the argument, due to (8.19).

The **assignment manifold** is defined as

$$(\mathcal{W}, g), \qquad \mathcal{W} = \mathcal{S} \times \cdots \times \mathcal{S}. \qquad (n = |\mathcal{I}| \text{ factors}) \tag{8.24}$$

Points $W \in \mathcal{W}$ are row-stochastic matrices $W \in \mathbb{R}^{n \times c}$ with row vectors

$$W_i \in \mathcal{S}, \quad i \in \mathcal{I} \tag{8.25}$$

that represent the assignments (8.1) for every $i \in \mathcal{I}$. The jth component of W_i is interchangeably denoted by W_i^j or as element $W_{i,j}$ of the matrix $W \in \mathcal{W}$.

We set

$$\mathcal{T}_0 = T_0 \times \cdots \times T_0 \qquad (n = |\mathcal{I}| \text{ factors}) \tag{8.26}$$

with tangent vectors $V \in \mathbb{R}^{n \times c},\ V_i \in T_0,\ i \in \mathcal{I}$. All the mappings defined above factorize in a natural way and apply row-wise, e.g. $\mathrm{Exp}_W = (\mathrm{Exp}_{W_1}, \ldots, \mathrm{Exp}_{W_n})$ etc.

Remark 8.1 (Early Related Work: Nonlinear Relaxation Labeling) Regarding *image labeling*, our work originates in the seminal work of Rosenfeld et al. [24, 43]. Similar to the early days of neural networks [46], this approach was not accepted by researchers focusing on applications. Rather, support vector machines [13] were prevailing later on in pattern recognition and machine learning due to the convexity of the training problem, whereas graph cuts [9] became the workhorse for image labeling (segmentation), for similar reasons.

Nowadays, deep networks predominate in any field due to its unprecedented performance in applications. And most practitioners, therefore, accept it and ignore the criticism in the past that has not become obsolete [2, 15].

Remark 8.2 (Related Work: Replicator Equation and Evolutionary Game Dynamics) The gradient flow

$$\dot{p} = \operatorname{grad}_p f, \qquad p(0) \in p_0 \in \mathcal{S} \tag{8.27}$$

evolving on $\mathcal{S}$, with $\operatorname{grad}_p f$ due to (8.17), is known as the *replicator equation* in connection with evolutionary dynamical games [21, 47]. More general 'payoff functions' replacing ∂f_p in (8.17) have been considered that may or may not derive from a potential.

In view of Remark 8.1, we point out that Pelillo [39] worked out connections to relaxation labeling from this angle and, later on, also to graph-based clustering [38]. In our opinion, a major reason for why these approaches fall short of the performance of alternative schemes is the absence of a spatial interaction mechanism that conforms with the underlying geometry of assignments. Such a mechanism basically defines the assignment flow to be introduced below.

8.2.2 The Assignment Flow

We introduce the assignment flow [3] and its components for supervised data labeling on a graph.

8.2.2.1 Likelihood Map

Let $i \in \mathcal{I}$ be any vertex and (recall (8.1))

$$D_i = \big(d_{\mathcal{F}}(f_i, f_1^*), \ldots, d_{\mathcal{F}}(f_i, f_c^*)\big)^\top, \qquad i \in \mathcal{I}. \tag{8.28}$$

Since the metric (feature) space $\mathcal{F}$ can be anything depending on the application at hand, we include a scaling parameter[2] $\rho > 0$ for normalizing the range of the components of D_i and define the **likelihood map** in terms of the **likelihood vectors**

$$L_i \colon \mathcal{S} \to \mathcal{S}, \qquad L_i(W_i) = \exp_{W_i}\Big(-\frac{1}{\rho} D_i\Big) = \frac{W_i e^{-\frac{1}{\rho} D_i}}{\langle W_i, e^{-\frac{1}{\rho} D_i}\rangle}, \qquad i \in \mathcal{I}. \tag{8.29}$$

By (8.23c) and (8.17), a likelihood vector (8.29) is formed by regarding D_i as gradient vector (see also Remark 8.4 below) and applying the exponential map Exp_{W_i}.

Using (8.29) we define the **single-vertex assignment flow**

[2] The sizes of the components D_i^j, $j \in \mathcal{J}$ *relative* to each other only matter.

$$\dot{W}_i = R_{W_i} L_i(W_i), \qquad W_i(0) = \mathbb{1}_{\mathcal{S}} \tag{8.30a}$$

$$= W_i\big(L_i(W_i) - \mathbb{E}_{W_i}[L_i(W_i)]\mathbb{1}\big), \quad i \in \mathcal{I}. \tag{8.30b}$$

We have

Proposition 8.3 *The solution to the system* (8.30) *satisfies*

$$\lim_{t\to\infty} W_i(t) = W_i^* = \frac{1}{|J_*|}\sum_{j\in J_*} e_j \in \arg\min_{W_i\in\Delta_c} \langle W_i, D_i\rangle, \qquad J_* = \arg\min_{j\in J} D_i^j. \tag{8.31}$$

In particular, if the distance vector D_i has a unique minimal component $D_i^{j_*}$, then $\lim_{t\to\infty} W_i(t) = e_{j_*}$.

Remark 8.4 (Data Term, Variational Continuous Cuts) A way to look at (8.29) that has proven to be useful for generalizations of the assignment flow (cf. Sect. 8.3.2), is to regard D_i as Euclidean gradient of the *data term*

$$W_i \mapsto \langle D_i, W_i\rangle \tag{8.32}$$

of established variational approaches (‘continuous cuts’) to image labeling, cf. [35, Eq. (1.2)] and [11, Thm. 2] for the specific binary case of $c = 2$ labels. Minimizing this data term over $W_i \in \Delta_c$ yields the result (8.31). In this sense, (8.29) and (8.30) provide a *smooth geometric* version of traditional data terms of variational approaches to data labeling and a dynamic ‘local rounding’ mechanism, respectively.

8.2.2.2 Similarity Map

The flow (8.30) does not interact with the flow at any other vertex $i' \in \mathcal{I}$. In order to couple these flows within each neighborhood $\mathcal{N}_i$ given by (8.2), we assign to each such neighborhood the positive weights[3]

$$\Omega_i = \Big\{ w_{i,k} \colon k \in \mathcal{N}_i,\ w_{i,k} > 0,\ \sum_{k\in\mathcal{N}_i} w_{i,k} = 1 \Big\}, \quad i \in \mathcal{I} \tag{8.33}$$

and define the **similarity map** in terms of the **similarity vectors**

$$S_i \colon \mathcal{W} \to \mathcal{S}, \qquad S_i(W) = \operatorname{Exp}_{W_i}\Big(\sum_{k\in\mathcal{N}_i} w_{i,k} \operatorname{Exp}_{W_i}^{-1}\big(L_k(W_k)\big)\Big) \tag{8.34a}$$

[3] Here we overload the symbol Ω which denotes the Euclidean domain covered by the graph $\mathcal{G}$, as mentioned after Eq. (8.2). Due to the subscripts Ω_i and the context, there should be no danger of confusion.

$$= \frac{\prod_{k\in\mathcal{N}_i} L_k(W_k)^{w_{i,k}}}{\big\langle \mathbb{1}, \prod_{k\in\mathcal{N}_i} L_k(W_k)^{w_{i,k}}\big\rangle}, \quad i \in \mathcal{I}. \tag{8.34b}$$

The meaning of this map is easy to see: The argument of (8.34a) in round brackets corresponds to the optimality condition that determines the Riemannian mean of the likelihood vectors L_k, $k \in \mathcal{N}_i$ with respect to the discrete measure Ω_i, if the exponential map of the Riemannian connection were used [27, Lemma 6.9.4]. Using instead the exponential map of the e-connection yields the closed-form formula (8.34b) that can be computed efficiently.

Remark 8.5 (Parameters) Two parameters have been introduced at this point: the *size* $|\mathcal{N}_i|$ of the neighborhoods (8.2) that we regard as a *scale parameter*, and the *weights* (8.33). How to turn the weights in *adaptivity parameters* and to learn them from data is discussed in Sect. 8.4.

8.2.2.3 Assignment Flow

The interaction of the single-vertex flows through the similarity map defines the **assignment flow**

$$\dot{W} = R_W S(W), \qquad W(0) = \mathbb{1}_{\mathcal{W}}, \tag{8.35a}$$

$$\dot{W}_i = R_{W_i} S_i(W), \qquad W_i(0) = \mathbb{1}_{\mathcal{S}}, \quad i \in \mathcal{I}, \tag{8.35b}$$

where $\mathbb{1}_{\mathcal{W}} \in \mathcal{W}$ denotes the barycenter of $\mathcal{W}$, each row of which is equal to $\mathbb{1}_{\mathcal{S}}$. System (8.35a) collects the local systems (8.35b), for each $i \in \mathcal{I}$, which are coupled through the neighborhoods $\mathcal{N}_i$ and the similarity map (8.34).

Observe the structural similarity of (8.30a) and (8.35) due to the *composition* of the likelihood and similarity maps, unlike the traditional *additive* combination of data and regularization terms.

Example Consider the case of two vertices $\mathcal{I} = \{1, 2\}$ and two labels $c = 2$. Parametrize the similarity vectors by

$$S_1 = (s_1, 1 - s_1)^\top, \qquad S_2 = (s_2, 1 - s_2)^\top, \qquad s_i \in (0, 1), \qquad i \in \mathcal{I} \tag{8.36a}$$

and the weights $\Omega_i = \{w_{i,1}, w_{i,2}\}$ by

$$w_{11} = w_1, \quad w_{12} = 1 - w_1, \quad w_{21} = 1 - w_2, \quad w_{22} = w_2, \qquad w_i \in (0, 1) \tag{8.36b}$$

for $i \in \mathcal{I}$. Due to this parametrization, one can show that the assignment flow for this special case is essentially governed by the system

$$\begin{pmatrix} \dot{s}_1 \\ \dot{s}_2 \end{pmatrix} = \begin{pmatrix} s_1(1-s_1)\big(w_1(2s_1-1) + (1-w_1)(2s_2-1)\big) \\ s_2(1-s_2)\big((1-w_2)(2s_1-1) + w_2(2s_2-1)\big) \end{pmatrix} \tag{8.37}$$

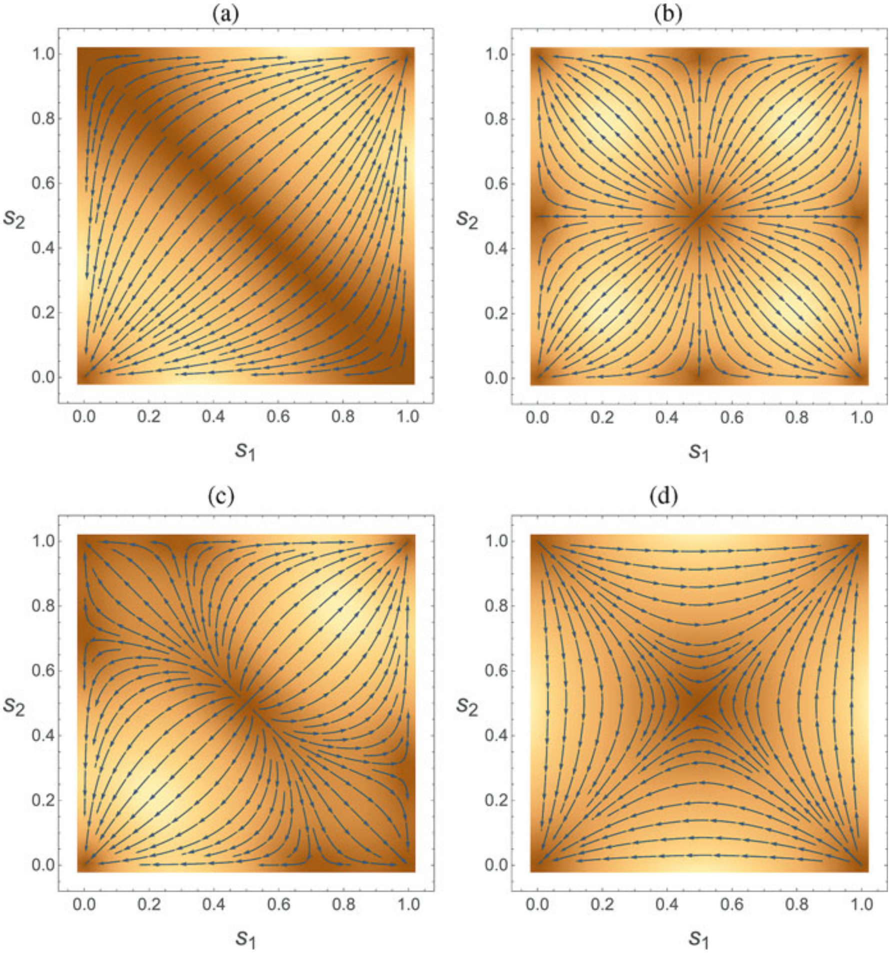


Fig. 8.1 Vector field on the right-hand side of the ODE-system (8.37) that represents the assignment flow for two vertices and two labels, for different weight values (w_1, w_2): (**a**) $(\frac{1}{2}, \frac{1}{2})$; (**b**) $(1, 1)$; (**c**) $(\frac{7}{10}, \frac{7}{10})$; (**d**) $(0, 0)$. Depending on these weights, we observe stable and unstable stationary points at the vertices that represent the possible labelings, and throughout unstable interior stationary points (including interior points of the facets) that correspond to ambiguous labelings.

with initial values $s_1(0), s_2(0)$ depending on the data D_1, D_2. Figure 8.1 illustrates that the weights control the stability of stationary points at the extreme points that correspond to unambiguous labelings to which the assignment flow may converge, and the regions of attraction. Interior fixed points exist as well, including interior points of the facets, but are unstable.

A corresponding study of the general case will be reported in future work.

8.2.2.4 Geometric Integration

We numerically compute the assignment flow by *geometric integration* of the system of ODEs (8.35). Among a range of possible methods [19], Lie group methods [26] are particularly convenient if they can be applied. This requires to point out a Lie group G and an action $\Lambda\colon \mathrm{G}\times\mathcal{M}\to\mathcal{M}$ of G on the manifold $\mathcal{M}$ at hand such that the ODE to be integrated can be represented by a corresponding Lie algebra action [26, Assumption 2.1].

In the case of the assignment flow, we simply identify $\mathrm{G} = T_0$ with the *flat* tangent space. One easily verifies that the action $\Lambda\colon T_0\times\mathcal{S}\to\mathcal{S}$ defined as

$$\Lambda(v, p) = \exp_p(v), \tag{8.38}$$

satisfies

$$\Lambda(0, p) = p, \tag{8.39a}$$

$$\Lambda(v_1 + v_2, p) = \frac{pe^{v_1+v_2}}{\langle p, e^{v_1+v_2}\rangle} = \Lambda(v_1, \Lambda(v_2, p)). \tag{8.39b}$$

Based on Λ the 'Lie machinery' can be applied [56, Section 3] and eventually leads to the following tangent space parametrization of the assignment flow.

Proposition 8.6 ([56]) *The solution $W(t)$ to assignment flow (8.35) emanating from any $W_0 = W(0)$ admits the representation*

$$W(t) = \exp_{W_0}\big(V(t)\big) \tag{8.40a}$$

where $V(t)\in\mathcal{T}_0$ solves

$$\dot{V} = \Pi_{\mathcal{T}_0} S\big(\exp_{W_0}(V)\big), \qquad V(0) = 0 \tag{8.40b}$$

and $\Pi_{\mathcal{T}_0}$ denotes the natural extension of the orthogonal projection (8.14) onto the tangent space (8.26).

We refer to [56] for an evaluation of geometric RKMK methods [37] including embedding schemes for adaptive stepsize selection and more. These algorithms efficiently integrate not only the basic assignment flow but also more involved extensions to unsupervised scenarios, as discussed in Sect. 8.3.

8.2.2.5 Evaluation of Discrete Graphical Models: Wasserstein Message Passing

In Sect. 8.1 the assignment flow was motivated and characterized as an approach that extends discrete graphical models in a systematic way. A natural question, therefore, is: How can one *evaluate* a *given* graphical model using the assignment flow?

This problem was studied in [22]. Let

$$\ell\colon \mathcal{I} \to J \tag{8.41}$$

denote a labeling variable defined on the graph $\mathcal{G}$. We regard ℓ both as a *function* $\mathcal{I} \ni i \mapsto \ell_i \in \mathcal{J}$ and as a *vector* $\mathcal{I} \ni i \mapsto \ell_i = (\ell_i^1, \ldots, \ell_i^c)^\top \in \{e_1, \ldots, e_{|\mathcal{J}|}\}$ depending on the context.

The basic MAP-inference problem (MAP = maximum a posteriori) amounts to minimize a given discrete energy function with arbitrary local functions E_i, E_{ik},

$$E(\ell) = \sum_{i\in\mathcal{I}} E_i(\ell_i) + \sum_{ik\in\mathcal{E}} E_{ik}(\ell_i, \ell_k), \tag{8.42}$$

which is a combinatorially hard problem. The local interaction functions are typically specified in terms of a metric $d_{\mathcal{J}}$ of the label space $(\mathcal{J}, d_{\mathcal{J}})$,

$$E_{ik}(\ell_i, \ell_k) = d_{\mathcal{J}}(\ell_i, \ell_k), \tag{8.43}$$

in which case the problem to minimize $E(\ell)$ is also called the *metric labeling problem* [29]. The basic idea of the approach [22] is

1. to rewrite the local energy terms in the form

$$E(\ell) = \sum_{i\in\mathcal{I}} \Big(\langle \theta_i, \ell_i \rangle + \frac{1}{2} \sum_{k\in\mathcal{N}_i} \langle \ell_i, \Theta_{ik} \ell_k \rangle \Big) \tag{8.44}$$

   with local parameter vectors θ_i and matrices Θ_{ik} given by

$$\langle \theta_i, e_j \rangle = E_i(j), \qquad \langle e_j, \Theta_{ik} e_{j'} \rangle = d_{\mathcal{J}}(j, j'), \qquad i, k \in \mathcal{I}, \quad j, j' \in \mathcal{J}; \tag{8.45}$$

2. to define the energy function (8.42) on the assignment manifold by substituting assignment variables for the labeling variables,

$$\ell \to W \in \mathcal{W}, \qquad \ell_i \to W_i \in \mathcal{S}, \qquad i \in \mathcal{I}; \tag{8.46}$$

   this constitutes a problem *relaxation*;
3. to turn the interaction term into *smoothed* local Wasserstein distances

$$d_{\Theta_{ik},\tau}(W_i, W_k) = \min_{W_{ik}\in\Pi(W_i,W_k)} \Big\{\langle\Theta_{ik}, W_{ik}\rangle + \tau \sum_{j,j'\in\mathcal{J}} W_{ik,jj'} \log W_{ik,jj'}\Big\} \tag{8.47a}$$

$$\text{subject to} \quad W_{ik}\mathbb{1} = W_i, \quad W_{ik}^{\top}\mathbb{1} = W_k \tag{8.47b}$$

between the assignment vectors considered as local marginal measures and using Θ_{ik} as costs for the 'label transport'. Problem (8.47) is a linear assignment problem regularized by the negative entropy which can be efficiently solved by iterative scaling of the coupling matrix W_{ik} [25].

As a result, one obtains the relaxed energy function

$$E_\tau(W) = \sum_{i\in\mathcal{I}} \Big(\langle\theta_i, W_i\rangle + \frac{1}{2}\sum_{k\in\mathcal{N}_i} d_{\Theta_{ik},\tau}(W_i, W_k)\Big) \tag{8.48}$$

with smoothing parameter $\tau > 0$, that properly takes into account the interaction component of the graphical model.

Objective function (8.48) is continuously differentiable. Replacing D_i in the likelihood map (8.29) by $\partial_{W_i}E_\tau(W)$—cf. the line of reasoning mentioned as Remark 8.4—bases the likelihood map on *state-dependent distances* that take into account the interaction of label assignment with the neighborhoods $\mathcal{N}_i$ of the underlying graph, as specified by the given graphical model. This regularizing component of $\partial_{W_i}E_\tau(W)$ replaces the geometric averaging (8.34). An entropy term $\alpha H(W)$ is added in order to gradually enforce an integral assignment W. Numerical integration yields $W(t)$ which converges to a local minimum of the *discrete* objective function (8.42) whose quality (energy value) depends on the tradeoff—controlled by the single parameter α—between minimizing the relaxed objective function (8.48) and approaching an integral solution (unambiguous labeling).

The corresponding 'data flow' along the edges of the underlying graph resembles established belief propagation algorithms [55], yet with significant conceptual differences. For example, the so-called local-polytope constraints of the standard polyhedral relaxation of discrete graphical models (cf. Sect. 8.2.2.6) are satisfied *throughout* the iterative algorithm, rather than *after* convergence only. This holds by construction due to the 'Wasserstein messages' the result from the local Wasserstein distances of (8.48), once the partial gradients $\partial_{W_i}E_\tau(W),\ i \in \mathcal{I}$ are computed. We refer to [22] for further details and discussion.

8.2.2.6 Global Static vs. Local Dynamically Interacting Statistical Models

The standard polyhedral convex relaxation [54] of the discrete optimization problem (8.42) utilizes a *linearization* of (8.44), rewritten in the form

$$E(\ell) = \sum_{i \in \mathcal{I}} \langle \theta_i, \ell_i \rangle + \sum_{ik \in \mathcal{E}} \sum_{j,j' \in \mathcal{J}} \Theta_{ik,jj'} \ell_i^j \ell_k^{j'}, \tag{8.49}$$

by introducing auxiliary variables $\ell_{ik,jj'}$ that replace the quadratic terms $\ell_i^j \ell_k^{j'}$. Collecting all variables $\ell_i^j, \ell_{ik,jj'}, \;\; i,k \in \mathcal{I}, \;\; j,j' \in \mathcal{J}$ into vectors $\ell_{\mathcal{I}}$ and $\ell_{\mathcal{E}}$ and similarly for the model parameters to obtain vectors $\theta_{\mathcal{I}}$ and $\theta_{\mathcal{E}}$, enables to write (8.49) as linear form

$$E(\ell) = \langle \theta_{\mathcal{I}}, \ell_{\mathcal{I}} \rangle + \langle \theta_{\mathcal{E}}, \ell_{\mathcal{E}} \rangle, \qquad \ell = (\ell_{\mathcal{I}}, \ell_{\mathcal{E}}) \tag{8.50}$$

and to define the probability distribution

$$p(\ell; \theta) = \exp\big(\langle \theta_{\mathcal{I}}, \ell_{\mathcal{I}} \rangle + \langle \theta_{\mathcal{E}}, \ell_{\mathcal{E}} \rangle - \psi(\theta)\big), \tag{8.51}$$

which is a member of the exponential family of distributions [5, 51]. $p(\ell; \theta)$ is the *discrete graphical model* corresponding to the discrete energy function (8.42) with log-partition function

$$\psi(\theta) = \log \sum_{\ell \in \text{labelings}} \exp\big(\langle \theta_{\mathcal{I}}, \ell_{\mathcal{I}} \rangle + \langle \theta_{\mathcal{E}}, \ell_{\mathcal{E}} \rangle\big). \tag{8.52}$$

The aforementioned polyhedral convex relaxation is based on the substitution (8.46) and replacing the integrality constraints on ℓ by

$$W_i \in \Delta_c, \quad i \in \mathcal{I} \tag{8.53a}$$

and further affine constraints

$$\sum_{j \in \mathcal{J}} W_{ik,jj'} = W_{k,j'}, \qquad \sum_{j' \in \mathcal{J}} W_{ik,jj'} = W_{i,j}, \qquad \forall i,k \in \mathcal{I},\; \forall j,j' \in \mathcal{J} \tag{8.53b}$$

that ensure local consistency of the linearization step from (8.49) to (8.50). While this so-called *local polytope relaxation* enables to compute good suboptimal minima of (8.42) by solving a (typically huge) linear program as defined by (8.50) and (8.53) using dedicated solvers [28], it has also a major mathematical consequence: the graphical model (8.51) is *overcomplete* or *non-minimally represented* [51] due to linear dependencies among the constraints (8.53b). For this reason the model (8.51) cannot be regarded as point on a *smooth* statistical *manifold* as outlined in Sect. 8.2.1.1.

In this context, the assignment flow may be considered as an approach that emerges from an antipodal starting point. Rather than focusing on the *static global* and *overcomplete* model of the exponential family (8.51) defined on the entire graph

$\mathcal{G}$, we assign to each vertex $i \in \mathcal{I}$ a discrete distribution $W_i = (W_i^1, \dots, W_i^c)^\top$, which by means of the parametrization (8.20a) can be recognized as *minimally represented* member of the exponential family

$$W_i^{\ell_i} = p(\ell_i; \theta_i) = \exp\big(\langle(\theta_i, 1), e_{\ell_i}\rangle - \psi(\theta_i)\big), \quad \ell_i \in \mathcal{J}, \quad i \in \mathcal{I} \tag{8.54a}$$

$$\psi(\theta_i) = \log(1 + \langle \mathbb{1}, e^{\theta_i}\rangle), \tag{8.54b}$$

and hence as point $W_i \in \mathcal{S}$ of the statistical manifold $\mathcal{S}$. These states of label assignments *dynamically* interact through the *smooth* assignment flow (8.35).

We point out that the parameters θ_i of (8.54) are the affine coordinates of $\mathcal{S}$ and have nothing to do with the model parameter $\theta_{\mathcal{I}}, \theta_{\mathcal{E}}$ of the graphical model (8.51). The counter part of $\theta_{\mathcal{I}}$ are the distance vectors D_i, $i \in \mathcal{I}$ (8.28) as part of the likelihood map (8.29), whereas the counterpart of $\theta_{\mathcal{E}}$ are the weights Ω_i, $i \in \mathcal{I}$ (8.33) as part of the similarity map (8.34). The parameters $\theta_{\mathcal{I}}, \theta_{\mathcal{E}}$ are *static (fixed)*, whereas the smooth geometric setting of the assignment flow facilitates computationally the *adaption* of D_i, Ω_i, $i \in \mathcal{I}$. Examples for the adaption of distances D_i are the state-dependent distances discussed in Sect. 8.2.2.5 (cf. the paragraph after Eq. (8.48)) and in the unsupervised scenario of Sect. 8.3.2. Adapting the weights Ω_i by learning from data is discussed in Sect. 8.4.

Regarding numerical computations, using discrete graphical models to cope with such tasks is more cumbersome.

8.3 Unsupervised Assignment Flow and Self-assignment

Two extensions of the assignment flow to unsupervised scenarios are considered in this section. The ability to adapt labels on a feature manifold, during the evolution of the assignment flow, defines the *unsupervised assignment flow* [57, 58] introduced in Sect. 8.3.1. On the other hand, learning labels directly from data without any prior information defines the *self-assignment flow* [59] introduced in Sect. 8.3.2.

8.3.1 Unsupervised Assignment Flow: Label Evolution

Specifying a proper set $\mathcal{F}_*$ of labels (prototypes) beforehand is often difficult in practice: Determining prototypes by clustering results in suboptimal quantizations of the underlying feature space $\mathcal{F}_n$. And carrying out this task without the context that is required for proper inference (label assignment) makes the problem ill-posed, to some extent.

In order to alleviate this issue, a natural approach is to *adapt* an initial label set *during* the evolution of the assignment flow. This is done by coupling label and assignment evolution with interaction in both directions: labels define a time-variant distance vector field that steers the assignment flow, whereas regularized assignments move labels to proper positions in the feature space $\mathcal{F}$.

In this section, we make the stronger assumption that $(\mathcal{F}, g_{\mathcal{F}})$ is a smooth Riemannian feature manifold with metric $g_{\mathcal{F}}$. The corresponding linear tangent-cotangent isomorphism $\widehat{g}_{\mathcal{F}}$ connecting differentials and gradients of smooth functions $f : \mathcal{F} \to \mathbb{R}$ is given by

$$\operatorname{grad} f = \widehat{g}_{\mathcal{F}}^{-1}(df). \tag{8.55}$$

Furthermore, we assume a smooth divergence function [6] to be given,

$$D_{\mathcal{F}} : \mathcal{F} \times \mathcal{F} \to \mathbb{R}, \qquad D_{\mathcal{F}}(f, f') \approx \frac{1}{2} d_{\mathcal{F}}(f, f')^2, \tag{8.56}$$

that approximates the squared Riemannian distance, including equality as special case. A proper choice of $D_{\mathcal{F}}$ is crucial for applications: It ensures that approximate Riemannian means can be computed efficiently. See [58, Section 5] for few scenarios worked out in detail.

Let

$$F^*(t) = \{f_1^*(t), \dots, f_c^*(t)\}, \qquad t \in [0, T] \tag{8.57}$$

denote set of evolving feature prototypes, with initial set $F^*(0) = F_0^*$ computed by any efficient method like metric clustering [20], and with the final set $F^*(T) = \mathcal{F}_*$ of *adapted* prototypes. In order to determine $F^*(t)$, the assignment flow (8.35) is extended to the system

$$\dot{F}^* = \mathcal{V}_{\mathcal{F}}(W, F^*), \qquad F^*(0) = F_0^*, \tag{8.58a}$$

$$\dot{W} = \mathcal{V}_{\mathcal{W}}(W, F^*), \qquad W(0) = \mathbb{1}_{\mathcal{W}}. \tag{8.58b}$$

The solution $F^*(t)$ to (8.58a) evolves on the feature manifold $\mathcal{F}$. It is driven by local Riemannian means that are regularized by the assignments $W(t)$. Equation (8.58b) is the assignment flow determining $W(t)$, based on a time-variant distance vector field in the likelihood map (8.29) due to the moving labels $F^*(t)$.

A specific formulation of (8.58) is worked out in [58] in terms of a one-parameter family of vector fields $(\mathcal{V}_{\mathcal{F}}, \mathcal{V}_{\mathcal{W}})$ that define the following **unsupervised assignment flow** for given data $\mathcal{F}_n = \{f_1, \dots, f_n\}$,

$$\dot{f}_j^* = -\alpha \sum_{i \in \mathcal{I}} \nu_{j|i}(W, F^*) \widehat{g}_{\mathcal{F}}^{-1}\big(d_2 D_{\mathcal{F}}(f_i, f_j^*)\big), \qquad f_j^*(0) = f_{0,j}^*, \quad j \in \mathcal{J}, \tag{8.59a}$$

$$\dot{W}_i = R_{W_i} S_i(W), \qquad W_i(0) = \mathbb{1}_{\mathcal{S}}, \quad i \in \mathcal{I}, \tag{8.59b}$$

with parameter $\alpha > 0$ controlling the speed of label vs. assignment evolution, and

$$\nu_{j|i}(W, F^*) = \frac{L^{\sigma}_{i,j}(W, F^*)}{\sum_{k\in\mathcal{I}} L^{\sigma}_{k,j}(W, F^*)}, \qquad L^{\sigma}_{i,j}(W, F^*) = \frac{W_{i,j} e^{-\frac{1}{\sigma} D_{\mathcal{F}}(f_i, f^*_j)}}{\sum_{l\in\mathcal{J}} W_{i,l} e^{-\frac{1}{\sigma} D_{\mathcal{F}}(f_i, f^*_l)}} \tag{8.60}$$

with parameter $\sigma > 0$ that smoothly 'interpolates' between two specific formulations of the coupled flow (8.58) (cf. [58]).

In Eq. (8.59a), the differential $d_2 D_{\mathcal{F}}(f_i, f^*_j)$ means $dD_{\mathcal{F}}(f_i, \cdot)|_{f^*_j(t)}$ which determines the evolution $f^*_j(t)$ by averaging geometrically data points $\mathcal{F} = \{f_i\}_{i\in\mathcal{I}}$, using weights $\nu_{j|i}\big(W(t), F^*(t)\big)$ due to (8.60) that represent the current assignments of data points f_i, $i \in \mathcal{I}$ to the labels $f^*_j(t)$, $j \in \mathcal{J}$. This dependency on $W(t)$ *regularizes* the evolution $F^*(t)$.

Conversely, the dependency of $W(t)$ on $F^*(t)$ due to the right-hand side of (8.58b) is *implicitly* given through the concrete formulation (8.59b) in terms of the time-variant distances

$$D_i(t) = \big(D_{\mathcal{F}}(f_i, f^*_1(t)), \ldots, D_{\mathcal{F}}(f_i, f^*_c(t))\big)^{\top} \tag{8.61}$$

that generalize the likelihood map (8.29) and in turn (8.59b), through the similarity map (8.34).

In applications, a large number c of labels (8.57) is chosen so as to obtain an 'overcomplete' initial dictionary $F^*(0)$ in a preprocessing step. This helps to remove the bias caused by imperfect clustering at this initial stage of the overall algorithm. The final *effective* number c of labels $F^*(T)$ is smaller, however, and mainly determined by the scale of the assignment flow (cf. Remark 8.5): The regularizing effect of the assignments $W(t)$ on the evolution of labels $F^*(t)$ causes many labels $f^*_j(t)$ to merge or to 'die out', which can be recognized by weights $\nu_{j|i}\big(W(t), F^*(t)\big)$ converging to 0. Extracting the effective labels from $F^*(T)$ determines $\mathcal{F}_*$.

The benefit of the *unsupervised* assignment flow (8.59) is that the remaining labels moved to positions $f^*_j(T) \in \mathcal{F}$ that are difficult to determine beforehand in *supervised* scenarios.

8.3.2 Self-assignment Flow: Learning Labels from Data

This section addresses the fundamental problem: How to determine labels $\mathcal{F}_*$ directly from given data $\mathcal{F}_n$ without any prior information? The resulting *self-assignment flow* generalizes the unsupervised assignment flow of Sect. 8.3.1 that is based on an initial label set $F^*(0)$ and label adaption.

A naive approach would set $F^*(0) = \mathcal{F}_n$ and apply the unsupervised assignment flow. In applications this is infeasible because n generally is large. We overcome this issue by marginalization as follows.

Let $\mathcal{F}_n' = \{f_1', \dots, f_n'\}$ denote a copy of the given data and consider them as initial labels by setting $F^*(0) = \mathcal{F}_n'$. We interpret

$$W_{i,j} = \Pr(j|i), \qquad W_i \in \mathcal{S},\ j \in \mathcal{J},\ i \in \mathcal{I} \tag{8.62}$$

as posterior probabilities of assigning label f_j' to datum f_i, as discussed in [3]. Adopting the uninformative prior $\Pr(i) = \frac{1}{|\mathcal{I}|}$, $i \in \mathcal{I}$ and Bayes rule, we compute

$$\Pr(i|j) = \frac{\Pr(j|i)\Pr(i)}{\sum_{k\in\mathcal{I}} \Pr(j|k)\Pr(k)} = \big(WC(W)^{-1}\big)_{i,j}, \quad C(W) = \mathrm{Diag}(W^\top \mathbb{1}). \tag{8.63}$$

Next we determine the *probabilities of self-assignments* $f_i \leftrightarrow f_k$ of data points by marginalizing over the labels (data copies $\mathcal{F}_n'$) to obtain the *self-assignment matrix*

$$A_{k,i}(W) = \sum_{j\in J} \Pr(k|j)\Pr(j|i) = \big(WC(W)^{-1}W^\top\big)_{k,i}. \tag{8.64}$$

Note that the initial labels are no longer involved. Rather, their evolution as *hidden* variables is *implicitly* determined by the evolving assignments $W(t)$ and (8.63).

Finally, we replace the data term $\langle D, W\rangle = \sum_{i\in\mathcal{I}}\langle D_i, W_i\rangle$ of supervised scenarios (cf. Remark 8.4) by

$$E(W) = \langle D, A(W)\rangle, \tag{8.65}$$

with $D_{i,k} = d_{\mathcal{F}}(f_i, f_j)$ and $A(W)$ given by (8.64). In other words, we replace the assignment matrix W by the self-assignment matrix $A(W)$ that is *parametrized* by the assignment matrix, in order to generalize the data term from supervised scenarios to the current completely unsupervised setting.

As a consequence, we substitute the Euclidean gradient $\partial_{W_i} E(W))$ for the distances vectors (8.28) on which the likelihood map (8.29) is based. These likelihood vectors in turn generalize the similarity map (8.34) and thus define the **self-assignment flow** (8.35).

The approach has attractive properties that enable interpretations from various viewpoints. We mention here only two of them and refer to [59] for further discussion and to the forthcoming report [60].

1. The self-assignment matrix $A(W)$ (8.64) may be seen as a weighted *adjacency matrix* of $\mathcal{G}$ and, in view of its entries, as a *self-affinity* matrix with respect to given data f_i, $i \in \mathcal{I}$ supported by $\mathcal{G}$. $A(W)$ is parametrized by $W(t)$ and (8.64) shows that it evolves in the cone of completely positive matrices [8]. This reflects the combinatorial nature of label learning problem, exhibits relations to *nonnegative matrix factorization* [12] and via convex duality to *graph partitioning* [42].
2. $A(W)$ is nonnegative, symmetric and doubly stochastic. Hence it may be seen as *transportation plan* corresponding to the *discrete optimal transport problem*

[40] of minimizing the objective function (8.65). Taking into account the factorization (8.64) and the parametrization by $W(t)$, minimizing the objective (8.65) may be interpreted as transporting the uniform prior measure $\Pr(i) = \frac{1}{|\mathcal{I}|}$, $i \in \mathcal{I}$ to the support of data points f_i that implicitly define latent labels. In this way, by means of the solution $W(t)$ to the self-assignment flow, labels $\mathcal{F}_*$ directly emerge from given data $\mathcal{F}_n$.

8.4 Regularization Learning by Optimal Control

A key component of the assignment flow is the similarity map (8.34) that couples single-vertex flows (8.30) within neighborhoods $\mathcal{N}_i$, $i \in \mathcal{I}$. Based on the 'context' in terms of data observed within these neighborhoods, the similarity map discriminates structure from noise that is removed by averaging. In this section we describe how the weights (8.33) that parametrize the similarity map can be estimated from data [23].

Our approach is based on an approximation of the assignment flow that is governed by an ODE defined on the tangent space $\mathcal{T}_0$ which linearly depends on the weights (Sect. 8.4.1). Using this representation, the learning problem is subdivided into two tasks (Sect. 8.4.2):

1. *Optimal weights* are computed from ground truth data and corresponding labelings.
2. A *prediction map* is computed in order to extrapolate the relation between observed data and optimal weights to novel data.

8.4.1 Linear Assignment Flow

We consider the following approximation of the assignment flow (8.35), introduced by Zeilmann et al. [56].

$$\dot{W} = R_W\Big(S(W_0) + dS_{W_0} R_{W_0} \log \frac{W}{W_0}\Big), \qquad W(0) = W_0 = \mathbb{1}_{\mathcal{W}}. \tag{8.66}$$

The 'working point' $W_0 \in \mathcal{W}$ can be arbitrary, in principle. Numerical experiments [56, Section 6.3.1] showed, however, that using the barycenter $W_0 = \mathbb{1}_{\mathcal{W}}$ suffices for our purposes.

Assuming that elements of the tangent space $V \in \mathcal{T}_0 \subset \mathbb{R}^{n \times c}$ are written as vectors by stacking row-wise the tangent vectors V_i, $i \in \mathcal{I}$, the Jacobian dS_{W_0} is given by the sparse block matrix

$$dS_{W_0} = \big(A_{i,k}(W_0)\big)_{i,k\in\mathcal{I}}, \qquad A_{i,k}(W_0) = \begin{cases} w_{i,k}R_{S_i(W_0)}\big(\frac{V_k}{W_{0,k}}\big), & \text{if } k\in\mathcal{N}_i,\\ 0, & \text{otherwise.}\end{cases} \tag{8.67}$$

We call the nonlinear ODE (8.66) *linear assignment flow* because it admits the parametrization [56, Prop. 4.2]

$$W(t) = \mathrm{Exp}_{W_0}\big(V(t)\big), \tag{8.68a}$$

$$\dot{V} = R_{W_0}\big(S(W_0) + dS_{W_0}V\big), \quad V(0) = 0. \tag{8.68b}$$

Equation (8.68b) is a *linear* ODE. In addition, Eq. (8.67) shows that it *linearly* depends on the weight parameters (8.33), which is convenient for estimating optimal values of these parameters.

8.4.2 Parameter Estimation and Prediction

Let

$$\mathcal{P} = \{\Omega_i : i\in\mathcal{I}\} \tag{8.69}$$

denote the parameter space comprising all ‘weight patches’ Ω_i according to (8.33), one patch assigned to every vertex $i\in\mathcal{I}$ within the corresponding neighborhood $\mathcal{N}_i$. Note that $\mathcal{P}$ is a parameter *manifold*: The space containing all feasible weight values of each patch Ω_i has the same structure (ignoring the different dimensions) as $\mathcal{S}$ given by (8.11).

Parameter estimation is done by solving the constrained optimization problem

$$\min_{\Omega\in\mathcal{P}} \; E\big(V(T)\big) \tag{8.70a}$$

$$\text{s.t.} \;\; \dot{V} = f(V,\Omega), \qquad t\in[0,T], \qquad V(0)=0, \tag{8.70b}$$

where (8.70b) denotes the linear ODE (8.68b) and the essential variables V and $\Omega = \{\Omega_1,\Omega_2,\dots,\Omega_n\}$ (all weight patches) in compact form. A basic instance of the objective function (8.70a) is

$$E\big(V(T)\big) = D_{\mathrm{KL}}\big(W^*, \mathrm{Exp}_{W_0}\big(V(t)\big)\big), \tag{8.71}$$

which evaluates the Kullback–Leibler divergence of the labeling induced by $V(T)$ by (8.68a) from a given ground-truth labeling $W^*\in\mathcal{W}$.

Problem (8.70) can be tackled in two ways as indicated by the following diagram.

$$\begin{tikzcd}
E\big(V(T)\big) \text{ s.t. } \dot{V} = f(V,\Omega) \arrow[r, "\text{differentiate}"] \arrow[d, "\text{discretize}"'] & \textbf{adjoint system} \arrow[d, "\text{discretize}"] \\
\textbf{nonlinear program} \arrow[r, "\text{differentiate}"'] & \textbf{sensitivity}
\end{tikzcd} \tag{8.72}$$

Differentiation yields the adjoint system which, together with the primal system (8.70b) and proper discretization, enables to compute the sensitivity $\frac{\mathrm{d}}{\mathrm{d}\Omega}E\big(V(T)\big)$ by numerical integration. Alternatively, one first selects a numerical scheme for integrating the primal system (8.70b) which turns (8.70) into a nonlinear program that can be tackled by established methods.

Most appealing are situations where these two approaches are equivalent, that is when the above diagram commutes [44]. A key aspect in this context concerns the symplectic numerical integration of the joint system. We refer to [23] for details and to [48] for the general background.

The weight parameters are updated by numerically integrating the Riemannian gradient descent flow

$$\dot{\Omega} = -\operatorname{grad}_{\mathcal{P}} E\big(V(T)\big) = -R_{\Omega}\frac{\mathrm{d}}{\mathrm{d}\Omega}E\big(V(T)\big), \quad \Omega(0) = \mathbb{1}_{\mathcal{P}}, \tag{8.73}$$

based on the sensitivities determined using either (equivalent) path of diagram (8.72). The linear map R_{Ω} factorizes according to (8.69) into components R_{Ω_i}, $i \in \mathcal{I}$ that are given by (8.18) and well-defined due to (8.33).

Running this algorithm for many instances of data $\mathcal{F}_n^1, \mathcal{F}_n^2, \dots$ and corresponding ground-truth labelings $W^{*1}, W^{*2}, \dots$ produces the optimal weights $\Omega^{*1}, \Omega^{*2}, \dots,$

$$\big\{\{\mathcal{F}_n^1, \mathcal{F}_n^2, \dots\}, \{W^{*1}, W^{*2}, \dots\}\big\} \quad \longrightarrow \quad \big\{\{\mathcal{F}_n^1, \mathcal{F}_n^2, \dots\}, \{\Omega^{*1}, \Omega^{*2}, \dots\}\big\}. \tag{8.74}$$

We rearrange the data *patch-wise* and denote them by $\mathcal{F}_1^*, \mathcal{F}_2^*, \dots$, i.e. $\mathcal{F}_i^*$ denotes a feature patch[4] extracted in *any* order from some $\mathcal{F}_n^k$. Grouping these feature patches with the corresponding optimal weight patches, extracted from $\Omega^{*1}, \Omega^{*2}, \dots$ in the *same* order, yields the input data

$$\big\{(\mathcal{F}_1^*, \Omega_1^*), \dots, (\mathcal{F}_N^*, \Omega_N^*)\big\} \tag{8.75}$$

for prediction, possible after data size reduction by condensing it to a coreset [41]. The predictor

$$\widehat{\omega}\colon \mathcal{F} \to \mathcal{P}, \qquad \mathcal{F}_i \mapsto \Omega_i \tag{8.76}$$

[4] Not to be confused with *labels* $\mathcal{F}_*$!

returns for any feature patch $\mathcal{F}_i \subset \mathcal{F}_n$ of *novel* data $\mathcal{F}_n$ a corresponding weight patch Ω_i (8.33) that controls the similarity map (8.34).

A basic example of a predictor map (8.76) is the *Nadaraya–Watson* kernel regression estimator [52, Section 5.4]

$$\widehat{\omega}(\mathcal{F}_i) = \sum_{k \in [N]} \frac{K_h(\mathcal{F}_i, \mathcal{F}_k^*)}{\sum_{k' \in [N]} K_h(\mathcal{F}_i, \mathcal{F}_{k'}^*)} \Omega_k^* \tag{8.77}$$

with a proper kernel function (Gaussian, Epanechnikov, etc.) and bandwidth parameter estimated, e.g., by cross-validation based on (8.75). We refer to [23] for numerical examples.

Remark 8.7 (Feasibility of Learning) The present notion of *context* is quite limited: it merely concerns the co-occurrence of features within local neighborhoods $\mathcal{N}_i$. This limits the scope of the assignment flow for applications, so far.

On the other hand, this limited scope enables to subdivide the problem of *learning* these contextual relationships into two *manageable tasks* (1), (2) mentioned in the first paragraph of this section: Subtask (1) can be solved using sound numerics (recall the discussion of (8.72)) without the need to resort to opaque toolboxes, as is common in machine learning. Subtask (2) can be solved using a range of state-of-the-art methods of computational statistics and machine learning, respectively.

The corresponding situation seems less clear for more complex networks that are empirically investigated in the current literature on machine learning. Therefore, the strategy to focus first on the relations between data, data structure and label assignments at *two adjacent scales* (vertices ↔ neighborhoods $\mathcal{N}_i$ ↔ neighborhoods of neighborhoods, and so forth) appears to be more effective, in the long run.

8.5 Outlook

This project has started about 2 years ago. Motivation arises from computational difficulties encountered with the evaluation of hierarchical discrete graphical models and from our limited mathematical understanding of deep networks. We outline our next steps and briefly comment on a long-term perspective.

Current Work Regarding *unsupervised learning*, we are focusing on the low-rank structure of the factorized self-assignment matrix (8.64) that is caused by the *regularization* of the assignment flow and corresponds to the reduction of the effective number of labels (cf. the paragraph below Eq. (8.61)). Our objective is to learn labels directly from data in terms of *patches of assignments* for any class of images at hand.

It is then a natural consequence to extend the objective (8.71) of *controlling* the assignment flow to such dictionaries of assignment patches, that encode image structure at the subsequent local scale (measured by $|\mathcal{N}_i|$). In addition, the prediction map (8.77) should be generalized to *feedback* control that not only takes into account feature similarities, but also similarities between the current state $W(t)$ of the assignment flow and assignment trajectories $W^{*k}(t)$. The latter are computed anyway when estimating the parameters on the right-hand side of (8.74) from the data on the left-hand side.

Coordinating in this way unsupervised learning and control using the assignment flow will satisfactorily solve our current core problem discussed as Remark 8.7.

Perspective In order to get rid of discretization parameters, we are currently studying variants of the assignment flow on continuous domains [50]. 'Continuous' here not only refers to the underlying Euclidean domain Ω replacing the graph $\mathcal{G}$, but also to the current *discrete* change of scale $i \to |\mathcal{N}_i|$, that should become infinitesimal and *continuous*. This includes a continuous-domain extension of the approach [49], where a variational formulation of the assignment flow was studied that is inline with the *additive* combination of data term and regularization in related work [7, 53]. Variational methods (Γ-convergence, harmonic maps) then may provide additional mathematical insight into the regularization property of the assignment flow, into a geometric characterization of partitions of the underlying domain, and into the pros and cons of the compositional structure of the assignment flow.

Acknowledgements I thank my students Ruben Hühnerbein, Fabrizio Savarino, Alexander Zeilmann, Artjom Zern, Matthias Zisler and my colleague Stefania Petra for many discussions and collaboration. We gratefully acknowledge support by the German Science Foundation (DFG), grant GRK 1653.

This work has also been stimulated by the recently established Heidelberg STRUCTURES Excellence Cluster, funded by the DFG under Germany's Excellence Strategy EXC-2181/1 - 390900948.

References

1. Amari, S.I., Nagaoka, H.: Methods of Information Geometry. American Mathematical Society/Oxford University Press, Providence/Oxford (2000)
2. Antun, V., Renna, F., Poon, C., Adcock, B., Hansen, A.C.: On instabilities of deep learning in image reconstruction: does AI come at a cost? (2019). arXiv preprint arXiv:abs/1902.05300
3. Åström, F., Petra, S., Schmitzer, B., Schnörr, C.: Image labeling by assignment. J. Math. Imaging Vis. **58**(2), 211–238 (2017)
4. Ay, N., Jost, J., Lê, H.V., Schwachhöfer, L.: Information Geometry. Springer, Berlin (2017)
5. Barndorff-Nielsen, O.E.: Information and Exponential Families in Statistical Theory. Wiley, Chichester (1978)
6. Basseville, M.: Divergence measures for statistical data processing—an annotated bibliography. Signal Proc. **93**(4), 621–633 (2013)
7. Bergmann, R., Tenbrinck, D.: A graph framework for manifold-valued data. SIAM J. Imaging Sci. **11**(1), 325–360 (2018)
8. Berman, A., Shaked-Monderer, N.: Completely Positive Matrices. World Scientific, Singapore (2003)

9. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. **23**(11), 1222–1239 (2001)
10. Calin, O., Udriste, C.: Geometric Modeling in Probability and Statistics. Springer, Berlin (2014)
11. Chan, T.F., Esedoglu, S., Nikolova, M.: Algorithms for finding global minimizers of image segmentation and denoising models. SIAM J. Appl. Math. **66**(5), 1632–1648 (2006)
12. Cichocki, A., Zdunek, A., Phan, A.H., Amari, S.I.: Nonnegative Matrix and Tensor Factorizations. Wiley, London (2009)
13. Cortes, C., Vapnik, V.: Support-vector networks. Mach. Learn. **20**, 273–297 (1995)
14. Cover, T.M., Thomas, J.A.: Elements of Information Theory, 2nd edn. Wiley, London (2006)
15. Elad, M.: Deep, deep trouble: deep learning's impact on image processing, mathematics, and humanity. SIAM News (2017)
16. Gary, R.M., Neuhoff, D.L.: Quantization. IEEE Trans. Inform. Theory **44**(6), 2325–2383 (1998)
17. Geman, S., Geman, D.: Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE Trans. Pattern Anal. Mach. Intell. **6**(6), 721–741 (1984)
18. Graf, S., Luschgy, H.: Foundations of Quantization for Probability Distributions. Lecture Notes in Mathematics, vol. 1730. Springer, Berlin (2000)
19. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration. Springer, Berlin (2006)
20. Har-Peled, S.: Geometric Approximation Algorithms. AMS, Providence (2011)
21. Hofbauer, J., Siegmund, K.: Evolutionary game dynamics. Bull. Am. Math. Soc. **40**(4), 479–519 (2003)
22. Hühnerbein, R., Savarino, F., Åström, F., Schnörr, C.: Image labeling based on graphical models using Wasserstein messages and geometric assignment. SIAM J. Imaging Sci. **11**(2), 1317–1362 (2018)
23. Hühnerbein, R., Savarino, F., Petra, S., Schnörr, C.: Learning adaptive regularization for image labeling using geometric assignment. In: Proceedings of the International Conference on Scale Space and Variational Methods in Computer Vision (SSVM). Springer, Berlin (2019)
24. Hummel, R.A., Zucker, S.W.: On the foundations of the relaxation labeling processes. IEEE Trans. Pattern Anal. Mach. Intell. **5**(3), 267–287 (1983)
25. Idel, M.: A Review of Matrix scaling and Sinkhorn's normal form for matrices and positive maps (2016). arXiv preprint arXiv:abs/1609.06349
26. Iserles, A., Munthe-Kaas, H.Z., Nørsett, S.P., Zanna, A.: Lie-group methods. Acta Numer. **14**, 1–148 (2005)
27. Jost, J.: Riemannian Geometry and Geometric Analysis, 7th edn. Springer, Berlin (2017)
28. Kappes, J., Andres, B., Hamprecht, F., Schnörr, C., Nowozin, S., Batra, D., Kim, S., Kausler, B., Kröger, T., Lellmann, J., Komodakis, N., Savchynskyy, B., Rother, C.: A comparative study of modern inference techniques for structured discrete energy minimization problems. Int. J. Comput. Vis. **115**(2), 155–184 (2015)
29. Kleinberg, J., Tardos, E.: Approximation algorithms for classification problems with pairwise relationships: metric labeling and Markov random fields. J. ACM **49**(5), 616–639 (2002)
30. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. MIT Press, Cambridge (2009)
31. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems (NIPS), pp. 1097–1105. ACM, New York (2012)
32. Lauritzen, S.L.: Chapter 4: statistical manifolds. In: Gupta, S.S., Amari, S.I., Barndorff-Nielsen, O.E., Kass, R.E., Lauritzen, S.L., Rao, C.R. (eds.) Differential Geometry in Statistical Inference, pp. 163–216. Institute of Mathematical Statistics, Hayward (1987)
33. Lauritzen, S.L.: Graphical Models. Clarendon Press, Oxford (1996)
34. Lee, J.M.: Introduction to Smooth Manifolds. Springer, Berlin (2013)
35. Lellmann, J., Schnörr, C.: Continuous multiclass labeling approaches and algorithms. SIAM J. Imag. Sci. **4**(4), 1049–1096 (2011)

36. Mézard, M., Montanari, A.: Information, Physics, and Computation. Oxford University Press, Oxford (2009)
37. Munthe-Kaas, H.: High order Runge-Kutta methods on manifolds. Appl. Numer. Math. **29**(1), 115–127 (1999)
38. Pavan, M., Pelillo, M.: Dominant sets and pairwise clustering. IEEE Trans. Pattern Anal. Mach. Intell. **29**(1), 167–172 (2007)
39. Pelillo, M.: The dynamics of nonlinear relaxation labeling processes. J. Math. Imaging Vision **7**, 309–323 (1997)
40. Peyré, G., Cuturi, M.: Computational Optimal Transport. CNRS, Paris (2018)
41. Phillips, J.: Coresets and sketches. In: Handbook of Discrete and Computational Geometry, chapter 48. CRC Press, Boca Raton (2016)
42. Povh, J., Rendl, F.: A copositive programming approach to graph partitioning. SIAM J. Optim. **18**(1), 223–241 (2007)
43. Rosenfeld, A., Hummel, R.A., Zucker, S.W.: Scene labeling by relaxation operations. IEEE Trans. Syst. Man Cybern. **6**, 420–433 (1976)
44. Ross, I.: A roadmap for optimal control: the right way to commute. Ann. N.Y. Acad. Sci. **1065**(1), 210–231 (2006)
45. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D **60**, 259–268 (1992)
46. Rumelhart, D.E., McClelland, J.L.: Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations. MIT Press, Boca Raton (1986)
47. Sandholm, W.H.: Population Games and Evolutionary Dynamics. MIT Press, Boca Raton (2010)
48. Sanz-Serna, J.: Symplectic Runge–Kutta schemes for adjoint equations, automatic differentiation, optimal control, and more. SIAM Rev. **58**(1), 3–33 (2016)
49. Savarino, F., Schnörr, C.: A variational perspective on the assignment flow. In: Proceedings of the International Conference on Scale Space and Variational Methods in Computer Vision (SSVM). Springer, Berlin (2019)
50. Savarino, F., Schnörr, C.: Continuous-domain assignment flows. Heidelberg University, October (2019). Preprint, submitted for publication
51. Wainwright, M.J., Jordan, M.I.: Graphical models, exponential families, and variational inference. Found. Trends Mach. Learn. **1**(1–2), 1–305 (2008)
52. Wasserman, L.: All of Nonparametric Statistics. Springer, Berlin (2006)
53. Weinmann, A., Demaret, L., Storath, M.: Total variation regularization for manifold-valued data. SIAM J. Imag. Sci. **7**(4), 2226–2257 (2014)
54. Werner, T.: A linear programming approach to max-sum problem: a review. IEEE Trans. Pattern Anal. Mach. Intell. **29**(7), 1165–1179 (2007)
55. Yedidia, J.S., Freeman, W.T., Weiss, Y.: Constructing free-energy approximations and generalized belief propagation algorithms. IEEE Trans. Inform. Theory **51**(7), 2282–2312 (2005)
56. Zeilmann, A., Savarino, F., Petra, S., Schnörr, C.: Geometric numerical integration of the assignment flow. Inverse Problems, https://doi.org/10.1088/1361-6420/ab2772 (2019, in press)
57. Zern, A., Zisler, M., Åström, F., Petra, S., Schnörr, C.: Unsupervised label learning on manifolds by spatially regularized geometric assignment. In: Proceedings of German Conference on Pattern Recognition (GCPR). Springer, Berlin (2018)
58. Zern, A., Zisler, M., Petra, S., Schnörr, C.: Unsupervised assignment flow: label learning on feature manifolds by spatially regularized geometric assignment (2019). arXiv preprint arXiv:abs/1904.10863
59. Zisler, M., Zern, A., Petra, S., Schnörr, C.: Unsupervised labeling by geometric and spatially regularized self-assignment. In: Proceedings of the Scale Space and Variational Methods in Computer Vision (SSVM). Springer, Berlin (2019)
60. Zisler, M., Zern, A., Petra, S., Schnörr, C.: Self-assignment flows for unsupervised data labeling on graphs. Heidelberg University, October (2019). Preprint, submitted for publication

Chapter 9
Geometric Methods on Low-Rank Matrix and Tensor Manifolds

André Uschmajew and Bart Vandereycken

Contents

A. Uschmajew
Max Planck Institute for Mathematics in the Sciences, Leipzig, Germany
e-mail: uschmajew@mis.mpg.de

B. Vandereycken (✉)
Section of Mathematics, University of Geneva, Geneva, Switzerland
e-mail: bart.vandereycken@unige.ch

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_9

Abstract In this chapter we present numerical methods for low-rank matrix and tensor problems that explicitly make use of the geometry of rank constrained matrix and tensor spaces. We focus on two types of problems: The first are optimization problems, like matrix and tensor completion, solving linear systems and eigenvalue problems. Such problems can be solved by numerical optimization for manifolds, called Riemannian optimization methods. We will explain the basic elements of differential geometry in order to apply such methods efficiently to rank constrained matrix and tensor spaces. The second type of problem is ordinary differential equations, defined on matrix and tensor spaces. We show how their solution can be approximated by the dynamical low-rank principle, and discuss several numerical integrators that rely in an essential way on geometric properties that are characteristic to sets of low rank matrices and tensors.

9.1 Introduction

The following chapter is an outline of Riemannian optimization and integration methods on manifolds of low-rank matrices and tensors. This field is relatively new. While the minimization of functions or the time evolution of dynamical systems under smooth manifold constraints is of course classical, and can be treated in a quite general context, there are specific peculiarities to sets of low-rank matrices and tensors that make Riemannian methods particularly amenable to these sets in actual algorithms. There are at least two main reasons for this.

The first is that manifolds of low-rank matrices or tensors are images of multilinear maps. This does not only have the advantage of having at hand an explicit global parametrization of the manifold itself, but also provides a simple representation of tangent vectors and tangent space projections by the product rule. The second reason is the singular value decomposition (SVD), which for matrices has the remarkable property of providing metric projections onto the non-convex sets of bounded rank matrices. As we will see, for certain low-rank tensor manifolds the SVD can be of a similar use.

A classical and powerful set of algorithms for handling low-rank constraints for matrices or tensors is based on eliminating the constraints by using the aforementioned multilinear parametrizations, and then optimize the block parameters separately, typically in the form of alternating optimization. In contrast, Riemannian methods try to take advantage of the actual geometry of the image, which for instance can overcome problems of ill-conditioning of the typically non-unique multilinear parametrizations. One of the earlier works where the tangent space geometry of non-symmetric fixed rank matrices was quite explicitly exploited in numerical algorithms is [59]. It introduced the dynamical low-rank approximation method for calculating low-rank approximations when integrating a matrix that satisfies a set of ordinary differential equations (ODEs), as we will explain in Sect. 9.5.1. In the context of finding rank bounded feasible points for linear matrix inequalities, a similar exploitation of the tangent space for fixed rank symmetric definite matrices already appeared in [84]. For optimization problems with rank constraints, several Riemannian optimization methods were first presented in [79, 98, 113] that each use slightly different geometries of the sets fixed rank matrices. However, all of them show in great detail how the geometry can be exploited in the algorithms, and [98, 113] also include Riemannian Hessians to obtain superlinear convergence. These algorithms fit in the general framework of optimization on manifolds, summarized in the monograph [2], which however does not deal with manifolds of fixed rank matrices. An influential earlier work using geometrical tools close to the subject of this chapter is [45] about the best rank approximation problem for matrices.

The geometric viewpoint on low-rank matrices can be carried over to low-rank tensors as well. Here, some of the main ideas emanated from mathematical physics, specifically spin systems and molecular dynamics which involves low-rank representation of high-dimensional functions [69]. The embedded geometry of tensor train and hierarchical Tucker manifolds has then been worked out in [46, 108] with the goal of providing the tool of Riemannian optimization also to problems of scientific computing and optimization with tensors. Some examples and references for successful application of such methods will be presented in some details later.

9.1.1 Aims and Outline

Our aim in this chapter is to provide a high-level overview of the main ideas and tools for optimization and time integration on low-rank manifolds. For this we decided to avoid formal definitions, assumptions or arguments that we considered too technical, and tried to develop the concepts in a more descriptive manner. As a result the chapter contains few rigorous theorems, but the provided references should enable the reader to look up most of the technical details. We also stick to a quite concrete 'matrix language' as much as possible and avoid abstract tensor product spaces. In this sense, a tensor will be just an array of numbers, and while this is often sufficient when dealing with practical problems, coordinate-free multilinear algebra can of course be essential for understanding the theoretical foundations, but is out of scope here.

There are several topics that will not be touched at all in this chapter. First of all, for tensors we have restricted to manifolds of tensors with fixed tensor train rank, because it can be quite easily presented. The two other tensor formats that allow for geometric methods in a similar spirit are the Tucker format (related to the multilinear rank) and its hierarchical version, the hierarchical Tucker format.

Another important ignored topic is about the choice of the rank. While we present methods for optimization and integration on manifolds of fixed rank matrices and tensors, the choice of the rank is quite problem dependent and needs to balance the reachable model error with the numerical complexity. This is often achieved adaptively. Of course, if a problem at hand does not allow for a 'low-rank' solution in the first place, the methods presented in this chapter are of limited use, albeit still mathematically interesting. Finding conditions that ensure low-rank solutions to a class of optimization problems or ODEs can be challenging and several questions in this context are still unanswered, especially for tensors.

Finally, the alternating optimization methods mentioned above, like the alternating least squares or DMRG algorithm, will not be further discussed in this chapter. Compared to Riemannian optimization, these classic approaches to low-rank optimization are much better known and have been used in countless applications. For further reading we would like to refer to the several overview articles taking different perspectives on low-rank optimization, see [6, 15–17, 37, 61, 100], and the monographs [39, 51, 53].

The chapter is structured as follows. In Sect. 9.2 we provide an elementary outline of the geometry of the set of fixed rank matrices as an embedded submanifold with focus on the geometric concepts that are needed in efficient algorithms. In Sect. 9.3 we introduce the tensor train format and show that its geometry shares many similarities to that of the matrix case. The next two Sects. 9.4 and 9.5, are devoted to optimization problems and the integration of ODEs over low-rank matrices and tensor train tensors. In both cases we will show how the geometry that was just derived plays a crucial role. Finally, in Sect. 9.6 we mention typical applications that can be treated well with low-rank tensor techniques and in particular with geometric methods.

9.2 The Geometry of Low-Rank Matrices

As motivated in the introduction, many approximation and identification problems involving low-rank matrices or tensors can be formulated as nonlinear, rank constrained optimization problems. To design and understand efficient geometric methods for their solution, it is therefore necessary to understand the geometry of sets of matrices and tensors of bounded rank. The most basic ingredients for such methods are the representation of tangent vectors, the computation of tangent space projections and the availability of retractions. In this section we present these concepts for the well known case of low-rank matrices in quite some detail as it features all the core ideas on an easily understandable level. We will then in the next section consider manifolds of tensors in low rank tensor train format as an exemplary case for tensors, since it is a tensor decomposition with many parallels to the matrix case.

We restrict the considerations to the linear space $\mathbb{R}^{m \times n}$ of real $m \times n$ matrices, although most of the following theory can be developed for complex matrices too.

The Euclidean structure of this space is given by the *Frobenius inner product* of two matrices,

$$(X, Y)_F = \mathrm{trace}(X^T Y) = \sum_{i_1=1}^{m} \sum_{i_2=1}^{n} X(i_1, i_2) Y(i_1, i_2),$$

which induces the *Frobenius norm* $\|X\|_F = (X, X)_F^{1/2}$.

As is well known, the *rank* of a matrix $X \in \mathbb{R}^{m \times n}$ is the smallest number $r = \mathrm{rank}(X)$ such that there exist a decomposition

$$X = GH^T, \qquad G \in \mathbb{R}^{m \times r},\ H \in \mathbb{R}^{n \times r}. \tag{9.1}$$

Necessarily, it holds $r \le \min(m, n)$. We call such a rank revealing decomposition of X the (G, H)*-format*.

Note that the decomposition (9.1) is not unique, since we may replace G with GA and H with HA^{-T}, where A is an invertible $r \times r$ matrix. This ambiguity can be removed by requiring additional constraints. A special case is the rank revealing QR decomposition $X = QR$, where $Q \in \mathbb{R}^{m \times r}$ has pairwise orthonormal columns, and $R \in \mathbb{R}^{r \times n}$ is an upper triangular matrix with positive diagonal entries. Such a decomposition can be computed by the column pivoted QR algorithm; see [35].

When m or n are very large, but r is small, it is obviously beneficial in computations to store the matrix X in the (G, H)-format (9.1): instead of storing mn entries of the full matrix X, we only need to know the $(m + n)r$ entries of the matrices G and H. When $(m + n)r$ is much smaller than mn, we may rightfully say that X is of *low rank*. The key idea of low-rank approximation is that in many applications X may not be of exact low rank, but still can be well approximated by a low-rank matrix.

9.2.1 Singular Value Decomposition and Low-Rank Approximation

The fundamental tool for low-rank approximation is the *singular value decomposition* (SVD). Let $\mathrm{rank}(X) \le r \le \min(m, n)$, then the SVD of X is a decomposition

$$X = U \Sigma V^T = \sum_{\ell=1}^{r} \sigma_\ell u_\ell v_\ell^T, \tag{9.2}$$

where $U = \begin{bmatrix} u_1 & \cdots & u_r \end{bmatrix} \in \mathbb{R}^{m \times r}$ and $V = \begin{bmatrix} v_1 & \cdots & v_r \end{bmatrix} \in \mathbb{R}^{n \times r}$ have orthonormal columns and $\Sigma \in \mathbb{R}^{r \times r}$ is a diagonal matrix. Its diagonal entries $\sigma_1, \ldots, \sigma_r$ are called the *singular values* of X and will always be taken to be nonnegative and ordered: $\sigma_1 \geq \cdots \geq \sigma_r \geq 0$. Note that if $k = \operatorname{rank}(X) < r$, then $\sigma_k > 0$, while $\sigma_{k+1} = \cdots = \sigma_r = 0$.

The discovery of the SVD is usually attributed to Beltrami and Jordan around 1873/1874, with important later contributions by Sylvester, Schmidt, and Weyl; see, e.g., [104] for a history. Its existence is not difficult to show when appealing to the spectral theorem for symmetric matrices. It is enough to consider $r = \operatorname{rank}(X)$. The positive semidefinite matrix XX^T then has r positive eigenvalues and admits an eigenvalue decomposition $XX^T = U\Lambda U^T$ with $\Lambda \in \mathbb{R}^{r\times r}$ being a diagonal matrix with a positive diagonal, and $U^TU = I_r$. The matrix UU^T is then the orthogonal projector on the column space of X, and hence $UU^TX = X$. Now setting $\Sigma = \Lambda^{1/2}$ and $V = X^TU\Sigma^{-1}$ we obtain $U\Sigma V^T = UU^TX = X$, that is, an SVD of X. Note that V indeed has orthonormal columns, as $V^TV = \Sigma^{-1}U^TXX^TU\Sigma^{-1} = \Sigma^{-1}\Lambda\Sigma^{-1} = I_r$.

The following theorem is the reason for the importance of the SVD in modern applications involving low rank approximation of matrices and—as we will explain later—of tensors.

Theorem 9.1 *Consider an SVD* (9.2) *of a matrix X with $\sigma_1 \geq \cdots \geq \sigma_r \geq 0$. For any $k < r$, the* truncated SVD

$$X_k = \sum_{\ell=1}^{k} \sigma_\ell u_\ell v_\ell^T$$

provides a matrix of rank at most k that is closest in Frobenius norm to X. The distance is

$$\|X - X_k\|_F = \min_{\operatorname{rank}(Y)\leq k} \|X - Y\|_F = \left(\sum_{\ell=k+1}^{r} \sigma_\ell^2 \right)^{1/2}. \tag{9.3}$$

If $\sigma_k > \sigma_{k+1}$, then X_k has rank k and is the unique best approximation of rank at most k.

This famous theorem is due to Schmidt [96] dating 1907 who proved it for compact integral operators. Later in 1936 it was rediscovered by Eckart and Young [25]. In 1937, Mirksy [80] proved a much more general version of this theorem stating that the same truncated SVD provides a best rank-k approximation in any unitarily invariant norm. A norm $\|\cdot\|$ on $\mathbb{R}^{m\times n}$ is called unitarily invariant if $\|X\| = \|QXP\|$ for all orthogonal Q and P. For such a norm it holds that $\|X\| = \|\Sigma\|$, that is, the norm is entirely defined by the vector of singular values.

The SVD of an $m \times n$ matrix can be computed from a symmetric eigenvalue problem or, better, using the Golub–Kahan algorithm [34]. The amount of work in double precision[1] when $m \geq n$ is $O(14mn^2 + 8n^3)$; see [35, Chapter 8.6]. For a large matrix X, computing the full SVD is prohibitively expensive if one is only interested in its low-rank approximation X_k and if $k \ll \min(m, n)$. To this end, there exist many so-called matrix-free methods based on Krylov subspaces or randomized linear algebra; see, e.g., [43, 67]. In general, these methods are less predictable than the Golub–Kahan algorithm and are not guaranteed to always give (good approximations of) X_k. They can, however, exploit sparsity since they only require matrix vector products with X and X^T.

Observe that the existence of a best approximation of any matrix X by another matrix of rank at most k implies that the set

$$\mathcal{M}_{\leq k} = \{X \in \mathbb{R}^{m\times n} \colon \operatorname{rank}(X) \leq k\} \tag{9.4}$$

is a closed subset of $\mathbb{R}^{m\times n}$. Therefore any continuous function $f \colon \mathbb{R}^{m\times n} \to \mathbb{R}$ with bounded sublevel sets attains a minimum on $\mathcal{M}_{\leq k}$. The formula (9.3) for the distance from $\mathcal{M}_{\leq k}$ implies that a matrix admits a good low-rank approximation in Frobenius norm if its singular values decay sufficiently fast. Consequently, low-rank optimization is suitable for such matrix problems, in which the true solution can be expected to have such a property.

9.2.2 Fixed Rank Manifold

Geometric optimization methods, like the ones we will discuss later, typically operate explicitly on smooth manifolds. The set $\mathcal{M}_{\leq k}$ of matrices of rank at most k is a real algebraic variety, but not smooth in those points X of rank strictly less than k. The good news is that the set $\mathcal{M}_{\leq k-1}$ of these points is of relative Lebesgue measure zero.

The *smooth part* of the variety $\mathcal{M}_{\leq k}$ is the set

$$\mathcal{M}_k = \{X \in \mathbb{R}^{m\times n} \colon \operatorname{rank}(X) = k\}$$

of matrices of fixed rank k. It is a folklore result in differential geometry (see, e.g., [66, Example 8.14]) that $\mathcal{M}_k$ is a C^∞ smooth embedded submanifold of $\mathbb{R}^{m\times n}$ of dimension

$$\dim(\mathcal{M}_k) = mn - (m-k)(n-k) = (m+n-k)k. \tag{9.5}$$

The easiest way to show this is by explicitly constructing $\mathcal{M}_k$ as the union of level sets of submersions. The idea is as follows.

[1] Like for eigenvalues, computing the SVD has to be done iteratively and hence will not terminate in finite time in exact arithmetic for a general matrix.

We partition the matrices in $\mathbb{R}^{m\times n}$ as

$$X = \begin{bmatrix} A & B \\ C & D \end{bmatrix}, \qquad A \in \mathbb{R}^{k\times k},$$

and consider the open set $\mathcal{U}$ of all matrices, for which block A is invertible. A matrix X in $\mathcal{U}$ then has rank k if and only if the Schur complement $F(X) = D - CA^{-1}B$ vanishes, that is, $\mathcal{M}_k \cap \mathcal{U} = F^{-1}(0)$. The function F is a submersion from $\mathcal{U}$ to $\mathbb{R}^{(m-k)\times(n-k)}$ because it is surjective (consider $B = C = 0$), and its partial derivative at any point $X \in \mathcal{U}$ with respect to D is the identity, hence the derivative $F'(X)$ at X is surjective. By the submersion theorem, the above preimage $\mathcal{M}_k \cap \mathcal{U}$ is therefore an embedded submanifold of the specified dimension (9.5), and it remains to note that the full set $\mathcal{M}_k$ is the finite union of such manifolds $\mathcal{M}_k \cap \mathcal{U}$ over all possible positions of a $k \times k$ invertible submatrix A.

As an alternative to the above proof, $\mathcal{M}_k$ can also be described as a smooth quotient manifold as in [82]; see also [1] for an overview.

Another important remark concerning optimization is that for $k < \min(m, n)$ both the sets $\mathcal{M}_k$ and $\mathcal{M}_{\le k}$ are simply connected. This follows from the rank revealing decomposition (9.1) and the connectivity of non-singular k frames in $\mathbb{R}^n$.

9.2.3 Tangent Space

The explicit knowledge of the tangent spaces and the efficient representation of tangent vectors is crucial for the practical implementation of geometric optimization methods on a manifold. For the fixed rank manifold we have several options for representing tangent vectors.

First of all, it follows from the bilinearity of the map $(G, H) \mapsto GH^T$ that matrices of the form

$$\xi = \dot{G}H^T + G\dot{H}^T, \qquad \dot{G} \in \mathbb{R}^{m\times k},\ \dot{H} \in \mathbb{R}^{n\times k}, \tag{9.6}$$

are tangent vectors to $\mathcal{M}_k$ at $X = GH^T$. Like the (G, H)-format, this representation of tangent vectors has the disadvantage of not being unique, and it might be sensitive to numerical errors when G or H are ill conditioned.

On the other hand, the representation (9.6) reveals that the tangent vector ξ lies in the sum of two overlapping linear spaces, namely, the subspaces of all matrices whose column (resp. row) space is contained in the column (resp. row) space of X. Based on this observation we can find another representation for ξ. Let $U \in \mathbb{R}^{m\times k}$ and $V \in \mathbb{R}^{n\times k}$ contain orthonormal bases for the column and row space of $X \in \mathcal{M}_k$. Then $X = USV^T$ for some $S \in \mathbb{R}^{k\times k}$ (a possible choice here is the SVD (9.2) of

X, that is, $S = \Sigma$). We choose corresponding orthonormal bases $U_\perp \in \mathbb{R}^{m\times(m-k)}$ and $V_\perp \in \mathbb{R}^{n\times(n-k)}$ for the orthogonal complements. Then the tangent vector ξ is an element of the linear space

$$T_X\mathcal{M}_k = \left\{ \begin{bmatrix} U & U_\perp \end{bmatrix} \begin{bmatrix} C_{11} & C_{12}^T \\ C_{21} & 0 \end{bmatrix} \begin{bmatrix} V & V_\perp \end{bmatrix}^T : \right. \\ \left. C_{11} \in \mathbb{R}^{k\times k}, C_{21} \in \mathbb{R}^{(m-k)\times k}, C_{12} \in \mathbb{R}^{(n-k)\times k} \right\}. \tag{9.7}$$

Vice versa, it is not too difficult to show that every element in $T_X\mathcal{M}_k$ can be written in the form (9.6) and hence is a tangent vector. Since the dimension of $T_X\mathcal{M}_k$ equals that of $\mathcal{M}_k$, it follows that in fact $T_X\mathcal{M}_k$ is equal to the tangent space to $\mathcal{M}_k$ at X.

In (9.7) we have decomposed the tangent space $T_X\mathcal{M}_k$ into three mutually orthogonal subspaces represented by the three matrices C_{11}, C_{21} and C_{12}. The orthogonal projection of any matrix $Z \in \mathbb{R}^{m\times n}$ onto $T_X\mathcal{M}_k$ is hence obtained by projecting on these three spaces separately. This gives

$$\mathcal{P}_X(Z) = P_U Z P_V + (I - P_U) Z P_V + P_U Z (I - P_V), \tag{9.8}$$

where $P_U = UU^T$ and $P_V = VV^T$ are the orthogonal projections onto the column and row space of X, respectively. Expanding this expression, gives the alternative formula

$$\mathcal{P}_X(Z) = P_U Z + Z P_V - P_U Z P_V. \tag{9.9}$$

While the characterization (9.7) of $T_X\mathcal{M}_k$ is very convenient for theoretical purposes, it is less suitable in calculations when k is small but m or n are very large, since then also one of the matrices $U_\perp$ or $V_\perp$ will be very large. In that situation, the factored representation proposed in [98, 113] is preferable:

$$\xi = U\dot{S}V^T + \dot{U}V^T + U\dot{V}^T, \tag{9.10}$$

where

$$\dot{S} = C_{11} \in \mathbb{R}^{k\times k}, \quad \dot{U} = U_\perp C_{21} \in \mathbb{R}^{m\times k}, \quad \dot{V} = V_\perp C_{12} \in \mathbb{R}^{n\times k}. \tag{9.11}$$

This only requires storing the smaller matrices $\dot{S}$, $\dot{U}$, and $\dot{V}$. Observe that the columns of $\dot{U}$ and $\dot{V}$ are orthogonal to the columns of U and V, respectively, which is also called a *gauging condition*.

To conclude, once U, S and V are chosen to represent $X = USV^T$, all the factored parametrizations of tangent vectors at X belong to the linear subspace[2]

$$H_{(U,S,V)} = \{(\overset{\bullet}{U}, \overset{\bullet}{S}, \overset{\bullet}{V}) \colon \overset{\bullet}{S} \in \mathbb{R}^{k \times k},\ \overset{\bullet}{U} \in \mathbb{R}^{n \times k},\ U^T\overset{\bullet}{U} = 0,\ \overset{\bullet}{V} \in \mathbb{R}^{m \times k},\ V^T\overset{\bullet}{V} = 0\}.$$

The representation of $T_X\mathcal{M}_k$ by $H_{(U,S,V)}$ is bijective. One can therefore directly compute the result of the projection $\mathcal{P}_X(Z)$ as a factored parametrization:

$$\overset{\bullet}{S} = U^T ZV, \quad \overset{\bullet}{U} = (I - P_U)ZV, \quad \overset{\bullet}{V} = (I - P_V)Z^T V. \tag{9.12}$$

Observe that this requires k matrix vector products with Z and Z^T, hence sparsity or a low rank of Z can be exploited nicely.

9.2.4 Retraction

The other main ingredient for efficient geometric methods are retractions. A retraction for a manifold $\mathcal{M}$ is a smooth map R on the tangent bundle $T\mathcal{M}$, and maps at every X the tangent space $T_X\mathcal{M}$ to $\mathcal{M}$. The decisive property of a retraction is that this mapping is exact to first order, that is,

$$R_X(\xi) = X + \xi + o(\|\xi\|). \tag{9.13}$$

Obviously, such a map will be useful in optimization methods for turning an increment $X + \xi$ on the affine tangent plane to a new point $R_X(\xi)$ on the manifold. For Riemannian manifolds it can be shown that retractions always exist. A very natural way from a differential geometry viewpoint is the so called *exponential map*, which maps along geodesics in direction of the tangent vector. In practice, the exponential map may be very complicated to compute. There are, however, alternative choices. Retractions in our current context[3] seem to be first introduced in [99]; see also [2] for more details.

For the embedded submanifold $\mathcal{M}_k$ (more precisely, for $\mathcal{M}_{\leq k}$) we are in the fortunate situation that, by Theorem 9.1, we can compute the metric projection (best approximation) in the ambient space equipped with the Frobenius norm as metric through the truncated SVD. It hence provides an easy-to-use retraction with respect to this metric. Note that in general for a C^m smooth embedded submanifold $\mathcal{M}$ of an Euclidean space with $m \geq 2$ and a point $X \in \mathcal{M}$, there exists an open neighborhood of $0 \in T_X\mathcal{M}$ on which a metric projection $\xi \mapsto \mathcal{P}_\mathcal{M}(X + \xi)$ is uniquely defined and satisfies the retraction property

[2] This subspace is a horizontal distribution for the smooth quotient manifold that factors out the freedom in the parametrization $X = USV^T = (UA)(A^{-1}SB^{-T})(VB)^T$; see [1].

[3] Not to be confused with a (deformation) retract from topology.

$$\|X + \xi - \mathcal{P}_{\mathcal{M}}(X + \xi)\| = o(\|\xi\|).$$

In addition, $\mathcal{P}_{\mathcal{M}}$ is C^{m-1} smooth on that neighborhood; see, e.g., [68, Lemma 2.1].

When using truncated SVD as a retraction for $\mathcal{M}_k$, the crucial question arises whether it can be computed efficiently. This indeed is the case. If $X = USV^T$ and $\xi \in T_X\mathcal{M}_k$ are represented in the factored form (9.10), we first compute QR decompositions of $\overset{\bullet}{U}$ and $\overset{\bullet}{V}$,

$$\overset{\bullet}{U} = Q_1R_1, \quad \overset{\bullet}{V} = Q_2R_2.$$

It then holds

$$X + \xi = \begin{bmatrix} U & \overset{\bullet}{U}_1 \end{bmatrix} \begin{bmatrix} SV^T + \overset{\bullet}{S}V^T + \overset{\bullet}{V}^T \\ V^T \end{bmatrix} = \begin{bmatrix} U & Q_1 \end{bmatrix} K \begin{bmatrix} V & Q_2 \end{bmatrix}^T \tag{9.14}$$

with the $2k \times 2k$ block matrix

$$K = \begin{bmatrix} S + \overset{\bullet}{S} & R_2^T \\ R_1 & 0 \end{bmatrix}.$$

Since the matrices $\begin{bmatrix} U & Q_1 \end{bmatrix}$ and $\begin{bmatrix} V & Q_2 \end{bmatrix}$ each have orthonormal columns (as before we assume that both U and V have orthonormal columns), we can obtain an SVD of the 'big' matrix $X + \xi$ from an SVD of the small matrix K, which can be done in $O(k^3)$ time.

9.3 The Geometry of the Low-Rank Tensor Train Decomposition

In this section we present the tensor train decomposition as a possible generalization of low-rank matrix decomposition to tensors. By tensors we simply mean higher-order analogs of matrices: an $n_1 \times \cdots \times n_d$ tensor X is an array of this size containing real valued entries $X(i_1, \ldots, i_d)$; see Fig. 9.1. Such data structures appear in many applications. Another way to see them is as multivariate functions depending on discrete variables/indices. The tensors of given size form a linear space denoted as $\mathbb{R}^{n_1 \times \cdots \times n_d}$. The number d of directions is called the *order* of the tensor. Matrices are hence tensors of order $d = 2$. As for matrices, it is common to also call the natural Euclidean inner product for tensors,

$$\langle X, Y \rangle_F = \sum_{i_1=1}^{n_1} \cdots \sum_{i_d=1}^{n_d} X(i_1, \ldots, i_d) Y(i_1, \ldots, i_d), \tag{9.15}$$

the Frobenius inner product, and it induces the Frobenius norm.

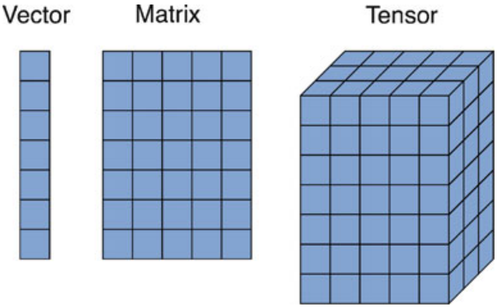


Fig. 9.1 Tensors of order one (vectors), two (matrices), and three

An $n \times \cdots \times n$ tensor has n^d entries, which can quickly become unmanageable in practice when d is large. This is sometimes called a *curse of dimensionality*. Besides other important reasons, the use of low-rank tensor formats provides a tool to circumvent this problem and deal with high dimensional data structures in practice. From a geometric viewpoint a low-rank tensor format defines a nonlinear subset in the space $\mathbb{R}^{n_1 \times \cdots \times n_d}$, like the sets $\mathcal{M}_{\le k}$ from (9.4) in the space of matrices, which can be conveniently represented as the image of a multilinear map. Several choices are possible here.

Let us recall the (G, H)-format (9.1) for a matrix. One way to look at it is as a separation of the variables/indices:

$$X(i_1, i_2) = \sum_{\ell=1}^{r} G(i_1, \ell) H(i_2, \ell). \tag{9.16}$$

The rank is the minimal number r needed for such a separation. A straightforward analog for tensors would be a decomposition

$$X(i_1, \dots, i_d) = \sum_{\ell=1}^{r} C_1(i_1, \ell) \cdots C_d(i_d, \ell)$$

with *factor matrices* $C_\mu \in \mathbb{R}^{n_\mu \times r}$, $\mu = 1, \dots, d$. This tensor format is called the *canonical polyadic (CP) format*. The minimal r required for such a decomposition is called the (canonical) *tensor rank* of X. As for matrices, if r is small then storing a tensor in the CP format is beneficial compared to storing all $n_1 \cdots n_d$ entries since one only needs to know the d factor matrices $C_1, \dots, C_d$.

The CP format has numerous useful applications in data science and scientific computing; see [61] for an overview. One major difference to the matrix case, however, is that the set of all tensors with canonical rank bounded by k is typically not closed. Moreover, while the closure of this set is an algebraic variety, its smooth part is in general not equal to the set of tensors of fixed rank k and does not admit an easy explicit description. An exception is the case of rank-one tensors ($k = 1$): the set of all *outer products* $X = c_1 \circ \cdots \circ c_d$, defined by

$X(i_1, \dots, i_d) = c_1(i_1) \cdots c_d(i_d)$, of nonzero vectors $c_\mu \in \mathbb{R}^{n_\mu}$, $\mu = 1, \dots, d$, is an embedded submanifold of dimension $(n_1 + \cdots + n_d) - (d - 1)$. (It is indeed a special case of manifolds of fixed tensor train rank to be introduced below.) Riemannian optimization in the CP format is hence possible by considering the d-fold sum of rank-one tensors as a manifold, as proposed in [13]. We will, however, not consider this format further in this chapter. Instead, we will present another way to separate the indices of a tensor, which leads to the tensor train format and yields smooth manifolds more similar to the matrix case.

9.3.1 The Tensor Train Decomposition

The *tensor train* (TT) format of a tensor $X \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ can be derived recursively. First, index i_1 is separated from the others, that is,

$$X(i_1, i_2, \dots, i_d) = \sum_{\ell_1=1}^{r_1} G_1(i_1, \ell_1) H_1(\ell_1, i_2, \dots, i_d). \tag{9.17}$$

Note that this is a usual matrix decomposition of the form (9.16) when treating the multi-index $(i_2, \dots, i_d)$ as a single index. Next, in the tensor H_1 the indices (ℓ_1, i_2) are separated from the rest, again by a matrix decomposition,

$$H_1(\ell_1, i_2, \dots, i_d) = \sum_{\ell_2=1}^{r_2} G_2(\ell_1, i_1, \ell_2) H_2(\ell_2, i_3, \dots, i_d), \tag{9.18}$$

yielding

$$X(i_1, i_2, \dots, i_d) = \sum_{\ell_1=1}^{r_1} \sum_{\ell_2=1}^{r_2} G_1(i_1, \ell_1) G_2(\ell_1, i_2, \ell_2) H_2(\ell_2, i_3, \dots, i_d). \tag{9.19}$$

Proceeding in this way, one arrives after d steps at a decomposition of the form

$$\begin{aligned} &X(i_1, \dots, i_d) = \\ &\sum_{\ell_1=1}^{r_1} \cdots \sum_{\ell_{d-1}=1}^{r_{d-1}} G_1(i_1, \ell_1) G_2(\ell_1, i_2, \ell_2) \cdots G_{d-1}(\ell_{d-2}, i_{d-1}, \ell_{d-1}) G_d(\ell_{d-1}, i_d), \end{aligned} \tag{9.20}$$

with *core tensors* $G_\mu \in \mathbb{R}^{r_{\mu-1} \times n_\mu \times r_\mu}$, $\mu = 1, \dots, d$, and $r_0 = r_d = 1$. (The third dummy mode was added to G_1 and G_d to unify notation.) The core tensors G_1 and G_d are hence just matrices, while $G_2, \dots, G_{d-1}$ are tensors of order three. A decomposition (9.20) is called a *tensor train* or *TT decomposition* of X.

The nested summation in formula (9.20) is in fact a long matrix product. If we denote by $G_\mu(i_\mu)$ the $r_{\mu-1} \times r_\mu$ matrix slices of G_μ, one gets the compact representation

$$X(i_1, \dots, i_d) = G_1(i_1)G_2(i_2) \cdots G_{d-1}(i_{d-1})G_d(i_d) \tag{9.21}$$

of the TT format, which explains the alternative name *matrix product state* (MPS) of this tensor decomposition common in physics. This formula clearly shows the multilinearity of the TT decomposition with respect to the core tensors. Also it is easy to see from (9.21) that a TT decomposition is never unique: we can insert the identity $A_\mu A_\mu^{-1}$ between any two matrix factors to obtain another decomposition. It will turn out below that this group action is essentially the only ambiguity.

In the numerical analysis community, the TT format was developed by Oseledets and Tyrtyshnikov in [86, 87] with related formats proposed in [36, 40]. In earlier work, it appeared in theoretical physics under a variety of different forms and names, but is now accepted as MPS; see [97] for an overview.

The number of parameters in the TT decomposition (9.20) is bounded by dnr^2 where $n = \max n_\mu$ and $r = \max r_\mu$. When $r \ll n^{d-2}$, this constitutes a great reduction compared to storing the $n_1 \cdots n_d$ entries in X explicitly. Hence the minimal possible choices for the 'ranks' r_μ appearing in the above construction are of interest. The crucial concept in this context is unfoldings of a tensor into matrices.

We define the *μth unfolding* of a tensor X as the matrix $X^{\langle \mu \rangle}$ of size $(n_1 \cdots n_\mu) \times (n_{\mu+1} \cdots n_d)$ obtained by taking the partial multi-indices $(i_1, \dots, i_\mu)$ as row indices, and $(i_{\mu+1}, \dots, i_d)$ as column indices.[4] In other words,

$$X^{\langle \mu \rangle}(i_1, \dots, i_\mu; i_{\mu+1}, \dots, i_d) = X(i_1, \dots, i_d)$$

where the semicolon indicates the separation between the row- and column indices. One can then show the following theorem.

Theorem 9.2 *In a TT decomposition* (9.20) *it necessarily holds that*

$$r_\mu \geq \operatorname{rank}(X^{\langle \mu \rangle}), \quad \mu = 1, \dots, d-1. \tag{9.22}$$

It is furthermore possible to obtain a decomposition such that equality holds.

To get an insight into why the above statement is true, first observe that, by isolating the summation over the index j_μ, the TT decomposition (9.20) is in fact equivalent to the simultaneous matrix decompositions

$$X^{\langle \mu \rangle} = G_{\leq \mu} G_{\geq \mu+1}^T, \quad \mu = 1, \dots, d-1, \tag{9.23}$$

[4] In the following, we silently assume that a consistent ordering of multi-indices is used.

with ‘partial’ TT unfoldings

$$[G_{\leq\mu}(i_1,\dots,i_\mu;\ell_\mu)] = [G_1(i_1)\cdots G_\mu(i_\mu)] \in \mathbb{R}^{n_1\cdots n_\mu\times r_\mu}$$

and

$$[G_{\geq\mu+1}(i_{\mu+1},\dots,i_d;\ell_\mu)] = [G_{\mu+1}(i_{\mu+1})\cdots G_d(i_d)]^T \in \mathbb{R}^{n_{\mu+1}\cdots n_d\times r_\mu}.$$

From (9.23) it follows immediately that the rank condition (9.22) is necessary. Equality can be achieved using the constructive procedure leading to (9.20) with minimal matrix ranks in every step. Let us explain this for the first two steps. Clearly, the first step (9.17) is a rank revealing decomposition of $X^{\langle 1\rangle}$, so the rank of that matrix can be used as r_1. The minimal admissible r_2 in the second step (9.19) is the rank of the second unfolding $H_1^{\langle 2\rangle}$ of tensor H_1. Let us show that this rank is not larger than the rank of $X^{\langle 1\rangle}$, and hence both are equal by (9.22). Indeed, if $z = [z(i_3,\dots,i_d)]$ is a vector of length $n_3\cdots n_d$ such that $X^{\langle 2\rangle}z = 0$ and $y = H^{\langle 2\rangle}z$, then (9.17) yields $0 = \sum_{\ell_1=1}^{r_1} G_1(i_1,\ell_1)y(\ell_1,i_2)$, which implies $y = 0$, since G_1 has rank r_1. This implies $\operatorname{rank}(H^{\langle 2\rangle}) \leq \operatorname{rank}(X^{\langle 2\rangle})$. One can proceed with a similar argument for the subsequent ranks $r_3,\dots,r_d$.

Theorem 9.2 justifies the following definition.

Definition 9.3 The vector $\mathbf{r} = (r_1,\dots,r_{d-1})$ with $r_\mu = \operatorname{rank}(X^{\langle\mu\rangle})$, $\mu = 1,\dots,d-1$ is called the *TT rank* of a tensor $X \in \mathbb{R}^{n_1\times\cdots\times n_d}$.

For matrices, the SVD-like decompositions $X = USV^T$ with U and V having orthonormal columns are often particularly useful in algorithms since they provide orthonormal bases for the row and column space. This was for instance important for the projection onto the tangent space $T_X\mathcal{M}_k$ at X, see (9.8) and (9.9). It is possible to impose similar orthogonality conditions in the TT decomposition. Recall, that the TT decomposition of a tensor X is obtained by subsequent rank-revealing matrix decompositions for separating the indices $i_1,\dots,i_d$ one from another. This can actually be done from left-to-right, from right-to-left, or from both directions simultaneously and stopping at some middle index i_μ. By employing QR (resp. LQ) matrix decompositions in every splitting step, it is not so difficult to show that one can find core tensors $U_1,\dots,U_{d-1}$, as well as $V_2,\dots,V_d$ such that for every μ between 1 and $d-1$ it holds

$$X^{\langle\mu\rangle} = U_{\leq\mu}S_\mu V_{\geq\mu+1}^T, \tag{9.24}$$

for some $S_\mu \in \mathbb{R}^{r_\mu\times r_\mu}$, and

$$U_{\leq\mu}^T U_{\leq\mu} = V_{\geq\mu+1}^T V_{\geq\mu+1} = I_{r_\mu}. \tag{9.25}$$

Note that these orthogonality conditions inductively imply that the unfoldings $U_\mu^{\langle 3\rangle}$ as well as $V_\mu^{\langle 1\rangle}$ of core tensors itself have orthonormal columns. In general, for a given μ, we call a TT decomposition with cores $G_\nu = U_\nu$ for $\nu < \mu$, $G_\mu(i_\mu) = U_\mu(i_\mu)S_\mu$ and $G_\nu = V_\nu$ for $\nu \geq \mu + 1$, and satisfying (9.25) a *μ-orthogonal TT decomposition* of X. It implies (9.24).

One advantage of such a μ-orthogonal TT decomposition is that it provides the orthogonal projections $U_{\leq\mu}U_{\leq\mu}^T$ and $V_{\geq\mu+1}V_{\geq\mu+1}^T$ for the column and row space of $X^{\langle\mu\rangle}$ in the form of partial TT unfoldings that are hence easily applicable to tensors in TT decomposition. From these projections it will be possible to construct the tangent space projectors to TT manifolds in Sect. 9.3.4.

Note that if a TT decomposition with *some* cores $G_1, \dots, G_d$ is already given, a μ-orthogonal decomposition can be obtained efficiently by manipulating cores in a left-to-right, respectively, right-to-left sweep, where each step consists of elementary matrix operations and QR decompositions and costs $O(dnr^4)$ operations in total. In particular, switching from a μ-orthogonal to a $(\mu + 1)$- or $(\mu - 1)$-orthogonal decomposition, only one such step is necessary costing $O(nr^4)$. Observe that the costs are linear in the order d and mode sizes n_μ but fourth-order in the ranks r_μ. In practice, this means the limit for r_μ is about 10^2 to 10^3, depending on the computing power. We refer to [46, 72, 85] for more details on the implementation and properties of the orthogonalization of TT decompositions.

We conclude with the general remark that algorithmically the TT tensor decomposition is characterized by the concept of *sweeping*, which means that most operations are performed recursively from left-to-right, then right-to-left, and so on. Furthermore, the manipulations on the cores of a TT are based on basic linear algebra. We have already seen that building the decomposition by itself or orthogonalizing a given decomposition can be achieved by a left-to-right sweep involving matrix decompositions only. Next we discuss the important operation of rank truncation that is also achieved in this recursive way.

9.3.2 TT-SVD and Quasi Optimal Rank Truncation

Instead of QR decompositions, one can also use singular value decompositions for constructing a μ-orthogonal TT representation (9.24). One then obtains

$$X^{\langle\mu\rangle} = U_{\leq\mu}\Sigma_\mu V_{\geq\mu+1}^T \tag{9.26}$$

with $\Sigma_\mu \in \mathbb{R}^{r_\mu \times r_\mu}$ being diagonal. In other words, (9.26) is an SVD of $X^{\langle\mu\rangle}$.

The advantage of using SVDs for constructing the TT decomposition is that they can be truncated ‘on the fly’, that is, the index splitting decompositions like (9.17) and (9.19) are replaced by truncated SVDs to enforce a certain rank. Specifically, in a left-to-right sweep, at the μth step, let us assume a partial decomposition

$$\widetilde{X}^{\langle \mu-1 \rangle} = U_{\leq \mu-1} H_{\mu-1}$$

with $U_{\leq \mu-1}$ having orthonormal columns has been constructed.[5] Here we write $\widetilde{X}$, since the tensor may not equal X anymore due to previous rank truncations. The next core U_μ is then obtained from the left singular vectors of a truncated SVD of $H_{\mu-1}^{\langle 2 \rangle}$. This procedure is called the *TT-SVD algorithm* [86, 88]. Note that since $U_{\leq \mu-1}$ is orthogonal, the truncated SVD of $H_{\mu-1}^{\langle 2 \rangle}$ is implicitly also a truncated SVD of $\widetilde{X}^{\langle \mu \rangle}$.

So if at every step of the TT-SVD algorithm instead of the exact rank r_μ a smaller rank k_μ is used, the result will be a tensor $X_{\mathbf{k}}$ of TT rank (at most) $\mathbf{k} = (k_1, \dots, k_{d-1})$ in d-orthogonal TT format. It now turns out that this result provides a *quasi-optimal* approximation of TT rank at most $\mathbf{k}$ to the initial tensor X. Thus the TT-SVD algorithm plays a similar role for TT tensors as the SVD truncation for matrices.

To state this result, let us define the sets

$$\mathcal{M}_{\leq \mathbf{k}} = \{X \in \mathbb{R}^{n_1 \times \cdots \times n_d} \colon \text{ TT-rank}(X) \leq \mathbf{k}\}$$

of tensors of TT rank at most $\mathbf{k} = (k_1, \dots, k_{d-1})$, where the inequality for the rank vector is understood pointwise. By Theorem 9.2, this set is an intersection of low-rank matrix varieties:

$$\mathcal{M}_{\leq \mathbf{k}} = \bigcap_{\mu=1}^{d-1} \{X \in \mathbb{R}^{n_1 \times \cdots \times n_d} \colon \text{ rank}(X^{\langle \mu \rangle}) \leq k_\mu\}. \tag{9.27}$$

Since each of the sets in this intersection is closed, the set $\mathcal{M}_{\leq \mathbf{k}}$ is also closed in $\mathbb{R}^{n_1 \times \cdots \times n_d}$. As a result, every tensor X admits a best approximation by a tensor in the set $\mathcal{M}_{\leq \mathbf{k}}$, which we denote by $X_{\mathbf{k}}^{\text{best}}$, that is,

$$\|X - X_{\mathbf{k}}^{\text{best}}\|_F = \min_{\text{TT-rank}(Y) \leq \mathbf{k}} \|X - Y\|_F.$$

The TT-SVD algorithm, on the other hand, can be seen as an alternating projection method for computing an approximation to X in the intersection (9.27).

The following theorem has been obtained in [88].

Theorem 9.4 *Let $X \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ have TT rank* $\mathbf{r}$ *and* $\mathbf{k} \leq \mathbf{r}$. *Denote by $X_{\mathbf{k}}$ the result of the TT-SVD algorithm applied to X with target rank* $\mathbf{k}$. *Let ε_μ be the error in Frobenius norm committed in the μth truncation step. Then the following estimates hold:*

[5] For consistency we set $U_{\leq 0} = 1$ and $X^{\langle 0 \rangle} = H_0 = X$.

$$\|X - X_{\mathbf{k}}\|_F^2 \le \sum_{\mu=1}^{d-1} \varepsilon_\mu^2, \tag{9.28}$$

and

$$\varepsilon_\mu^2 \le \sum_{\ell > k_\mu} (\sigma_\ell^\mu)^2 \le \|X - X_{\mathbf{k}}^{\text{best}}\|_F^2, \tag{9.29}$$

where σ_ℓ^μ are the singular values of the μth unfolding $X^{\langle\mu\rangle}$.

The theorem has two immediate and equally important corollaries. The first of them is that the sequential rank truncation performed by the TT-SVD is, as announced above, a quasi-optimal projection:

$$\|X - X_{\mathbf{k}}\|_F \le \sqrt{d-1}\|X - X_{\mathbf{k}}^{\text{best}}\|_F. \tag{9.30}$$

The second corollary is a complete characterization of low-rank approximability in the TT format. Since $\|X - X_{\mathbf{k}}^{\text{best}}\|_F \le \|X - X_{\mathbf{k}}\|_F$, the above inequalities imply

$$\|X - X_{\mathbf{k}}^{\text{best}}\|_F^2 \le \sum_{\mu=1}^{d-1} \sum_{\ell_\mu > k_\mu} (\sigma_{\ell_\mu}^\mu)^2.$$

A tensor X will therefore admit good approximation by TT tensors of small rank if the singular values of all the unfoldings $X^{\langle 1\rangle}, \dots, X^{\langle d-1\rangle}$ decay sufficiently fast to zero. By (9.29) such a decay is also a necessary condition. Similar to the comment on matrix problems, the low-rank TT format is hence suitable in practice for tensor problems where the solution has such a property. Justifying this a-priori can be, however, a difficult task, especially for very large problems, and will not be discussed.

We now sketch a proof of Theorem 9.4. The main argument is the observation that while the best rank-k truncation of a matrix is a nonlinear operation, it is for every input indeed performing a *linear* orthogonal projection that can be realized by multiplying from the left an orthogonal projector onto the subspace spanned by the dominant k left singular vectors of the input. Therefore, before the μth truncation step, the current μth unfolding is the result of *some* $\mu - 1$ previous orthogonal projections

$$\widetilde{X}^{\langle\mu\rangle} = \widetilde{P}_{\mu-1} \cdots \widetilde{P}_1 X^{\langle\mu\rangle}, \tag{9.31}$$

which, however, have all been achieved by a matrix multiplication from the left (since only indices $i_1, \dots, i_{\mu-1}$ have been separated at this point). By comparing to the projected best rank-k_μ approximation of $X^{\langle\mu\rangle}$, it is then easy to prove that $\widetilde{X}^{\langle\mu\rangle}$

has no larger distance (in Frobenius norm) to the set of rank-k_μ matrices than $X^{\langle k\rangle}$ itself. Hence

$$\varepsilon_\mu \le \min_{\operatorname{rank}(Y^{\langle\mu\rangle})\le k_\mu} \|X^{\langle\mu\rangle} - Y^{\langle\mu\rangle}\|_F \le \min_{\text{TT-rank}(Y)\le \mathbf{k}} \|X - Y\|_F = \|X - X_{\mathbf{k}}^{\text{best}}\|_F,$$

where the second inequality is due to (9.27). Since the squared Frobenius distance of $X^{\langle\mu\rangle}$ to $\mathcal{M}_{\le k_\mu}$ equals $\sum_{\ell>k_\mu}(\sigma_\ell^\mu)^2$, this proves the second statement (9.29) of the theorem.

Showing the first statement (9.28) is more subtle. One writes $X_{\mathbf{k}}$ as the result of corresponding $d-1$ orthogonal projections in tensor space:

$$X_{\mathbf{k}} = \mathcal{P}_{d-1}\cdots\mathcal{P}_1 X.$$

The error can then be decomposed into

$$X - X_{\mathbf{k}} = (\mathcal{P}_{d-2}\cdots\mathcal{P}_1 X - \mathcal{P}_{d-1}\cdots\mathcal{P}_1 X) + (X - \mathcal{P}_{d-2}\cdots\mathcal{P}_1 X).$$

The Frobenius norm of the first term is precisely ε_{d-1}. One now has to show that both terms are orthogonal to proceed by induction. Indeed, an easy way to see that for every $\mu = 1,\dots,d-1$ the result $\mathcal{P}_\mu\cdots\mathcal{P}_1 X$ after the μth truncation is still in the range of the operator $\mathcal{P}_{\mu-1}\cdots\mathcal{P}_1$ is that the rank truncation of $\widetilde{X}^{\langle\mu\rangle}$ as given by (9.31) may equally be achieved by multiplying from the *right* an orthogonal projector on the dominant k_μ right singular values. Then it is clear that multiplying $\widetilde{P}_{\mu-1}\cdots\widetilde{P}_1$ from the left again will have no effect.

We conclude with two remarks. The first is that the TT-SVD algorithm can be implemented very efficiently if X is already given in a μ-orthogonal TT decomposition as in (9.24), say, with $\mu = 1$, with moderate TT rank. Then in a left-to-right sweep it is sufficient to compute SVDs of single cores, which is computationally feasible if ranks are not too large. This is important in practice when using the TT-SVD algorithm as a retraction as explained below.

The second remark is that the target ranks in the TT-SVD procedure can be chosen adaptively depending on the desired accuracies ε_μ. Thanks to Theorem 9.4 this gives full control of the final error. In this scenario the algorithm is sometimes called *TT-rounding* [86].

9.3.3 Manifold Structure

It may appear at this point that it is difficult to deal with the TT tensor format (and thus with its geometry) computationally, but this is not the case. Tensors of low TT rank can be handled very well by geometric methods in a remarkably analogous way as to low-rank matrices. To do so, one first needs to reveal the geometric structure.

Similar to matrices, the set $\mathcal{M}_{\le \mathbf{k}}$ of tensors of TT rank bounded by $\mathbf{k} = (k_1, \dots, k_{d-1})$ is a closed algebraic variety but not a smooth manifold. Let us assume that the set of tensors of *fixed* TT rank $\mathbf{k}$, that is, the set

$$\mathcal{M}_{\mathbf{k}} = \{X \in \mathbb{R}^{n_1 \times \cdots \times n_d} : \text{ TT-rank}(X) = \mathbf{k}\},$$

is not empty (the conditions for this are given in (9.32) below). Based on Theorem 9.2 it is then easy to show that $\mathcal{M}_{\mathbf{k}}$ is relatively open and dense in $\mathcal{M}_{\le \mathbf{k}}$. One may rightfully conjecture that $\mathcal{M}_{\mathbf{k}}$ is a smooth embedded manifold in $\mathbb{R}^{n_1 \times \cdots \times n_d}$. Note that while $\mathcal{M}_{\mathbf{k}}$ is the intersection of smooth manifolds (arising from taking the conditions $\text{rank}(X^{\langle \mu \rangle}) = k_\mu$ in (9.27)), this by itself does not prove that $\mathcal{M}_{\mathbf{k}}$ is a smooth manifold.

Instead, one can look again at the global parametrization $(G_1, \dots, G_d) \mapsto X$ of TT tensors given in (9.20) but with ranks k_μ. This is a multilinear map τ from the linear parameter space $\mathcal{W}_{\mathbf{k}} = \mathbb{R}^{k_0 \times n_1 \times k_1} \times \cdots \times \mathbb{R}^{k_{d-1} \times n_d \times k_d}$ (with $k_0 = k_d = 1$) to $\mathbb{R}^{n_1 \times \cdots \times n_d}$ and its image is $\mathcal{M}_{\le \mathbf{k}}$. One can now show that the condition $\text{TT-rank}(X) = \mathbf{k}$ is equivalent to the conditions $\text{rank}(G_\mu^{\langle 1 \rangle}) = k_{\mu-1}$ and $\text{rank}(G_\mu^{\langle 2 \rangle}) = k_\mu$ on the unfoldings of core tensors, which defines a subset $\mathcal{W}_{\mathbf{k}}^*$ of parameters. The conditions

$$k_{\mu-1} \le n_\mu k_\mu, \quad k_\mu \le n_\mu k_{\mu-1}, \quad \mu = 1, \dots, d, \tag{9.32}$$

are necessary and sufficient for the existence of such cores, and hence for $\mathcal{M}_{\mathbf{k}}$ being non-empty. Given these conditions the set $\mathcal{W}_{\mathbf{k}}^*$ is open and dense in $\mathcal{W}_{\mathbf{k}}$ and its image under τ is $\mathcal{M}_{\mathbf{k}}$. Yet this parametrization is not injective. From the compact matrix product formula (9.21), we have already observed that the substitution

$$G_\mu(i_\mu) \to A_{\mu-1}^{-1} G_\mu(i_1) A_\mu, \tag{9.33}$$

where A_μ are invertible $r_\mu \times r_\mu$ matrices, does not change the resulting tensor X. One can show that this is the only non-uniqueness in case that X has exact TT rank $\mathbf{k}$, basically by referring to the equivalence with the simultaneous matrix decompositions (9.23). After removing this ambiguity by suitable gauging conditions, one obtains a locally unique parametrization of $\mathcal{M}_{\mathbf{k}}$ and a local manifold structure [46].

An alternative approach, that provides a global embedding of $\mathcal{M}_{\mathbf{k}}$, is to define an equivalence relation of equivalent TT decompositions of a tensor $X \in \mathcal{M}_{\mathbf{k}}$. The equivalence classes match the orbits of the Lie group $\mathcal{G}_{\mathbf{k}}$ of tuples $(A_1, \dots, A_{d-1})$ of invertible matrices acting on $\mathcal{W}_{\mathbf{k}}^*$ through (9.33). One can then apply a common procedure in differential geometry and first establish that the quotient space $\mathcal{W}_{\mathbf{k}}^*/\mathcal{G}_{\mathbf{k}}$ possesses a smooth manifold structure such that the quotient map $\mathcal{W}_{\mathbf{k}}^* \to \mathcal{W}_{\mathbf{k}}^*/\mathcal{G}_{\mathbf{k}}$ is a submersion. As a second step, one shows that the parametrization $\mathcal{W}_{\mathbf{k}}^*/\mathcal{G}_{\mathbf{k}} \to \mathcal{M}_{\mathbf{k}}$ by the quotient manifold is an injective immersion and a homeomorphism in the topology of the ambient space $\mathbb{R}^{n_1 \times \cdots \times n_d}$. It then follows from standard results (see, e.g., [66, Prop. 8.3]), that $\mathcal{M}_{\mathbf{k}}$ is an embedded submanifold of $\mathbb{R}^{n_1\times\cdots\times n_d}$ and its dimension is

$$\dim(\mathcal{M}_{\mathbf{k}}) = \dim(\mathcal{W}^*_{\mathbf{k}}) - \dim(\mathcal{G}_{\mathbf{k}}) = 1 + \sum_{\mu=1}^{d} r_{\mu-1}n_\mu r_\mu - r_\mu^2. \tag{9.34}$$

The details of this construction can be found in [108].

9.3.4 Tangent Space and Retraction

In view of the practical geometric methods on the manifold $\mathcal{M}_{\mathbf{k}}$ to be described later, we now consider the efficient representation of tangent vectors and the computation of retractions. These are quite analogous to the matrix case. First of all, using, e.g., the compact notation (9.21) for the multilinear and surjective parametrization $(G_1,\dots,G_d) \mapsto X$ of the TT manifold $\mathcal{W}^*_{\mathbf{k}}$, it is clear that the tangent space $T_X\mathcal{M}_{\mathbf{k}}$ at a point $X \in \mathcal{M}_{\mathbf{k}}$ with TT-cores $(G_1,\dots,G_d) \in \mathcal{W}^*_{\mathbf{k}}$ (see Sect. 9.3.3) consists of all tensors ξ of the form

$$\xi(i_1,\dots,i_d) = \sum_{\mu=1}^{d} G_1(i_1)\cdots G_{\mu-1}(i_{\mu-1})\dot{G}_\mu(i_\mu)G_{\mu+1}(i_{\mu+1})\cdots G_d(i_d), \tag{9.35}$$

where the cores $\dot{G}_\mu$ at position μ can be chosen freely. In view of (9.34), this representation has too many degrees of freedom, even when fixing the TT decomposition $G_1,\dots,G_d$ of X, but this redundancy can be removed by gauging conditions.

A very reasonable way to do this is the following [56, 103]. We assume that the cores $U_1,\dots,U_{d-1}$ and $V_2,\dots,V_d$ for the orthogonal decompositions (9.24)–(9.25) are available. Then, since the $\dot{G}_\mu$ in (9.35) are entirely free, we do not loose generality by orthogonalizing every term of the sum around $\dot{G}_\mu$:

$$\xi(i_1,\dots,i_d) = \sum_{\mu=1}^{d} U_1(i_1)\cdots U_{\mu-1}(i_{\mu-1})\dot{G}_\mu(i_\mu)V_{\mu+1}(i_{\mu+1})\cdots V_d(i_d). \tag{9.36}$$

We now can add the gauging conditions

$$(U_\mu^{\langle 2\rangle})^T \dot{G}_\mu^{\langle 2\rangle} = 0, \quad \mu = 1,\dots,d-1, \tag{9.37}$$

which remove r_μ^2 degrees of freedom from each of the cores $\dot{G}_1,\dots,\dot{G}_{d-1}$. The last core $\dot{G}_d$ is not constrained.

What this representation of tangent vectors achieves is that all d terms in (9.36) now reside in mutually orthogonal subspaces $T_1, \dots, T_d$. In other words, the tangent space $T_X \mathcal{M}_{\mathbf{k}}$ is orthogonally decomposed:

$$T_X \mathcal{M}_{\mathbf{k}} = T_1 \oplus \cdots \oplus T_d.$$

This allows to write the orthogonal projection onto $T_X \mathcal{M}_{\mathbf{k}}$ as a sum of orthogonal projections onto the spaces $T_1, \dots, T_d$. To derive these projections, consider first the operators that realize the orthogonal projection onto the row and column space of the unfoldings $X^{\langle \mu \rangle}$. They read

$$\mathcal{P}_{\leq \mu}(Z) = \operatorname{Ten}_\mu(U_{\leq \mu} U_{\leq \mu}^T Z^{\langle \mu \rangle}) \quad \text{and} \quad \mathcal{P}_{\geq \mu+1}(Z) = \operatorname{Ten}_\mu(Z^{\langle \mu \rangle} V_{\geq \mu+1} V_{\geq \mu+1}^T), \tag{9.38}$$

where Ten_μ denotes the inverse operation of the μth unfolding so that $\mathcal{P}_{\leq \mu}$ and $\mathcal{P}_{\geq \mu+1}$ are in fact orthogonal projectors in the space $\mathbb{R}^{n_1 \times \cdots \times n_d}$. Note that $\mathcal{P}_{\leq \mu}$ and $\mathcal{P}_{\geq \nu}$ commute when $\mu < \nu$. Furthermore, $\mathcal{P}_{\leq \mu} \mathcal{P}_{\leq \nu} = \mathcal{P}_{\leq \nu}$ and $\mathcal{P}_{\geq \nu} \mathcal{P}_{\geq \mu} = \mathcal{P}_{\geq \mu}$ if $\mu < \nu$.

By inspecting the different terms in (9.36) and taking the gauging (9.37) into account, it is not so difficult to verify that the projection on T_1 is given by

$$Z \mapsto (\mathcal{I} - \mathcal{P}_{\leq 1}) \mathcal{P}_{\geq 2} Z,$$

the projection on T_2 is given by

$$Z \mapsto \mathcal{P}_{\leq 1}(\mathcal{I} - \mathcal{P}_{\leq 2}) \mathcal{P}_{\geq 3} Z = (\mathcal{P}_{\leq 1} - \mathcal{P}_{\leq 2}) \mathcal{P}_{\geq 3} Z$$

and so forth. Setting $\mathcal{P}_{\leq 0} = \mathcal{P}_{\geq d+1} = \mathcal{I}$ (identity) for convenience, the overall projector $\mathcal{P}_X$ onto the tangent space $T_X \mathcal{M}_{\mathbf{k}}$ is thus given in one of the two following forms [72]:

$$\begin{aligned} \mathcal{P}_X &= \sum_{\mu=1}^{d-1} (\mathcal{P}_{\leq \mu-1} - \mathcal{P}_{\leq \mu}) \mathcal{P}_{\geq \mu+1} + \mathcal{P}_{\leq d-1} \\ &= \mathcal{P}_{\geq 2} + \sum_{\mu=2}^{d} \mathcal{P}_{\leq \mu-1} (\mathcal{P}_{\geq \mu+1} - \mathcal{P}_\mu). \end{aligned} \tag{9.39}$$

The formulas (9.39) for the projector on the tangent space are conceptually insightful but still extrinsic. An efficient implementation of this projection for actually getting the gauged components $\dot{G}_\mu$ that represent the resulting tangent vector is possible if Z is itself a TT tensor of small ranks or a very sparse tensor. For example, due to the partial TT structure of projectors (9.38), when computing $\mathcal{P}_{\leq \mu+1} Z$, the partial result from $\mathcal{P}_{\leq \mu} Z$ can be reused and so on. The full details are cumbersome to explain so we do not present them here and refer to [73, §7] and [103, §4].

It is also interesting to note that the tangent space $T_X\mathcal{M}_{\mathbf{k}}$ itself contains only tensors of TT rank at most $2\mathbf{k}$. This is due to the structure (9.35) of tangent vectors as sums of TT decompositions that vary in a single core each [50]. Since X itself is in $T_X\mathcal{M}_{\mathbf{k}}$, we directly write the TT decomposition of $X+\xi$, since this will be the tensors that need to be retracted in optimization methods. In terms of the left- and right-orthogonal cores $U_1,\dots,U_{d-1}$ and $V_2,\dots,V_d$ from (9.25) we have [103]

$$(X+\xi)(i_1,\dots,i_d) = W_1(i_1)W_2(i_2)\cdots W_{d-1}(i_{d-1})W_d(i_d), \tag{9.40}$$

with the cores

$$W_1(i_1) = \begin{bmatrix} U_1(i_1) & \dot{G}_1(i_1)\end{bmatrix}, \quad W_\mu(i_\mu) = \begin{bmatrix} U_\mu(i_\mu) & \dot{G}_\mu(i_\mu) \\ 0 & V_\mu(i_\mu)\end{bmatrix}$$

for $\mu = 2,\dots,d-1$, and

$$W_d(i_d) = \begin{bmatrix} S_d V_d(i_d) + \dot{G}_d(i_d) \\ V_d(i_d)\end{bmatrix},$$

where S_d is the matrix from the d-orthogonal decomposition (9.24) of X. The formula (9.40) is the TT analog to (9.14).

Finally we mention that since $\mathcal{M}_{\mathbf{k}}$ is a smooth manifold, the best approximation of $X+\xi$ would be in principle a feasible retraction from the tangent space to the manifold. It is, however, computationally not available. The TT-SVD algorithm applied to $X+\xi$ with target ranks $\mathbf{k}$ is a valid surrogate, which due to the TT representation (9.40) of tangent vectors is efficiently applicable. As discussed in Sect. 9.3.2 the TT-SVD procedure is essentially a composition of nonlinear projections on low-rank matrix manifolds, which are locally smooth around a given $X \in \mathcal{M}_{\mathbf{k}}$. This provides the necessary smoothness properties of the TT-SVD algorithm when viewed as a projection on $\mathcal{M}_{\mathbf{k}}$. On the other hand, the quasi-optimality of this projection as established in (9.30) implies the retraction property (9.13); see [103] for the details.

9.3.5 Elementary Operations and TT Matrix Format

Provided that the ranks are small enough, the TT representation introduced above allows to store very high-dimensional tensors in practice and to access each entry individually by computing the matrix product (9.21). Furthermore, it is possible to efficiently perform certain linear algebra operations. For instance the sum of two TT tensors X and $\hat{X}$ with TT cores $G_1,\dots,G_d$ and $\hat{G}_1,\dots,\hat{G}_d$ has the matrix product representation

$$(X + \hat{X})(i_1, \dots, i_d) =$$
$$\begin{bmatrix} G_1(i_1) \ \hat{G}_1(i_1) \end{bmatrix} \begin{bmatrix} G_2(i_2) & 0 \\ 0 & \hat{G}_2(i_2) \end{bmatrix} \cdots \begin{bmatrix} G_{d-1}(i_{d-1}) & 0 \\ 0 & \hat{G}_{d-1}(i_{d-1}) \end{bmatrix} \begin{bmatrix} G_d(i_d) \\ \hat{G}_d(i_d) \end{bmatrix}.$$

Hence the core tensors are simply augmented, and no addition at all is required when implementing this operation. Note that this shows that the TT rank of $X + \hat{X}$ is bounded by the (entry-wise) sum of TT ranks of X and $\hat{X}$.

As another example, the Frobenius inner product of X and $\hat{X}$ can be implemented by performing the nested summation in

$$\langle X, \hat{X} \rangle_F = \sum_{i_1=1}^{n_1} \cdots \sum_{i_d=1}^{n_d} G_1(i_1) \cdots G_d(i_d) \hat{G}_d(i_d)^T \cdots \hat{G}_1(i_1)^T$$

sequentially: first, the matrix

$$Z_d = \sum_{i_d=1}^{n_d} G_d(i_d) \hat{G}_d(i_d)^T$$

is computed, then

$$Z_{d-1} = \sum_{i_{d-1}=1}^{n_{d-1}} G_{d-1}(i_{d-1}) Z \hat{G}_{d-1}(i_{d-1})^T$$

and so on. These computations only involve matrix products and the final result Z_1 will be the desired inner product. The computational complexity for computing inner products is hence $O(dnr^3)$ with $n = \max n_\mu$ and $r = \max\{r_\mu, \hat{r}_\mu\}$, where $\mathbf{r}$ and $\hat{\mathbf{r}}$ are the TT-ranks of X and $\hat{X}$, respectively. As a special case, the Frobenius norm of a TT tensor can be computed.

Obviously, these elementary operations are crucial for applying methods from numerical linear algebra and optimization. However, in many applications the most important operation is the computation of the 'matrix-vector-product', that is, in our case the action of a given linear operator $\mathcal{A}$ on a tensor X. In order to use low-rank techniques like Riemannian optimization it is mandatory that the given operator $\mathcal{A}$ can be applied efficiently. In some applications, sparsity of $\mathcal{A}$ makes this possible. More naturally, most low-rank formats for tensors come with a corresponding low-rank format for linear operators acting on such tensors that enable their efficient application. For the TT format, the corresponding operator format is called the *TT matrix format* [86] or *matrix product operator* (MPO) format [115].

A linear map $\mathcal{A}\colon \mathbb{R}^{n_1 \times \cdots \times n_d} \to \mathbb{R}^{n_1 \times \cdots \times n_d}$ can be identified with an $(n_1 \cdots n_d) \times (n_1 \cdots n_d)$ matrix with entries $[\mathcal{A}(i_1, \dots, i_d; j_1, \dots, j_d)]$, where both the rows and columns are indexed with multi-indices. The operator $\mathcal{A}$ is then said to be in the TT matrix format with *TT matrix ranks* $(R_1, \dots, R_{d-1})$ if its entries can be written as

$$\mathcal{A}(i_1, \dots, i_d; j_1, \dots, j_d) = O_1(i_1, j_1)O_2(i_2, j_2) \cdots O_d(i_d, j_d),$$

where $O_\mu(i_\mu, j_\mu)$ are matrices of size $R_{\mu-1} \times R_\mu$ ($R_0 = R_d = 1$). Clearly, the TT matrix format becomes the usual TT format when treating $\mathcal{A}$ as an $n_1^2 \times \cdots \times n_d^2$ tensor.

Note that if $\mathcal{A}$ is an operator on matrices, that is, in the case $d = 2$, $O_1(i_\mu, j_\mu)$ and $O_2(i_\mu, j_\mu)$ are just vectors of length $R_1 = R$, and the formula can be written as

$$\mathcal{A}(i_1, i_2; j_1, j_2) = \sum_{\ell=1}^{R} O_{1,\ell}(i_1, j_1)O_{2,\ell}(i_2, j_2).$$

In other words, such an operator $\mathcal{A}$ is a sum

$$\mathcal{A} = \sum_{\ell=1}^{R} A_\ell \otimes B_\ell$$

of Kronecker products of matrices $[A_\ell(i, j)] = [O_{1,\ell}(i, j)]$ and $[B_\ell(i, j)] = [O_{2,\ell}(i, j)]$.

An operator in the TT matrix format can be efficiently applied to a TT tensor, yielding a result in the TT format again. Indeed, let $Y = \mathcal{A}(X)$, then a TT decomposition of Y can be found using the properties of the Kronecker product $\otimes$ of matrices [86]:

$$\begin{aligned} Y(i_1, \dots, i_d) &= \sum_{j_1=1}^{n_1} \cdots \sum_{j_d=1}^{n_d} \mathcal{A}(i_1, \dots, i_d; j_1, \dots, j_d)X(j_1, \dots, j_d) \\ &= \sum_{j_1=1}^{n_1} \cdots \sum_{j_d=1}^{n_d} \big(O_1(i_1, j_1) \cdots O_d(i_d, j_j)\big) \otimes \big(G_1(j_1) \cdots G_d(j_d)\big) \\ &= \sum_{j_1=1}^{n_1} \cdots \sum_{j_d=1}^{n_d} \big(O_1(i_1, j_1) \otimes G_1(j_1)\big) \cdots \big(O_d(i_d, j_d) \otimes G_d(j_d)\big) \\ &= \hat{G}_1(i_1) \cdots \hat{G}_d(i_d) \end{aligned}$$

with resulting TT cores

$$\hat{G}_\mu(i_\mu) = \sum_{j_\mu=1}^{n_\mu} O_\mu(i_\mu, j_\mu) \otimes G_\mu(j_\mu), \quad \mu = 1, \dots, d.$$

Forming all these cores has a complexity of $O(dnr^2R^2)$, where $R = \max R_\mu$.

Note that $G_\mu(i_\mu)$ is a matrix of size $r_{\mu-1}R_{\mu-1} \times r_\mu R_\mu$ so the TT ranks of $\mathcal{A}$ and X are multiplied when applying $\mathcal{A}$ to X. In algorithms where this operation is performed several times it therefore can become necessary to apply the TT-SVD procedure to the result as a post-processing step for reducing ranks again. This is akin to rounding in floating point arithmetic and is therefore also called TT-rounding.

9.4 Optimization Problems

As we have explained above, the sets of matrices of fixed rank k and tensors of fixed TT rank $\mathbf{k}$ are smooth submanifolds $\mathcal{M}_k \subset \mathbb{R}^{m\times n}$ and $\mathcal{M}_{\mathbf{k}} \subset \mathbb{R}^{n_1\times\cdots\times n_d}$, respectively. In this section we will see how to efficiently exploit these smooth structures in optimization problems.

Here and in the following $\mathbb{V}$ denotes a finite dimensional real vector space, that depending on the context, can be just $\mathbb{R}^N$, a space $\mathbb{R}^{m\times n}$ of matrices, or a space $\mathbb{R}^{n_1\times\cdots\times n_d}$ of tensors.

9.4.1 Riemannian Optimization

We start with a relatively general introduction to local optimization methods on smooth manifolds; see [2] for a broader but still self-contained treatment of this topic.

Let $\mathcal{M}$ be a smooth submanifold in $\mathbb{V}$, like $\mathcal{M}_k$ or $\mathcal{M}_{\mathbf{k}}$. Since $\mathcal{M} \subset \mathbb{V}$, we can represent a point X on $\mathcal{M}$ as an element of $\mathbb{V}$. We can do the same for its tangent vectors $\xi \in T_X\mathcal{M}$ since $T_X\mathcal{M} \subset T_X\mathbb{V} \simeq \mathbb{V}$. This allows us to restrict any smoothly varying inner product on $\mathbb{V}$ to $T_X\mathcal{M}$ and obtain a Riemannian metric $(\cdot,\cdot)_X$ on $\mathcal{M}$. For simplicity, we choose the Euclidean metric:

$$(\xi,\eta)_X = \xi^T\eta, \qquad \xi,\eta \in T_X\mathcal{M} \subset \mathbb{V}.$$

Consider now a smooth objective function $f\colon \mathbb{V} \to \mathbb{R}$. If we restrict its domain to $\mathcal{M}$, we obtain an optimization problem on a Riemannian manifold:

$$\min f(X) \quad \text{s.t.} \quad X \in \mathcal{M}. \tag{9.41}$$

The aim of a Riemannian optimization method is to generate iterates $X_1, X_2, \ldots$ that remain on $\mathcal{M}$ and converge to a (local) minimum of f constrained to $\mathcal{M}$; see Fig. 9.2. It uses only local knowledge of f, like first and second-order derivatives. It thus belongs to the family of feasible methods for constrained optimization, which is a very useful property in our setting since general tensors or matrices in $\mathbb{V}$ with arbitrary rank might otherwise be too large to store. A distinctive difference with other methods for constrained optimization is that a Riemannian optimization method has a detailed geometric picture of the constraint set $\mathcal{M}$ at its disposal.

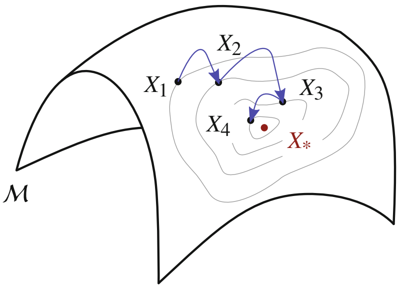


Fig. 9.2 A Riemannian optimization method generates iterates X_ℓ starting from X_1 to minimize f on a manifold $\mathcal{M}$. The thin gray lines are level sets of f and X_* is a (local) minimum of f on $\mathcal{M}$

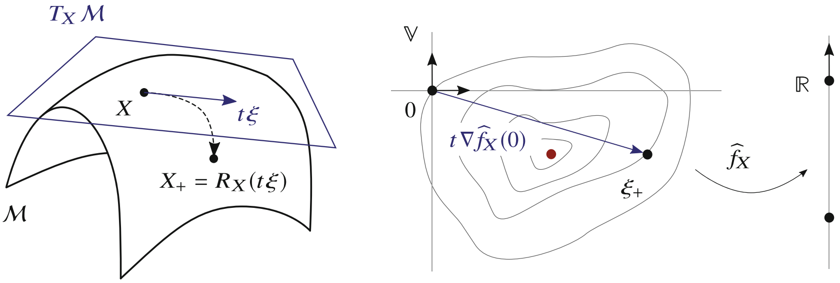


Fig. 9.3 One step of a typical Riemannian optimization method with step direction ξ on the submanifold (left). Example of one step of steepest descent on the pullback (right)

In its most basic form, a Riemannian optimization method is the update formula

$$X_+ = R_X(t\,\xi), \tag{9.42}$$

that is then repeated after replacing X by X_+. The formula (9.42) is defined by the following 'ingredients'; see also the left panel of Fig. 9.3.

1. The *search direction* $\xi \in T_X\mathcal{M}$ that indicates the direction of the update. Similar as in Euclidean unconstrained optimization, the search direction can be obtained from first-order (gradient) or second-order (Hessian) information.[6] Generally, f will locally decrease in the direction of ξ, that is, the directional derivative satisfies $f'(X)\,\xi < 0$.
2. As explained in Sect. 9.2.4, the *retraction* $R_X : T_X\mathcal{M} \to \mathcal{M}$ is a smooth map that replaces the usual update $X + t\,\xi$ from Euclidean space to the manifold setting. Running over t, we thus replace a straight ray with a curve that (locally) lies on $\mathcal{M}$ by construction. By the retraction property (9.13), the curve is rigid at $t = 0$, which means that $R_X(0) = X$ and $\frac{d}{dt}R(t\xi)|_{t=0} = \xi$ for all $\xi \in T_X\mathcal{M}$.
3. The *step size* $t > 0$ is usually chosen to guarantee sufficient decrease of f in X_+, although non-monotone strategies also exist. Given ξ, the step size is typically

[6] We stick here to more standard smooth optimization on purpose but also nonsmooth and stochastic methods are possible for Riemannian manifolds; see [38, 47, 48, 95].

found by line search strategies like backtracking, whereas an *exact* line search would provide a global minimum along direction ξ, if it exists. As an alternative one can use the trust-region mechanism to generate $t\,\xi$.

To explain the possible search directions ξ at a point $X \in \mathcal{M}$, we take a slight detour and consider the *pullback* of f at X:

$$\widehat{f_X} = f \circ R_X \colon T_X\mathcal{M} \to \mathcal{M}.$$

Since $\widehat{f_X}$ is defined on the linear subspace $T_X\mathcal{M}$, we can for example minimize it by the standard steepest descent method; see the right panel of Fig. 9.3. Observe that rigidity of R_X implies $\widehat{f_X}(0) = f(X)$. Hence, the starting guess is the zero tangent vector, which will get updated as

$$\xi_+ = 0 - \beta^\ell \beta_0 \, \nabla \widehat{f_X}(0)$$

and *Armijo backtracking* determines the smallest $\ell = 0, 1, \ldots$ such that

$$\widehat{f_X}(\xi_+) \le \widehat{f_X}(0) - c\,\beta^\ell \beta_0 \|\nabla \widehat{f_X}(0)\|_F^2. \tag{9.43}$$

Here, $\beta = 1/2$, $\beta_0 = 1$, and $c = 0.99$ are standard choices. We could keep on iterating, but the crucial point is that in Riemannian optimization, we perform such a step only once, and then redefine the pullback function for $X_+ = R_X(\xi_+)$ before repeating the procedure.

Formally, the iteration just described is clearly of the form as (9.42), but it is much more fruitful to regard this procedure from a geometric point of view. To this end, observe that rigidity of R_X also implies

$$\widehat{f}_X'(0)\,\xi = f'(X)\,R_X'(0)\,\xi = f'(X)\,\xi \qquad \text{for all } \xi \in T_X\mathcal{M}.$$

With $\mathcal{P}_X \colon \mathbb{V} \to T_X\mathcal{M}$ the orthogonal projection, we thus obtain

$$(\nabla \widehat{f_X}(0), \xi)_F = (\nabla f(X), \mathcal{P}_X(\xi))_F = (\mathcal{P}_X(\nabla f(X)), \xi)_F. \tag{9.44}$$

These identities allow us to define the *Riemannian gradient* of f at X to $\mathcal{M}$ simply as the tangent vector $\mathcal{P}_X(\nabla f(X))$. This vector is conveniently also a direction of steepest ascent among all tangent vectors at X with the same length. We can thus define the *Riemannian steepest descent method* as

$$X_+ = R_X(-t\,\mathcal{P}_X(\nabla f(X))), \quad \text{with } t = \beta^\ell \beta_0. \tag{9.45}$$

Here, Armijo backtracking picks again the smallest ℓ (since $0 < \beta < 1$) such that

$$f(R_X(X_+)) \le f(X) - c\,\beta^\ell \beta_0 \|\mathcal{P}_X \nabla f(X)\|_F^2.$$

Observe that we have arrived at the same iteration as above but instead of using a pullback we derived it directly from geometric concepts, where we have benefited from choosing the Euclidean metric on $T_X\mathcal{M}$ for obtaining the simple formula (9.44) for the Riemannian gradient. Using the notion of second-order retractions, one can in this way derive the *Riemannian Newton method* either using the Riemannian Hessian with pullbacks or directly with the Riemannian connection. We refer to [2] for details, where also trust-region strategies are discussed.

The 'recipe' above leaves a lot of freedom, which can be used to our advantage to choose computational efficient components that work well in practice. Below we will focus on approaches that are 'geometric versions' of classical, non-Riemannian algorithms, yet can be implemented efficiently on a manifold so that they become competitive.

9.4.2 Linear Systems

We now explain how Riemannian optimization can be used to solve very large linear systems. Given a linear operator $\mathcal{L}\colon \mathbb{V} \to \mathbb{V}$ and a 'right-hand side' $B \in \mathbb{V}$, the aim is to calculate any X_{ex} that satisfies the equation

$$\mathcal{L}(X_{\mathrm{ex}}) = B.$$

Since our strategy is optimization, observe that X_{ex} can also be found as a global minimizer of the residual objective function

$$f_{LS}(X) = (\mathcal{L}(X) - B, \mathcal{L}(X) - B)_F = \|\mathcal{L}(X) - B\|_F^2.$$

If in addition $\mathcal{L}$ is symmetric and positive semi-definite on $\mathbb{V}$, the same is true for the energy norm function

$$\begin{aligned} f_{\mathcal{L}}(X) &= (X, \mathcal{L}(X))_F - 2(X, B)_F \\ &= (X - X_{\mathrm{ex}}, \mathcal{L}(X - X_{\mathrm{ex}}))_F - (X_{\mathrm{ex}}, \mathcal{L}(X_{\mathrm{ex}}))_F. \end{aligned}$$

The second identity shows that $f_{\mathcal{L}}(X)$ is indeed, up to a constant, the square of the error in the induced $\mathcal{L}$-(semi)norm. In the following, we will assume that $\mathcal{L}$ is positive semi-definite and focus only on $f = f_{\mathcal{L}}$ since it leads to better conditioned problems compared to f_{LS}.

When X_{ex} is a large matrix or tensor, we want to approximate it by a low-rank matrix or tensor. Since we do not know X_{ex} we cannot use the quasi-best truncation procedures as explained in Sect. 9.3. Instead, we minimize the restriction of $f = f_{\mathcal{L}}$ onto an approximation manifold $\mathcal{M} = \mathcal{M}_k$ or $\mathcal{M} = \mathcal{M}_{\mathbf{k}}$:

$$\min f(X) \quad \text{s.t.} \quad X \in \mathcal{M}.$$

This is exactly a problem of the form (9.41) and we can, for example, attempt to solve it with the Riemannian steepest descent algorithm. With $X \in \mathcal{M}_{\mathbf{k}}$ and the definition of $f_{\mathcal{L}}$, this iteration reads

$$X_+ = R_X(-t\,\mathcal{P}_X(\mathcal{L}(X) - B)). \tag{9.46}$$

When dealing with ill-conditioned problems, as they occur frequently with discretized PDEs, it is advisable to include some preconditioning. In the Riemannian context, one way of doing this is by modifying (9.46) to

$$X_+ = R_X(-t\,\mathcal{P}_X(\mathcal{Q}(\mathcal{L}(X) - B))), \tag{9.47}$$

where $\mathcal{Q}\colon \mathbb{V} \to \mathbb{V}$ is a suitable preconditioner for $\mathcal{L}$. This iteration is called *truncated Riemannian preconditioned Richardson iteration* in [65] since it resembles a classical Richardson iteration.

9.4.3 Computational Cost

Let us comment which parts of (9.46) are typically the most expensive. Since the retraction operates on a tangent vector, it is cheap both for matrices and tensors in TT format as long as their ranks are moderate; see Sect. 9.3. The remaining potentially expensive steps are therefore the application of the projector $\mathcal{P}_X$ and the computation of the step size t.

Let $Z = \mathcal{L}(X) - B$ be the residual. Recall that the projected tangent vector $\xi = \mathcal{P}_X(Z)$ will be computed using (9.12) for matrices and (9.36)–(9.37) for TT tensors. As briefly mentioned before, these formulas are essentially many (unfolded) matrix multiplications that can efficiently be computed if Z is a *sparse* or *low rank* matrix/tensor.

Sparsity occurs for example in the matrix and tensor completion problems (see Sect. 9.6 later) where $\mathcal{L}$ is the orthogonal projector $\mathcal{P}_\Omega$ onto a sampling set $\Omega \subset \{1, \dots, n_1\} \times \cdots \times \{1, \dots, n_d\}$ of known entries of an otherwise unknown matrix/tensor $X_{\mathrm{ex}} \in \mathbb{V}$. The matrix/tensor B in this problem is then the sparse matrix/tensor containing the known entries of X_{ex}. Then if, for example, $X = USV^T \in \mathcal{M}_k$ is a matrix in SVD-like format, the residual $Z = \mathcal{P}_\Omega(X) - B$ is also a sparse matrix whose entries are computed as

$$Z(i_1, i_2) = \begin{cases} \sum_{\ell=1}^{r} U(i_1, \ell) S(\ell, \ell) V(i_2, \ell) - B(i_1, i_2) & \text{if } (i_1, i_2) \in \Omega, \\ 0 & \text{otherwise.} \end{cases}$$

Hence the computation of $\mathcal{P}_X(Z)$ now requires two sparse matrix multiplications ZU and $Z^T V$; see [112]. For tensor completion, a little bit more care is needed but an efficient implementation for applying the tangent space projector exists; see [103, §4.2]. In all cases, the computation becomes cheaper the sparser Z is.

If on the other hand $\mathcal{L}$ is a low-rank TT matrix operator as explained in Sect. 9.3.5, and B is a low-rank TT tensor, then $Z = \mathcal{L}(X) - B$ will be also of low-rank since $X \in \mathcal{M}_{\mathbf{k}}$. This makes the tangent space projection $\mathcal{P}_X(Z)$ efficiently applicable afterwards as explained before. Operators with TT matrix structure are the most typical situation when TT tensors are used for parametric PDEs and for the Schrödinger equation; see again Sect. 9.6 later.

Regarding the computation of the step size t, we can approximate an exact line search method by minimizing the first-order approximation

$$g(t) = f(X - t\,\xi) \approx f(R_X(-t\,\xi)).$$

For quadratic functions f, the function $g(t)$ is a quadratic polynomial in t and can thus be exactly minimized. For instance, with $f_{\mathcal{L}}$ it satisfies

$$g(t) = (\xi, \mathcal{L}(\xi))_F\, t^2 - 2(\xi, \mathcal{L}(X) - B)_F\, t + \text{constant}.$$

Recall that, by (9.40), the matrix or TT rank of a tangent vector ξ is bounded by two times that of X. Hence, in the same situation as for $\mathcal{L}$ above, these inner products can be computed very efficiently. It has been observed in [112] that with this initialization of the step size almost no extra backtracking is needed.

9.4.4 Difference to Iterative Thresholding Methods

A popular algorithm for solving optimization problems with low-rank constraints, like matrix completion [49] and linear tensor systems [8, 54], is *iterative hard thresholding* (IHT).[7] It is an iteration of the form

$$X_+ = \mathcal{P}_{\mathcal{M}}(X - t\,\nabla f(X)),$$

where $\mathcal{P}_{\mathcal{M}} \colon \mathbb{V} \to \mathcal{M}$ denotes the (quasi) projection on the set $\mathcal{M}$, like the truncated SVD for low-rank matrices and TT-SVD for tensors as explained in Sects. 9.2.1 and 9.3.2. Variations of this idea also include alternating projection schemes like in [101]. Figure 9.4 compares IHT to Riemannian steepest descent. The main difference between the two methods is the extra tangent space projection $\mathcal{P}_X$ of the negative gradient $-\nabla f(X)$ for the Riemannian version. Thanks to this projection, the truncated SVD in the Riemannian case has to be applied to a tangent vector which can be implemented cheaply with direct linear algebra and is thus very reliable, as explained in Sects. 9.2.4 and 9.3.4. In IHT on the other hand, the

[7] Also called *singular value projection* and *truncated Richardson iteration*.

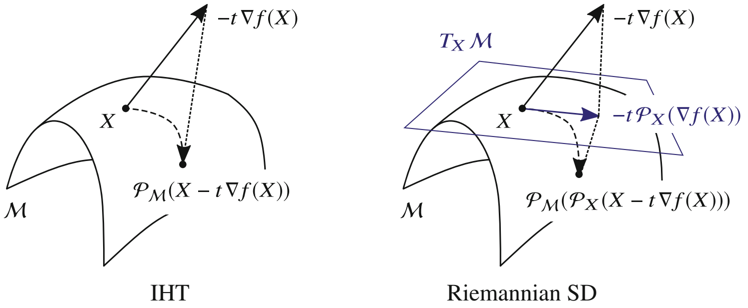

$-t\nabla f(X)$
X
$\mathcal{P}_{\mathcal{M}}(X - t\nabla f(X))$
$\mathcal{M}$
IHT
$T_X\mathcal{M}$
$-t\nabla f(X)$
X
$-t\mathcal{P}_X(\nabla f(X))$
$\mathcal{P}_{\mathcal{M}}(\mathcal{P}_X(X - t\nabla f(X)))$
$\mathcal{M}$
Riemannian SD

Fig. 9.4 Iterative hard thresholding (IHT) and Riemannian steepest descent (SD) for fixed-rank matrices and tensors

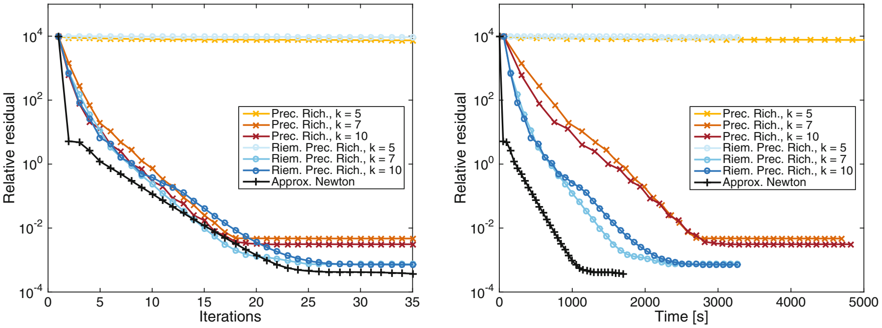


Fig. 9.5 Convergence of the Riemannian and non-Riemannian versions of preconditioned Richardson iteration. The approximation quality of the preconditioner is proportional to the k value. (We do not explain the other Riemannian method “approx. Newton”.) Picture taken from [65]. Copyright 2016 Society for Industrial and Applied Mathematics. Reprinted with permission. All rights reserved

truncated SVD is applied to a generally unstructured search direction and needs to be implemented with sparse or randomized linear algebra, which are typically less reliable and more expensive.

This difference becomes even more pronounced with preconditioning for linear systems $\mathcal{L}(X) = B$ as in (9.47). As approximate inverse of $\mathcal{L}$, the operator Q there has typically high TT matrix rank and so the additional tangent space projector in (9.47) is very beneficial compared to the seemingly more simpler *truncated preconditioned Richardson method*

$$X_+ = \mathcal{P}_{\mathcal{M}}(X - t\,Q(\mathcal{L}(X) - B)).$$

The numerical experiments from [65] confirm this behavior. For example, in Fig. 9.5 we see the convergence history when solving a Laplace-type equation with Newton potential in the low-rank Tucker format, which has not been discussed, but illustrates the same issue. Since the Newton potential is approximated by a rank 10 Tucker

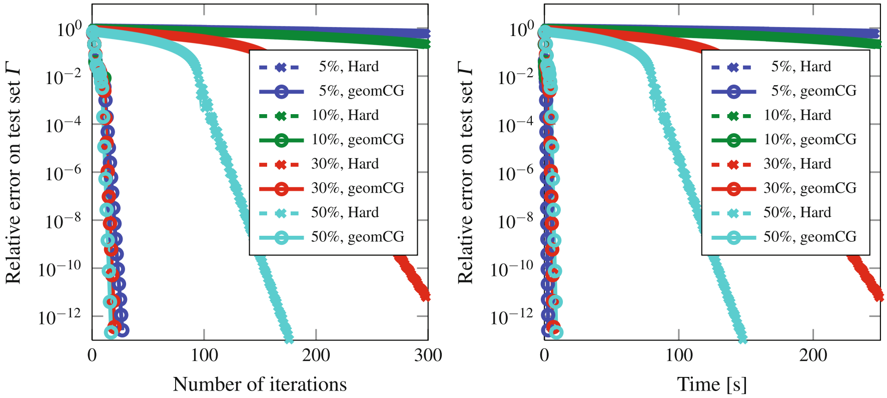


Fig. 9.6 Relative testing error in function of number of iterations (left) and time (right). For an IHT algorithm (denoted "Hard") and a Riemannian method (denoted by "geomCG") for different sampling sizes when solving the tensor completion problem. Picture taken from [64]

matrix, applying $\mathcal{QL}$ greatly increases the rank of the argument. Thanks to the tangent space projections, the time per iteration is reduced significantly and there is virtually no change in the number of iterations needed.

There is another benefit of Riemannian algorithms over more standard rank truncated schemes. Thanks to the global smoothness of the fixed-rank manifolds $\mathcal{M}$, it is relatively straightforward to accelerate manifold algorithms using non-linear CG or BFGS, and perform efficient line search. For example, Fig. 9.6 compares the Riemannian non-linear CG algorithm from [64] to a specific IHT algorithm based on nuclear norm relaxation from [101] for the low-rank tensor completion problem as explained in Sect. 9.6.3. We can see that the Riemannian algorithm takes less iterations and less time. While this example is again for fixed-rank Tucker tensors, the same conclusion is also valid for fixed-rank matrices and TT tensors; see, e.g., [112, Fig. 5.1].

9.4.5 *Convergence*

Theoretical results for Riemannian optimization parallel closely the results from Euclidean unconstrained optimization. In particular, with standard line search or trust-region techniques, limit points are guaranteed to be critical points, and additional Hessian information can enforce attraction to local minimal points; see [2]. For example, when the initial point X_1 is sufficiently close to a strict local minimizer X_* of f on $\mathcal{M}$, Riemannian gradient descent will converge exponentially fast. Specifically, if the Riemannian Hessian of f at X_* has all positive eigenvalues $\lambda_p \geq \cdots \geq \lambda_1 > 0$, then the iterates X_ℓ with exact line search satisfy the following asymptotic Q-linear convergence rate [74]:

$$\lim_{\ell \to \infty} \frac{\|X_{\ell+1} - X_*\|_F}{\|X_\ell - X_*\|_F} = \frac{\lambda_p - \lambda_1}{\lambda_p + \lambda_1} \leq 1 - \kappa, \qquad \text{with } \kappa = \frac{\lambda_1}{\lambda_p}.$$

With more practical line searches, like those that ensure the Armijo condition (9.43), this rate deteriorates but remains $1 - O(\kappa)$; see [2]. As in the Euclidean non-convex case, non-asymptotic results that are valid for arbitrary X_1 can only guarantee algebraic rates; see [12]. If however X_1 is in a region where f is locally convex, then also fast exponential convergence is guaranteed; see [107]. Results of this kind but specific to matrix completion are available in [117].

For particular problems, one can show that gradient schemes converge to the global minimum when started at any X_1. The main idea is that, while these problems are not convex, their optimization landscape is still favorable for gradient schemes in the sense that all critical points are either strict saddle points or close to a global minimum. Strict saddles are characterized as having directions of sufficient negative curvature so that they push away the iterates of a gradient scheme that might be attracted to such a saddle [76]. This property has been established in detail for matrix sensing with RIP (restricted isometry property) operators, which are essentially very well-conditioned when applied to low-rank matrices. Most of the results are formulated for particular non-Riemannian algorithms (see, e.g., [90]), but landscape properties can be directly applied to Riemannian algorithms as well; see [18, 110]. As far as we know, such landscape results have not been generalized to TT tensors but related work on completion exists [93].

9.4.6 Eigenvalue Problems

Another class of optimization problems arises when computing extremal eigenvalues of Hermitian operators. This is arguably the most important application of low-rank tensors in theoretical physics since it includes the problem of computing ground-states (eigenvectors of minimal eigenvalues) of the Schrödinger equation.

The main idea is similar to the previous section. Suppose we want to compute an eigenvector $X \in \mathbb{V}$ of a minimal eigenvalue of the Hermitian linear operator $\mathcal{H} \colon \mathbb{V} \to \mathbb{V}$. Then, instead of minimizing the Rayleigh function on $\mathbb{V}$, we restrict the optimization space to an approximation manifold:

$$\min \rho(X) = \frac{(X, \mathcal{H}(X))_F}{(X, X)_F} \quad \text{s.t.} \quad X \in \mathcal{M}.$$

Since f is homogeneous in X, the normalization $(X, X)_F = 1$ can also be imposed as a constraint:

$$\min \widetilde{\rho}(X) = (X, \mathcal{H}(X))_F \quad \text{s.t.} \quad X \in \widetilde{\mathcal{M}} = \mathcal{M} \cap \{X \colon (X, X)_F = 1\}.$$

This intersection is transversal in cases when $\mathcal{M}$ is a manifold of low-rank matrices or tensors, so $\widetilde{\mathcal{M}}$ is again a Riemannian submanifold of $\mathbb{V}$ with a geometry very similar to that of $\mathcal{M}$; see [91] for details on the matrix case. One can now proceed and apply Riemannian optimization to either problem formulation.

Standard algorithms for eigenvalue problems typically do not use pure gradient schemes. Thanks to the specific form of the problem, it is computationally feasible to find the global minimum of ρ on a small subspace of $\mathbb{V}$. This allows to enrich the gradient direction with additional directions in order to accelerate convergence. Several strategies of this type exist of which LOBPCG and Jacob–Davidson have been extended to low-rank matrices and tensors. In particular, thanks to the multilinear structure of the TT format, it is feasible to minimize globally over a subspace in one of the TT cores. Proceeding in a sweeping manner, one can mimic the Jacob–Davidson method to TT tensors; see [91, 92].

9.5 Initial Value Problems

Instead of approximating only a single (very large) matrix or tensor X by low rank, we now consider the task of approximating a *time-dependent* tensor $X(t)$ directly by a low-rank tensor $Y(t)$. The tensor $X(t)$ is either given explicitly, or more interesting, as the solution of an initial value problem (IVP)

$$\overset{\bullet}{X}(t) = F(X(t)), \quad X(t_0) = X_0 \in \mathbb{V}, \tag{9.48}$$

where $\overset{\bullet}{X}$ means dX/dt. As it is usual, we assume that F is Lipschitz continuous with constant Λ,

$$\|F(X) - F(Z)\| \leq \Lambda \, \|X - Z\| \quad \text{for all } X, Z \in \mathbb{V}, \tag{9.49}$$

so that the solution to (9.48) exists at least on some interval $[t_0, T]$. We took F autonomous, which can always be done by adding t as an extra integration parameter. For simplicity, we assume that the desired rank for the approximation $Y(t)$ is known and constant. In most applications, it will be important however that the numerical method that computes $Y(t)$ is robust to overestimation of the rank and/or allows for adapting the rank to improve the accuracy.

The aim is to obtain good approximations of $X(t)$ on the whole interval $[t_0, T]$. This is usually done by computing approximations $X_\ell \approx X(t_0 + \ell h)$ with h the time step. Classical time stepping methods for this include Runge–Kutta and BDF methods. Sometimes, one is only interested in the steady-state solution, that is, $X(t)$ for $t \to \infty$. This is for example the case for gradient flows, where F is the negative gradient of an objective function $f \colon \mathbb{V} \to \mathbb{R}$. The steady state solution of (9.48) is then a critical point of f, for example, a local minimizer. However, in such situations, it may be better to directly minimize f using methods from numerical optimization as explained in Sect. 9.4.

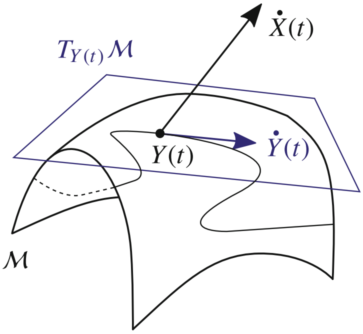


Fig. 9.7 Graphical depiction of the dynamical low-rank approximation $Y(t)$ at time t

9.5.1 Dynamical Low-Rank Approximation

We now explain how to obtain a low-rank approximation to (9.48) without needing to first solve for $X(t)$. Given an approximation submanifold $\mathcal{M} = \mathcal{M}_k$ or $\mathcal{M} = \mathcal{M}_{\mathbf{k}}$ of fixed-rank matrices or tensors, the idea is to replace $\dot{X}$ in (9.48) by the tangent vector in $\mathcal{M}$ that is closest to $F(X)$; see also Fig. 9.7. It is easy to see that for the Frobenius norm, this tangent vector is $\mathcal{P}_X(F(X))$ where $\mathcal{P}_X \colon \mathbb{V} \to T_X\mathcal{M}$ is the orthogonal projection. Applying this substitution at every time t, we obtain a new IVP

$$\dot{Y}(t) = \mathcal{P}_{Y(t)}F(Y(t)), \quad Y(t_0) = Y_0 \in \mathcal{M}, \tag{9.50}$$

where $Y_0 = \mathcal{P}_{\mathcal{M}}(X_0)$ is a quasi-best approximation of X_0 in $\mathcal{M}$. In [59], the IVP (9.50) (or its solution) is aptly called the *dynamical low-rank approximation* (DLRA) of $X(t)$. Thanks to the tangent space projection, the solution $Y(t)$ will belong to $\mathcal{M}$ as long as $\mathcal{P}_{Y(t)}$ exists, that is, until the rank of $Y(t)$ drops. In the following we assume that (9.50) can be integrated on $[t_0, T]$.

The DLRA (9.50) can equivalently be defined in weak form as follows: find, for each $t \in [t_0, T]$, an element $Y(t) \in \mathcal{M}$ such that

$$(\dot{Y}(t), Z)_F = (F(Y(t)), Z)_F \quad \text{for all } Z \in T_{Y(t)}\mathcal{M}, \tag{9.51}$$

and $Y(0) = Y_0 \in \mathcal{M}$. Observe that this can be seen as a time-dependent Galerkin condition since $T_{Y(t)}\mathcal{M}$ is a linear subspace that varies with t.

In the concrete case of low-rank matrices, DLRA appeared first in [59]. The same approximation principle, called *dynamically orthogonal* (DO), was also proposed in [94] for time-dependent stochastic PDEs. It was shown in [32, 83] that DO satisfies (9.50) after discretization of the stochastic and spatial domain. In theoretical physics, the *time-dependent variational principle* (TDVP) from [41] seems to be the first application of DLRA for simulating spin systems with uniform MPS, a variant of TT tensors. It is very likely similar ideas appeared well before since obtaining approximations in a manifold from testing with tangent vectors as in (9.51) goes back as far as 1930 with the works of Dirac [22] and Frenkel [33]. We refer to [69] for a mathematical overview of this idea in quantum physics.

9.5.2 Approximation Properties

The local error at t of replacing (9.48) by (9.50) is minimized in Frobenius norm by the choice $\dot{Y}$; see also Fig. 9.7. In order to quantity the effect of this approximation on the global error at the final time T, the simplest analysis is to assume as in [57, 58] that the vector field F is ε close to the tangent bundle of $\mathcal{M}$, that is,

$$\|F(Y(t)) - \mathcal{P}_{Y(t)}F(Y(t))\|_F \leq \varepsilon \quad \text{for all } t \in [t_0, T].$$

A simple comparison of IVPs then gives

$$\|Y(t) - X(t)\|_F \leq e^{\lambda t}\delta + (e^{\lambda t} - 1)\lambda^{-1}\varepsilon = O(\varepsilon + \delta), \tag{9.52}$$

where $\|X_0 - Y_0\|_F \leq \delta$ and λ is a *one-sided* Lipschitz constant of F satisfying[8]

$$(X - Z, F(X) - F(Z))_F \leq \lambda \, \|X - Z\|_F^2 \quad \text{for all } X, Z \in \mathbb{V}.$$

From (9.52), we observe that $Y(t)$ is guaranteed to be a good approximation of $X(t)$ but only for (relatively) short time intervals when $\lambda > 0$.

Alternatively, one can compare $Y(t)$ with a quasi-best approximation $Y_{\mathrm{qb}}(t) \in \mathcal{M}$ to $X(t)$. Assuming $Y_{\mathrm{qb}}(t)$ is continuously differentiable on $[t_0, T]$, this can be done by assuming that $\mathcal{M}$ is not too curved along $Y_{\mathrm{qb}}(t)$. In the matrix case, this means that the kth singular value of $Y_{\mathrm{qb}}(t)$ is bounded from below, i.e., there exists $\rho > 0$ such that $\sigma_k(Y_{\mathrm{qb}}(t)) \geq \rho$ for $t \in [t_0, T]$. Now a typical result from [59] is as follows: Let F be the identity operator and assume $\|\dot{X}(t)\|_F \leq c$ and $\|X(t) - Y_{\mathrm{qb}}(t)\|_F \leq \rho/16$ for $t \in [t_0, T]$. Then,

$$\|Y(t) - Y_{\mathrm{qb}}(t)\|_F \leq 2\beta e^{\beta t} \int_0^t \|Y_{\mathrm{qb}}(s) - X(s)\|_F ds \qquad \text{with } \beta = 8c\rho^{-1}$$

for $t_0 \leq t \leq \min(T, 0.55\beta^{-1})$. Hence, the approximation $Y(t)$ stays close to $Y_{\mathrm{qb}}(t)$ for short times. We refer to [59] for additional results that also include the case of F not the identity. Most of the analysis was also extended to manifolds of fixed TT rank (as well as to Tucker and hierarchical) tensors in [60, 73] and to Hilbert spaces [83].

[8] We remark that λ can be negative and is bounded from above by Λ from (9.49).

We remark that these a-priori results only hold for (very) short times. In practice, they are overly pessimistic and in actual problems the accuracy is typically much higher than theoretically predicted; see [57, 71, 72, 83, 94], and the numerical example from Fig. 9.8 further below.

9.5.3 Low-Dimensional Evolution Equations

The dynamical low-rank problem (9.50) is an IVP that evolves on a manifold $\mathcal{M}$ of fixed-rank matrices or tensors. In relevant applications, the rank will be small and hence we would like to integrate (9.50) by exploiting that $\mathcal{M}$ has low dimension.

Let us explain how this is done for $m \times n$ matrices of rank k, that is, for the manifold $\mathcal{M}_k$. Then $\operatorname{rank}(Y(t)) = k$ and we can write $Y(t) = U(t)S(t)V(t)^T$ where $U(t) \in \mathbb{R}^{m\times k}$ and $V(t) \in \mathbb{R}^{m\times k}$ have orthonormal columns and $S(t) \in \mathbb{R}^{k\times k}$. This is an SVD-like decomposition but we do not require $S(t)$ to be diagonal. The aim is now to formulate evolution equations for $U(t)$, $S(t)$, and $V(t)$.

To this end, recall from (9.10) that for fixed U, S, V every tangent vector $\dot{Y}$ has a *unique* decomposition

$$\dot{Y} = \dot{U}SV^T + U\dot{S}V^T + USV̇^T \quad \text{with } U^T\dot{U} = 0,\ V^T\dot{V} = 0.$$

Since $\dot{Y} = \mathcal{P}_Y(F(Y))$, we can isolate $\dot{U}$, $\dot{S}$, $\dot{V}$ by applying (9.12) with $Z = F(Y)$. The result is a new IVP equivalent to (9.50) but formulated in the factors:

$$\begin{cases} \dot{U} = (I - UU^T)F(Y)VS^{-1}, \\ \dot{S} = U^TF(Y)V, \\ \dot{V} = (I - VV^T)F(Y)^TUS^{-T}. \end{cases} \tag{9.53}$$

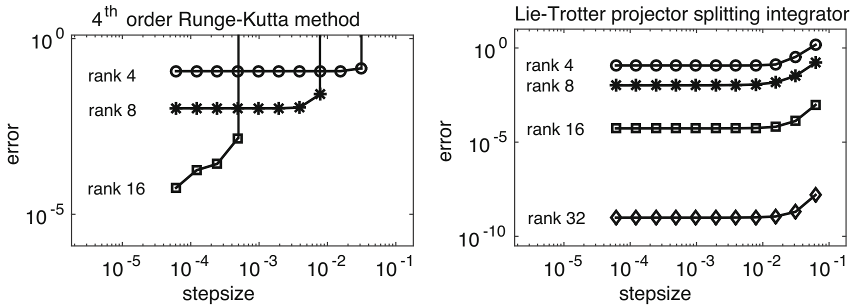


Fig. 9.8 Errors of the dynamical low-rank approximation for (9.55) integrated by a standard explicit Runge–Kutta scheme for (9.53) and the projector-splitting integrator (9.58). Picture taken from [58]. Copyright 2016 Society for Industrial and Applied Mathematics. Reprinted with permission. All rights reserved

(Here, U, S, V all depend on t.) Observe that this is a set of coupled non-linear ODEs, parametrized using $(m+n)k+k^2$ entries.

The ODE (9.53) appeared in [59, Prop. 2.1]. For DO [94], one uses the parametrization $Y(t) = U(t)M(t)^T$ where only U has orthonormal columns and obtains

$$\begin{cases} \dot{U} = (I - UU^T)F(UM^T)M(M^TM)^{-1}, \\ \dot{M} = F(UM^T)^T U. \end{cases} \tag{9.54}$$

These two coupled non-linear ODEs are very similar to (9.53) with respect to theoretical and numerical behavior. In particular, they also involve the normalization condition $U^T\dot{U} = 0$ and an explicit inverse $(M^TM)^{-1}$.

The derivation of these ODEs can be generalized to TT tensors with factored and gauged parametrizations for the tangent vectors. The equations are more tedious to write down explicitly, but relatively easy to implement. We refer to [73] for details. See also [4, 60] for application to the (hierarchical) Tucker tensor format.

For matrices and for tensors, the new IVPs have the advantage of being formulated in low dimensional parameters. However, they both suffer from a major problem: the time step in explicit methods needs to be in proportion to the smallest positive singular value of (each unfolding) of $Y(t)$. If these singular values become small (which is typically the case, since the DLRA approach by itself is reasonable for those applications where the true solution exhibits fast decaying singular values), Eq. (9.53) is very stiff. The presence of the terms S^{-1} in (9.53) and $(M^TM)^{-1}$ in (9.54) already suggests this and numerical experiments make this very clear. In Fig. 9.8, we report on the approximation errors for DLRA applied to the explicit time-dependent matrix

$$A(t) = \exp(tW_1)\,\exp(t)\,D\,\exp(tW_2), \qquad 0 \le t \le 1, \tag{9.55}$$

with W_1, W_2 being skew-symmetric of size 100×100 and D a diagonal matrix with entries $2^{-1}, \cdots, 2^{-100}$; see [58] for details. The left panel shows the results of a Runge–Kutta method applied to the resulting system (9.53). The method succeeds in computing a good low-rank approximation when the step size h is sufficiently small, but becomes unstable when h is larger than the smallest singular value of $Y(t)$. Due to this step-size restriction it hence becomes very expensive when aiming for accurate low-rank approximations. See also [57, Fig. 3] for similar results.

One solution would be to use expensive implicit methods or an ad-hoc regularization of S^{-1}. In the next subsection, a different approach is presented that is based on a splitting of the tangent space projector, and is robust to small singular values.

9.5.4 Projector-Splitting Integrator

Instead of immediately aiming for an ODE in the small factors U, S, V, the idea of the splitting integrator of [71] is to first apply a Lie splitting to the orthogonal projector $\mathcal{P}_Y$ in

$$\dot{Y}(t) = \mathcal{P}_{Y(t)} F(Y(t)), \quad Y(t_0) = Y_0 \in \mathcal{M}, \tag{9.56}$$

and then—thanks to some serendipitous observation—obtain low dimensional ODEs at a later stage. For instance, in the matrix case, as stated in (9.9), the projector can be written as

$$\mathcal{P}_Y(Z) = ZVV^T - UU^T ZVV^T + UU^T V \quad \text{with } Y = USV^T. \tag{9.57}$$

When we integrate each of these three terms consecutively (labeled a, b, c) from t_0 to $t_1 = t_0 + h$, we obtain the following scheme (all matrices depend on time):

$$\begin{cases} \dot{Y}_a = F(Y_a) V_a V_a^T, & Y_a(t_0) = Y_0 = U_a S_a V_a^T, \\ \dot{Y}_b = -U_b U_b^T F(Y_b) V_b V_b^T, & Y_b(t_0) = Y_a(t_1) = U_b S_b V_b^T, \\ \dot{Y}_c = U_c U_c^T F(Y_c), & Y_c(t_0) = Y_b(t_1) = U_c S_c V_c^T. \end{cases} \tag{9.58}$$

Here, all U_x and V_x are matrices with orthonormal columns. Observe the minus sign for Y_b. The result $Y_c(t_1)$ is an $O(h^2)$ approximation to $Y(t_1)$. We then repeat this scheme starting at $Y_c(t_1)$ and integrate from t_1 to $t_2 = t_1 + h$, and so on. By standard theory for Lie splittings, this scheme is first-order accurate for (9.56), that is, $\|Y_c(T) - Y(T)\|_F = O(h)$ where $T = \ell h$.

To integrate (9.58) we will first write it using much smaller matrices. To this end, observe that with exact integration $Y_a(t_1) \in \mathcal{M}_k$ since $\dot{Y}_a \in T_{Y_a}\mathcal{M}_k$ and $Y_a(t_0) \in \mathcal{M}_k$. Hence, we can substitute the ansatz $Y_a(t) = U_a(t) S_a(t) V_a(t)^T$ in the first substep and obtain

$$\dot{Y}_a = \frac{d}{dt}[U_a(t) S_a(t)] V_a(t)^T + U_a(t) S_a(t) \dot{V}_a(t)^T = F(Y_a(t)) V_a(t) V_a(t)^T.$$

Judiciously choosing $\dot{V}_a(t) = 0$, we can simplify to

$$V_a(t) = V_a(t_0), \quad \frac{d}{dt}[U_a(t) S_a(t)] = F(Y_a(t)) V_a(t_0).$$

Denoting $K(t) = U_a(t) S_a(t)$, the first substep is therefore equivalent to

$$\dot{K}(t) = F(K(t) V_a(t_0)^T) V_a(t_0), \quad K(t_0) = U_a(t_0) S(t_0). \tag{9.59}$$

Contrary to the earlier formulation, this is an IVP for an $n \times k$ matrix $K(t)$. The orthonormal matrix U_b for the next substep can be computed in $O(nk^2)$ work by a QR decomposition of $K(t_1)$.

The second and third substeps can be integrated analogously in terms of evolution equations only for $S_b(t)$ and $L_c(t) = V_c(t) S_c(t)$. Also note that we can take $V_b = V_a$ and $U_c = U_b$. We thus get a scheme, called KSL, that integrates in order K, S, and L. A second-order accurate scheme is the symmetric Strang splitting: one step consists of computing the K, S, L substeps for F with $h/2$ and afterwards the L, S, K substeps for the adjoint of F with $h/2$.

In both versions of the splitting scheme, care must be taken in the integration of the substeps since they are computationally the most expensive part. Fortunately, the ODEs in the substeps are formally of the same form as the original equation for the vector field $F(Y)$ since the projected subspace is constant; see, e.g., $V_a(t_0)$ in (9.59). This means that one can usually adapt specialized integrators for $F(Y)$. In [28], for example, the substeps arising from the Vlasov–Poisson equations in plasma physics (see also Sect. 9.6.5) can be integrated by spectral or semi-Lagrangian methods. In addition, when F is linear and has low TT matrix rank, the large matrix $K(t)V_a(t_0)^T$ in (9.59), for example, does not need to be formed explicitly when evaluating F. As illustration, for the Lyapunov operator $F(Z) = LZ + ZL^T$, the equation for K becomes

$$\dot{K}(t) = F(K(t)V_a(t_0)^T)V_a(t_0) = LK(t) + K(t)L_a, \quad L_a = V_a(t_0)^T L V_a(t_0)$$

where $L \in \mathbb{R}^{n\times n}$ is large but usually sparse, and $L_a \in \mathbb{R}^{k\times k}$ is small. Hence, an exponential integrator with a Krylov subspace method is ideally suited to integrate $K(t)$; see, e.g., [70].

Let us finish by summarizing some interesting properties of the splitting integrator for matrices. Let Y_ℓ be the solution after ℓ steps of the scheme explained above with step size h. For simplicity, we assume that each substep is solved exactly (or sufficiently accurately). Recall that $X(t)$ is the solution to the original ODE (9.48) that we approximate with the dynamical low-rank solution $Y(t)$ of (9.50).

(a) *Exactness [71, Thm. 4.1]*: If the solution $X(t)$ of the original ODE lies on $\mathcal{M}_k$, then $Y_\ell = X(t_\ell)$ when F equals the identity.

(b) *Robustness to small singular values [58, Thm. 2.1]*: There is no step size restriction due to small singular values. Concretely, under the same assumptions that lead to (9.52), the approximation error satisfies

$$\|Y_\ell - X(t_0 + \ell h)\|_F \le C(\delta + \varepsilon + h) \quad \text{for all } \ell, h \text{ such that } t_0 + \ell h \le T$$

where C only depends on F and T. Observe that this only introduced the time step error h. For the integration error $\|Y_\ell - Y(t_0 + \ell h)\|_F$, a similar bound exists in [71, Thm. 4.2] but it requires rather technical assumptions on F.

(c) *Norm and energy conservation [70, Lemma 6.3]*: If the splitting scheme is applied to complex tensors for a Hamiltonian problem $F(X) = -i\mathcal{H}(X)$ with complex Hermitian $\mathcal{H}$, the Frobenius norm and energy are preserved:

$$\|Y_\ell\|_F = \|Y_0\|_F \quad \text{and} \quad \langle Y_\ell, \mathcal{H}(Y_\ell)\rangle_F = \langle Y_0, \mathcal{H}(Y_0)\rangle_F \quad \text{for all } n.$$

In the case of real tensors, the norm is preserved if $\langle F(Y), Y\rangle_F = 0$.

Property (a) does not seem very useful but it is key to showing the much more relevant property (b). All three properties are not shared when solving (9.53) by a standard integrator, like explicit Runge–Kutta. Even more, properties (a) and (b) are also lost for a different ordering of the splitting scheme, like KLS, even though that would still result in a first-order scheme. We remark that these properties also hold for the solution $Y(t)$ of the continuous problem by (formally) replacing h by 0.

To extend the idea of projector splitting to TT tensors $Y(t) \in \mathcal{M}_{\mathbf{k}}$, the correct splitting of the tangent space projector $\mathcal{P}_Y : \mathbb{V} \to T_Y \mathcal{M}_{\mathbf{k}}$ has to be determined. The idea in [72] is to take the sum expression (9.39) and split it as

$$\mathcal{P}_Y = \mathcal{P}_1^+ - \mathcal{P}_1^- + \mathcal{P}_2^+ - \mathcal{P}_2^- \cdots - \mathcal{P}_{d-1}^- + \mathcal{P}_d^+ \tag{9.60}$$

where

$$\mathcal{P}_\mu^+(Z) = \mathcal{P}_{\le \mu-1}(\mathcal{P}_{\ge \mu+1}(Z)) \quad \text{and} \quad \mathcal{P}_\mu^-(Z) = \mathcal{P}_{\le \mu}(\mathcal{P}_{\ge \mu+1}(Z)).$$

Observe that $\mathcal{P}_\mu^\pm$ depends on Y and that this splitting reduces to the matrix case in (9.57) when $d = 2$. The projector-splitting integrator for TT is now obtained by integrating each term in (9.60) from left to right:

$$\begin{cases} \dot{Y}_1^+ = \mathcal{P}_1^+(F(Y_1^+)), & Y_1^+(t_0) = Y_0, \\ \dot{Y}_1^- = -\mathcal{P}_1^-(F(Y_1^-)), & Y_1^-(t_0) = Y_1^+(t_1), \\ \dot{Y}_2^+ = \mathcal{P}_2^+(F(Y_2^+)), & Y_2^+(t_0) = Y_1^-(t_1), \\ \quad\vdots & \quad\vdots \\ \dot{Y}_d^+ = \mathcal{P}_d^+(F(Y_d^+)), & Y_d^+(t_0) = Y_{d-1}^-(t_1). \end{cases} \tag{9.61}$$

Quite remarkably, this splitting scheme for TT tensors shares many of the important properties from the matrix case. In particular, it allows for an efficient integration since only one core varies with time in each substep (see [72, Sec. 4]) and it is robust to small singular values in each unfolding (see [58, Thm. 3.1]). We refer to [42] for more details on its efficient implementation and its application to quantum spin systems in theoretical physics.

9.6 Applications

In this section, we explain different types of problems that have been solved by low-rank matrix and tensor methods in the literature. We will in particular focus on problems that can be approached by the geometry-oriented methods considered in this chapter, either via optimization on low-rank manifolds or via dynamical low-rank integration. Our references to the literature are meant to give a broad and recent view of the usefulness of these methods, but we do not claim they are exhaustive.

9.6.1 Matrix Equations

In control and systems theory (see, e.g., [3]), a number of applications requires solving the following types of matrix equations:

$$\begin{aligned} \text{Lyapunov:} \quad & AX + XA^T = C, \\ \text{Sylvester:} \quad & AX + XB = C, \\ \text{Riccati:} \quad & AX + XA^T + XBX = C. \end{aligned} \tag{9.62}$$

Here, A, B, C are given matrices and X is the unknown matrix (of possible different size in each equation). The first two equations are linear, whereas the second is quadratic. For simplicity, we assume that these equations are uniquely solvable but there exist detailed results about conditions for this.

In large-scale applications, the matrix X is typically dense and too large to store. Under certain conditions, one can prove that X has fast decaying singular values and can thus be well approximated by a low-rank matrix; see [102] for an overview. For the linear equations, one can then directly attempt the optimization strategy explained in Sect. 9.4.2 and minimize the residual function or the energy-norm error. The latter is preferable but only possible when A and B are symmetric and positive definite; see [111] for a comparison. If the underlying matrices are ill-conditioned, as is the case with discretized PDEs, a simple Riemannian gradient scheme will not be effective and one needs to precondition the gradient steps or perform a quasi-Newton method. For example, in case of the Lyapunov equation, it is shown in [113] how to efficiently solve the Gauss–Newton equations for the manifold $\mathcal{M}_k$. If the Riccati equation is solved by Newton's method, each step requires solving a Sylvester equation [102]. When aiming for low-rank approximations, the latter can again be solved by optimization on $\mathcal{M}_k$; see [81]. We remark that while most methods for calculating low-rank approximations to (9.62) are based on Krylov subspaces and rational approximations, there exists a relation between both approaches; see [11].

The matrix equations from above have direct time-dependent versions. For example, the differential Riccati equation is given by

$$\dot{X}(t) = AX(t) + X(t)A^T + G(t, X(t)), \quad X(t_0) = X_0, \tag{9.63}$$

where $G(t, X(t)) = C - X(t)BX(t)$. Uniqueness of the solution $X(t)$ for all $t \geq t_0$ is guaranteed when X_0, C, and B are symmetric and positive semi-definite [21]. In optimal control, the linear quadratic regulator problem with finite time horizon requires solving (9.63). In the large-scale case, it is typical that X_0 and C are low rank and it has been observed [20, 76] that $X(t)$ has then fast decaying singular values, even on infinite time horizons.

Other examples are the differential Lyapunov equation ($G(t, X) = 0$) and the generalized differential Riccati equation ($G(t, X(t)) = C + \sum_{j=1}^{J} D_j^T X(t) D_j - X(t) B X(t)$); see, e.g. [20, 75], for applications. When matrices are large, it is important to exploit that applying the right hand side in (9.63) does not increase the rank of $X(t)$ too much, which is guaranteed here, if J is not too large and the matrix C is of low rank. In [89] a low-rank approximation to $X(t)$ is obtained with the dynamical low-rank algorithm. Like in the time-independent case, discretized PDEs might need special treatment to cope with the stiff ODEs. In particular, an exponential integrator can be combined with the projector-splitting integrator by means of an additional splitting of the vector field for the stiff part; see [89] for details and analysis.

9.6.2 Schrödinger Equation

Arguably the most typical example involving tensors of very high order is the time-dependent Schrödinger equation,

$$\dot{\psi} = -i\mathsf{H}(\psi),$$

where H is a self-adjoint Hamiltonian operator acting on a (complex-valued) multi-particle wave function $\psi(x_1, \dots, x_d, t)$ with $x_\mu \in \mathbb{R}^p$, $p \leq 3$. This equation is fundamental in theoretical physics for the simulation of elementary particles and molecules. Employing a Galerkin discretization with $i = 1, \dots, n_\mu$ basis functions $\varphi_i^{(\mu)}$ in each mode $\mu = 1, \dots, d$, the wave function will be approximated as

$$\psi(x_1, \dots, x_d, t) \approx \sum_{i_1}^{n_1} \cdots \sum_{i_d}^{n_d} X(i_1, \dots, i_d; t)\, \varphi_{i_1}^{(1)}(x_1) \cdots \varphi_{i_d}^{(d)}(x_d).$$

By regarding the unknown complex coefficient $X(i_1, \dots, i_d; t)$ as the $(i_1, \dots, i_d)$th element of the time-dependent tensor $X(t)$ of size $n_1 \times \cdots \times n_d$, we obtain the linear differential equation

$$\dot{X}(t) = -i\mathcal{H}(X(t)) \tag{9.64}$$

where $\mathcal{H}$ is the Galerkin discretization of the Hamiltonian H. More complicated versions of this equation allow the Hamiltonian to be time-dependent.

The size of the tensor $X(t)$ will be unmanageable for large d but, fortunately, certain systems allow it to be approximated by a low-rank tensor. For example,

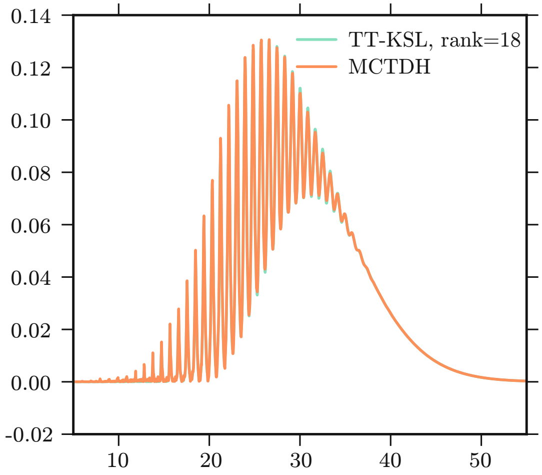


Fig. 9.9 Spectrum computed by the second-order splitting integrator and by the MCTDH package. Picture taken from [72]. Copyright 2015 Society for Industrial and Applied Mathematics. Reprinted with permission. All rights reserved

in the multilayer multiconfiguration time-dependent Hartree model (ML-MCDTH) in [78, 116] for simulating quantum dynamics in small molecules, the wave functions are approximated by hierarchical Tucker tensors. Spin systems in theoretical physics, on the other hand, employ the TT format and can simulate systems of very large dimension (since n_μ are small); see [97] for an overview. For both application domains, the solution of (9.64) can be obtained by applying dynamical low-rank; see, e.g., [77] for MCDTH and [41] for spin systems.

Numerical experiments for (9.64) with the Henon–Heiles potential were performed in [72]. There the second-order splitting integrator with a fixed time step $h = 0.01$ and a fixed TT rank of 18 was compared to an adaptive integration of the gauged ODEs, similar to (9.53). In particular, the ML-MCDTH method [10] was used in the form of the state-of-the art code `mcdth v8.4`. Except for the slightly different tensor formats (TT versus hierarchical Tucker) all other modeling parameters are the same. For similar accuracy, a 10 dimensional problem is integrated by `mcdth` in 54 354 s, whereas the TT splitting integrator required only 4425 s. The reason for this time difference was mainly due to the ill conditioned ODEs in the gauged representation. In addition, there was no visible difference in the Fourier transform of the auto-correlation functions; see Fig. 9.9.

There is an interesting link between the computation of the ground state (eigenvector of the minimal eigenvalue) of $\mathcal{H}$ via the minimization of the Rayleigh quotient

$$\rho(X) = \frac{(X, \mathcal{H}(X))_F}{(X, X)_F}$$

and so-called *imaginary time evolution* [41] for a scaled version of (9.64) that conserves unit norm. The latter is a formal way to obtain a gradient flow for $\rho(X)$ using imaginary time $\tau = -it$ by integrating

$$\dot{X} = -\mathcal{H}(X) + (X, \mathcal{H}(X))_F \, X.$$

For both approaches, we can approximate their solutions with low-rank tensors, as we explained before, either via optimization or dynamical low rank on $\mathcal{M}_{\mathbf{k}}$. However, methods based on optimization of the multilinear TT representation of X remain the more popular approach since they easily allow to reuse certain techniques from standard eigenvalue problems, like subspace corrections, as is done in the DMRG [118] or AMEn [23, 63] algorithm.

For an overview on tensor methods in quantum physics and chemistry we refer to [51, 97, 105].

9.6.3 Matrix and Tensor Completion

Let $\Omega \subset \{1, \dots, m\} \times \{1, \dots, n\}$ be a sampling set. The problem of matrix completion consists of recovering an unknown $m \times n$ matrix M of rank k based only on the values $M(i, j)$ for all $(i, j) \in \Omega$. Remarkably, this problem has a unique solution if $|\Omega| \approx O(\dim \mathcal{M}_k) = O(k(m+n))$ and under certain randomness conditions on Ω and M; see [14]. If the rank k is known, this suggests immediately the strategy of recovering M by minimizing the least-squares fit

$$f(X) = \sum_{i=1}^{m} \sum_{j=1}^{n} (X(i, j) - M(i, j))^2 = \|\mathcal{P}_\Omega(X - M)\|_F^2$$

on the manifold $\mathcal{M}_k$, where $\mathcal{P}_\Omega$ is the orthogonal projection onto matrices that vanish outside of Ω. Since $\mathcal{P}_\Omega$ is well-conditioned on $\mathcal{M}_k$ when the iterates satisfy an incoherence property, the simple Riemannian gradient schemes that we explained above perform very well in recovering M; see, e.g., [82, 112].

The problem of matrix completion can be generalized to tensors, and Riemannian methods for tensor completion have been developed for the Tucker format in [64], and for the TT format in [103].

In addition, instead of element-wise sampling, the observations can also be constructed from a general linear operator $\mathcal{S} \colon \mathbb{V} \to \mathbb{R}^q$. This problem remains well-posed under certain randomness conditions on $\mathcal{S}$ and also Riemannian optimization performs well if applied to the least-square version of the problem for which $\mathcal{L} = \mathcal{S}^T \mathcal{S}$; see [117].

9.6.4 Stochastic and Parametric Equations

Other interesting applications for low-rank tensors arise from stochastic or parametric PDEs [7, 9, 24, 26, 31, 54, 55]. For simplicity, suppose that the system matrix of a finite-dimensional linear system $Ax = b$ of equations depends on p parameters $\omega^{(1)}, \ldots, \omega^{(p)}$, that is,

$$A(\omega^{(1)}, \ldots, \omega^{(p)})\, x(\omega^{(1)}, \ldots, \omega^{(p)}) = b. \tag{9.65}$$

One might be interested in the solution $x \in \mathbb{R}^n$ for some or all choices of parameters, or, in case the parameters are random variables, in expectation values of certain quantities of interest.

By discretizing each parameter $\omega^{(\mu)}$ with m_μ values, we can gather all the $m_1 \cdots m_p$ solution vectors x into one tensor

$$[X(j, i_1, \ldots, i_p)] = [x_j(\omega^{(1)}_{i_1}, \ldots, \omega^{(p)}_{i_p})]$$

of order $p + 1$ and size $n \times m_1 \times \cdots \times m_p$. When A depends analytically on $\omega = (\omega^{(1)}, \ldots, \omega^{(p)})$, the tensor X can be shown [62] to be well approximated by low TT rank and it satisfies a very large linear system $\mathcal{L}(X) = B$. If $\mathcal{L}$ is a TT matrix of low rank, we can then approximate X on $\mathcal{M}_{\mathbf{k}}$ by the optimization techniques we discussed in Sect. 9.4. This is done, for example, in [65] with an additional preconditioning of the gradient.

A similar situation arises for elliptic PDEs with stochastic coefficients. After truncated Karhunen–Loève expansion, one can obtain a deterministic PDE of the same form as (9.65); see [106]. The following time-dependent example with random variable ω,

$$\partial_t \psi + v(t, x; \omega) \cdot \nabla \psi = 0, \qquad \psi(0, x) = x,$$

was solved by dynamical low-rank in [32].

9.6.5 Transport Equations

Transport equations describe (densities of) particles at position $x \in \mathbb{R}^p$ and velocity $v \in \mathbb{R}^p$. They are typically more challenging to integrate than purely diffusive problems. For example, the Vlasov equation

$$\partial_t u(t, x, v) + v \cdot \nabla_x u(t, x, v) - F(u) \cdot \nabla_v u(t, x, v) = 0 \tag{9.66}$$

is a kinetic model for the density u of electrons in plasma. The function F is a nonlinear term representing the force. These equations can furthermore be coupled with Maxwell's equations resulting in systems that require specialized integrators to preserve conservation laws in the numerical solution. After spatial discretization on a tensor product grid, Eq. (9.66) becomes a differential equation for a large tensor of order $d = 6$. In the case of the Vlasov–Poisson and Vlasov–Maxwell equations, [28, 30] show the splitting integrator gives very good approximations with modest TT rank, even over relatively large time intervals. In addition, the numerical integration of the substeps can be modified to ensure better preservation of some conservation laws; see [29, 30].

Similar approaches appear for weakly compressible fluid flow with the Boltzmann equation in [27] and stochastic transport PDEs in [32]. The latter also shows that numerical filters can be used in combination with dynamical low-rank to successfully reduce artificial oscillations.

9.7 Conclusions

In this chapter we have shown how the geometry of low-rank matrices and TT tensors can be exploited in algorithms. We focused on two types of problems: Riemannian optimization for solving large linear systems and eigenvalue problems, and dynamical low-rank approximation for initial value problems. Our aim was to be sufficiently explanatory without sacrificing readability and we encourage the interested reader to refer to the provided references for a more in depth treatment of these subjects.

Several things have not been discussed in this introductory chapter. The most important issue is arguably the rank adaptation during the course of the algorithms to match the desired tolerance at convergence. For this, truncation of singular values with a target error instead of a target rank can be used both for matrices and TT tensors, but from a conceptual perspective such an approach is at odds with algorithms that are defined on manifolds of fixed rank matrices or tensors. However, it is possible to combine geometric methods with rank adaptivity as in [109] for greedy rank-one optimization and in [42] for a two-site version of the splitting scheme for time integration, yet many theoretical and implementation questions remain. Other important topics not covered are the problem classes admitting a-priori low-rank approximability [19, 44], the application of low-rank formats to seemingly non-high dimensional problems like quantized TT (QTT) [52, 53], the efficient numerical implementation of truly large-scale and stiff problems, schemes with guaranteed and optimal convergence as in [5], and more general tensor networks like PEPS [114].

References

1. Absil, P.A., Oseledets, I.V.: Low-rank retractions: a survey and new results. Comput. Optim. Appl. **62**(1), 5–29 (2015)
2. Absil, P.A., Mahony, R., Sepulchre, R.: Optimization Algorithms on Matrix Manifolds. Princeton University Press, Princeton (2008)
3. Antoulas, A.C.: Approximation of Large-Scale Dynamical Systems. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2005)

4. Arnold, A., Jahnke, T.: On the approximation of high-dimensional differential equations in the hierarchical Tucker format. BIT **54**(2), 305–341 (2014)
5. Bachmayr, M., Dahmen, W.: Adaptive near-optimal rank tensor approximation for high-dimensional operator equations. Found. Comput. Math. **15**(4), 839–898 (2015)
6. Bachmayr, M., Schneider, R., Uschmajew, A.: Tensor networks and hierarchical tensors for the solution of high-dimensional partial differential equations. Found. Comput. Math. **16**(6), 1423–1472 (2016)
7. Bachmayr, M., Cohen, A., Dahmen, W.: Parametric PDEs: sparse or low-rank approximations? IMA J. Numer. Anal. **38**(4), 1661–1708 (2018)
8. Ballani, J., Grasedyck, L.: A projection method to solve linear systems in tensor format. Numer. Linear Algebra Appl. **20**(1), 27–43 (2013)
9. Ballani, J., Grasedyck, L.: Hierarchical tensor approximation of output quantities of parameter-dependent PDEs. SIAM/ASA J. Uncertain. Quantif. **3**(1), 852–872 (2015)
10. Beck, M.H., Jäckle, A., Worth, G.A., Meyer, H.D.: The multiconfiguration time-dependent Hartree (MCTDH) method: a highly efficient algorithm for propagating wavepackets. Phys. Rep. **324**(1), 1–105 (2000)
11. Benner, P., Breiten, T.: On optimality of approximate low rank solutions of large-scale matrix equations. Syst. Control Lett. **67**, 55–64 (2014)
12. Boumal, N., Absil, P.A., Cartis, C.: Global rates of convergence for nonconvex optimization on manifolds. IMA J. Numer. Anal. **39**(1), 1–33 (2019)
13. Breiding, P., Vannieuwenhoven, N.: A Riemannian trust region method for the canonical tensor rank approximation problem. SIAM J. Optim. **28**(3), 2435–2465 (2018)
14. Candès, E.J., Tao, T.: The power of convex relaxation: near-optimal matrix completion. IEEE Trans. Inform. Theory **56**(5), 2053–2080 (2010)
15. Cichocki, A., Mandic, D., De Lathauwer, L., Zhou, G., Zhao, Q., Caiafa, C., Phan, H.A.: Tensor decompositions for signal processing applications: from two-way to multiway component analysis. IEEE Signal Proc. Mag. **32**(2), 145–163 (2015)
16. Cichocki, A., Lee, N., Oseledets, I., Phan, A.H., Zhao, Q., Mandic, D.P.: Tensor networks for dimensionality reduction and large-scale optimization. Part 1: low-rank tensor decompositions. Found. Trends Mach. Learn. **9**(4–5), 249–429 (2016)
17. Cichocki, A., Phan, A.H., Zhao, Q., Lee, N., Oseledets, I., Sugiyama, M., Mandic, D.P.: Tensor networks for dimensionality reduction and large-scale optimization. Part 2: applications and future perspectives. Found. Trends Mach. Learn. **9**(6), 431–673 (2017)
18. Criscitiello, C., Boumal, N.: Efficiently escaping saddle points on manifolds (2019). arXiv:1906.04321
19. Dahmen, W., DeVore, R., Grasedyck, L., Süli, E.: Tensor-sparsity of solutions to high-dimensional elliptic partial differential equations. Found. Comput. Math. **16**(4), 813–874 (2016)
20. Damm, T., Mena, H., Stillfjord, T.: Numerical solution of the finite horizon stochastic linear quadratic control problem. Numer. Linear Algebra Appl. **24**(4), e2091, 11 (2017)
21. Dieci, L., Eirola, T.: Positive definiteness in the numerical solution of Riccati differential equations. Numer. Math. **67**(3), 303–313 (1994)
22. Dirac, P.A.M.: Note on exchange phenomena in the Thomas atom. Proc. Camb. Philos. Soc. **26**, 376–385 (1930)
23. Dolgov, S.V., Savostyanov, D.V.: Alternating minimal energy methods for linear systems in higher dimensions. SIAM J. Sci. Comput. **36**(5), A2248–A2271 (2014)
24. Dolgov, S., Khoromskij, B.N., Litvinenko, A., Matthies, H.G.: Polynomial chaos expansion of random coefficients and the solution of stochastic partial differential equations in the tensor train format. SIAM/ASA J. Uncertain. Quantif. **3**(1), 1109–1135 (2015)
25. Eckart, C., Young, G.: The approximation of one matrix by another of lower rank. Psychometrika **1**(3), 211–218 (1936)
26. Eigel, M., Pfeffer, M., Schneider, R.: Adaptive stochastic Galerkin FEM with hierarchical tensor representations. Numer. Math. **136**(3), 765–803 (2017)

27. Einkemmer, L.: A low-rank algorithm for weakly compressible flow. SIAM J. Sci. Comput. **41**(5), A2795–A2814 (2019)
28. Einkemmer, L., Lubich, C.: A low-rank projector-splitting integrator for the Vlasov-Poisson equation. SIAM J. Sci. Comput. **40**(5), B1330–B1360 (2018)
29. Einkemmer, L., Lubich, C.: A quasi-conservative dynamical low-rank algorithm for the Vlasov equation. SIAM J. Sci. Comput. **41**(5), B1061–B1081(2019)
30. Einkemmer, L., Ostermann, A., Piazzola, C.: A low-rank projector-splitting integrator for the Vlasov–Maxwell equations with divergence correction (2019). arXiv:1902.00424
31. Espig, M., Hackbusch, W., Litvinenko, A., Matthies, H.G., Wähnert, P.: Efficient low-rank approximation of the stochastic Galerkin matrix in tensor formats. Comput. Math. Appl. **67**(4), 818–829 (2014)
32. Feppon, F., Lermusiaux, P.F.J.: A geometric approach to dynamical model order reduction. SIAM J. Matrix Anal. Appl. **39**(1), 510–538 (2018)
33. Frenkel, J.: Wave Mechanics: Advanced General Theory. Clarendon Press, Oxford (1934)
34. Golub, G., Kahan, W.: Calculating the singular values and pseudo-inverse of a matrix. SIAM J. Numer. Anal. **2**(2), 205–224 (1965)
35. Golub, G.H., Van Loan, C.F.: Matrix Computations, 4th edn. Johns Hopkins University Press, Baltimore (2013)
36. Grasedyck, L.: Hierarchical singular value decomposition of tensors. SIAM J. Matrix Anal. Appl. **31**(4), 2029–2054 (2010)
37. Grasedyck, L., Kressner, D., Tobler, C.: A literature survey of low-rank tensor approximation techniques. GAMM-Mitt. **36**(1), 53–78 (2013)
38. Grohs, P., Hosseini, S.: Nonsmooth trust region algorithms for locally Lipschitz functions on Riemannian manifolds. IMA J. Numer. Anal. **36**(3), 1167–1192 (2016)
39. Hackbusch, W.: Tensor Spaces and Numerical Tensor Calculus. Springer, Heidelberg (2012)
40. Hackbusch, W., Kühn, S.: A new scheme for the tensor representation. J. Fourier Anal. Appl. **15**(5), 706–722 (2009)
41. Haegeman, J., Cirac, I., Osborne, T., Piźorn, I., Verschelde, H., Verstraete, F.: Time-dependent variational principle for quantum lattices. Phys. Rev. Lett. **107**(7), 070601 (2011)
42. Haegeman, J., Lubich, C., Oseledets, I., Vandereycken, B., Verstraete, F.: Unifying time evolution and optimization with matrix product states. Phys. Rev. B **94**(16), 165116 (2016)
43. Halko, N., Martinsson, P.G., Tropp, J.A.: Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions. SIAM Rev. **53**(2), 217–288 (2011)
44. Hastings, M.B.: An area law for one-dimensional quantum systems. J. Stat. Mech. Theory Exp. **2007**, P08024 (2007)
45. Helmke, U., Shayman, M.A.: Critical points of matrix least squares distance functions. Linear Algebra Appl. **215**, 1–19 (1995)
46. Holtz, S., Rohwedder, T., Schneider, R.: On manifolds of tensors of fixed TT-rank. Numer. Math. **120**(4), 701–731 (2012)
47. Hosseini, S., Uschmajew, A.: A Riemannian gradient sampling algorithm for nonsmooth optimization on manifolds. SIAM J. Optim. **27**(1), 173–189 (2017)
48. Hosseini, S., Huang, W., Yousefpour, R.: Line search algorithms for locally Lipschitz functions on Riemannian manifolds. SIAM J. Optim. **28**(1), 596–619 (2018)
49. Jain, P., Meka, R., Dhillon, I.S.: Guaranteed rank minimization via singular value projection. In: Advances in Neural Information Processing Systems, vol. 23, pp. 937–945 (2010)
50. Kazeev, V.A., Khoromskij, B.N.: Low-rank explicit QTT representation of the Laplace operator and its inverse. SIAM J. Matrix Anal. Appl. **33**(3), 742–758 (2012)
51. Khoromskaya, V., Khoromskij, B.N.: Tensor Numerical Methods in Quantum Chemistry. De Gruyter, Berlin (2018)
52. Khoromskij, B.N.: $O(d \log N)$-quantics approximation of N-d tensors in high-dimensional numerical modeling. Constr. Approx. **34**(2), 257–280 (2011)
53. Khoromskij, B.N.: Tensor Numerical Methods in Scientific Computing. De Gruyter, Berlin (2018)

54. Khoromskij, B.N., Oseledets, I.: Quantics-TT collocation approximation of parameter-dependent and stochastic elliptic PDEs. Comput. Methods Appl. Math. **10**(4), 376–394 (2010)
55. Khoromskij, B.N., Schwab, C.: Tensor-structured Galerkin approximation of parametric and stochastic elliptic PDEs. SIAM J. Sci. Comput. **33**(1), 364–385 (2011)
56. Khoromskij, B.N., Oseledets, I.V., Schneider, R.: Efficient time-stepping scheme for dynamics on TT-manifolds (2012). MPI MiS Preprint 24/2012
57. Kieri, E., Vandereycken, B.: Projection methods for dynamical low-rank approximation of high-dimensional problems. Comput. Methods Appl. Math. **19**(1), 73–92 (2019)
58. Kieri, E., Lubich, C., Walach, H.: Discretized dynamical low-rank approximation in the presence of small singular values. SIAM J. Numer. Anal. **54**(2), 1020–1038 (2016)
59. Koch, O., Lubich, C.: Dynamical low-rank approximation. SIAM J. Matrix Anal. Appl. **29**(2), 434–454 (2007)
60. Koch, O., Lubich, C.: Dynamical tensor approximation. SIAM J. Matrix Anal. Appl. **31**(5), 2360–2375 (2010)
61. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. SIAM Rev. **51**(3), 455–500 (2009)
62. Kressner, D., Tobler, C.: Low-rank tensor Krylov subspace methods for parametrized linear systems. SIAM J. Matrix Anal. Appl. **32**(4) (2011)
63. Kressner, D., Steinlechner, M., Uschmajew, A.: Low-rank tensor methods with subspace correction for symmetric eigenvalue problems. SIAM J. Sci. Comput. **36**(5), A2346–A2368 (2014)
64. Kressner, D., Steinlechner, M., Vandereycken, B.: Low-rank tensor completion by Riemannian optimization. BIT **54**(2), 447–468 (2014)
65. Kressner, D., Steinlechner, M., Vandereycken, B.: Preconditioned low-rank Riemannian optimization for linear systems with tensor product structure. SIAM J. Sci. Comput. **38**(4), A2018–A2044 (2016)
66. Lee, J.M.: Introduction to Smooth Manifolds. Springer, New York (2003)
67. Lehoucq, R.B., Sorensen, D.C., Yang, C.: ARPACK users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (1998)
68. Lewis, A.S., Malick, J.: Alternating projections on manifolds. Math. Oper. Res. **33**(1), 216–234 (2008)
69. Lubich, C.: From Quantum to Classical Molecular Dynamics: Reduced Models and Numerical Analysis. European Mathematical Society (EMS), Zürich (2008)
70. Lubich, C.: Time integration in the multiconfiguration time-dependent Hartree method of molecular quantum dynamics. Appl. Math. Res. Express. AMRX **2015**(2), 311–328 (2015)
71. Lubich, C., Oseledets, I.: A projector-splitting integrator for dynamical low-rank approximation. BIT **54**(1), 171–188 (2014)
72. Lubich, C., Oseledets, I., Vandereycken, B.: Time integration of tensor trains. SIAM J. Numer. Anal. **53**(2), 917–941 (2015)
73. Lubich, C., Rohwedder, T., Schneider, R., Vandereycken, B.: Dynamical approximation of hierarchical Tucker and tensor-train tensors. SIAM J. Matrix Anal. Appl. **34**(2), 470–494 (2013)
74. Luenberger, D.G.: The gradient projection method along geodesics. Manage. Sci. **18**, 620–631 (1972)
75. Mena, H., Pfurtscheller, L.: An efficient SPDE approach for El Niño. Appl. Math. Comput. **352**, 146–156 (2019)
76. Mena, H., Ostermann, A., Pfurtscheller, L.M., Piazzola, C.: Numerical low-rank approximation of matrix differential equations. J. Comput. Appl. Math. **340**, 602–614 (2018)
77. Meyer, H.D.: Studying molecular quantum dynamics with the multiconfiguration time-dependent Hartree method. Wiley Interdiscip. Rev. Comput. Mol. Sci. **2**(2), 351–374 (2012)
78. Meyer, H., Manthea, U., Cederbauma, L.S.: The multi-configurational time-dependent Hartree approach. Chem. Phys. Lett. **165**(1), 73–78 (1990)

79. Meyer, G., Journée, M., Bonnabel, S., Sepulchre, R.: From subspace learning to distance learning: a geometrical optimization approach. In: Proceedings of the IEEE/SP 15th Workshop on Statistical Signal Processing, pp. 385–388 (2009)
80. Mirsky, L.: Symmetric gauge functions and unitarily invariant norms. Quart. J. Math. Oxf. Ser. (2) **11**, 50–59 (1960)
81. Mishra, B., Vandereycken, B.: A Riemannian approach to low-rank Algebraic Riccati equations. In: 21st International Symposium on Mathematical Theory of Networks and Systems, pp. 965–968 (2014)
82. Mishra, B., Meyer, G., Bonnabel, S., Sepulchre, R.: Fixed-rank matrix factorizations and Riemannian low-rank optimization. Comput. Stat. **29**(3–4), 591–621 (2014)
83. Musharbash, E., Nobile, F., Zhou, T.: Error analysis of the dynamically orthogonal approximation of time dependent random PDEs. SIAM J. Sci. Comput. **37**(3), A776–A810 (2015)
84. Orsi, R., Helmke, U., Moore, J.B.: A Newton–like method for solving rank constrained linear matrix inequalities. In: Proceedings of the 43rd IEEE Conference on Decision and Control, pp. 3138–3144 (2004)
85. Oseledets, I.V.: Approximation of $2^d \times 2^d$ matrices using tensor decomposition. SIAM J. Matrix Anal. Appl. **31**(4), 2130–2145 (2010)
86. Oseledets, I.V.: Tensor-train decomposition. SIAM J. Sci. Comput. **33**(5), 2295–2317 (2011)
87. Oseledets, I.V., Tyrtyshnikov, E.E.: Breaking the curse of dimensionality, or how to use SVD in many dimensions. SIAM J. Sci. Comput. **31**(5), 3744–3759 (2009)
88. Oseledets, I., Tyrtyshnikov, E.: TT-cross approximation for multidimensional arrays. Linear Algebra Appl. **432**(1), 70–88 (2010)
89. Ostermann, A., Piazzola, C., Walach, H.: Convergence of a low-rank Lie-Trotter splitting for stiff matrix differential equations. SIAM J. Numer. Anal. **57**(4), 1947–1966 (2019)
90. Park, D., Kyrillidis, A., Carmanis, C., Sanghavi, S.: Non-square matrix sensing without spurious local minima via the Burer-Monteiro approach. In: Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, pp. 65–74 (2017)
91. Rakhuba, M.V., Oseledets, I.V.: Jacobi-Davidson method on low-rank matrix manifolds. SIAM J. Sci. Comput. **40**(2), A1149–A1170 (2018)
92. Rakhuba, M., Novikov, A., Oseledets, I.: Low-rank Riemannian eigensolver for high-dimensional Hamiltonians. J. Comput. Phys. **396**, 718–737 (2019)
93. Rauhut, H., Schneider, R., Stojanac, Ž.: Low rank tensor recovery via iterative hard thresholding. Linear Algebra Appl. **523**, 220–262 (2017)
94. Sapsis, T.P., Lermusiaux, P.F.J.: Dynamically orthogonal field equations for continuous stochastic dynamical systems. Phys. D **238**(23–24), 2347–2360 (2009)
95. Sato, H., Kasai, H., Mishra, B.: Riemannian stochastic variance reduced gradient algorithm with retraction and vector transport. SIAM J. Optim. **29**(2), 1444–1472 (2019)
96. Schmidt, E.: Zur Theorie der linearen und nichtlinearen Integralgleichungen. Math. Ann. **63**(4), 433–476 (1907)
97. Schollwöck, U.: The density-matrix renormalization group in the age of matrix product states. Ann. Phys. **326**(1), 96–192 (2011)
98. Shalit, U., Weinshall, D., Chechik, G.: Online learning in the manifold of low-rank matrices. In: Advances in Neural Information Processing Systems, vol. 23, pp. 2128–2136 (2010)
99. Shub, M.: Some remarks on dynamical systems and numerical analysis. In: Dynamical systems and partial differential equations (Caracas, 1984), pp. 69–91. University Simon Bolivar, Caracas (1986)
100. Sidiropoulos, N.D., De Lathauwer, L., Fu, X., Huang, K., Papalexakis, E.E., Faloutsos, C.: Tensor decomposition for signal processing and machine learning. IEEE Trans. Signal Process. **65**(13), 3551–3582 (2017)
101. Signoretto, M., Tran Dinh, Q., De Lathauwer, L., Suykens, J.A.K.: Learning with tensors: a framework based on convex optimization and spectral regularization. Mach. Learn. **94**(3), 303–351 (2014)
102. Simoncini, V.: Computational methods for linear matrix equations. SIAM Rev. **58**(3), 377–441 (2016)

103. Steinlechner, M.: Riemannian optimization for high-dimensional tensor completion. SIAM J. Sci. Comput. **38**(5), S461–S484 (2016)
104. Stewart, G.W.: On the early history of the singular value decomposition. SIAM Rev. **35**(4), 551–566 (1993)
105. Szalay, S., Pfeffer, M., Murg, V., Barcza, G., Verstraete, F., Schneider, R., Legeza, O.: Tensor product methods and entanglement optimization for ab initio quantum chemistry. Int. J. Quantum Chem. **115**(19), 1342–1391 (2015)
106. Todor, R.A., Schwab, C.: Convergence rates for sparse chaos approximations of elliptic problems with stochastic coefficients. IMA J. Numer. Anal. **27**(2), 232–261 (2007)
107. Udrişte, C.: Convex functions and optimization methods on Riemannian manifolds. Kluwer Academic Publishers Group, Dordrecht (1994)
108. Uschmajew, A., Vandereycken, B.: The geometry of algorithms using hierarchical tensors. Linear Algebra Appl. **439**(1), 133–166 (2013)
109. Uschmajew, A., Vandereycken, B.: Greedy rank updates combined with Riemannian descent methods for low-rank optimization. In: 2015 International Conference on Sampling Theory and Applications (SampTA), pp. 420–424 (2015)
110. Uschmajew, A., Vandereycken, B.: On critical points of quadratic low-rank matrix optimization problems (2018). MPI MiS Preprint 58/2018
111. Vandereycken, B.: Riemannian and multilevel optimization for rank-constrained matrix problems. Ph.D. thesis, Department of Computer Science, KU Leuven (2010)
112. Vandereycken, B.: Low-rank matrix completion by Riemannian optimization. SIAM J. Optim. **23**(2), 1214–1236 (2013)
113. Vandereycken, B., Vandewalle, S.: A Riemannian optimization approach for computing low-rank solutions of Lyapunov equations. SIAM J. Matrix Anal. Appl. **31**(5), 2553–2579 (2010)
114. Verstraete, F., Cirac, J.I.: Renormalization algorithms for quantum-many body systems in two and higher dimensions (2004). arXiv:cond-mat/0407066
115. Verstraete, F., García-Ripoll, J.J., Cirac, J.I.: Matrix product density operators: simulation of finite-temperature and dissipative systems. Phys. Rev. Lett. **93**(20), 207204 (2004)
116. Wang, H., Thoss, M.: Multilayer formulation of the multiconfiguration time-dependent Hartree theory. J. Chem. Phys. **119**(3), 1289–1299 (2003)
117. Wei, K., Cai, J.F., Chan, T.F., Leung, S.: Guarantees of Riemannian optimization for low rank matrix recovery. SIAM J. Matrix Anal. Appl. **37**(3), 1198–1222 (2016)
118. White, S.R.: Density-matrix algorithms for quantum renormalization groups. Phys. Rev. B **48**(14), 10345 (1993)

Part III
Statistical Methods and Non-linear Geometry

Chapter 10
Statistical Methods Generalizing Principal Component Analysis to Non-Euclidean Spaces

Stephan Huckemann and Benjamin Eltzner

Contents

Abstract Very generally speaking, statistical data analysis builds on descriptors reflecting data distributions. In a linear context, well studied nonparametric descriptors are means and PCs (principal components, the eigenorientations of covariance matrices). In 1963, T.W. Anderson derived his celebrated result of joint asymptotic normality of PCs under very general conditions. As means and PCs can also be defined geometrically, there have been various generalizations of PC analysis (PCA) proposed for manifolds and manifold stratified spaces. These generalizations play an increasingly important role in statistical dimension reduction of non-Euclidean data. We review their beginnings from Procrustes analysis (GPA), over principal geodesic analysis (PGA) and geodesic PCA (GPCA) to principal nested spheres (PNS), horizontal PCA, barycentric subspace analysis (BSA) and backward nested descriptors analysis (BNDA). Along with this, we review the current state of the art of their asymptotic statistical theory and applications for statistical testing, including open challenges, e.g. new insights into scenarios of nonstandard rates and asymptotic nonnormality.

S. Huckemann (✉) · B. Eltzner
Felix-Bernstein-Institute for Mathematical Statistics in the Biosciences, University of Göttingen, Göttingen, Germany
e-mail: huckeman@math.uni-goettingen.de; beltzne@math.uni-goettingen.de

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_10

10.1 Introduction

The *mean* and the *covariance* are among the most elementary statistical descriptors describing a distribution in a nonparametric way, i.e. in the absence of a distributional model. They can be used for dimension reduction and for statistical testing based on their asymptotics. Extending these two quantities to non-Euclidean random deviates and designing statistical methods for these has been the subject of intense research in the last 50 years, beginning with *Procrustes analysis* introduced by Gower [23] and the *strong law of large numbers* for Fréchet means by Ziezold [51]. This chapter intends to provide a brief review of the development of this research until now and to put it into context.

We begin with the Euclidean version including classical PCA, introduce the more general concept of generalized Fréchet ρ-means, their strong laws and recover *general Procrustes analysis* (GPA) as a special case. Continuing with *principal geodesic analysis* we derive a rather general central limit theorem for generalized Fréchet ρ-means and illustrate how to recover from this Anderson's asymptotic theorem for the classical first PC and the CLT for Procrustes means. Next, as another application of our CLT we introduce *geodesic principal component analysis* (GPCA), which, upon closer inspection, turns out to be a nested descriptor. The corresponding *backward nested descriptor analysis* (BNDA) requires a far more complicated CLT, which we state. We put the rather recently developed methods of *principal nested spheres* (PNS), *horizontal PCA* and *barycentric subspace analysis* (BSA) into context and conclude with a list of open problems in the field.

10.2 Some Euclidean Statistics Building on Mean and Covariance

Asymptotics and the Two-Sample Test

Let $X_1, \dots, X_n \overset{i.i.d.}{\sim} X$ be random vectors in $\mathbb{R}^D$, $D \in \mathbb{N}$, with existing *population mean* $\mathbb{E}[X]$. Denoting the *sample mean* by

$$\bar{X}_n = \frac{1}{n} \sum_{j=1}^{n} X_j \,,$$

the *strong law of large numbers* (SLLN) asserts that (e.g. [8, Chapter 22])

$$\bar{X}_n \overset{\text{a.s.}}{\to} \mathbb{E}[X] \,.$$

Upon existence of the second moment $\mathbb{E}[\|X\|^2]$, the *covariance* $\operatorname{cov}[X]$ exists and the *central limit theorem* (CLT) asserts that the fluctuation between sample and population mean is asymptotically normal (e.g. [14, Section 9.5]), namely that

$$\sqrt{n}\big(\bar{X}_n - \mathbb{E}[X]\big) \overset{\mathcal{D}}{\to} \mathcal{N}\big(0, \operatorname{cov}[X]\big)\,. \tag{10.1}$$

Using the *sample covariance*

$$\hat{\Sigma} = \frac{1}{n-1}\sum_{j=1}^{n}(X_j - \mathbb{E}[X])(X_j - \mathbb{E}[X])^T$$

as a *plugin estimate* for $\operatorname{cov}[X]$ in (10.1), asymptotic confidence bands for $\mathbb{E}[X]$ can be obtained as well as corresponding tests.

A particularly useful test is the *two-sample* test, namely that for random vectors $X_1, \ldots, X_n \overset{i.i.d.}{\sim} X$ in $\mathbb{R}^D$ and independent random vectors $Y_1, \ldots, Y_m \overset{i.i.d.}{\sim} Y$ in $\mathbb{R}^D$ with full rank population and sample covariance matrices, $\operatorname{cov}[X]$ and $\operatorname{cov}[Y]$, $\hat{\Sigma}^X_n$ and $\hat{\Sigma}^Y_m$, respectively,

$$T^2 = \frac{n+m-2}{\frac{1}{n}+\frac{1}{m}}(\bar{X}_n - \bar{Y}_m)^T\left((n-1)\hat{\Sigma}^X_n + (m-1)\hat{\Sigma}^Y_m\right)^{-1}(\bar{X}_n - \bar{Y}_m) \tag{10.2}$$

follows a *Hotelling* distribution if X and Y are multivariate normal, cf. [40, Section 3.6.1]. More precisely, $T^2\,\frac{nm(n+m-D-1)}{(n+m)(n+m-2)D}$ follows a $F_{D,n+m-D-1}$-distribution. Remarkably, this holds also asymptotically under nonnormality of X and Y, if $\operatorname{cov}[X] = \operatorname{cov}[Y]$ or $n/m \to 1$, cf. [45, Section 11.3].

Principal Component Analysis (PCA)

Consider again random vectors $X_1, \ldots, X_n \overset{i.i.d.}{\sim} X$ in $\mathbb{R}^D$, $D \in \mathbb{N}$, with sample covariance matrix $\hat{\Sigma}$ and existing population covariance $\Sigma = \operatorname{cov}[X]$. Further let $\Sigma = \Gamma\Lambda\Gamma^T$ and $\hat{\Sigma} = \hat{\Gamma}\hat{\Lambda}\hat{\Gamma}^T$ be spectral decompositions, i.e. $\Gamma = (\gamma_1, \ldots, \gamma_D)$, $\hat{\Gamma} = (\hat{\gamma}_1, \ldots, \hat{\gamma}_D) \in SO(D)$ and $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_D)$, $\hat{\Lambda} = \operatorname{diag}(\hat{\lambda}_1, \ldots, \hat{\lambda}_D)$ with $\lambda_1 \geq \ldots \lambda_D \geq 0$ and $\hat{\lambda}_1 \geq \ldots \hat{\lambda}_D \geq 0$, respectively. Then the vectors γ_j $(j = 1, \ldots, D)$ are called *population principal components* and $\hat{\gamma}_j$ $(j = 1, \ldots, D)$ are called *sample principal components*, abbreviated as PCs. These PCs can be used for *dimension reduction*, namely considering instead of $X_1, \ldots, X_n \in \mathbb{R}^D$ their projections, also called *scores*,

$$\left(X_1^T\hat{\gamma}_j\right)_{j=1}^{J}, \ldots, \left(X_n^T\hat{\gamma}_j\right)_{j=1}^{J} \in \mathbb{R}^J\,,$$

to the first $1 \leq J \leq D$ PCs. The *variance explained* by the first J PCs is

$$\hat{\lambda}_1 + \ldots + \hat{\lambda}_J\,.$$

Due to the seminal result by Anderson [1], among others, there is a CLT for $\hat{\gamma}_j$ $(1 \leq j \leq D)$, stating that if X is multivariate normal, $\Sigma > 0$ and λ_j simple,

$$\sqrt{n}(\hat{\gamma}_j - \gamma_j) \overset{\mathcal{D}}{\to} \mathcal{N}\left(0, \sum_{j\neq k=1}^{D} \frac{\lambda_j\lambda_k}{\lambda_j - \lambda_k}\gamma_k\gamma_k^T\right). \tag{10.3}$$

Here, we have assumed, w.l.o.g., that $\gamma_j^T\hat{\gamma}_j \geq 0$.

This CLT has been extended to nonnormal X with existing fourth moment $\mathbb{E}[\|X\|^4]$ by Davis [11] with a more complicated covariance matrix in (10.3). With little effort we reproduce the above result in Corollary 10.4 for $j = 1$ in the context of generalized Fréchet ρ-means.

10.3 Fréchet ρ-Means and Their Strong Laws

What is a good analog to $\mathbb{E}[X]$ when data are no longer vectors but points on a sphere, as are principal components, say? More generally, one may want to statistically assess points on manifolds or even on stratified spaces. For example, data on stratified spaces are encountered in modeling three-dimensional landmark-based shapes by Kendall [36] (cf. Sect. 10.4) or in modeling *phylogenetic descendants trees* in the space introduced by Billera et al. [7].

For a vector-valued random variable X in $\mathbb{R}^D$, upon existence of second moments $\mathbb{E}[\|X\|^2]$ note that,

$$\mathbb{E}[X] = \operatorname*{argmin}_{x\in\mathbb{R}^D} \mathbb{E}\left[\|X - x\|^2\right].$$

For this reason, [21] generalized the classical Euclidean expectation to random deviates X taking values in a metric space (Q, d) via

$$E(X) = \operatorname*{argmin}_{q\in Q} \mathbb{E}\left[d(X, q)^2\right]. \tag{10.4}$$

In contrast to the Euclidean expectation, $E(X)$ can be set-valued, as is easily seen by a symmetry argument for $Q = \mathbb{S}^{D-1}$ equipped with the spherical metric d and X uniform on $\mathbb{S}^{D-1}$. Then $E(X) = \mathbb{S}^{D-1}$.

Revisiting PCA, note that PCs are not elements of the data space $Q = \mathbb{R}^D$ but elements of $\mathbb{S}^{D-1}$, or more precisely, elements of real projective space of dimension $D - 1$

$$\mathbb{R}P^{D-1} = \{[x] : x \in \mathbb{S}^{D-1}\} = \mathbb{S}^{D-1}/\mathbb{S}^0 \text{ with } [x] = \{-x, x\}.$$

Moreover, the PCs (as elements in $\mathbb{S}^{D-1}$) are also solutions to a minimization problem, e.g. for the first PC we have

$$\gamma_1 \in \operatorname*{argmin}_{\gamma \in \mathbb{S}^{D-1}} \Big(\mathrm{var}[X] - \gamma^T \mathrm{cov}[X]\gamma \Big) . \tag{10.5}$$

Since $\mathrm{var}[X] - \gamma^T \mathrm{cov}[X]\gamma = \mathbb{E}\Big[\|X - \gamma^T X\gamma\|^2 \Big]$, in case of $\mathbb{E}[X] = 0$, this motivates the following distinction between *data space* and *descriptor space* leading to *Fréchet ρ-means*.

Definition 10.1 (Generalized Fréchet Means) Let Q, P be topological spaces and let $\rho : Q \times P \to \mathbb{R}$ be continuous. We call Q the *data space*, P the *descriptor space* and ρ the *link function*. Suppose that $X_1, \ldots, X_n \overset{i.i.d.}{\sim} X$ are random elements on Q with the property that

$$F(p) = \mathbb{E}[\rho(X, p)^2], \quad F_n(p) = \frac{1}{n} \sum_{j=1}^{n} \rho(X_j, p)^2 ,$$

called the *population* and *sample Fréchet functions*, are finite for all $p \in P$. Every minimizer of the population Fréchet function is called a *population Fréchet mean* and every minimizer of the sample Fréchet function is called a *sample Fréchet mean*. The corresponding sets are denoted by

$$E = \operatorname*{argmin}_{p \in P} F(p), \quad E_n = \operatorname*{argmin}_{p \in P} F_n(p) . \qquad \square$$

Remark By construction, E_n and E are closed sets, but they may be empty without additional assumptions. □

For the following we require that the topological space P is equipped with a *loss function* d, i.e.

1. $d : P \times P \to [0, \infty)$, is a continuous function
2. that vanishes only on the diagonal, that is $d(p, p') = 0$ if and only if $p = p'$.

We now consider the following two versions of a set valued strong law,

$$\bigcap_{n=1}^{\infty} \overline{\bigcup_{k=n} E_k} \subseteq E \;\text{ a.s.} \tag{10.6}$$

$$\forall \epsilon > 0\, \exists N \in \mathbb{N} \text{ such that } E_n \subseteq \{p \in P : d(E, p) < \epsilon\}\, \forall n \geq N \text{ a.s.} \tag{10.7}$$

In (10.7), N is random as well.

Ziezold [51] established (10.6) for separable $P = Q$ and $\rho = d$ a quasi-metric. Notably, this also holds in case of void E. Bhattacharya and Patrangenaru [5] proved (10.7) under the additional assumptions that $E \neq \emptyset$, $\rho = d$ is a metric and $P = Q$ satisfies the Heine–Borel property (stating that every closed bounded subset is compact). Remarkably, (10.6) implies (10.7) for compact spaces P; this has been observed by Bhattacharya and Patrangenaru [5, Remark 2.5] for $P = Q$ and $\rho = d$ a metric and their argument carries over at once to the general case.

For generalized Fréchet ρ-means we assume the following strongly relaxed analogs of the triangle inequality for (quasi-)metrics.

Definition 10.2 Let Q, P be topological spaces with link function ρ and let d be a loss function on P. We say that (ρ, d) is *uniform* if

$$\begin{aligned}&\forall p \in P, \epsilon > 0\ \exists \delta = \delta(\epsilon, p) > 0 \text{ such that}\\&|\rho(x, p') - \rho(x, p)| < \epsilon\ \forall x \in Q, p' \in P \text{ with } d(p, p') < \delta.\end{aligned}$$

Further, we say that (ρ, d) is *coercive*, if $\forall p_0, p^* \in P$ and $p_n \in P$ with $d(p^*, p_n) \to \infty$,

$$\begin{aligned}&d(p_0, p_n) \to \infty \ \text{ and } \exists C > 0 \text{ such that}\\&\rho(x, p_n) \to \infty\ \forall x \in Q \text{ with } \rho(x, p_0) < C\end{aligned}$$

Theorem ([27]) *With the notation of Definition 10.1 we have (10.6) if (ρ, d) is uniform and P is separable. If additionally (ρ, d) is coercive, $E \neq \emptyset$ and $\cap_{n=1}^{\infty}\overline{\cup_{k=n} E_k}$ satisfies the Heine Borel property with respect to d then (10.7) holds true.* □

Let us conclude this section with another example. In biomechanics, e.g. traversing skin markers placed around the knee joint (e.g. [49]), or in medical imaging, modeling deformation of internal organs via skeletal representations (cf. [47]), typical motion of markers occurs naturally along small circles in $\mathbb{S}^2$, c.f. [46]. For a fixed number $k \in \mathbb{N}$, considering k markers as one point $q = (q_1, \ldots, q_k) \in (\mathbb{S}^2)^k$, define the descriptor space P of k concentric small circles $p = (p_1, \ldots, p_k)$ defined by a common axis $w \in \mathbb{S}^2$ and respective latitudes $0 < \theta_1 < \ldots < \theta_k < \pi$. Setting

$$\rho(q, p) = \sqrt{\sum_{j=1}^{k} \min_{y \in p_j} \arccos^2 y^T q_j}$$

and

$$d(p, p') = \sqrt{\arccos^2(w^T w') + \sum_{j=1}^{k} (\theta_j - \theta'_j)^2}$$

we obtain a link ρ and a loss d which form a uniform and coercive pair. Moreover, even P satisfies the Heine–Borel property.

10.4 Procrustes Analysis Viewed Through Fréchet Means

Long before the notion of Fréchet means entered the statistics of shape, *Procrustes analysis* became a tool of choice and has been ever after for the statistical analysis of shape.

Kendall's Shape Spaces
Consider n geometric objects in $\mathbb{R}^m$, each described by k landmarks ($n, k, m \in \mathbb{N}$), i.e. every object is described by a matrix $X_j \in \mathbb{R}^{m\times k}$ ($1 \le j \le n$), the columns of which are the k landmark vectors in $\mathbb{R}^m$ of the j-th object. When only the *shape* of the objects is of concern, consider every

$$\lambda_j R_j (X_j - a_j\, 1_k^T/n)$$

equivalent with X_j, where $\lambda_j \in (0,\infty)$ reflects size, $R_j \in SO(m)$ rotation and $a_j \in \mathbb{R}^m$ translation. Here, 1_k is the k-dimensional column vector with all entries equal to 1. Note that the canonical quotient topology of $\mathbb{R}^{m\times k}/(0,\infty)$ gives a non-Hausdorff space which is a dead end for statistics, because all points have zero distance from one another. For this reason, one projects instead to the unit sphere $\mathbb{S}^{m\times k-1}$ and the canonical quotient

$$\Sigma_m^k = \mathbb{S}^{m\times k-1}/SO(m)/\mathbb{R}^m \cong \mathbb{S}^{m\times(k-1)-1}/SO(m)$$

is called *Kendall's shape space*, for details see [13].

Procrustes Analysis
Before the introduction of Kendall's shape spaces, well aware that the canonical quotient is statistically meaningless, [23] suggested to minimize the *Procrustes sum of squares*

$$\sum_{j=1}^{n} \left\| \lambda_i R_i (X_i - a_i\, 1_k^T/n) - \lambda_j R_j (X_j - a_j\, 1_k^T/n) \right\|^2$$

over $\lambda_j, R_j, a_j \in (0,\infty)\times SO(m)\times\mathbb{R}^m$ ($1 \le j \le n$) under the constraining condition

$$\left\| \sum_{j=1}^{n} \lambda_j R_j (X_j - a_j\, 1_k^T/n) \right\| = 1\,.$$

It turns out that the minimizing a_j are the mean landmarks, so for the following, we may well assume that every X_j is centered, i.e. $X_j 1_k = 0$ and dropping one landmark, e.g. via Helmertizing, i.e. by multiplying each X_j with a *sub-Helmert* matrix $\mathcal{H}$

$$\mathcal{H} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & \cdots & \frac{1}{\sqrt{k(k-1)}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & \cdots & \frac{1}{\sqrt{k(k-1)}} \\ 0 & -\frac{2}{\sqrt{6}} & \cdots & \frac{1}{\sqrt{k(k-1)}} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -\frac{k-1}{\sqrt{k(k-1)}} \end{pmatrix} \in M(k, k-1)$$

from the right, see [13], we may even more assume that $X_j \in \mathbb{R}^{m\times(k-1)}$ $(j = 1, \ldots, n)$. Further, with minimizing λ_j, R_j, every *Procrustes mean*

$$\mu = \frac{1}{n}\sum_{j=1}^{n} \lambda_j R_j X_j$$

is also a representative of a Fréchet mean on $Q = P = \Sigma_m^k$ using the canonical quotient $\rho = d$ of the residual quasi-metric

$$\widetilde{\rho}(X, Y) = \|X - (X^T Y)Y\| \tag{10.8}$$

on $\mathbb{S}^{m\times(k-1)-1}$, in this context, called the *pre-shape space*, see [26] for a detailed discussion.

If μ is a Procrustes mean with minimizing λ_j, R_j $(j = 1, \ldots, n)$, notably, this implies $\operatorname{trace}(R_j X_j^T \mu) = \lambda_j$, then

$$\lambda_j R_j X_j - \mu$$

are called the *Procrustes residuals*. By construction they live in the tangent space $T_\mu \mathbb{S}^{m\times(k-1)-1}$ of $\mathbb{S}^{m\times(k-1)-1}$ at μ. In particular, this is a linear space and hence, the residuals can be subjected to PCA. Computing the Procrustes mean and performing PCA for the Procrustes residuals is *full Procrustes analysis* as proposed by Gower [23].

Note that at this point, we have neither a CLT for Procrustes means nor can we apply the CLT (10.3) because the tangent space is random.

This nested randomness can be attacked directly by nested subspace analysis in Sect. 10.7 or circumvented by the approach detailed in the Sect. 10.6. Let us conclude the present section by briefly mentioning an approach for Riemannian manifolds similar to Procrustes analysis.

Principal Geodesic Analysis

Suppose that $Q = P$ is a Riemannian manifold with intrinsic geodesic distance $\rho = d$. Fréchet means with respect to ρ are called *intrinsic means* and [20] compute an intrinsic mean μ and perform PCA with the data mapped under the inverse exponential at μ to the tangent space $T_\mu Q$ of Q at μ. Again, the base point of the tangent space is random, prohibiting the application of the CLT (10.3).

10.5 A CLT for Fréchet ρ-Means

For this section we require the following assumptions.

(A1) $X_1, \ldots, X_n \overset{i.i.d.}{\sim} X$ are random elements in a topological data space Q, which is linked to a topological descriptor space P via a continuous function $\rho : Q \times P \to \mathbb{R}$, featuring a unique Fréchet ρ-mean $\mu \in P$.

(A2) There is a loss function $d : P \times P \to [0, \infty)$ and P has locally the structure of a D-dimensional manifold near μ, i.e. there is an open set $U \subset P$, $\mu \in U$ and a homeomorphism $\phi : U \to V$ onto an open set $V \subset \mathbb{R}^D$. W.l.o.g. assume that $\phi(\mu) = 0 \in V$.

(A3) In local coordinates the population Fréchet function is twice differentiable at μ with non-singular Hessian there, i.e. for $p \in U$, $x = \phi^{-1}(p)$,

$$F(p) = F(\mu) + x^T \frac{1}{2} \operatorname{Hess}\left(F \circ \phi^{-1}\right)(0)\, x + o(\|x\|^2)\,,$$
$$H := \operatorname{Hess}\left(F \circ \phi^{-1}\right)(0) > 0\,.$$

(A4) The gradient $\dot{\rho}_0(X) := \operatorname{grad}_x \rho(X, \phi^{-1}(x))^2|_{x=0}$ exists almost surely and there is a measurable function $\dot{\rho} : Q \to \mathbb{R}$, satisfying $\mathbb{E}[\dot{\rho}(X)^2] < \infty$, such that the following Lipschitz condition

$$|\rho(X, \phi^{-1}(x_1))^2 - \rho(X, \phi^{-1}(x_2))^2| \le \dot{\rho}(X)\|x_1 - x_2\| \text{ a.s.}$$

holds for all $x_1, x_2 \in U$.

Theorem 10.3 *Under the above Assumptions (A1)–(A4), if $\mu_n \in E_n$ is a measurable selection of sample Fréchet ρ-means with $\mu_n \overset{\mathbb{P}}{\to} \mu$, then*

$$\sqrt{n}\phi^{-1}(\mu_n) \overset{\mathcal{D}}{\to} \mathcal{N}\left(0, H^{-1}\operatorname{cov}[\dot{\rho}_0(X)]H^{-1}\right).$$

Proof We use [17, Theorem 2.11] for $r = 2$. While this theorem has been formulated for intrinsic means on manifolds, upon close inspection, the proof utilizing empirical process theory from [50], rests only on the above assumptions, so that it can be transferred word by word to the situation of Fréchet ρ-means. □

Remark Since the seminal formulation of the first version of the CLT for intrinsic means on manifolds by Bhattacharya and Patrangenaru [6] there has been a vivid discussion on extensions and necessary assumptions (e.g. [2–4, 19, 24, 25, 33, 37, 39, 41]). Recently it has been shown that the rather complicated assumptions originally required by Bhattacharya and Patrangenaru [6] could be relaxed to the above. Further relaxing the assumption $H > 0$ yields so-called *smeary* CLTs, cf. [17]. □

Classical PCA as a Special Case of Fréchet ρ-Means

As an illustration how asymptotic normality of PCs shown by Anderson [1] in an elaborate proof follows simply from Theorem 10.3 we give the simple argument for the first PC.

Corollary 10.4 *Suppose that* $X_1, \ldots, X_n \overset{i.i.d.}{\sim} X$ *are random vectors in* $\mathbb{R}^D$ *with* $\mathbb{E}[X] = 0$, *finite fourth moment and orthogonal PCs* $\gamma_1, \ldots, \gamma_D \in S^{D-1}$ *to descending eigenvalues* $\lambda_1 > \lambda_2 \geq \ldots \geq \lambda_D > 0$. *Further let* $\hat{\gamma}_1$ *be a first sample PC with* $\hat{\gamma}_1^T \gamma_1 \geq 0$ *and local coordinates* $\hat{x}_n = \hat{\gamma}_1 - \gamma_1^T \hat{\gamma}_1\, \gamma_1$. *Then, with* $H^{-1} = \sum_{k=2}^{D} \frac{1}{\lambda_1 - \lambda_k}\, \gamma_k \gamma_k^T$,

$$\sqrt{n}\hat{x}_n \;\overset{\mathcal{D}}{\to}\; \mathcal{N}(0, H^{-1}\mathrm{cov}[XX^T\gamma_1]H^{-1})\,,$$

If X *is multivariate normal then the covariance of the above r.h.s. is given by the r.h.s. of (10.3) for* $j = 1$. □

Proof With the representation $\gamma = x + \sqrt{1 - \|x\|^2}\, \gamma_1 \in \mathbb{S}^{D-1}$, $\gamma_1 \perp x \in U \subset T_{\gamma_1} S^{D-1} \subset \mathbb{R}^D$, we have that the link function underlying (10.5) is given by

$$\begin{aligned}\rho(X, x)^2 &= \|X - \gamma^T X\,\gamma\|^2 \;=\; \|X\|^2 - (\gamma^T X)^2\\ &= \|X\|^2 - (x^T X + \sqrt{1 - \|x\|^2}\, \gamma_1^T X)^2\\ &= \|X\|^2 - x^T X X^T x - (1 - \|x\|^2)(\gamma_1^T X)^2 - 2x^T X \sqrt{1 - \|x\|^2}\, \gamma_1^T X\,.\end{aligned}$$

From

$$\mathrm{grad}_x \rho(X, x)^2 = -2XX^T x + 2x(\gamma_1^T X)^2 - 2\left(\sqrt{1 - \|x\|^2}X - \frac{x^T X x}{\sqrt{1 - \|x\|^2}}\right)\gamma_1^T X$$

and, with the unit matrix I,

$$\mathrm{Hess}_x \rho(X, x)^2 = -2XX^T + 2I(\gamma_1^T X)^2 + 2\left(\frac{Xx^T + xX^T - Xx^T}{\sqrt{1 - \|x\|^2}} + \frac{x^T X x x^T}{(1 - \|x\|^2)^{3/2}}\right)\gamma_1^T X,$$

verify that it satisfies Assumption (A4) with $\dot{\rho}_0(X) = -2XX^T\gamma_1$ and $\dot{\rho}(X) = 4\|XX^T\gamma_1\|$ for U sufficiently small, which is square integrable by hypothesis. Since $\mathrm{Hess}_x \rho(X, x)^2|_{x=0} = 2(\gamma_1^T XX^T \gamma_1 I - XX^T)$, with

$$H = 2\mathbb{E}[\gamma_1^T XX^T \gamma_1 I - XX^T] = 2\sum_{k=2}^{D}(\lambda_1 - \lambda_k)\gamma_k\gamma_k^T\,,$$

which is, by hypothesis, positive definite in $T_{\gamma_1}\mathbb{S}^{D-1}$, we obtain the first assertion of Theorem 10.3. Since in case of multivariate normality $X = \sum_{k=1}^{D} c_k \gamma_k$ with independent real random variables $c_1, \ldots, c_D$, the second assertion follows at once. □

The CLT for Procrustes Means

For $m \geq 3$, Kendall's shape spaces are stratified as follows. There is an open and dense manifold part $(\Sigma_m^k)^*$ and a lower dimensional rest $(\Sigma_m^k)^0$ that is similarly stratified (comprising a dense manifold part and a lower dimensional rest, and so on), e.g. [9, 32, 38]. For a precise definition of stratified spaces, see the following Sect. 10.6.

As a toy example one may think of the unit two-sphere $\mathbb{S}^2 = \{x \in \mathbb{R}^3 : \|x\| = 1\}$ on which $SO(2) \subset SO(3)$ acts via

$$\begin{pmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} x_1 \cos\phi + x_2 \sin\phi \\ -x_1 \sin\phi + x_2 \cos\phi \\ x_3 \end{pmatrix} .$$

The canonical quotient space has the structure of the closed interval $\mathbb{S}^2/SO(2) \cong [-1, 1]$ in which $(-1, 1)$ is an open dense one-dimensional manifold and $\{1, -1\}$ is the rest, a zero-dimensional manifold.

Let $Z_1, \ldots, Z_n \overset{i.i.d.}{\sim} Z$ be random configurations of m-dimensional objects with k landmarks, with pre-shapes $X_1, \ldots, X_n \overset{i.i.d.}{\sim} X$ in $\mathbb{S}^{m\times(k-1)-1}$ and shapes $\xi_1, \ldots, \xi_n \overset{i.i.d.}{\sim} \xi$ in Σ_m^k with the link function ρ given by the Procrustes metric from the pre-shape (i.e. residual) quasi-metric (10.8).

Theorem (Manifold Stability, cf. [28, 29]) *If, with the above setup, $\mathbb{P}\{\xi \in (\Sigma_m^k)^*\} > 0$ and if the probability that two shapes are maximally remote is zero then every Procrustes mean μ is assumed on the manifold part.* □

In consequence, for $Q = P = \Sigma_m^k$, if the manifold part is assumed at all, Assumption (A1) implies Assumption (A2). With the same reasoning as in the proof of Corollary 10.4, Assumption (A4) is verified. This yields the following.

Corollary *Let $Z_1, \ldots, Z_n \overset{i.i.d.}{\sim} Z$ be random configurations of m-dimensional objects with k landmarks, with pre-shapes $X_1, \ldots, X_n \overset{i.i.d.}{\sim} X$ in $\mathbb{S}^{D\times(k-1)-1}$ and shapes $\xi_1, \ldots, \xi_n \overset{i.i.d.}{\sim} \xi$ in Σ_m^k such that*

- $\mathbb{P}\{\xi \in (\Sigma_m^k)^*\} > 0$*, the probability that two shapes are maximally remote is zero and*
- *Assumptions (A1) and (A3) are satisfied.*

Then, every measurable selection μ_n of Procrustes sample means satisfies a CLT as in Theorem 10.3. □

10.6 Geodesic Principal Component Analysis

In this section we assume that random deviates $X_1, \ldots, X_n \overset{i.i.d.}{\sim} X$ take values in a Riemann stratified space Q.

Definition (Stratified Space) A stratified space Q of dimension m embedded in a Euclidean space can be defined as a direct sum

$$Q = \bigcup_{j=1}^{k} Q_{d_j}$$

such that $0 \leq d_1 < \ldots < d_k = m$, each Q_{d_j} is a d_j-dimensional manifold and $Q_{d_j} \cap Q_{d_l} = \emptyset$ for $j \neq l$.

A stratified space $Q \subset \mathbb{R}^s$ is called *Whitney stratified*, if for every $j < l$

(i) If $Q_{d_j} \cap \overline{Q_{d_l}} \neq \emptyset$ then $Q_{d_j} \subset \overline{Q_{d_l}}$.

(ii) For sequences $x_1, x_2, \ldots \in Q_{d_j}$ and $y_1, y_2, \ldots \in Q_{d_l}$ which converge to the same point $x \in Q_{d_j}$ such that the sequence of secant lines c_i between x_i and y_i converges to a line c as $i \to \infty$, and such that the sequence of tangent planes $T_{y_i} Q_{d_l}$ converges to a d_l dimensional plane T as $i \to \infty$, the line c is contained in T.

We call a Whitney stratified space *Riemann stratified*, if

(iii) for every $j < l$ and sequence $y_1, y_2, \ldots \in Q_{d_l}$ which converges to the point $x \in Q_{d_j}$ the Riemannian metric tensors $g_{l,y_i} \in T^2_{y_i} Q_{d_l}$ converge to a rank two tensor $g_{l,x} \in T \otimes T$ and the Riemannian metric tensor $g_{j,x} \in T^2_x Q_{d_j}$ is given by the restriction $g_{j,x} = g_{l,x}|_{T^2_x Q_{d_j}}$.

□

Geodesics, i.e. curves of locally minimal length, exist locally in every stratum Q_{d_j}. Due to the Whitney condition, a geodesic can also pass through strata of different dimensions if these strata are connected. Property (ii) is called Whitney condition B and it follows from this condition that $T_x Q_{d_j} \subset T$, which is called Whitney condition A, e.g. [22].

Of course, all Riemannian manifolds are stratified spaces. Typical examples for stratified spaces that are not Riemannian manifolds are Kendall's shape spaces Σ^k_m for $m \geq 3$ dimensional objects with $k \geq 4$ landmarks or the BHV space of phylogenetic descendants trees $\mathcal{T}_n$ with $n \geq 3$ leaves.

Let $\Gamma(Q)$ be the space of point sets of maximal geodesics in Q. With the intrinsic geodesic metric d_Q on Q we have the link function

$$\rho : Q \times \Gamma(Q) \to [0, \infty), \quad (q, \gamma) \mapsto \inf_{q' \in \gamma} d_Q(q, q') \,.$$

Further, we assume that $\Gamma(Q)$ also carries a metric d_Γ. This can be either Hausdorff distance based on d_Q, or a quotient metric, e.g. induced from $\Gamma(Q) = (Q \times Q)/\sim$ with a suitable equivalence relation. An example for the latter is the identification of $\Gamma(\mathbb{S}^{D-1})$ with $G(D, 2)$, the Grassmannian structure of the space of two-dimensional linear subspaces in $\mathbb{R}^D$ (every geodesic on $\mathbb{S}^{D-1}$ is a great circle which is the intersection with $\mathbb{S}^{D-1} \subset \mathbb{R}^D$ of a plane through the origin).

Definition 10.5 (cf. [32]) With the above assumptions, setting $P_0 = \Gamma(Q)$, every population Fréchet ρ-mean on $P = P_0$ is a first population *geodesic principal component* (GPC) and every such sample mean is a first sample GPC.

Given a unique first population GPC γ_1, setting $P_1 = \{\gamma \in \Gamma(Q) : \gamma \cap \gamma_1 \neq \emptyset \text{ and } \gamma \perp \gamma_1 \text{ there}\}$, every population Fréchet ρ-mean on $P = P_1$ is a second population GPC.

Higher order population GPCs are defined by requiring them to pass through a common point $p \in \gamma_1 \cap \gamma_2$ and being orthogonal there to all previous unique population GPCs.

Similarly, for the second sample GPC, for a given unique first GPC $\hat{\gamma}_1$ use $P = \hat{P}_1 = \{\gamma \in \Gamma(Q) : \gamma \cap \gamma_1 \neq \emptyset \text{ and } \gamma \perp \hat{\gamma}_1 \text{ there}\}$ and higher order sample GPCs are defined by requiring them to pass through a common point $\hat{p} \in \hat{\gamma}_1 \cap \hat{\gamma}_2$ and being orthogonal there to all previous unique sample GPCs.

The GPC scores are the orthogonal projections of X, or of the data, respectively, to the respective GPCs. □

Remark In case of valid assumptions (A1) – (A4) the CLT from Theorem 10.3 yields asymptotic $\sqrt{n}$-normality for the first PC in a local chart. An example and an application to $Q = \Sigma_2^k$ $(k \geq 3)$ can be found in [27]. □

Obviously, there are many other canonical intrinsic generalizations of PCA to non-Euclidean spaces, e.g. in his *horizontal PCA* [48] defines the second PC by a parallel translation of a suitable tangent space vector, orthogonally along the first PC. One difficulty is that GPCs usually do not define subspaces, as classical PCs do, which define affine subspaces. However, there are stratified spaces which have rich sets of preferred subspaces.

Definition (Totally Geodesic Subspace) A Riemann stratified space $S \subset Q$ with Riemannian metric induced by the Riemannian metric of a Riemann stratified space Q is called *totally geodesic* if every geodesic of S is a geodesic of Q. □

The totally geodesic property is transitive in the following sense. Consider a sequence of Riemann stratified subspaces $Q_1 \subset Q_2 \subset Q_3$ where Q_1 is totally geodesic with respect to Q_2 and Q_2 is totally geodesic with respect to Q_3. Then Q_1 is also totally geodesic with respect to Q_3.

In the following, we will use the term *rich space of subspaces* for a space of k-dimensional Riemann stratified subspaces of an m-dimensional Riemann stratified space, if it has dimension at least $(m - k)(k + 1)$. This means that the space of k-dimensional subspaces has at least the same dimension as the space of affine k-dimensional subspaces in $\mathbb{R}^m$. If a Riemann stratified space has a rich space of sequences of totally geodesic subspaces $Q_0 \subset Q_1 \subset \cdots \subset Q_{m-1} \subset Q_m = Q$ where every Q_j is a Riemann stratified space of dimension j, a generalization which is very close in spirit to PCA can be defined. This is especially the case, if Q has a rich space of $(m-1)$-dimensional subspaces which are of the same class as Q. For example, the sphere S^m has a rich space of great subspheres S^{m-1}, which are totally geodesic submanifolds. Therefore, spheres are well suited to introduce an analog of PCA, and [34, 35] have defined *principal nested spheres* (PNS) which even exist as *principal nested small spheres*, which are not totally geodesic, however. In the latter case the dimension of the space of k-dimensional submanifolds is even $(m-k)(k+2)$, cf. [30].

Generalizing this concept [43], has introduced *barycentric subspaces*, cf. Chapter 18 of this book.

In the following penultimate section we develop an inferential framework for such nested approaches.

10.7 Backward Nested Descriptors Analysis (BNDA)

As seen in Definition 10.5, higher order GPCs depend on lower order GPCs and are hence defined in a nested way. More generally, one can consider sequences of subspaces of descending dimension, where every subspace is also contained in all higher dimensional subspaces. Here we introduce the framework of *backward nested families of descriptors* to treat such constructions in a very general way.

In Sect. 10.9 we introduce several examples of such backward nested families of descriptors.

Definitions and Assumptions 10.6 Let Q be a separable topological space, called the *data space* and let $\{P_j\}_{j=0}^m$ be a family of separable topological spaces called *descriptor spaces*, each equipped with a loss function $d_j : P_j \times P_j \to [0, \infty)$ (i.e. it is continuous and vanishes exactly on the diagonal) with $P_m = \{Q\}$ $(j = 1, \dots, m)$.

Next, assume that every $p \in P_j$ $(j = 1, \dots, m)$ is itself a topological space giving rise to a topological space $\emptyset \neq S_p \subseteq P_{j-1}$ with a continuous function $\rho_p : p \times S_p \to [0, \infty)$ called a *link function*.

Further, assume that for all $p \in P_j$ $(j = 1, \dots, m)$ and $s \in S_p$ there exists a measurable mapping $\pi_{p,s} : p \to s$ called *projection*.

Then for $j \in \{1, \dots, m\}$ every

$$f = \{p^m, \dots, p^j\}, \text{ with } p^{l-1} \in S_{p^l}, l = j+1, \dots, m$$

is a *backward nested family of descriptors* (BNFD) from P_m to P_j which lives in the space

$$P_{m,j} = \left\{ f = \{p^l\}_{l=m}^{j} : p^{l-1} \in S_{p^l}, l = j+1, \dots, m \right\},$$

with *projection* along each descriptor

$$\pi_f = \pi_{p^{j+1},p^j} \circ \ldots \circ \pi_{p^m,p^{m-1}} : p^m \to p^j$$

For another BNFD $f' = \{p'^l\}_{l=m}^j \in T_{m,j}$ set

$$d^j(f, f') = \sqrt{\sum_{l=m}^{j} d_j(p^l, p'^l)^2}\,.$$

Definition With the above Definitions and Assumptions 10.6, random elements $X_1, \ldots, X_n \overset{i.i.d.}{\sim} X$ on the data space Q admitting BNFDs give rise to *backward nested population* and *sample means* (BN-means) $f = (p^m, \ldots, p^j)$ and $f_n = (p_n^m, \ldots, p_n^j)$, respectively, recursively defined via $f^m = \{Q\} = f_n^m$, i.e. $p^m = Q = p_n^m$ and for $j = m, \ldots, 2$,

$$p^{j-1} \in \operatorname*{argmin}_{s \in S_{p^j}} \mathbb{E}[\rho_{p^j}(\pi_{f^j} \circ X, s)^2], \qquad f^{j-1} = \{p^l\}_{l=m}^{j-1}$$

$$p_n^{j-1} \in \operatorname*{argmin}_{s \in S_{p_n^j}} \sum_{i=1}^{n} \rho_{p_n^j}(\pi_{f_n^j} \circ X_i, s)^2, \qquad f_n^{j-1} = \{p_n^l\}_{l=m}^{j-1}\,.$$

If all of the population minimizers are unique, we speak of *unique BN-means*.

Remark A nested sequence of subspaces is desirable for various scenarios. Firstly, it can serve as a basis for dimension reduction as is also often done using PCA in Euclidean space. Secondly, the residuals of projections along the BNFD can be used as residuals of orthogonal directions in Euclidean space in order to achieve a "Euclideanization" of the data (e.g. [44]). Thirdly, lower dimensional representations of the data or scatter plots of residuals can be used for more comprehensible visualization.

Backward approaches empirically achieve better results than forward approaches, starting from a point and building up spaces of increasing dimension, in terms of data fidelity. The simplest example, determining the intrinsic mean first and then requiring the geodesic representing the one-dimensional subspace to pass through it, usually leads to higher residual variance than fitting the principal geodesic without reference to the mean. □

For a strong law and a CLT for BN-means we require assumptions corresponding to Definition 10.2 and corresponding to assumptions in [4]. Both sets of assumptions are rather complicated, so that they are only referenced here.

(B1) Assumptions 3.1–3.6 from [31]

(B2) Assumption 3.10 from [31]

To the best knowledge of the authors, instead of (B2), more simple assumptions corresponding to (A1)–(A4) from Sect. 10.5 have not been derived for the backward nested descriptors scenario.

Theorem ([31]) *If the BN population mean $f = (p^m, \dots, p^j)$ is unique and if $f_n = (p_n^m, \dots, p_n^j)$ is a measurable selection of BN sample means then under (B1),*

$$f_n \to f \textit{ a.s.}$$

i.e. there is $\Omega' \subseteq \Omega$ measurable with $\mathbb{P}(\Omega') = 1$ such that for all $\epsilon > 0$ and $\omega \in \Omega'$, there is $N(\epsilon, \omega) \in \mathbb{N}$ with

$$d(f_n, f) < \epsilon \quad \textit{for all } n \geq N(\epsilon, \omega)\,.$$

Theorem 10.7 ([31]) *Under Assumptions (B2), with unique BN population mean $f \in P = P_{m,j}$ and local chart ϕ with $\phi^{-1}(0) = f$, for every measurable selection f_n of BN sample means $f_n \overset{\mathbb{P}}{\to} f$, there is a symmetric positive definite matrix B_ϕ such that*

$$\sqrt{n}\,\phi^{-1}(f_n) \overset{\mathcal{D}}{\to} \mathcal{N}(0, B_\phi)\,.$$

Remark 10.8 Under *factoring charts* as detailed in [31], asymptotic normality also holds for the last descriptor $p_n^{j-1} \overset{\text{a.s.}}{\to} p^{j-1}$,

$$\sqrt{n}\phi^{-1}(p_n^{j-1}) \overset{\mathcal{D}}{\to} \mathcal{N}(0, C_\phi)$$

with a suitable local chart ϕ such that $\phi^{-1}(0) = p^{j-1}$ and a symmetric positive definite matrix C_ϕ. □

10.8 Two Bootstrap Two-Sample Tests

Exploiting the CLT for ρ-means, BN-means or the last descriptor of BN-means (cf. Remark 10.8) in order to obtain an analog of the two-sample test (10.2), we inspect its ingredients. Suppose that $X_1, \dots, X_n \overset{i.i.d.}{\sim} X$ and $Y_1, \dots, Y_m \overset{i.i.d.}{\sim} Y$ are independent random elements on Q. In case of ρ-means, we assume that Assumptions (A1)–(A4) from Sect. 10.5 are valid and in case of BN-means (or a last descriptor thereof) assume that Assumption (B2) from Sect. 10.7 is valid for X and Y, in particular that unique means μ^X and μ^Y lie within one single open set $U \subset P$ that homeomorphically maps to an open set $V \subset \mathbb{R}^D$ under ϕ. With measurable selections $\hat{\mu}_n^X$ and $\hat{\mu}_m^Y$ of sample means, respectively, replace $\bar{X}_n - \bar{Y}_m$ with $\phi^{-1}\big(\hat{\mu}_n^X\big) - \phi^{-1}\big(\hat{\mu}_m^Y\big) \in \mathbb{R}^D$.

Obviously, $\hat{\Sigma}_n^X$ and $\hat{\Sigma}_m^Y$ are not directly assessable, however. If one had a large number B of samples $\{X_{1,1}, \dots, X_{1,n}\}, \dots, \{X_{B,1}, \dots, X_{B,n}\}$, one could calculate the descriptors $\phi^{-1}\big(\hat{\mu}_{n,1}^X\big), \dots, \phi^{-1}\big(\hat{\mu}_{n,B}^X\big)$ and estimate the covariance of these. But since we only have one sample, we use the bootstrap instead. The idea of the n out-of n non-parametric bootstrap (e.g. [12, 15]) is to generate a large number B of *bootstrap samples* $\{X_1^{*1}, \dots, X_n^{*1}\}, \dots, \{X_1^{*B}, \dots, X_n^{*B}\}$ of the same size n by drawing with replacement from the sample $X_1, \dots, X_n$. From each of these bootstrap samples one can calculate estimators $\phi^{-1}\big(\mu_n^{X,*1}\big), \dots, \phi^{-1}\big(\mu_n^{X,*B}\big)$, which serve as so-called *bootstrap estimators* of μ. From these, one can now calculate the estimator for the covariance of $\hat{\mu}_n^X$

$$\Sigma_n^{X,*} := \frac{1}{B}\sum_{j=1}^{B}\Big(\phi^{-1}(\mu_n^{X,*j}) - \phi^{-1}(\hat{\mu}_n^X)\Big)\Big(\phi^{-1}(\mu_n^{X,*j}) - \phi^{-1}(\hat{\mu}_n^X)\Big)^T \tag{10.9}$$

For the First Test

Perform B_X times n out-of n bootstrap from $X_1, \dots, X_n$ to obtain Fŕechet ρ-means $\mu_n^{X,*1}, \dots, \mu_n^{X,*B_X}$ and replace $\hat{\Sigma}_n^X$ with the n-fold of the bootstrap covariance $\Sigma_n^{X,*}$ as defined in Eq. (10.9). With the analog m out-of m bootstrap, replace $\hat{\Sigma}_m^Y$ with $m\Sigma_m^{Y,*}$.

Then, under $H_0 : \mu^X = \mu^Y$, if $m/n \to 1$ or $n \operatorname{cov}\big[\phi^{-1}(\mu_n^X)\big] = m \operatorname{cov}\big[\phi^{-1}(\mu_m^Y)\big]$, under typical regularity conditions, e.g. [10], the statistic

$$T^2 = (\phi^{-1}(\hat{\mu}_n^X) - \phi^{-1}(\hat{\mu}_m^Y))^T\left(\left(\frac{1}{n} + \frac{1}{m}\right)\Sigma_p^*\right)^{-1}(\phi^{-1}(\hat{\mu}_n^X) - \phi^{-1}(\hat{\mu}_m^Y)) \tag{10.10}$$

$$\Sigma_p^* := \frac{1}{n+m-2}\Big(n(n-1)\Sigma_n^{X,*} + m(m-1)\Sigma_m^{Y,*}\Big)$$

adapted from Eq. (10.2), is asymptotically Hotelling distributed as discussed in Sect. 10.2.

For the Second Test

Observe that, alternatively, the test statistic

$$T^2 = \Big(\phi^{-1}(\mu_n^{\widetilde{X}}) - \phi^{-1}(\mu_m^{\widetilde{Y}})\Big)^T\Big(\Sigma_n^{X,*} + \Sigma_m^{Y,*}\Big)^{-1}\Big(\phi^{-1}(\mu_n^{\widetilde{X}}) - \phi^{-1}(\mu_m^{\widetilde{Y}})\Big),$$

can be used. Notably, this second test for $H_0 : \mu_X = \mu_Y$ does not rely on $n/m \to 1$ or equal covariances, as does the first. However, the test statistic is only approximately F distributed, even for normally distributed data and the parameters of the F distribution have to be determined by an approximation procedure.

To enhance the power of either test, quantiles can be determined using the bootstrap. A naive approach would be to pool samples and use $\widetilde{X}$ for the first n

data points, $\widetilde{Y}$ for the last m data points of bootstrapped samples from the pooled data $X_1, \dots, X_n, Y_1, \dots, Y_m$. However, it turns out that this approach suffers from significantly diminished power.

Instead, we generate the same number B of bootstrap samples from $X_1, \dots, X_n$ and $Y_1, \dots, Y_m$ separately, thus getting $\mu_n^{X,*1}, \dots, \mu_n^{X,*B}$ and $\mu_m^{Y,*1}, \dots, \mu_m^{Y,*B}$. Due to the CLT 10.7 and Remark 10.8, $\phi^{-1}(\mu_n^{X,*1}), \dots, \phi^{-1}(\mu_n^{X,*B})$ are samples from a distribution which is close to normal with mean $\phi^{-1}(\hat{\mu}_n^X)$. The analog holds for Y. As a consequence, the residuals $d_n^{X,*j} = \phi^{-1}(\mu_n^{X,*j}) - \phi^{-1}(\hat{\mu}_n^X)$ are close to normally distributed with mean 0. To simulate quantiles from the null hypothesis $\mu_n^X = \mu_m^Y$, we therefore only use the residuals $d_{n,*j}^X$ and $d_{m,*j}^Y$ and calculate

$$T_j^2 = \left(d_n^{X,*j} - d_m^{Y,*j})\right)^T \left(\Sigma_n^{X,*} + \Sigma_m^{Y,*}\right)^{-1} \left(d_n^{X,*j} - d_m^{Y,*j}\right) . \tag{10.11}$$

Then we order these values ascendingly and use them as $(j - 1/2)/B$-quantiles as usual for empirical quantiles. Tuning the corresponding test to the right level, its power is usually larger than using the F-quantiles corresponding to the Hotelling distribution.

For a detailed discussion and justification see [16, 31].

10.9 Examples of BNDA

Scenarios of BNDA are given by *flags*, namely, by nested subspaces,

$$Q \supseteq p^m \supseteq p^{m-1} \supseteq \dots \supseteq p^0 \in Q .$$

We give three examples.

The Intrinsic Mean on the First GPC

It is well known, that the intrinsic mean usually comes not to lie on the first GPC. For example a distribution on $\mathbb{S}^2$ that is uniform on a great circle has this great circle as its first GPC with respect to the spherical metric. The Fréchet mean with respect to this metric is given, as is easily verified, by the two poles having this great circle as the equator. In order to enforce nestedness, we consider the first GPC on a Riemannian manifold and the intrinsic mean on it. The corresponding descriptor spaces are

$$P_2 = \{Q\},\ S_Q = P_1 = \left\{\gamma_{q,v} : (q, v) \in TQ, v \neq 0\right\} / \sim,\ P_0 = Q .$$

with the *tangent bundle* TQ over Q, the maximal geodesic $\gamma_{q,v}$ through q with initial velocity $v \in T_qQ$ and $\gamma_{q,v} \sim \gamma_{q',v'}$ if the two geodesics agree as point sets. Denoting the class of $\gamma_{q,v}$ by $[\gamma_{q,v}]$, it turns out that

$$T_{2,0} = \{(p, s) : p = [\gamma_{q,v}] \in P_1,\ s \in p\} \cong PQ,$$
$$([\gamma_{q,v}], s) \cong \left(s, \left\{\frac{w}{\|w\|}, -\frac{w}{\|w\|}\right\}\right),$$

where $[\gamma_{q,v}] = [\gamma_{s,w}]$ and PQ denotes the *projective bundle* over Q. With the local trivialization of the tangent bundle one obtains a local trivialization of the projective bundle and thus factoring charts, so that, under suitable conditions, Theorem 10.7 and Remark 10.8 are valid. In fact, this construction also works for suitable Riemann stratified spaces, e.g. also for $Q = \Sigma_m^k$ with $m \geq 3$, cf. [31].

Principal Nested Spheres (PNS)
For the special case of $Q = \mathbb{S}^{D-1} \subset \mathbb{R}^D$ let P_j be the space of all j-dimensional unit-spheres ($j = 1, \ldots, D-1$) and $P_{D-1} = \mathbb{S}^{D-1}$. Note that P_j can be given the manifold structure of the Grassmannian $G(D, j+1)$ of $(j+1)$-dimensional linear subspaces in $\mathbb{R}^D$. The corresponding BNDA has been introduced by Jung et al. [34, 35] as *principal nested great spheres analysis* (PNGSA) in contrast to *principal nested small sphere analysis* (PNSSA), when allowing also small subspheres in every step. Notably, estimation of small spheres involves a test for great spheres to avoid overfitting, cf. [18, 34].

Furthermore, PNSSA offers more flexibility than PNGSA because the family of all j-dimensional small subspheres in $\mathbb{S}^{D-1}$ has dimension $\dim\big(G(D, j+1)\big) + D - j$, cf. [18].

As shown in [31], under suitable conditions, Theorem 10.7 and Remark 10.8 are valid for both versions of PNS.

Extensions of PNS to general Riemannian manifolds can be sought by considering flags of totally geodesic subspaces. While there are always geodesics, which are one-dimensional geodesic subspaces, there may be none for a given dimension j. And even, if there are, for instance on a torus, totally geodesic subspaces winding around infinitely are statistically meaningless because they approximate any given data set arbitrarily well. As a workaround, tori can be topologically and geometrically deformed into stratified spheres and on these PNS with all of its flexibility, described above, can be performed, as in [18].

Barycentric Subspace Analysis (BSA) by Pennec [43] constitutes another extension circumventing the above difficulties. Here P_j is the space of *exponential spans* of any $j+1$ points in general position. More precisely, with the geodesic distance d on Q, for $q_1, \ldots, q_{j+1} \in Q$, define their exponential span by

$$\mathfrak{M}(q_1, \ldots, q_{j+1}) = \left\{ \operatorname*{argmin}_{q \in Q} \sum_{k=1}^{j+1} a_k d(q_k, q)^2 : a_1, \ldots, a_{j+1} \in \mathbb{R},\ \sum_{k=1}^{j+1} a_k = 1 \right\}.$$

For an m-dimensional manifold Q a suitable choice of $m+1$ points $q_1, \ldots, q_{m+1} \in Q$ thus yields the flag

$$Q = \mathfrak{M}(q_1, \dots, q_{m+1}) \supset \mathfrak{M}(q_1, \dots, q_m) \supset \dots \supset \mathfrak{M}(q_1, q_2) \supset \{q_1\}\,.$$

For the space of phylogenetic descendants tree by Billera et al. [7] in a similar approach by Nye et al. [42] the *locus of the Fréchet mean* of a given point set has been introduced along with corresponding optimization algorithms.

Barycentric subspaces and similar constructions are the subject of Chapter 11.

To the knowledge of the authors, there have been no attempts, to date, to investigate applicability of Theorem 10.7 and Remark 10.8 to BSA.

10.10 Outlook

Beginning with Anderson's CLT for PCA, we have sketched some extensions of PCA to non-Euclidean spaces and have come up with a rather general CLT, the assumptions of which are more general than those of [4, 6]. Let us conclude with listing a number of open tasks, which we deem of high importance for the development of suitable statistical non-Euclidean tools.

1. Formulate the CLT for BNFDs in terms of assumptions corresponding to Assumptions (A1)–(A4).
2. Apply the CLT for BNFDs to BSA if possible.
3. Formulate BNFD not as a sequential but as a simultaneous optimization problem, derive corresponding CLTs and apply them to BSA with simultaneous estimation of the entire flag.
4. In some cases we have no longer $\sqrt{n}$-Gaussian CLTs but so-called *smeary* CLTs which feature a lower rate, cf. [17]. Extend the CLTs presented here to the general smeary scenario.
5. Further reduce and generalize Assumptions (A1)–(A4), especially identify necessary and sufficient conditions for (A3).

Acknowledgements The authors acknowledge support from the Niedersachen Vorab of the Volkswagen Foundation and support from DFG HU 1575-7.

References

1. Anderson, T.: Asymptotic theory for principal component analysis. Ann. Math. Statist. **34**(1), 122–148 (1963)
2. Barden, D., Le, H., Owen, M.: Central limit theorems for Fréchet means in the space of phylogenetic trees. Electron. J. Probab. **18**(25), 1–25 (2013)
3. Barden, D., Le, H., Owen, M.: Limiting behaviour of fréchet means in the space of phylogenetic trees. Ann. Inst. Stat. Math. **70**(1), 99–129 (2018)
4. Bhattacharya, R., Lin, L.: Omnibus CLTs for Fréchet means and nonparametric inference on non-Euclidean spaces. Proc. Am. Math. Soc. **145**(1), 413–428 (2017)

5. Bhattacharya, R.N., Patrangenaru, V.: Large sample theory of intrinsic and extrinsic sample means on manifolds I. Ann. Stat. **31**(1), 1–29 (2003)
6. Bhattacharya, R.N., Patrangenaru, V.: Large sample theory of intrinsic and extrinsic sample means on manifolds II. Ann. Stat. **33**(3), 1225–1259 (2005)
7. Billera, L., Holmes, S., Vogtmann, K.: Geometry of the space of phylogenetic trees. Adv. Appl. Math. **27**(4), 733–767 (2001)
8. Billingsley, P.: Probability and Measure, vol. 939. Wiley, London (2012)
9. Bredon, G.E.: Introduction to Compact Transformation Groups. Pure and Applied Mathematics, vol. 46. Academic Press, New York (1972)
10. Cheng, G.: Moment consistency of the exchangeably weighted bootstrap for semiparametric m-estimation. Scand. J. Stat. **42**(3), 665–684 (2015)
11. Davis, A.W.: Asymptotic theory for principal component analysis: non-normal case. Aust. J. Stat. **19**, 206–212 (1977)
12. Davison, A.C., Hinkley, D.V.: Bootstrap Methods and Their Application, vol. 1. Cambridge University Press, Cambridge (1997)
13. Dryden, I.L., Mardia, K.V.: Statistical Shape Analysis. Wiley, Chichester (2014)
14. Durrett, R.: Probability: Theory and Examples. Cambridge University Press, Cambridge (2010)
15. Efron, B., Tibshirani, R.J.: An Introduction to the Bootstrap. CRC Press, Boca Raton (1994)
16. Eltzner, B., Huckemann, S.: Bootstrapping descriptors for non-Euclidean data. In: Geometric Science of Information 2017 Proceedings, pp. 12–19. Springer, Berlin (2017)
17. Eltzner, B., Huckemann, S.F.: A smeary central limit theorem for manifolds with application to high dimensional spheres. Ann. Stat. **47**, 3360–3381 (2019)
18. Eltzner, B., Huckemann, S., Mardia, K.V.: Torus principal component analysis with applications to RNA structure. Ann. Appl. Statist. **12**(2), 1332–1359 (2018)
19. Eltzner, B., Galaz-García, F., Huckemann, S.F., Tuschmann, W.: Stability of the cut locus and a central limit theorem for Fréchet means of Riemannian manifolds (2019). arXiv: 1909.00410
20. Fletcher, P.T., Lu, C., Pizer, S.M., Joshi, S.C.: Principal geodesic analysis for the study of nonlinear statistics of shape. IEEE Trans. Med. Imag. **23**(8), 995–1005 (2004)
21. Fréchet, M.: Les éléments aléatoires de nature quelconque dans un espace distancié. Ann. Inst. Henri Poincare **10**(4), 215–310 (1948)
22. Goresky, M., MacPherson, R.: Stratified Morse Theory. Springer, Berlin (1988)
23. Gower, J.C.: Generalized Procrustes analysis. Psychometrika **40**, 33–51 (1975)
24. Hotz, T., Huckemann, S.: Intrinsic means on the circle: uniqueness, locus and asymptotics. Ann. Inst. Stat. Math. **67**(1), 177–193 (2015)
25. Hotz, T., Huckemann, S., Le, H., Marron, J.S., Mattingly, J., Miller, E., Nolen, J., Owen, M., Patrangenaru, V., Skwerer, S.: Sticky central limit theorems on open books. Ann. Appl. Probab. **23**(6), 2238–2258 (2013)
26. Huckemann, S.: Inference on 3D Procrustes means: Tree boles growth, rank-deficient diffusion tensors and perturbation models. Scand. J. Stat. **38**(3), 424–446 (2011)
27. Huckemann, S.: Intrinsic inference on the mean geodesic of planar shapes and tree discrimination by leaf growth. Ann. Stat. **39**(2), 1098–1124 (2011)
28. Huckemann, S.: Manifold stability and the central limit theorem for mean shape. In: Gusnanto, A., Mardia, K.V., Fallaize, C.J. (eds.) Proceedings of the 30th LASR Workshop, pp. 99–103. Leeds University Press, Leeds (2011)
29. Huckemann, S.: On the meaning of mean shape: Manifold stability, locus and the two sample test. Ann. Inst. Stat. Math. **64**(6), 1227–1259 (2012)
30. Huckemann, S., Eltzner, B.: Polysphere PCA with applications. In: Proceedings of the Leeds Annual Statistical Research (LASR) Workshop, pp. 51–55. Leeds University Press, Leeds (2015)
31. Huckemann, S.F., Eltzner, B.: Backward nested descriptors asymptotics with inference on stem cell differentiation. Ann. Stat. **46**(5), 1994–2019 (2018)
32. Huckemann, S., Hotz, T., Munk, A.: Intrinsic shape analysis: Geodesic principal component analysis for Riemannian manifolds modulo Lie group actions (with discussion). Stat. Sin. **20**(1), 1–100 (2010)

33. Huckemann, S., Mattingly, J.C., Miller, E., Nolen, J.: Sticky central limit theorems at isolated hyperbolic planar singularities. Electron. J. Probab. **20**(78), 1–34 (2015)
34. Jung, S., Foskey, M., Marron, J.S.: Principal arc analysis on direct product manifolds. Ann. Appl. Stat. **5**, 578–603 (2011)
35. Jung, S., Dryden, I.L., Marron, J.S.: Analysis of principal nested spheres. Biometrika **99**(3), 551–568 (2012)
36. Kendall, D.G.: The diffusion of shape. Adv. Appl. Probab. **9**, 428–430 (1977)
37. Kendall, W.S., Le, H.: Limit theorems for empirical Fréchet means of independent and non-identically distributed manifold-valued random variables. Braz. J. Probab. Stat. **25**(3), 323–352 (2011)
38. Kendall, D.G., Barden, D., Carne, T.K., Le, H.: Shape and Shape Theory. Wiley, Chichester (1999)
39. Le, H., Barden, D.: On the measure of the cut locus of a Fréchet mean. Bull. Lond. Math. Soc. **46**(4), 698–708 (2014)
40. Mardia, K.V., Kent, J.T., Bibby, J.M.: Multivariate Analysis. Academic Press, New York (1980)
41. McKilliam, R.G., Quinn, B.G., Clarkson, I.V.L.: Direction estimation by minimum squared arc length. IEEE Trans. Signal Process. **60**(5), 2115–2124 (2012)
42. Nye, T.M., Tang, X., Weyenberg, G., Yoshida, R.: Principal component analysis and the locus of the fréchet mean in the space of phylogenetic trees. Biometrika **104**(4), 901–922 (2017)
43. Pennec, X.: Barycentric subspace analysis on manifolds. Ann. Stat. **46**(6A), 2711–2746 (2018)
44. Pizer, S.M., Jung, S., Goswami, D., Vicory, J., Zhao, X., Chaudhuri, R., Damon, J.N., Huckemann, S., Marron, J.: Nested sphere statistics of skeletal models. In: Innovations for Shape Analysis, pp. 93–115. Springer, Berlin (2013)
45. Romano, J.P., Lehmann, E.L.: Testing Statistical Hypotheses. Springer, Berlin (2005)
46. Schulz, J.S., Jung, S., Huckemann, S., Pierrynowski, M., Marron, J., Pizer, S.: Analysis of rotational deformations from directional data. J. Comput. Graph. Stat. **24**(2), 539–560 (2015)
47. Siddiqi, K., Pizer, S.: Medial Representations: Mathematics, Algorithms and Applications. Springer, Berlin (2008)
48. Sommer, S.: Horizontal dimensionality reduction and iterated frame bundle development. In: Geometric Science of Information, pp. 76–83. Springer, Berlin (2013)
49. Telschow, F.J., Huckemann, S.F., Pierrynowski, M.R.: Functional inference on rotational curves and identification of human gait at the knee joint (2016). arXiv preprint arXiv:1611.03665
50. van der Vaart, A.: Asymptotic Statistics. Cambridge University Press, Cambridge (2000)
51. Ziezold, H.: Expected figures and a strong law of large numbers for random elements in quasi-metric spaces. In: Transaction of the 7th Prague Conference on Information Theory, Statistical Decision Function and Random Processes, pp. 591–602. Springer, Berlin (1977)

Chapter 11
Advances in Geometric Statistics for Manifold Dimension Reduction

Xavier Pennec

Contents

Abstract Geometric statistics aim at shifting the classical paradigm for inference from points in a Euclidean space to objects living in a non-linear space, in a consistent way with the underlying geometric structure considered. In this chapter, we illustrate some recent advances of geometric statistics for dimension reduction in manifolds. Beyond the mean value (the best zero-dimensional summary statistics of our data), we want to estimate higher dimensional approximation spaces fitting our data. We first define a family of natural parametric geometric subspaces in manifolds that generalize the now classical geodesic subspaces: barycentric subspaces are implicitly defined as the locus of weighted means of $k + 1$ reference points with positive or negative weights summing up to one. Depending on the definition of the mean, we obtain the Fréchet, Karcher or Exponential Barycentric subspaces (FBS/KBS/EBS). The completion of the EBS, called the affine span of the points in a manifold is the most interesting notion as it defines complete sub-(pseudo)-spheres in constant curvature spaces. Barycentric subspaces can be characterized

X. Pennec (✉)
Universtié Côte d'Azur, Nice, France

Inria, Sophia-Antipolis, France
e-mail: Xavier.Pennec@inria.fr

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_11

very similarly to the Euclidean case by the singular value decomposition of a certain matrix or by the diagonalization of the covariance and the Gram matrices. This shows that they are stratified spaces that are locally manifolds of dimension k at regular points. Barycentric subspaces can naturally be nested by defining an ordered series of reference points in the manifold. This allows the construction of inductive forward or backward properly nested sequences of subspaces approximating data points. These flags of barycentric subspaces generalize the sequence of nested linear subspaces (flags) appearing in the classical Principal Component Analysis. We propose a criterion on the space of flags, the accumulated unexplained variance (AUV), whose optimization exactly lead to the PCA decomposition in Euclidean spaces. This procedure is called barycentric subspace analysis (BSA). We illustrate the power of barycentric subspaces in the context of cardiac imaging with the estimation, analysis and reconstruction of cardiac motion from sequences of images.

11.1 Introduction

Statistical computing on simple manifolds like spheres or flat tori raises problems due to the non-linearity of the space. For instance, averaging points on a sphere using the properties of the embedding Euclidean space leads to a point located inside the sphere and not on its surface. More generally, the classical mean value of random numeric values with distribution P is defined through an integral $\bar{x} = \int x dP(x)$, which can be rewritten as an implicit barycentric equation $\int (x - \bar{x}) dP(x)$. Notice that this notion is intrinsically affine. However, since an integral is a linear operator, this definition of the mean is bound to fail for general non-linear spaces. Geometric statistics were born out of this observation by Maurice Fréchet in the 1940's.

Geometric statistics is a rapidly evolving domain at the confluent of several mathematical and application domains. It was driven in the 1980s by Kendall's shape spaces, which encode the configuration of k points under a transformation group, often rigid body transformations or similarities, see e.g. [9, 24, 29, 46]. Applied mathematicians and computer scientists got interested in the 1990s in computing and optimizing on specific matrix manifolds, like rotations, rigid body transformations, Stiefel and Grassmann manifolds [10, 16, 17, 34, 42]. In the context of computer vision and medical image analysis applications, the Fréchet mean was used to develop a practical toolbox for statistics on manifolds in the 1990s [36, 37], with applications to the uncertainty of medical image registration and statistics on shapes. For instance, statistical distances such as the Mahalanobis distance were developed to define some simple statistical tests on manifolds. With the example of diffusion tensor images, this statistical toolbox was generalized in [43] to many manifold valued image processing algorithms such as interpolation, filtering, diffusion and restoration of missing data. Some of these statistics were generalized to even more non-Euclidean data like trees and graphs with object-oriented data analysis [32].

In this chapter, we illustrate some recent advances for dimension reduction in manifolds. Beyond the mean value, which is a good zero-dimensional summary statistic of our data, we want to estimate higher dimensional approximation spaces fitting our data. In [39], we have proposed a new and general family of subspaces in manifolds, *barycentric subspaces*, implicitly defined as the locus of weighted means of $k + 1$ reference points. Barycentric subspaces can naturally be nested and allow the construction of inductive forward or backward nested subspaces approximating data points. We can also consider the whole hierarchy of embedded barycentric subspaces defined by an ordered series of points in the manifold (a flag of affine spans): optimizing the accumulated unexplained variance (AUV) over all the subspaces actually generalizes PCA to non Euclidean spaces, a procedure named Barycentric Subspaces Analysis (BSA). We illustrate the power of barycentric subspaces in the context of cardiac imaging with the estimation, analysis and reconstruction of cardiac motion from sequences of images.

11.2 Means on Manifolds

The Fréchet Mean

Maurice Fréchet was the first to try to generalize the notion of mean, median and other types of typical central values to abstract spaces for statistical purposes. In a preparatory work [14] he first investigated different ways to compute mean values of random triangles, independently of their position and orientation in space. In the same paper he reports a series of experiments with careful measurements to reproduce the theoretical values. In this respect, he was really pioneering the statistical study of shapes. In a second study, motivated by the study of random curves, he first introduced a mean value and a law of large numbers defined by a generalization of the integral to normed vector (Wiener of Banach) spaces. Finally, Fréchet considered in [15] the generalization of many central values (including the mean and the median) to random elements in abstract metric spaces.

Definition 11.1 (Fréchet Mean in a Metric Space [15, p. 233]) The p-mean (typical position of order p according to Fréchet) of a distribution μ (a random element) in an abstract metric space $\mathcal{M}$ is set of minimizers of the mean p-distance (MpD):

$$\mathrm{Mean}_p(\mu) = \left\{ \arg\min_{y \in \mathcal{M}} \frac{1}{p} \int_{\mathcal{M}} \mathrm{dist}(x, y)^p \, d\mu(x) \right\}. \tag{11.1}$$

In a Euclidean space, this defines the usual arithmetic mean for $p = 2$, the median (*valeur équiprobable* in Fréchet's words) for $p = 1$ and the barycenter of the support of the distribution for $p = \infty$.

The first key contribution of Fréchet was to consider many different types of typical elements, including of course the mean but also the median. Fréchet considered mainly the case $p \geq 1$, but he observed that many of the properties could be also generalized to $0 < p < 1$. The second key contribution of Fréchet was to consider the mean as a set of elements rather than one unique element.[1] This key innovation was later developed by Ziezold [54] with a strong law of large numbers for sets of random elements in separable finite quasi-metric spaces. These two innovations justify the name naming of Fréchet mean that is used in geometric statistics.

The Riemannian Center of Mass

In a complete metric space, the existence of the Fréchet p-mean is ensured if the MpD is finite at one point, thus at all points thanks to the triangle inequality. The uniqueness is a much more difficult problem. For smooth differential geometric spaces like Riemannian spaces, and restricting to the classical mean with $p = 2$, it has been argued in [2, p. 235] that *the existence of a unique center of mass in the large for manifolds with non-positive curvature was proven and used by Élie Cartan back in the 1920's.* In order to find the fixed point of a group of isometries, Cartan indeed showed in [4][2] that the sum of the square distance to a finite number of points has a unique minimum in simply connected Riemannian manifolds with non-positive curvature (now called Hadamard spaces). This result was extended in [5] to closed subgroups of isometries[3]: "Let us apply to the point at origin O the displacements defined by the transformations of γ. The group γ being closed, we obtain a closed manifold V (that can be reduced to a point). But in a Riemann space without singular point at finite distance, simply connected, with negative or zero curvature, given points in finite number, we can find a fixed point invariant by all the displacements that exchanges these points: this is the point for which the sum of square distances to the given point is minimal [4, p. 267]. This property is still true if, instead of a finite number of points, we have an infinite number forming a closed manifold: we arrive at the conclusion that the group γ, which leave evidently the manifold V invariant, also leaves invariant a fixed point of the space. Thus, this point belongs to the group or (isometric) rotations around this point. But this group is homologous to g in the continuous adjoint group, which demonstrate the

[1] Page 259: "It is not certain that such an element exists nor that it is unique."

[2] Note III on normal spaces with negative or null Riemannian curvature, p. 267.

[3] Appliquons au point origine O les différents déplacements définis par les transformations de γ. Le groupe γ étant clos, nous obtenons ainsi une variété fermée V (qui peut se réduire à un point). Or, dans un espace de Riemann sans point singulier à distance finie, simplement connexe, a courbure negative ou nulle, on peut trouver, étant donnés des points en nombre fini, un point fixe invariant par tous les déplacements qui échangent entre eux les points donnés: c'est le point pour lequel la somme des carrés des distances au point donné est minima [4, p. 267]. Cette propriété est encore vraie si, au lieu d'un nombre fini de points, on en a une infinité formant une variété fermée: nous arrivons donc à la conclusion que le groupe γ qui laisse évidemment invariante la variété V, laisse invariant un point fixe de l'espace, il fait done partie du groupe des rotations (isométriques) autour de ce point. Mais ce groupe est homologue à g dans le groupe adjoint continu, ce qui démontre le théoreme.

theorem." It is obvious in this text that Cartan is only using the uniqueness of the sum of square distances as a tool in the specific case of negative curvature Riemannian manifolds and not as a general definition of the mean on manifolds as is intended in probability or statistics.

In 1973, for similar purposes, Grove and Karcher extended Cartan's idea to positive curvature manifolds. However, they restricted their definition to distribution with sufficiently small support so that the mean exists and is unique [18]. The notion was coined as The *Riemannian center of mass*. In this publication and in successive ones, Karcher and colleagues used Jacobi field estimates to determine the largest convex geodesic ball that support this definition. The Riemannian barycenter is commonly referred to as being introduced in [22]. However, the most complete description of the related properties is certainly found in [3], where a notion of barycenter in affine connection manifolds is also worked out. A good historical note on the history of the Riemannian barycenter is given in [1] and by Karcher himself in [23].

Exponential Barycenters

In all the above works, the Riemannian center of mass is by definition unique. Considering a set-valued barycenter on an affine connection manifold was the contribution of Emery and Mokobodzki [11]. In a Riemannian manifold, at the points $x \in \mathcal{M}$ where the cut locus has null measure for the measure μ, the critical points of the mean square distance are characterized by the barycentric equation:

$$\mathfrak{M}_1(x) = \int_{\mathcal{M}} \log_x(y) d\mu(y) = 0, \tag{11.2}$$

where $\log_x(z)$ is the initial tangent vector of the minimizing geodesic joining x to z (the Riemannian log). This barycentric equation was taken as the definition of *exponential barycenters* of the probability measure [11] in more general affine connection spaces. Notice that the notion remains purely affine, provided that the distribution has a support on a convex neighborhood in which the logarithm can be defined uniquely. The non-uniqueness of the expectation of a random variable considerably extended the usability of this definition, in particular in positive curvature spaces. In the Riemannian case, exponential barycenters contain in particular the smooth local (and the global) minimizers of the mean square distance (MSD) $\sigma^2(x) = \int_M \text{dist}^2(x, y)\mu(dy)$ (except those at which the MSD is not differentiable). Exponential barycenters were later used in [7, 35] and in [31, 40, 41] for bi-invariant means on Lie groups endowed with the canonical symmetric Cartan-Schouten connection.

Uniqueness of the Fréchet Mean

Conditions for the uniqueness of the minimum of the sum of square distance have been studied by Karcher [3, 22] and further optimized in [25, 27, 28].

Theorem 11.2 (Karcher and Kendall Concentration (KKC) Conditions) *Let $\mathcal{M}$ be a geodesically complete Riemannian manifold with injection radius inj(x)*

at x. Let μ is a probability distribution on $\mathcal{M}$ whose support is contained a closed regular geodesic ball $\bar{B}(x, r)$ of radius $r < \frac{1}{2} inj(x)$. We assume moreover that the upper bound $\kappa = \sup_{x \in B(x,r)}(\kappa(x))$ of the sectional curvatures in that ball satisfies $\kappa < \pi^2/(4r)^2$. This second condition is always true on spaces of negative curvature and specifies a maximal radius $r^* = \frac{\pi}{2\sqrt{\kappa}}$ when there is positive sectional curvature. These concentration assumptions ensure that the Mean square distance has a unique global minimum that belongs to the ball $\bar{B}(x, r)$.

The result has been extended to Fréchet p-means defined as the minimizers of the mean p-distance in [1, 51, 52].

In order to differentiate the different notions of means in Riemannian manifolds, it has been agreed in geometric statistics to name Fréchet mean the set of global minimizers of the MSD, Karcher mean the set of local minimizers, and exponential barycenters the set of critical points satisfying the implicit equation $\mathfrak{M}_1(x) = 0$. It is clear that all these definition boils down for the classical 2-mean to the same unique point within the ball $B(x, r)$ of the KKC conditions. Notice that we may still have local minima and critical points located outside this ball.

11.3 Statistics Beyond the Mean Value: Generalizing PCA

The mean value is just a zero-dimensional summary statistics of the data. If we want to retain more information, we need to add more degrees of freedom, and Principal Component Analysis (PCA) is the ubiquitous tool for that. In a Euclidean space, the principal k-dimensional affine subspace of the PCA procedure is equivalently defined by minimizing the MSD of the residuals (the projection of the data point to the subspace) or by maximizing the explained variance within that affine subspace. This double interpretation is available through Pythagoras' theorem, which does not hold in more general manifolds. A second important observation is that principal components of different orders are nested, enabling the forward or backward construction of nested principal components.

Tangent PCA

Generalizing PCA to manifolds and to potentially more general stratified spaces is currently a very active research topic. The first step is the generalization of affine subspaces in manifolds. For the zero-dimensional subspace, the Fréchet mean is the natural intrinsic generalization of the mean around which PCA is performed. The one-dimensional component can naturally be a geodesic passing through the mean point. Higher-order components are more difficult to define. The simplest generalization is tangent PCA (tPCA), which amounts unfolding the whole distribution in the tangent space at the mean, and computing the principal components of the covariance matrix in the tangent space. The method is thus based on the maximization of the explained variance, which is consistent with the entropy maximization definition of a Gaussian on a manifold proposed by Pennec

[37]. tPCA is actually implicitly used in most statistical works on shape spaces and Riemannian manifolds because of its simplicity and efficiency. However, if tPCA is good for analyzing data which are sufficiently centered around a central value (unimodal or Gaussian-like data), it is often not sufficient for distributions which are multimodal or supported on large compact subspaces (e.g. circles or spheres).

Principal Geodesic Analysis (PGA)
Instead of an analysis of the covariance matrix, [13] proposed the minimization of squared distances to subspaces which are totally geodesic at a point, a procedure coined Principal Geodesic Analysis (PGA). These Geodesic Subspaces (GS) are spanned by the geodesics going through a point with tangent vector restricted to a linear subspace of the tangent space. In fact, the tangent vectors also need to be restricted to the interior of the tangential cut locus within this linear subspace if we want to generate a submanifold of the original manifold [39]. The idea of minimizing the unexplained variance (i.e. the norm of the residuals) is really interesting and corresponds exactly to what we want to do in manifold dimension reduction. However, the non-linear least-squares procedure to optimize the geodesic subspace is computationally expensive, so that [13] approximated in practice PGA with tPCA. This is really unfortunate because this led many people to confuse PGA ant tPCA. A real implementation of the original PGA procedure was only recently provided by Sommer et al. [48]. PGA is allowing to build a flag (sequences of embedded subspaces) of principal geodesic subspaces consistent with a forward component analysis approach. Components are built iteratively from the mean point by selecting the tangent direction that optimally reduces the square distance of data points to the geodesic subspace.

Geodesic PCA
In the PGA procedure, the mean always belongs to geodesic subspaces even when it is out of the distribution support. To alleviate this problem Huckemann [19, 20] proposed to start at the first order component directly with the geodesic that best fits the data, which is not necessarily going through the mean. The second principal geodesic is chosen orthogonally at a point of the first one, and higher order components are added orthogonally at the crossing point of the first two components. The method was named Geodesic PCA (GPCA). Further relaxing the assumption that second and higher order components should cross at a single point Sommer proposed a parallel transport of the second direction along the first principal geodesic to define the second coordinates, and iteratively define higher order coordinates through horizontal development along the previous modes [47].

Principal Nested Spheres
All the previous extensions of PCA are intrinsically forward methods that build successively larger approximation spaces for the data. A notable exception to this principle is the concept of Principal Nested Spheres (PNS), proposed by Jung et al. [21] in the context of planar landmarks shape spaces. A backward analysis approach determines a decreasing family of nested subspheres by slicing a higher dimensional sphere with affine hyperplanes. In this process, the nested subspheres are not of

radius one, unless the hyperplanes pass through the origin. Damon and Marron have recently generalized this approach to manifolds with the help of a “nested sequence of relations” [6]. However, such a sequence was only known so far for spheres or Euclidean spaces.

11.3.1 Barycentric Subspaces in Manifolds

The geodesic subspaces used for tPCA, PGA and GPCA are described by one point and k tangent vectors at that point. This give a special role to this reference point which is not found in the higher dimensional descriptors. Moreover, these spaces are totally geodesic at the reference point but generally nowhere else. This asymmetry may not be optimal for multimodal distributions that do not have a single ‘pole’.

Fréchet, Karcher and Exponential Barycentric Subspaces
In order to have a fully symmetric and ‘multi-pole’ description of subspaces, we have proposed in [39] a new and more general type of subspaces in manifolds: *barycentric subspaces* are implicitly defined as the locus of weighted means of $k+1$ reference points with positive or negative weights summing up to one. This time the descriptors are fully symmetric (they are all standard points). Depending on the definition of the mean, we obtain the Fréchet, Karcher or exponential barycentric subspaces (FBS/KBS/EBS). The Fréchet (resp. Karcher) barycentric subspace of the points $(x_0, \dots x_k) \in \mathcal{M}^{k+1}$ is the locus of weighted Fréchet (resp. Karcher) means of these points, i.e. the set of global (resp. local) minimizers of the weighted mean square distance: $\sigma^2(x, \lambda) = \frac{1}{2} \sum_{i=0}^{k} \lambda_i \operatorname{dist}^2(x, x_i)$:

$$\mathrm{FBS}(x_0, \dots x_k) = \left\{ \arg\min_{x \in \mathcal{M}} \sigma^2(x, \lambda), \lambda \in R^{k+1}, \mathbb{1}^\top \lambda = 1 \right\}.$$

The EBS is the locus of weighted exponential barycenters of the reference points (critical points of the weighted MSD) defined outside their cut-locus as follows.

Definition 11.3 (Exponential Barycentric Subspace (EBS) and Affine Span) A set of $k+1$ points $\{x_0, \dots x_k\} \in \mathcal{M}^{k+1}$ is affinely independent if no point is in the cut-locus of another and if all the $k+1$ sets of k vectors $\{\log_{x_i}(x_j)\}_{0 \le j \ne i \le k} \in T_{x_i}\mathcal{M}^k$ are linearly independent.

The EBS of $k+1$ affinely independent points $(x_0, \dots x_k)$ is the locus of weighted exponential barycenters of the reference points:

$$\mathrm{EBS}(x_0, \dots x_k) = \{x \in \mathcal{M} \setminus \mathrm{Cut}(x_0, \dots x_k) \,|\, \exists \lambda \in R^{k+1}, \mathbb{1}^\top \lambda = 1 : \textstyle\sum_i \lambda_i \log_x(x_i) = 0\}.$$

The affine span is the closure of the EBS in $\mathcal{M}$: $\mathrm{Aff}(x_0, \dots x_k) = \overline{\mathrm{EBS}}(x_0, \dots x_k)$. Because we assumed that $\mathcal{M}$ is geodesically complete, this is equivalent to the metric completion of the EBS.

Thus, outside the cut locus of the reference points, we clearly see the inclusion $FBS \subset KBS \subset EBS$. The completeness of the affine span allows ensuring that a closest point exists on the subspace, which is fundamental in practice for optimizing the subspaces by minimizing the distance of the data to their projection. This definition works on metric spaces more general than Riemannian manifolds. In stratified metric spaces, the barycentric subspace spanned by points belonging to different strata naturally maps over several strata [50].

Barycentric subspaces can be characterized very similarly to the Euclidean case by the singular value decomposition of a certain matrix or by the diagonalization of the covariance and the Gram matrices. Let $Z(x) = [\log_x(x_0), \dots \log_x(x_k)]$ be the matrix field of the log of the reference points x_i in a local coordinate system. This is a smooth field outside the cut locus of the reference points. The EBS is the zero level-set of the smallest singular value of $Z(x)$. The associated right singular vector gives the weights λ that satisfy the barycentric equation $\sum_i \lambda_i \log_x(x_i) = 0$. Denoting $G(x)$ the matrix expression of the Riemannian metric, we can also define the smooth $(k+1) \times (k+1)$ Gram matrix field $\Omega(x) = Z(x)^{\mathrm{T}} G(x) Z(x)$ with components $\Omega_{ij}(x) = \left\langle \overrightarrow{xx_i} \mid \overrightarrow{xx_j} \right\rangle_x$ and the (scaled) $n \times n$ covariance matrix field of the reference points $\Sigma(x) = \sum_{i=0}^{k} \overrightarrow{xx_i}\, \overrightarrow{xx_i}^{\mathrm{T}} = Z(x) Z(x)^{\mathrm{T}}$. With these notations, $\mathrm{EBS}(x_0, \dots x_k)$ is the zero level-set of $\det(\Omega(x))$, the minimal eigenvalue σ_{k+1}^2 of $\Omega(x)$, the $k+1$ eigenvalue (in decreasing order) of the covariance $\Sigma(x)$.

Example in Constant Curvature Spaces

It is interesting to look at the barycentric subspaces in the most simple non-linear spaces: constant curvature spaces. The sphere can be represented by the classical embedding of the unit sphere $\mathcal{S}_n \subset \mathbb{R}^{n+1}$ and the hyperbolic space as the unit pseudo-sphere of the Minkowski space $\mathbb{R}^{1,n}$. With this model, the affine span of $k+1$ reference points is particularly simple: it is the great subsphere (resp great sub-pseudosphere) obtained by the intersection of the (pseudo) sphere with the hyperplane generated by the reference points in the embedding space. Notice that the EBS in the sphere case is the great subsphere (the affine span) minus the antipodal points (the cut locus) of the reference points. However, even if the affine span is simple, understanding the Fréchet/Karcher barycentric subspaces is much more involved. In order to find the local minima among the critical points, we have to compute the Hessian matrix of the weighted MSD, and to look at the sign of its eigenvalues: at the critical points with a non-degenerate Hessian (regular points), local minima are characterized by a positive definite Hessian. In fact, the zero-eigenvalues of the Hessian subdivide the EBS into a cell complex according to the index (the number of positive eigenvalues) of the Hessian. This index is illustrated on Fig. 11.1 for a few configuration of 3 affinely independent reference points on the 2-sphere: we clearly see that the positive points of the KBS do not in general cover the full subsphere containing the reference points. It may even be disconnected, contrarily to the affine span which consistently covers the whole subsphere. For subspace definition purposes, this shows that the affine span is a more interesting definition than KBS/FBS.

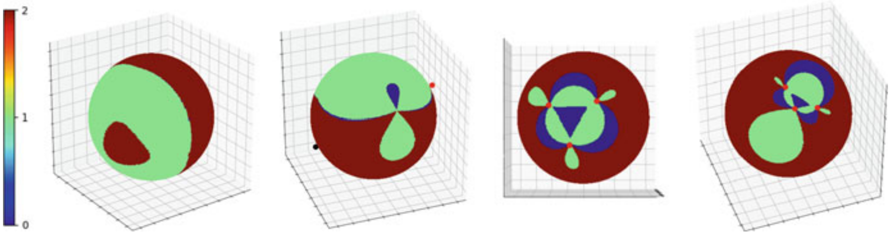


Fig. 11.1 Signature of the weighted Hessian matrix for different configurations of 3 reference points (in black, antipodal point in red) on the 2-sphere: the locus of local minima (KBS) in brown does not cover the whole sphere and can even be disconnected (first example). Reproduced from [39]

Properties of Barycentric Subspaces

For $k+1$ affinely independent reference points, the affine span is a smooth submanifold of dimension k in a neighborhood of each of the reference points. Further away from the reference points, one can show that the EBS is a stratified spaces of maximal dimension k at the regular points [39]. At the degenerate points where the Hessian has null eigenvalues, pathological behavior with an increasing dimension may perhaps appear although this has not been observed in constant curvature spaces.

When the reference points belong to a sufficiently small regular geodesic ball (typically satisfying the KKC conditions of Theorem 11.2), then the FBS with positive weights, called the barycentric simplex, is the graph of a k dimensional function. The barycentric k-simplex contains all the reference points and the geodesics segments between the reference points. The $(k-l)$-faces of a simplex are the simplices defined by the barycentric subspace of $k-l+1$ points among the $k+1$. They are obtained by imposing the l remaining barycentric coordinates to be zero. Barycentric simplexes were investigated in [50] as extensions of principal subspaces in the negatively curved metric space of trees under the name Locus of Fréchet mean (LFM).

Making the $k+1$ reference points become closer together, we may imagine that at the limit they coalesce at one point x along k directions $(v_1, \dots v_k) \in T_x\mathcal{M}^k$. In that case, one can show that the EBS converges to the geodesic subspace generated by geodesics starting at x with tangent vectors that are linear combination of the vectors $(v_1, \dots v_k)$ (up to the cut locus of x). More generally, the reference points may converge to a non local jet,[4] which may give a new way to define multi-dimensional splines in manifolds.

[4] p-jets are equivalent classes of functions up to order p. Thus, a p-jet specifies the Taylor expansion of a smooth function up to order p. Non-local jets, or multijets, generalize subspaces of the tangent spaces to higher differential orders with multiple base points.

11.3.2 From PCA to Barycentric Subspace Analysis

The nestedness of approximation spaces is one of the most important characteristics for generalizing PCA to more general spaces [6]. Barycentric subspaces can easily be nested by adding or removing one or several points at a time, which corresponds to put the barycentric weight of this (or these) point(s) to zero. This gives a family of embedded submanifolds called a flag because this generalizes flags of vector spaces.

Forward and Backward Approaches

With a forward analysis, we compute iteratively the flag of affine spans by adding one point at a time keeping the previous ones fixed. Thus, we begin by computing the optimal barycentric subspace $\mathrm{Aff}(x_0) = \{x_0\}$, which is the set of exponential barycenters. This may be a Fréchet mean or more generally a critical value of the unexplained variance. Adding a second point amounts to computing the geodesic passing through the mean that best approximates the data. Adding a third point now generally differs from PGA. In practice, the forward analysis should be stopped at a fixed number or when the variance of the residuals reaches a threshold (typically 5% of the original variance). We call this method the forward barycentric subspace (FBS) decomposition of the manifold data. Due to the greedy nature of this forward method, the affine span of dimension k defined by the first $k + 1$ points is not in general the optimal one minimizing the unexplained variance.

The backward analysis consists in iteratively removing one dimension. Starting with a set of points $x_0, \dots x_n$ which generates the full manifold, we could start to choose which one to remove. However, as the affine span of $n + 1$ linearly independent points generate the full manifold, the optimization really begins with n points. Once they are fixed, the optimization boils down to test which point should be removed. In practice, we may rather optimize $k + 1$ points to find the optimal k-dimensional affine span, and then reorder the points using a backward sweep to find inductively the one that least increases the unexplained variance. We call this method the k-dimensional pure barycentric subspace with backward ordering (k-PBS). With this method, the k-dimensional affine span is optimizing the unexplained variance, but there is no reason why any of the lower dimensional ones should.

Barycentric Subspace Analysis: Optimizing a Criterion on the Whole Flag of Subspaces

In order to obtain optimal subspaces which are embedded consistency across dimensions, it is necessary to define a criterion which depends on the whole flag of subspaces and not on each of the subspaces independently. In PCA, one often plots the unexplained variance as a function of the number of modes used to approximate the data. This curve should decreases as fast as possible from the variance of the data (for 0 modes) to 0 (for n modes). A standard way to quantify the decrease consists in summing the values at all steps. This idea gives the Accumulated Unexplained Variances (AUV) criterion [39], which sums the unexplained variance (the sum of squared norm of the residuals) by all the subspaces of the flag. AUV is analogous to the Area-Under-the-Curve (AUC) in Receiver Operating Characteristic

(ROC) curves. In practice, one can stop at a maximal dimension k like for the forward analysis in order to limit the computational complexity. This analysis limited to a flag defined by $k + 1$ points is denoted k-BSA. Let us denote by $\sigma^2_{out}(\text{Aff}(x_0, ..x_k))$ the sum of the squared distance of the data points to the affine subspace $\text{Aff}(x_0, ..x_k)$ (the variance unexplained by this subspace). The AUV of the flag $\text{Aff}(x_0) \subset \text{Aff}(x_0, x_1) \subset \ldots \text{Aff}(x_0, ..x_k)$ is

$$AUV(x_0, x_1, \ldots, x_k) = \sigma^2_{out}(\text{Aff}(x_0)) + \sigma^2_{out}(\text{Aff}(x_0, x_1)) + \ldots + \sigma^2_{out}(\text{Aff}(x_0, ..x_k)). \tag{11.3}$$

In a Euclidean space, minimizing the unexplained variance with respect to a k-dimensional affine subspace leads to select the hyperplane going through the mean of the data and spanned by the eigenvectors $(v_1, \ldots v_k)$ of the covariance matrix associated to the largest k eigenvalues. However, this subspace is independent of the ordering of the k selected eigenvectors, and nothing in this criterion helps us to select an optimal smaller dimensional subspace. Thus, the PCA ordering of the subspaces that produces a hierarchy of embedded subspaces of larger dimension is not specified by the unexplained variance criterion (nor by the explained variance) alone. If we now consider the AUV criterion on a flag of subspaces (say up to order k), then one can show that the unique solution is the flag generated with the increasing ordering of the eigenvalues. This shows that PCA is actually optimizing the AUV criterion on the space of affine flags. The generalization to the optimization of the AUV criterion on the space of flags of affine spans in Riemannian manifolds is called Barycentric Subspaces Analysis (BSA).

Forward, Backward and Optimal Flag Estimation

In summary, we may consider three main algorithms to estimate a flag of barycentric subspaces of maximal dimension k:

- The Forward Barycentric Subspace decomposition (k-FBS) iteratively adds the point that minimizes the unexplained variance up to $k + 1$ points. With this method, only the first (usually zero or one dimensional) subspace optimized is optimal and all the larger ones are suboptimal for the unexplained variance.
- The optimal Pure Barycentric Subspace with backward reordering (k-PBS) estimates the $k + 1$ points that minimize the unexplained variance for a subspace of dimension k, and then reorders the points accordingly for lower dimensions. Here only the largest subspace si optimal, and all the smaller ones are suboptimal.
- The Barycentric Subspace Analysis of order k (k-BSA) looks for the flag of affine spans defined by $k + 1$ ordered points that optimized the AUV. Here none of the subspaces are optimal for the unexplained variance of the corresponding dimension, but the whole flag is optimal for the AUV criterion.

11.3.3 Sample-Limited L_p Barycentric Subspace Inference

Sample-Limited Inference

In order to compute the optimal subspaces or flags of subspaces, we need to set-up an optimization procedure. This can be realized by standard gradient descent techniques for optimization on manifolds. However, to avoid gradient computations but also to avoid finding optima that are far away from any data point, it has been proposed to limit the inference of the Fréchet mean to the data-points only. For instance, in neuroimaging studies, the individual image minimizing the sum of square deformation distance to other subject images was found to be a good alternative to the mean template (a Fréchet mean in deformation and intensity space) because it conserves the original characteristics of a real subject image [30]. Beyond the Fréchet mean, Feragen et al. proposed to define the first principal component mode as the unexplained variance minimizing geodesic going through two of the data points [12]. The method named *set statistics* was aiming to accelerate the computation of statistics on tree spaces. Zhai [53] further explored this idea under the name of *sample-limited geodesics* in the context of PCA in phylogenetic tree space. In both cases, defining principal modes of order larger than two was seen as an unsolved challenging research topic.

With barycentric subspaces, the idea of sample-limited statistics naturally extends to any dimension by restricting the search to the (flag of) affine spans that are parametrized by points sampled from the data. The implementation boils down to an enumeration problem. With this technique, the reference points are never interpolated as they are by definition sampled from the data. This is a important advantage for interpreting the modes of variation since we may go back to other information about the samples like the medical history and disease type. The search can be done exhaustively for a small number of reference points. The main drawback is the combinatorial explosion of the computational complexity with the dimension for the optimal order-k flag of affine spans, which is involving $O(N^{k+1})$ operations, where N is the number of data points. In [38] we perform an exhaustive search, but approximate optima can be sought using a limited number of randomly sampled points [12].

Adding Robustness with L_p Norms

Since barycentric subspaces minimize the weighted MSD, one could think of taking a power p of the metric to define the mean p-distance (MpD) $\sigma^p(x) = \frac{1}{p}\sum_{i=0}^{k} \operatorname{dist}^p(x, x_i)$. We recall that the global minimizers of this MdP defines the Fréchet median for $p = 1$, the Fréchet mean for $p = 2$ and the barycenter of the support of the distribution for $p = \infty$. This suggests to further generalize barycentric subspaces by taking the locus of the minima of the weighted unexplained MpD $\sigma^p(x, \lambda) = \frac{1}{p}\sum_{i=0}^{k} \lambda_i \operatorname{dist}^p(x, x_i)$. However, it turns out that the critical points of the weighted this criterion are necessarily included in the EBS: since the gradient of the criterion is (except at the reference points and their cut loci):

$$\nabla_x \sigma^p(x, \lambda) = \nabla_x \frac{1}{p} \sum_{i=0}^k \lambda_i (\operatorname{dist}^2(x, x_i))^{p/2} = -\sum_{i=0}^k \lambda_i \operatorname{dist}^{p-2}(x, x_i) \log_x(x_i).$$

Thus, we see that the critical points of the MpD satisfy the equation $\sum_{i=0}^k \lambda'_i \log_x(x_i) = 0$ for the new weights $\lambda'_i = \lambda_i \operatorname{dist}^{p-2}(x, x_i)$. Thus, they are also elements of the EBS and changing the power of the metric just amounts to a reparameterization of the barycentric weights. This stability of the EBS/affine span with respect to the power of the metric p shows that the affine span is really a central notion.

While changing the power does not change the subspace definition, it has a drastic impact on its estimation: minimizing the sum of L_p distance to the subspace for non-vanishing residuals obviously changes the relative influence of points [38]. It is well known that medians are more robust than least-squares estimators: the intuitive idea is to minimize the power of residuals with $1 \leq p \leq 2$ to minimize the influence of outliers. For $0 < p < 1$, the influence of the closest points becomes predominant, at the cost of non-convexity. In general, this is a problem for optimization. However, there is no gradient estimation in the sample-limited setting as we have to rely on an exhaustive search for the global minimum or on a stochastic approximation by testing only a subset of reference configurations. At the limit of $p = 0$, all the barycentric subspaces containing $k + 1$ points (i.e. all the sample-limited barycentric subspaces of dimension k that we consider) have the same L_0 sum of residuals, which is a bit less interesting.

For a Euclidean space, minimizing the sum of L_p norm of residuals under a rank k constraint is essentially the idea of the robust R1-PCA [8]. However, an optimal rank k subspace is not in general a subspace of the optimal subspace of larger ranks: we loose the nestedness property. An alternative PCA-L1 approach, which maximizes the L_1 dispersion within the subspace, was proposed in [26]. On manifolds, this would lead to a generalization of tangent-PCA maximizing the explained p-variance. In contrast, we proposed in [38] to minimize the Accumulated Unexplained p-Variance (L_p AUV) over all the subspaces of the flag under consideration. Since the subspace definition is not affected by the power p, we can compare the subspaces' parameters (the reference points) for different powers. It also allows to simplify the algorithms: as the (positive) power of a (positive) distance is monotonic, the closest point to an affine span for the 2-distance remains the closest point for the p-distance. Thus, the forward barycentric subspace analysis (k-FBS), the pure subspace with backward reordering analysis (k-PBS) and the barycentric subspace analysis (k-BSA) can be seamlessly generalized to their robust L_p version in the sample-limited setting.

11.4 Example Applications of Barycentric Subspace Analysis

11.4.1 Example on Synthetic Data in a Constant Curvature Space

The L_p variant of the forward (FBS), backward (PBS) and BSA algorithms were evaluated on synthetically generated data on spheres and hyperbolic spaces in [38]. The projection of a point of a sphere on a subsphere is almost always unique and corresponds to the renormalization of the projection on the Euclidean subspace containing the subsphere. The same property can be established for hyperbolic spaces, which can be viewed as pseudo-spheres embedded in a Minkowski space. Affine spans are great pseudo-spheres (hyperboloids) generated by the intersection of the plane containing the reference points with the pseudo-sphere, and the closest point on the affine span is the renormalization of the unique Minkowski projection on that plane [39]. In both cases, implementing the Riemannian norm of the residuals is very easy and the difficulty of sample-limited barycentric subspace algorithms analysis resides in the computational complexity of the exhaustive enumeration of tuples of points.

We illustrate in Fig. 11.2 the results of the L_p barycentric subspace algorithms on a set of 30 points in the 5D hyperbolic space generated at follows: we draw 5 random points (tangent Gaussian with variance 0.015) around each vertex of an equilateral triangle of length 1.57 centered at the bottom of the 5D hyperboloid embedded in the (1,5)-Minkowski space. As outliers, we add 15 points drawn according to a tangent Gaussian of variance 1.0 truncated at a maximum distance of 1.5 around the bottom of the 5D hyperboloid. This simulates three clusters living on a lower dimensional 2-pseudo-sphere with 50% of outliers (Fig. 11.2). With the L_2 hyperbolic distance, the 1-FBS and 1-BSA methods select outliers for their two reference points. 1-PBS manages to get one point in a cluster. For the two dimensional approximation

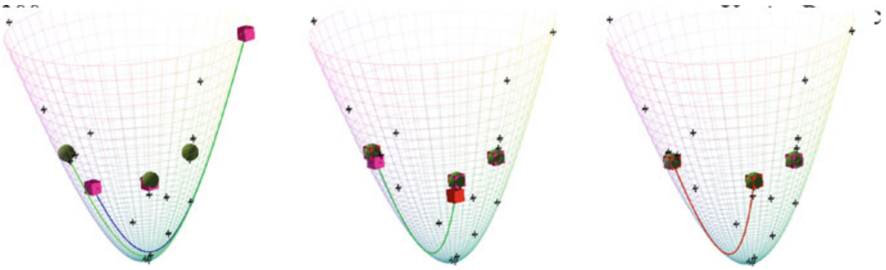


Fig. 11.2 Analysis of 3 clusters on a 5D hyperbolic space, projected to the expected 2-pseudo-sphere, with $p = 2$ (**left**), $p = 1$ (**middle**) and $p = 0.5$ (**right**). For each method (FBS in blue, 1-PBS in green, 1-BSA in red), the 1d mode is figured as a geodesic joining the two reference point (unseen red, blue or green geodesics are actually hidden by another geodesic). The three reference points of 2-PBS are represented with dark green solid circles, and the ones of 2-BSA with deep pink solid boxes. Reproduced, with permission from [38]

again with the L_2 distance, the 2-FBS and the 2-PBS select only one reference points within the clusters while 2-BSA correctly finds the clusters (Fig. 11.2 left, dark green points). With the L_1 distance, FBS, PBS and BSA select 3 very close points within the three clusters (Fig. 11.2 center). Lowering the power to $p = 0.5$ leads to selecting exactly the same points optimally centered within the 3 clusters for all the methods (Fig. 11.2 right). Thus, it seems that we could achieve some kind of *Principal Cluster Analysis* with the sample-limited L_p barycentric subspace analysis.

11.4.2 A Symmetric Group-Wise Analysis of Cardiac Motion in 4D Image Sequences

Understanding and analyzing the cardiac motion pattern in a patient is an important task in many clinical applications. The cardiac motion is usually studied by finding correspondences between each of the frames of the sequence and the first frame corresponding to the end-diastole (ED) image. This image registration process yields a dense displacement field that tracks the motion of the myocardium over the image sequence. Taking the ED image as a reference is natural as it is the starting point of the contraction of the heart which is the most important phase in evaluating the efficiency of the cardiac function. However, this specific choice can lead to important biases in quantifying the motion, especially at end-systole (ES) where the deformations to be evaluated are large. In [45], we proposed to build a multi-reference registration to a barycentric subspaces of the space of images representing cardiac motion instead of taking a unique reference image to evaluate the motion.

In the context of diffeomorphic medical image registration, 3D images are the "points" of our manifold while "geodesics" are the optimal deformations found by image registration to map one image to the other. In the Large Diffeomorphic Metric Mapping (LDDMM) at well as in the Stationary Velocity Field (SFV) registration frameworks, diffeomorphic deformations are obtained by the flow of velocity fields, and the tangent vector (i.e. a vector field over the image) at the initial point of the deformation trajectory registering image I to image J may be interpreted as $\log_I(J)$. We refer the reader to [31, 41] for a more in-depth description of the SVF framework and its application in medical image registration.

The barycentric subspace of dimension k spanned by $k + 1$ reference images R_1 to R_{k+1} is then defined as the set of images $\hat{I}$ for which there exists weights λ_i such that $\sum_{j=1}^{k+1} \lambda_j \log_{\hat{I}}(R_j) = 0$. Thus, projecting image I on this subspace amounts to find the smallest SVF $\hat{V}$ deforming image I to image $\hat{I}$ such that the SVFs $\hat{V}_1$,$\hat{V}_2$ and $\hat{V}_3$ encoding the deformation from this projected image to the three references R_1, R_2 and R_3 are linearly dependent (Fig. 11.3). The weights λ_j are the barycentric coordinates of image I in the "basis" (R_1, R_2, R_3). This process can be repeated for each image I_1 to I_N of the temporal sequence of one subject. This allows us to compute the unexplained variance $\sigma^2(R_1, R_2, R_3) = \sum_{i=1}^{N} \|\hat{V}(I_i)\|^2$ and to choose

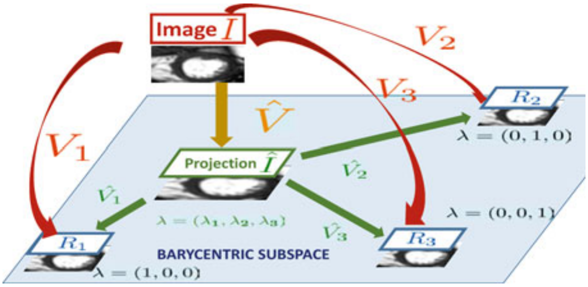


Fig. 11.3 Barycentric subspace of dimension 2 built from 3 references images (R_1, R_2, R_3). $\hat{I}$ is the projection of the image I within the barycentric subspace such that $\| \hat{V} \|^2$ is minimal under the conditions $\sum_j \lambda_j \hat{V}_j = 0$ and $\hat{V} + \hat{V}_j = V_j$. Reproduced from [45]

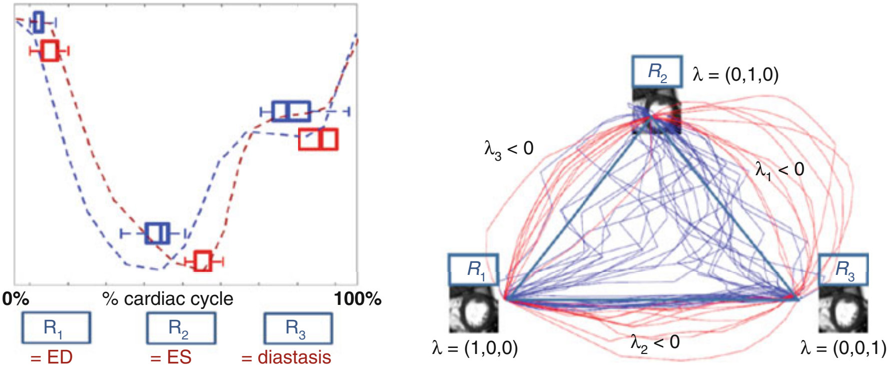


Fig. 11.4 Cardiac motion signature of 10 sequences of control subjects (in blue) and 16 sequences of Tetralogy of Fallot subjects (in red). Left: the time-index of the three optimal references superimposed on the average cardiac volume curve for each population. Right: the curve of barycentric coordinates of the images along the sequence of each subject, projected on the plane $\sum_i \lambda_i = 1$. Modified from [44, 45]

the optimal basis by repeating the experiment for all possible triplets of reference images within our sequence.

This methodology was used in [45] to compare the cardiac motion signature of two different populations. The first group consists of 10 healthy subjects from the STACOM 2011 cardiac motion tracking challenge dataset [49], and the second group is made of 10 Tetralogy of Fallot (ToF) patients [33]. Short axis cine MRI sequences were acquired with $T = 15$–30 frames. For each subject, we projected each of the frames of the cardiac sequence to a barycentric subspace of dimension 2 built by 3 reference images belonging to that sequence. A significant differences in the time-index of the optimal references frames can be seen between the two populations (Fig. 11.4, left). In particular, the second reference—corresponding to the end-systole—is significantly later for the ToF patients showing that this population has on average longer systolic contraction. The barycentric coefficients

(Fig. 11.4, right) also show significant differences between the groups, especially in the region $\lambda_1 < 0$. This demonstrates that this signature of the motion is encoding relevant features. Last but not least, the reconstruction based on the 3 reference images and the 30 barycentric coefficients along the sequence (a compression rate of 1/10) turned out to achieve a reconstruction error in intensity which is 40% less than the one of a classical tangent PCA with two deformation modes at the mean image of the sequence (compression rate 1/4 only). This demonstrates that the multi-reference approach based on barycentric subspaces can outperform the classical statistical shape analysis methods on real medical imaging problems.

11.5 Conclusion and Perspectives

We have drafted in this chapter a summary of some of the recent advances on manifold dimension reduction and generalization of principal component analysis to Riemannian manifolds. The first observation is that generalizing affine subspaces to manifolds with a minimization procedure (i.e. Fréchet or Karcher barycentric subspaces) leads to small disconnected patches that do not cover complete lower dimensional subspheres (resp. sub-pseudospheres) in constant curvature spaces. Considering the completion (the affine span) of all critical points (the exponential barycentric subspace) is needed to cover the full sub-(pseudo)-sphere. The fact that changing the MSD for the MpD does not change the affine span is an unexpected stability result which suggests that the notion is quite central. The second important point is that geodesic subspaces that were previously proposed for PGA or GPCA are actually particular cases of barycentric subspaces that can be obtained by taking reference points highly concentrated with respect to the distribution of our data points. The framework is thus more general. Last but not least, following [6], any interesting generalization of PCA should rely on sequences of properly nested spaces. Generalizing linear flags in Euclidean space, an ordering of the reference points naturally defines a flag of nested affine spans in a manifold. Now instead of defining the subspaces of the flag iteratively in a forward or backward manner, which is sub-optimal for each subspace considered independently, it turns out that PCA can be rephrase as an optimization of the AUV criterion (the sum of all unexplained variances by all the subspaces of the hierarchy) on the space of flags. Such a method coined Barycentric Subspace Analysis can be naturally extended to the L_p norm of residuals to account for outliers or different type of noises. BSA can also be performed by restricting reference points defining the subspaces to be a subset of the data points, thus considerably extending the toolbox of sample-limited statistics to subspaces of dimension larger than one, and an example application on 3D image sequences of the heart showed that many insights can be brought by this new methodology.

Acknowledgements This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant G-Statistics No 786854).

References

1. Afsari, B.: Riemannian L^p center of mass: existence, uniqueness, and convexity. Proc. Am. Math. Soc. **139**(2), 655–673 (2011)
2. Berger, M.: A Panoramic View of Riemannian Geometry. Springer, Berlin (2003)
3. Buser, P., Karcher, H.: Gromov's Almost Flat Manifolds. Number 81 in Astérisque. Société mathématique de France (1981)
4. Cartan, E.: Leçons sur la géométrie des espaces de Riemann. Gauthier-Villars, Paris (1928)
5. Cartan, E.: Groupes simples clos et ouverts et géométrie riemannienne. J. Math. Pures Appl. 9e série(tome 8), 1–34 (1929)
6. Damon, J., Marron, J.S.: Backwards principal component analysis and principal nested relations. J. Math. Imaging Vision **50**(1–2), 107–114 (2013)
7. Darling, R.W.R.: Geometrically intrinsic nonlinear recursive filters II: foundations (1998). arXiv:math/9809029
8. Ding, C., Zhou, D., He, X., Zha, H.: R1-PCA: Rotational invariant L1-norm principal component analysis for robust subspace factorization. In: Proceedings of the 23rd International Conference on Machine Learning, ICML '06, pp. 281–288. ACM, New York (2006)
9. Dryden, I., Mardia, K.: Theoretical and distributional aspects of shape analysis. In: Probability Measures on Groups, X (Oberwolfach, 1990), pp. 95–116, Plenum, New York (1991)
10. Edelman, A., Arias, T., Smith, S.: The geometry of algorithms with orthogonality constraints. SIAM J. Matrix Anal. Appl. **20**(2), 303–353 (1998)
11. Emery, M., Mokobodzki, G.: Sur le barycentre d'une probabilité dans une variété. In: Séminaire de Probabilités XXV, vol. 1485, pp. 220–233. Springer, Berlin (1991)
12. Feragen, A., Owen, M., Petersen, J., Wille, M.M.W., Thomsen, L.H., Dirksen, A., de Bruijne, M.: Tree-space statistics and approximations for large-scale analysis of anatomical trees. In: Gee, J.C., Joshi, S., Pohl, K.M., Wells, W.M., Zöllei, L. (eds.) Information Processing in Medical Imaging, pp. 74–85. Springer, Berlin (2013)
13. Fletcher, P., Lu, C., Pizer, S., Joshi, S.: Principal geodesic analysis for the study of nonlinear statistics of shape. IEEE Trans. Med. Imaging **23**(8), 995–1005 (2004)
14. Fréchet, M.: Valeurs moyennes attachées a un triangle aléatoire. La Revue Scientifique, Fascicule **10**, 475–482 (1943)
15. Fréchet, M.: Les éléments aléatoires de nature quelconque dans un espace distancié. Ann. Inst. Henri Poincare **10**, 215–310 (1948)
16. Gramkow, C.: On averaging rotations. Int. J. Comput. Vis. **42**(1–2), 7–16 (2001)
17. Grenander, U., Miller, M., Srivastava, A.: Hilbert-Schmidt lower bounds for estimators on matrix Lie groups for ATR. IEEE Trans. Pattern Anal. Mach. Intell. **20**(8), 790–802 (1998)
18. Grove, K., Karcher, H.: How to conjugate C1-close group actions. Math. Z. **132**(1), 11–20 (1973)
19. Huckemann, S., Ziezold, H.: Principal component analysis for Riemannian manifolds, with an application to triangular shape spaces. Adv. Appl. Probab. **38**(2), 299–319 (2006)
20. Huckemann, S., Hotz, T., Munk, A.: Intrinsic shape analysis: Geodesic principal component analysis for Riemannian manifolds modulo Lie group actions. Stat. Sin. **20**, 1–100 (2010)
21. Jung, S., Dryden, I.L., Marron, J.S.: Analysis of principal nested spheres. Biometrika **99**(3), 551–568 (2012)
22. Karcher, H.: Riemannian center of mass and mollifier smoothing. Commun. Pure Appl. Math. **30**(5), 509–541 (1977)
23. Karcher, H.: Riemannian center of mass and so called Karcher mean (2014). arXiv:1407.2087

24. Kendall, D.: A survey of the statistical theory of shape (with discussion). Stat. Sci. **4**, 87–120 (1989)
25. Kendall, W.: Probability, convexity, and harmonic maps with small image I: uniqueness and fine existence. Proc. Lond. Math. Soc. **61**(2), 371–406 (1990)
26. Kwak., N.: Principal component analysis based on L1-norm maximization. IEEE Trans. Pattern Anal. Mach. Intell. **30**(9), 1672–1680 (2008)
27. Le, H.: Locating Fréchet means with application to shape spaces. Adv. Appl. Probab. **33**, 324–338 (2001)
28. Le, H.: Estimation of Riemannian barycenters. LMS J. Comput. Math. **7**, 193–200 (2004)
29. Le, H., Kendall, D.: The Riemannian structure of Euclidean shape space: a novel environment for statistics. Ann. Stat. **21**, 1225–1271 (1993)
30. Leporé, N., Brun, C., Chou, Y.-Y., Lee, A., Barysheva, M., Pennec, X., McMahon, K., Meredith, M., De Zubicaray, G., Wright, M., Toga, A.W., Thompson, P.: Best individual template selection from deformation tensor minimization. In: Proceedings of the 2008 IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI'08), Paris, pp. 460–463 (2008)
31. Lorenzi, M., Pennec, X.: Geodesics, Parallel transport and one-parameter subgroups for diffeomorphic image registration. Int. J. Comput. Vis. **105**(2), 111–127 (2013)
32. Marron, J.S., Alonso, A.M.: Overview of object oriented data analysis. Biom. J. **56**(5), 732–753 (2014)
33. Mcleod, K., Sermesant, M., Beerbaum, P., Pennec, X.: Spatio-temporal tensor decomposition of a polyaffine motion model for a better analysis of pathological left ventricular dynamics. IEEE Trans. Med. Imaging **34**(7), 1562–1675 (2015)
34. Moakher, M.: Means and averaging in the group of rotations. SIAM J. Matrix Anal. Appl. **24**(1), 1–16 (2002)
35. Oller, J., Corcuera, J.: Intrinsic analysis of statistical estimation. Ann. Stat. **23**(5), 1562–1581 (1995)
36. Pennec, X.: Probabilities and statistics on Riemannian manifolds: basic tools for geometric measurements. In: Cetin, A.E., Akarun, L., Ertuzun, A., Gurcan, M.N., Yardimci, Y. (eds.) Proceedings of Nonlinear Signal and Image Processing (NSIP'99), vol. 1, pp. 194–198. IEEE-EURASIP, Antalya (1999)
37. Pennec, X.: Intrinsic Statistics on Riemannian Manifolds: Basic Tools for Geometric Measurements. J. Math. Imaging Vis. **25**(1), 127–154 (2006). A preliminary appeared as INRIA RR-5093, 2004
38. Pennec, X.: Sample-limited L p barycentric subspace analysis on constant curvature spaces. In: Geometric Sciences of Information (GSI 2017), vol. 10589, pp. 20–28. Springer, Berlin (2017)
39. Pennec, X: Barycentric subspace analysis on manifolds. Ann. Stat. **46**(6A), 2711–2746 (2018)
40. Pennec, X., Arsigny, V.: Exponential Barycenters of the Canonical Cartan Connection and Invariant Means on Lie Groups. In: Barbaresco, F., Mishra, A., Nielsen, F. (eds.) Matrix Information Geometry, pp. 123–168. Springer, Berlin (2012)
41. Pennec, X., Lorenzi, M.: Beyond Riemannian Geometry The affine connection setting for transformation groups chapter 5. In: Pennec, S.S.X., Fletcher, T. (eds.) Riemannian Geometric Statistics in Medical Image Analysis. Elsevier, Amsterdam (2019)
42. Pennec, X., Guttmann, C.R., Thirion, J.-P.: Feature-based registration of medical images: estimation and validation of the pose accuracy. In: Proceedings of First International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'98). LNCS, vol. 1496, pp. 1107–1114. Springer, Berlin (1998)
43. Pennec, X., Fillard, P., Ayache, N.: A Riemannian framework for tensor computing. Int. J. Comput. Vis. **66**(1), 41–66 (2006). A preliminary version appeared as INRIA Research Report 5255, 2004
44. Rohé, M.-M., Sermesant, M., Pennec, X.: Barycentric subspace analysis: a new symmetric group-wise paradigm for cardiac motion tracking. In: 19th International Conference on Medical Image Computing and Computer Assisted Intervention, MICCAI 2016. Lecture Notes in Computer Science, vol. 9902, pp. 300–307, Athens (2016)

45. Rohé, M.-M., Sermesant, M., Pennec, X.: Low-dimensional representation of cardiac motion using barycentric subspaces: a new group-wise paradigm for estimation, analysis, and reconstruction. Med. Image Anal. **45**, 1–12 (2018)
46. Small, C.: The Statistical Theory of Shapes. Springer Series in Statistics. Springer, Berlin (1996)
47. Sommer, S.: Horizontal Dimensionality Reduction and Iterated Frame Bundle Development. In: Nielsen, F., Barbaresco, F. (eds.) Geometric Science of Information. Lecture Notes in Computer Science, vol. 8085, pp. 76–83. Springer, Berlin (2013)
48. Sommer, S., Lauze, F., Nielsen, M.: Optimization over geodesics for exact principal geodesic analysis. Adv. Comput. Math. **40**(2), 283–313 (2013)
49. Tobon-Gomez, C., De Craene, M., Mcleod, K., Tautz, L., Shi, W., Hennemuth, A., Prakosa, A., Wang, H., Carr-White, G., Kapetanakis, S., Lutz, A., Rasche, V., Schaeffter, T., Butakoff, C., Friman, O., Mansi, T., Sermesant, M., Zhuang, X., Ourselin, S., Peitgen, H.O., Pennec, X., Razavi, R., Rueckert, D., Frangi, A.F., Rhode, K.: Benchmarking framework for myocardial tracking and deformation algorithms: an open access database. Med. Image Anal. **17**(6), 632–648 (2013)
50. Weyenberg, G.S.: Statistics in the Billera–Holmes–Vogtmann treespace. PhD thesis, University of Kentucky, 2015
51. Yang, L.: Riemannian median and its estimation. LMS J. Comput. Math. **13**, 461–479 (2010)
52. Yang, L.: Medians of probability measures in Riemannian manifolds and applications to radar target detection. PhD thesis, Poitier University, 2011
53. Zhai, H.: Principal component analysis in phylogenetic tree space. PhD thesis, University of North Carolina at Chapel Hill, 2016.
54. Ziezold, H.: On Expected figures and a strong law of large numbers for random elements in quasi-metric spaces. In: Kožešnik, J. (ed.) Transactions of the Seventh Prague Conference on Information Theory, Statistical Decision Functions, Random Processes and of the 1974 European Meeting of Statisticians, vol. 7A, pp. 591–602. Springer, Netherlands (1977)

Chapter 12
Deep Variational Inference

Iddo Drori

Contents

Abstract This chapter begins with a review of variational inference (VI) as a fast approximation alternative to Markov Chain Monte Carlo (MCMC) methods, solving an optimization problem for approximating the posterior. VI is scaled to stochastic variational inference and generalized to black-box variational inference (BBVI). Amortized VI leads to the variational auto-encoder (VAE) framework which is introduced using deep neural networks and graphical models and used for learning representations and generative modeling. Finally, we explore generative flows, the latent space manifold, and Riemannian geometry of generative models.

12.1 Variational Inference

We begin with observed data x, continuous or discrete, and suppose that the process generating the data involved hidden latent variables z. For example, x may be an image of a face and z a hidden vector describing latent variables such as pose, illumination, gender, or emotion. A probabilistic model is a joint density $p(z, x)$

I. Drori (✉)
Columbia University, New York, NY, USA
e-mail: idrori@cs.columbia.edu

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_12

of the hidden variables z and the observed variables x. Our goal is to estimate the posterior $p(z|x)$ to explain the observed variables x by the hidden variables z. For example, answering the question what are the hidden latent variables z for a given image x. Inference about the hidden variables is given by the posterior conditional distribution $p(z|x)$ of hidden variables given observations. By definition:

$$p(z, x) = p(z|x)p(x) = p(x|z)p(z) = p(x, z), \tag{12.1}$$

where $p(z, x)$ is the joint density, $p(z|x)$ the posterior, $p(x)$ the evidence or marginal density, $p(z)$ the prior density, and $p(x|z)$ the likelihood function. We may extend $p(x|z)p(z)$ to multiple layers by:

$$p(x|z_1)p(z_1|z_2)\cdots p(z_{l-1}|z_l)p(z_l), \tag{12.2}$$

by using deep generative models. For now we will focus on a single layer $p(x|z)p(z)$. Rearranging terms we get Bayes rule:

$$p(z|x) = \frac{p(x|z)p(z)}{p(x)}. \tag{12.3}$$

For most models the denominator $p(x)$ is a high dimensional intractable integral which requires integrating over an exponential number of terms for z:

$$p(x) = \int p(x|z)p(z)dz. \tag{12.4}$$

Therefore, instead of computing $p(z|x)$ the key insight of variational inference (VI) [6, 7, 29, 30, 39, 58, 61] is to approximate the posterior by a variational distribution $q_\phi(z)$ from a family of distributions Q, defined by variational parameters ϕ such that $q_\phi(z) \in Q$. In summary, we choose a parameterized family of distributions Q and find the distribution $q_{\phi^\star}(z) \in Q$ which is closest to $p(z|x)$. Once found, we use this approximation $q_{\phi^\star}(z)$ instead of the true posterior $p(z|x)$ as illustrated in Fig. 12.1.

Compared with this formulation, methods such as mean-field variational inference (MFVI) [23, 39] and Markov Chain Monte Carlo (MCMC) sampling have several shortcomings. The mean-field method [40] assumes that the variational distribution may be factorized, however such a factorization of variables is often inaccurate. Stochastic variational inference scales MFVI to large datasets [27]. MCMC sampling methods [10], such as the Metropolis Hastings algorithm generate a sequence of samples with a distribution which converges to a probability distribution, such as the posterior distribution. The Metropolis Hastings algorithm decides which proposed values to accept or reject, even though the functional form of the distribution is unknown, by using the ratio between consecutive samples. The limitations of MCMC methods are that they may not be scalable to very large datasets and require manually specifying a proposal distribution. Integrated nested

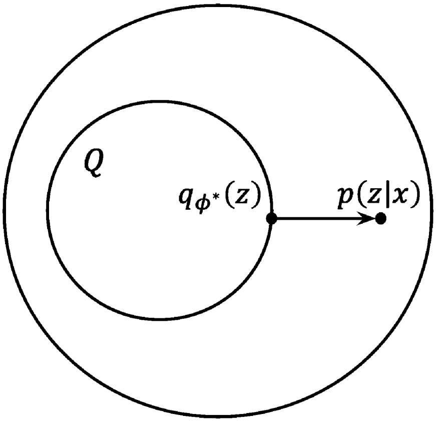


Fig. 12.1 Variational inference: optimizing an approximation $q_{\phi^\star}(z) \in Q$ closest to the posterior $p(z|x)$

Laplace approximations (INLA) [33, 49] are faster and more accurate than MCMC, however are used for the class of latent Gaussian models.

A choice for Q is the exponential family of distributions and a choice for closeness of distributions is the relative entropy also known as the Kullback–Leibler (KL) divergence , defined by:

$$KL(q(x)||p(x)) = \int q(x) \log \frac{q(x)}{p(x)} dx, \tag{12.5}$$

and other divergences may be used. The KL divergence is the special case of the α-divergence [35] with $\alpha = 1$, and the special case of the Bregman divergence generated by the entropy function. Making the choice of an exponential family and KL divergence, we minimize the KL divergence between $q(z)$ and $p(z|x)$:

$$\underset{\phi}{\text{minimize}}\; KL(q_\phi(z)||p(z|x)), \tag{12.6}$$

to find the approximate posterior:

$$q_{\phi^\star}(z) = \underset{q_\phi(z)}{\arg\min}\; KL(q_\phi(z)||p(z|x)), \tag{12.7}$$

as illustrated in Fig. 12.1. Notice that the KL divergence is non-negative $KL(q||p) \geq 0$ and is not symmetric $KL(q||p) \neq KL(p||q)$. Using the definition of the KL divergence in Eq. (12.5) for the variational distribution and posterior we get:

$$KL(q(z)||p(z|x)) = \int q(z) \log \frac{q(z)}{p(z|x)} dz. \tag{12.8}$$

Unfortunately, the denominator contains the posterior $p(z|x)$, which is the term that we would like to approximate. So how can we get close to the posterior without knowing the posterior? By using Bayes rule, replacing the posterior in Eq. (12.8) using Eq. (12.1) we get:

$$\int q(z)\log\frac{q(z)}{p(z|x)}dz = \int q(z)\log\frac{q(z)p(x)}{p(z,x)}dz \tag{12.9}$$

Separating the $\log p(x)$ term and replacing the log of the ratio with a difference yields:

$$\int q(z)\log\frac{q(z)p(x)}{p(z,x)}dz = \log p(x) - \int q(z)\log\frac{p(z,x)}{q(z)}dz. \tag{12.10}$$

In summary minimizing the KL divergence between $q(z)$ and $p(z|x)$ is equivalent to minimizing the difference:

$$\log p(x) - \int q(z)\log\frac{p(z,x)}{q(z)}dz \geq 0, \tag{12.11}$$

which is non-negative since the KL divergence is non-negative. Rearranging terms we get:

$$\log p(x) \geq \int q(z)\log\frac{p(z,x)}{q(z)}dz := \mathcal{L}. \tag{12.12}$$

The term on the right, denoted by $\mathcal{L}$, is known as the evidence lower bound (ELBO). Therefore, minimizing the KL divergence is equivalent to maximizing the ELBO. We have turned the problem of approximating the posterior $p(z|x)$ into an optimization problem of maximizing the ELBO. The ELBO is a non-convex function which consists of two terms:

$$\mathcal{L} = \mathbb{E}_{q_\phi(z)}[\log p(x,z)] - \mathbb{E}_{q_\phi(z)}[\log q_\phi(z)]. \tag{12.13}$$

The term on the left is the expected log likelihood, and the term on the right is the entropy. Therefore, when optimizing the ELBO, there is a trade-off between these two terms. The first term places mass on the MAP estimate; whereas the second term encourages diffusion, or spreading the variational distribution. In variational inference (VI) we maximize the ELBO in Eq. (12.13) to find $q_{\phi^\star}(z) \in Q$ closest to the posterior $p(z|x)$.

12.1.1 Score Gradient

Now that our objective is to maximize the ELBO, we turn to practical optimization methods. The ELBO is not convex so we can hope to find a local minimum. We would like to scale up to large data x with many hidden variables z. A practical optimization method which scales to large data is stochastic gradient descent [8, 47]. Gradient descent optimization is a first order method which requires computing the gradient. Therefore our problem is of computing the gradient of the ELBO:

$$\nabla_\phi \mathcal{L} = \nabla \mathbb{E}_{q_\phi(z)}[\log p(x, z) - \log q_\phi(z)]. \tag{12.14}$$

We would like to compute the gradients of the expectations $\nabla_\phi \mathbb{E}_{q_\phi(z)}[f_\phi(z)]$ for a cost function $f_\phi(z) = \log p(x, z) - \log q_\phi(z)$. Expanding the gradient results in:

$$\nabla_\phi \mathbb{E}_{q_\phi(z)}[f_\phi(z)] = \nabla_\phi \int q_\phi(z) f_\phi(z) dz, \tag{12.15}$$

and using the chain rule yields:

$$\nabla_\phi \int q_\phi(z) f_\phi(z) dz = \int (\nabla_\phi q_\phi(z)) f_\phi(z) + q_\phi(z)(\nabla_\phi f_\phi(z)) dz. \tag{12.16}$$

We cannot compute the expectation with respect to $q_\phi(z)$, which involves the unknown term $\nabla_\phi q_\phi(z)$, and therefore we will take Monte Carlo estimates of the gradient by sampling from q and use the score function estimator as described next.

Score Function
The score function is the derivative of the log-likelihood function:

$$\nabla_\phi \log q_\phi(z) = \frac{\nabla_\phi q_\phi(z)}{q_\phi(z)}. \tag{12.17}$$

Score Function Estimator
Using Eq. (12.15) and multiplying by the identity we get:

$$\nabla_\phi \int q_\phi(z) f_\phi(z) dz = \int \frac{q_\phi(z)}{q_\phi(z)} \nabla_\phi q_\phi(z) f_\phi(z) dz, \tag{12.18}$$

and plugging in Eq. (12.17) we derive:

$$\int \frac{q_\phi(z)}{q_\phi(z)} \nabla_\phi q_\phi(z) f_\phi(z) dz = \int q_\phi(z) \nabla_\phi \log q_\phi(z) f_\phi(z) dz, \tag{12.19}$$

which equals:

$$\int q_\phi(z) \nabla_\phi \log q_\phi(z) f_\phi(z) dz = \mathbb{E}_{q_\phi(z)}[f_\phi(z) \nabla_\phi \log q_\phi(z)]. \tag{12.20}$$

In summary, by using the score function, we have passed the gradient through the expectation:

$$\nabla_\phi \mathbb{E}_{q_\phi(z)}[f_\phi(z)] = \mathbb{E}_{q_\phi(z)}[f_\phi(z)\nabla_\phi \log q_\phi(z)]. \tag{12.21}$$

Score Gradient

The gradient of the ELBO with respect to the variational distribution $\nabla_\phi \mathcal{L}$ is computed using Eq. (12.21) as:

$$\nabla_\phi \mathcal{L} = E_{q_\phi(z)}[(\log p(x, z) - \log q_\phi(z))\nabla_\phi \log q_\phi(z)]. \tag{12.22}$$

Now that the gradient is inside the expectation we can evaluate using Monte Carlo sampling. For stochastic gradient descent we average over samples z_i from $q_\phi(z)$ to get:

$$\nabla_\phi \mathcal{L} = \frac{1}{k}\sum_{i=1}^{k}[(\log p(x, z_i) - \log q_\phi(z_i)\nabla_\phi \log q_\phi(z_i)], \tag{12.23}$$

where $\nabla_\phi \log q_\phi(z_i)$ is the score function. The score gradient works for both discrete and continuous models and a large family of variational distributions and is therefore widely applicable [42]. The problem with the score function gradient is that the noisy gradients have a large variance. For example, if we use Monte Carlo sampling for estimating a mean and there is high variance we would require many samples for a good estimate of the mean.

12.1.2 Reparametrization Gradient

Distributions can be represented by transformations of other distributions. We therefore express the variational distribution $z \sim q_\phi(z) = \mathcal{N}(\mu, \sigma)$ by a transformation:

$$z = g(\epsilon, \phi), \tag{12.24}$$

where $\epsilon \sim s(\epsilon)$ for a fixed distribution $s(\epsilon)$ independent of ϕ, and get an equivalent way of describing the same distribution:

$$z \sim q_\phi(z). \tag{12.25}$$

For example, instead of $z \sim q_\phi(z) = \mathcal{N}(\mu, \sigma)$ we use:

$$z = \mu + \sigma \odot \epsilon, \tag{12.26}$$

where $\epsilon \sim \mathcal{N}(0, 1)$ is a distribution w.r.t. which the others are parametrized, to get the same distribution:

$$z \sim \mathcal{N}(\mu, \sigma). \tag{12.27}$$

Although these are two different ways of describing the same distribution, the advantages of this transformation are that we can (1) express the gradient of the expectation, (2) achieve a lower variance than the score function estimator, and (3) differentiate through the latent variable z to optimize by backpropagation.

We reparameterize $\nabla_\phi \mathbb{E}_{q_\phi(z)}[f_\phi(z)]$, and by a change of variables Eq. (12.15) becomes:

$$\nabla_\phi \mathbb{E}_{q_\phi(z)}[f_\phi(z)] = \nabla_\phi \int s(\epsilon)\frac{d\epsilon}{dz} f(g(\epsilon,\phi))g'(\epsilon,\phi)d\epsilon, \tag{12.28}$$

and:

$$\nabla_\phi \int s(\epsilon)\frac{d\epsilon}{dz} f(g(\epsilon,\phi))g'(\epsilon,\phi)d\epsilon = \nabla_\phi \mathbb{E}_{s(\epsilon)}[f(g(\phi,\epsilon)] = \mathbb{E}_{s(\epsilon)}[\nabla_\phi f(g(\phi,\epsilon)], \tag{12.29}$$

passing the gradient through the expectation:

$$\nabla_\phi \mathbb{E}_{q_\phi(z)}[f_\phi(z)] = \mathbb{E}_{s(\epsilon)}[\nabla_\phi f(g(\phi,\epsilon))]. \tag{12.30}$$

Since the gradient is inside the expectation, we can use Monte Carlo sampling to estimate $\mathbb{E}_{s(\epsilon)}[\nabla_\phi f(g(\phi,\epsilon))]$. The reparameterization method given by Eq. (12.30) has a lower variance compared with the score function estimator given in Eq. (12.21).

In the case of the ELBO $\mathcal{L}$, the reparameterized gradient [32, 46] is given by:

$$\nabla_\phi \mathcal{L} = \mathbb{E}_{s(\epsilon)}[\nabla_\phi[\log p(x,z) - \log q_\phi(z)]\nabla_\phi g(\epsilon,\phi)], \tag{12.31}$$

and re-writing the expectation:

$$\nabla_\phi \mathcal{L} = \frac{1}{k}\sum_{i=1}^{k}(\nabla_\phi[\log p(x, g(\epsilon_i,\phi)) - \log q_\phi(g(\epsilon_i,\phi))], \tag{12.32}$$

provided the entropy term has an analytic derivation and $\log p(x,z)$ and $\log q(z)$ are differentiable with respect to z. Similarly, the reparametrization gradient in Eq. (12.31) has a lower variance than the score gradient in Eq. (12.22). In addition, we can use auto-differentiation for computing the gradient and reuse different transformations [34]. The gradient variance is further reduced by changing the computation graph in automatic differentiation [48]. However, a limitation of the reparameterization gradient is that it requires a differentiable model, works only for continuous models [22], and is computationally more expensive.

12.2 Variational Autoencoder

Instead of optimizing a separate parameter for each example, amortized variational inference (AVI) approximates the posterior across all examples together [32, 46]. Meta amortized variational inference (meta-AVI) goes a step further and approximates the posterior across models [13]. Next, we give a formulation of autoencoders, which motivates the AVI algorithm of variational autoencoders (VAE).

12.2.1 Autoencoder

An autoencoder is a neural network which performs non-linear principle component analysis (PCA) [20, 26]. Non-linear PCA extracts useful features from unlabeled data by minimizing:

$$\underset{W}{\text{minimize}} \sum_{i=1}^{m} \|x_i - W^T g(W x_i)\|_2^2 \tag{12.33}$$

where for single layer networks, W and W^T are matrices which are the networks parameters and g is a pointwise non-linear function. An autoencoder is composed of two neural networks. The first maps an input x by matrix multiplication and a non-linearity to a low dimensional variable z, and the second reconstructs the input as $\tilde{x}$ using W^T (Fig. 12.2). When g is the identity this is equivalent to PCA.

The goal of variational inference is to find a distribution q which approximates the posterior $p(z|x)$, and a distribution $p(x)$ which represents the data well. Maximizing the ELBO minimizes the KL divergence between q and the posterior and approximately maximizes the marginal likelihood. Motivated by autoencoders, we represent q and p using two neural networks and optimize for the variational parameters ϕ and the model parameters θ simultaneously. An encoder network

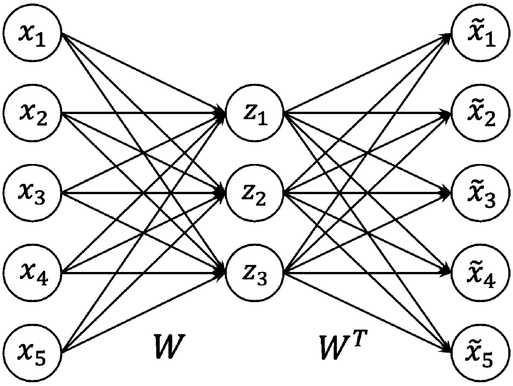


Fig. 12.2 Autoencoder: input x is passed through a low dimensional bottleneck z and reconstructed to form $\tilde{x}$ minimizing a loss between the input and output. The parameters W of the encoder and decoder are optimized end-to-end

represents q and a decoder network represents p. These neural networks are non-linear functions F which are a composition of functions $F(x) = f(f(\cdots f(x)))$, where each individual function f has a linear and non-linear component, and the function F is optimized given a large datasets by stochastic gradient descent (SGD).

12.2.2 Variational Autoencoder

The ELBO as defined in Eq. (12.12) can be re-written as:

$$\mathcal{L} = \int q(z) \log p(x|z) dz - \int q(z) \log \frac{p(z)}{q(z)} dz, \tag{12.34}$$

which is the lower bound consisting of two terms:

$$\mathcal{L} = \mathbb{E}_{q(z)}[\log p(x|z)] - KL(q(z)||p(z)). \tag{12.35}$$

Reconstruction Error The first term $\log p(x|z)$, on the left, is the log-likelihood of the observed data x given the sampled latent variable z. This term measures how well the samples from $q(z)$ explain the data x. The goal of this term is to reconstruct x from z and therefore is called the reconstruction error, representing a decoder which is implemented by a deep neural network.

Regularization The second term, on the right, consists of sampling $z \sim q(z|x)$, representing an encoder which is also implemented by a deep neural network. This term ensures that the explanation of the data does not deviate from the prior beliefs $p(z)$ and is called the regularization term, defined by the KL divergence between q and the prior $p(z)$.

The objective function in Eq. (12.35) is analogous to the formulation of autoencoders, and therefore gives rise to the variational autoencoder (VAE) [32]. The VAE is a deep learning algorithm, rather than a model, which is used for learning latent representations. The VAE algorithm is considered amortized variational inference (AVI) since it shares the variational parameters across all data examples. The learnt representations can be used for applications such as synthesizing examples or interpolation between samples, of different modalities such as images [25], video, audio, geometry, and text [9].

The variational autoencoder algorithm is defined by two back-to-back neural networks as illustrated in Fig. 12.3. The first is an encoder neural network which infers a hidden variable z from an observations x (Fig. 12.4). The second is a decoder neural network which reconstructs an observation $\tilde{x}$ from a hidden variable z. The encoder q_ϕ and decoder p_θ are trained end-to-end, optimizing for both the encoder parameters ϕ and decoder parameters θ. Neural networks are optimized by backpropagation [50] which is a special case of differentiable programming [4, 60]. Differentiation in a backward pass using the chain rule, results in the partial derivative of the output with respect to all input variables, namely the gradient, in a single pass which is highly efficient.

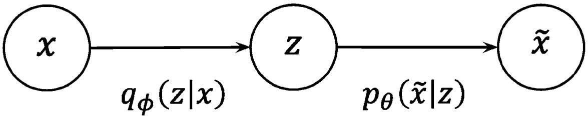
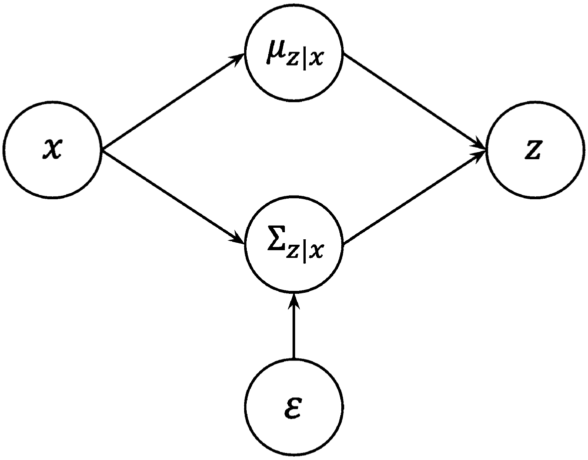


Fig. 12.3 Variational autoencoder (VAE): the input x is passed through a low dimensional bottleneck z and reconstructed to form $\tilde{x}$ minimizing a loss between the input and output. The variational parameters ϕ of the encoder q_ϕ and the model parameters θ of the decoder p_θ neural networks are optimized simultaneously end-to-end by backpropagation

Fig. 12.4 Variational Encoder: rather than sampling directly $z \sim \mathcal{N}(\mu, \sigma)$ in the latent space, reparameterization allows for backpropagation through the latent variable $z = \mu + \sigma \odot \epsilon$, which is a sum of the mean μ and covariance. The covariance σ is pointwise multiplied by noise $\epsilon \sim \mathcal{N}(0, \mathcal{I})$ sampled from a normal distribution



If we assume $q(z|x)$ and $p(x|z)$ are normally distributed then q is represented by:

$$q(z|x) = \mathcal{N}(\mu(x), \sigma(x) \odot \mathcal{I}), \tag{12.36}$$

for deterministic functions $\mu(x)$ and $\sigma(x)$, and p is represented by:

$$p(x|z) = \mathcal{N}(\mu(z), \sigma(z) \odot \mathcal{I}), \tag{12.37}$$

and

$$p(z) = \mathcal{N}(0, \mathcal{I}). \tag{12.38}$$

The variational predictive natural gradient [54] rescales the gradient to capture the curvature of variational inference. The correlated VAE [55] extends the VAE to learn pairwise variational distribution estimations which capture the correlation between data points.

In practice, very good synthesis results for different modalities are achieved using a vector quantized variational autoencoder (VQ-VAE) [57] which learns a discrete latent representation. Using an autoregressive decoder or prior with VQ-VAE [15, 44] generates photorealistic high resolution images [43].

12.3 Generative Flows

This section describes transformations of simple posterior distribution approximations to complex distributions by normalizing flows [45]. We would like to improve our variational approximation $q_\phi(z)$ to the posterior $p(z|x)$. An approach for achieving this goal is to transform a simple density, such as a Gaussian, to a complex density using a sequence of invertible transformations, also known as normalizing flows [17, 31, 45]. Instead of parameterizing a simple distribution directly, a change of variables allows us to define a complex distribution by warping $q(z)$ using an invertible function f. Given a random variable $z \sim q_\phi(z)$ the log density of $x = f(z)$ is:

$$\log p(x) = \log p(z) - \log\det\left|\frac{\partial f(z)}{\partial z}\right|. \tag{12.39}$$

A composition of multiple invertible functions results in a sequence of transformations, called normalizing flows. These transformations may be implemented by neural networks, performing end-to-end optimization of the network parameters. For example, for a planar flow family of transformations:

$$f(z) = z + uh(w^T z + b), \tag{12.40}$$

where h is a smooth differentiable non-linear function, and the log-det Jacobian is computed by:

$$\psi(z) = h'(w^T z + b)w, \tag{12.41}$$

and

$$\left|det\frac{\partial f}{\partial z}\right| = \left|I + u^T\psi(z)\right|. \tag{12.42}$$

If z is a continuous random variable $z(t)$ depending on time t with distribution $p(z(t))$ then for the differential equation $\frac{dz}{dt} = f(z(t), t)$ the change in log probability is:

$$\frac{\partial \log p(z(t))}{\partial t} = -\mathrm{tr}\left(\frac{\partial f}{\partial z(t)}\right), \tag{12.43}$$

and the change in log density is:

$$\log p(z(t_1)) = \log p(z(t_0)) - \int_{t_0}^{t_1} \operatorname{tr}\left(\tfrac{\partial f}{\partial z(t)}\right). \tag{12.44}$$

also known as continuous normal flows [11, 24].

For the planar flow family of transformations:

$$\frac{dz(t)}{dt} = uh(w^T z(y) + b), \tag{12.45}$$

and

$$\frac{\log p(z(t))}{\partial t} = -u^T \frac{\partial h}{\partial z(t)}, \tag{12.46}$$

such that given $p(z(0))$, $p(z(t))$ is sampled and the density evaluated by solving an ODE [11]. Finally, invertible ResNets [3] use continuous normal flows to define an invertible model for both discriminative and generative applications.

12.4 Geometric Variational Inference

This section generalizes variational inference and normalizing flows from Euclidean to Riemannian spaces [37], describing families of distributions which are compatible with a Riemannian geometry and metric [1, 14, 28, 51]. Finally, we consider the geometry of the latent space in variational autoencoders [12, 52, 59].

We briefly define a Riemannian manifold and metric, geodesic, tangent space, exponential and logarithm maps [18, 19, 38, 53, 56]. A manifold of dimension d has at each $p_0 \in \mathcal{M}$ a tangent space $T_{p_0}\mathcal{M}$ of dimension d consisting of vectors θ corresponding to derivatives of smooth paths $p(t) \in \mathcal{M}$, $t \in [0, 1]$, with $p(0) = p_0$. A Riemannian manifold has a metric on the tangent space. If for tangent vectors θ we adopt a specific coordinate representation θ_i, this quadratic form can be written as $\sum_{ij} g_{ij}(p)\theta_i\theta_j$. Between any two points p_0 and p_1 in the manifold, there is at least one shortest path, having arc length $\ell(p_0, p_1)$. Such a geodesic has an initial position, p_0, an initial direction, $\frac{\theta}{\|\theta\|_2}$ and an initial speed $\|\theta\|_2$. The procedure of fixing a vector in $\theta \in T_p\mathcal{M}$ as an initial velocity for a constant speed geodesic establishes an association between $T_{p_0}\mathcal{M}$ and a neighborhood of $p \in \mathcal{M}$. This association is one-to-one over a ball of sufficiently small size. The association is formally defined by the exponential map $p_1 = \exp_{p_0}(\theta)$. Within an appropriate neighborhood p_0, the inverse mapping is called the logarithm map and is defined by $\theta = \log_{p_0}(p_1)$ as illustrated in Fig. 12.5.

Normalizing flows have been extended from Euclidean space to Riemannian space [37]. A simple density on a manifold $\mathcal{M}$ is mapped to the tangent space $T_p\mathcal{M}$. Normalizing flow transformations are then applied to the mapped density in the tangent space, and the resulting density is mapped back to the manifold.

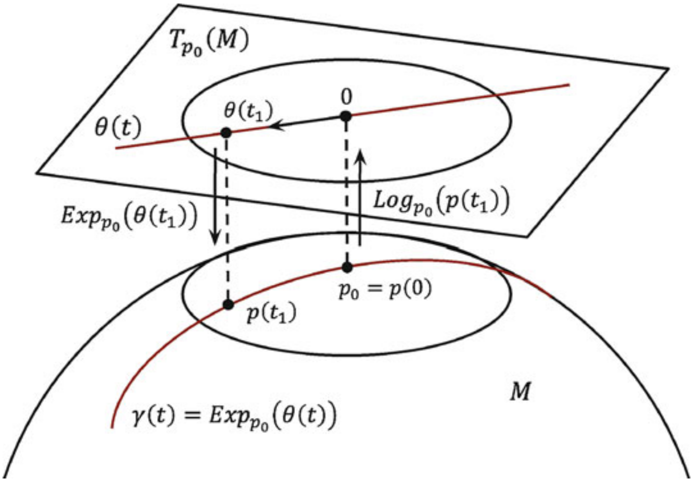


Fig. 12.5 Manifold and tangent plane: exponential and logarithm maps between the tangent plane and the manifold. A line in the tangent plane corresponds to a geodesic in the manifold

In variational inference, several transformation choices of a family of distributions are compatible with a Riemannian geometry [14, 21, 28, 51]. For example, transforming a distribution by the square-root to the positive orthant of the sphere, results in the square-root density of probability distributions. Probability distributions are then represented by square-root densities, and the geodesic distance is defined by the shortest arc length. Again, $p_1 = \exp_{p_0}(\theta)$ maps the tangent space to the sphere, and $\theta = \log_{p_0}(p_1)$ maps the sphere to the tangent space. Densities are represented in the tangent space, and in a similar fashion to normalizing flows, parallel transport is used to map one tangent space to another.

The decoder in the variational autoencoder is used for both reconstruction and synthesis, generating new samples x from latent variables z. In the past decade, generating a sequence of samples which smoothly morph or warp graphical objects required meticulously specifying correspondence between landmarks on the objects. In contrast, using the decoder as a generator and interpolating between hidden variables in latent space allows to perform this transformation without specifying correspondence. A question which arises is whether performing linear interpolation is suitable in the latent space? Interpolation may be performed by walking along a manifold rather than linear interpolation in the latent space. Specifically the latent space of a variational autoencoder can be considered as a Riemannian space [12]. Using a Riemannian metric rather than an Euclidean metric in the latent space provides better distance estimates [2, 36] which improve interpolation and synthesis results [52], as well as text generation results [59], increasing the mutual information between the latent and observed variables.

12.5 Summary

In this chapter we introduced variational inference (VI) and its extension to black-box variational inference (BBVI) which is used in practice for inference on large datasets. The variational autoencoders (VAE) algorithm consists of an encoder neural network for inference and decoder network for generation, trained end-to-end by backpropagation. We described a way in which the variational approximation of the posterior is improved using a series of invertible transformations, known as normalizing flows, in both discrete and continuous domains. Finally, we explored the latent space manifold and extended variational inference and normalizing flows to Riemannian manifolds. Scalable implementations of variational inference and variational autoencoders are available as part of Google's Tensorflow Probability library [16] and Uber's Pyro library [5] for Facebook's PyTorch deep learning platform [41].

References

1. Arvanitidis, G., Hansen, L.K., Hauberg, S.: Latent space oddity: on the curvature of deep generative models. In: International Conference on Learning Representations (2018)
2. Arvanitidis, G., Hauberg, S., Hennig, P., Schober, M.: Fast and robust shortest paths on manifolds learned from data. In: International Conference on Artificial Intelligence and Statistics (2019)
3. Behrmann, J., Duvenaud, D., Jacobsen, J.H.: Invertible residual networks. In: International Conference on Machine Learning (2019)
4. Bellman, R.E., Kagiwada, H., Kalaba, R.E.: Wengert's numerical method for partial derivatives, orbit determination and quasilinearization. Commun. ACM **8**(4), 231–232 (1965)
5. Bingham, E., Chen, J.P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P., Horsfall, P., Goodman, N.D.: Pyro: deep universal probabilistic programming. J. Mach. Learn. Res. **20**(1), 973–978 (2019)
6. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, Berlin (2006)
7. Blei, D.M., Kucukelbir, A., McAuliffe, J.D.: Variational inference: a review for statisticians. J. Am. Stat. Assoc. **112**(518), 859–877 (2017)
8. Bottou, L.: Large-scale machine learning with stochastic gradient descent. In: International Conference on Computational Statistics, pp. 177–186. Springer, Berlin (2010)
9. Bowman, S.R., Vilnis, L., Vinyals, O., Dai, A.M., Jozefowicz, R., Bengio, S.: Generating sentences from a continuous space. In: Conference on Computational Natural Language Learning (2016)
10. Brooks, S., Gelman, A., Jones, G., Meng, X.-L.: Handbook of Markov Chain Monte Carlo. CRC Press, Boca Raton (2011)
11. Chen, T.Q., Rubanova, Y., Bettencourt, J., Duvenaud, D.K.: Neural ordinary differential equations. In: Advances in Neural Information Processing Systems, pp. 6571–6583 (2018)
12. Chen, N., Ferroni, F., Klushyn, A., Paraschos, A., Bayer, J., van der Smagt, P.: Fast approximate geodesics for deep generative models (2019). arXiv preprint arXiv:1812.08284
13. Choi, K., Wu, M., Goodman, N., Ermon, S.: Meta-amortized variational inference and learning. In: International Conference on Learning Representations (2019)
14. Davidson, T.R., Falorsi, L., De Cao, N., Kipf, T., Tomczak, J.M.: Hyperspherical variational auto-encoders. In: Conference on Uncertainty in Artificial Intelligence (2018)

15. De Fauw, J., Dieleman, S., Simonyan, K.: Hierarchical autoregressive image models with auxiliary decoders (2019). arXiv preprint arXiv:1903.04933
16. Dillon, J.V., Langmore, I., Tran, D., Brevdo, E., Vasudevan, S., Moore, D., Patton, B., Alemi, A., Hoffman, M., Saurous, R.A.: Tensorflow distributions (2017). arXiv preprint arXiv:1711.10604
17. Dinh, L., Sohl-Dickstein, J., Bengio, S.: Density estimation using real NVP. In: International Conference on Learning Representations (2017)
18. Do Carmo, M.P.: Riemannian Geometry. Birkhäuser, Basel (1992)
19. Do Carmo, M.P.: Differential Geometry of Curves and Surfaces, 2nd edn. Courier Dover Publications, New York (2016)
20. Efron, B., Hastie, T.: Computer Age Statistical Inference. Cambridge University Press, Cambridge (2016)
21. Falorsi, L., de Haan, P., Davidson, T.R., Forré, P.: Reparameterizing distributions on Lie groups. In: Proceedings of Machine Learning Research, pp. 3244–3253 (2019)
22. Figurnov, M., Mohamed, S., Mnih, A.: Implicit reparameterization gradients. In: Advances in Neural Information Processing Systems, pp. 441–452 (2018)
23. Giordano, R., Broderick, T., Jordan, M.I.: Covariances, robustness and variational Bayes. J. Mach. Learn. Res. **19**(1), 1981–2029 (2018)
24. Grathwohl, W., Chen, R.T., Betterncourt, J., Sutskever, I., Duvenaud, D.: FFJORD: free-form continuous dynamics for scalable reversible generative models. In: International Conference on Learning Representations (2019)
25. Gregor, K., Danihelka, I., Graves, A., Rezende, D.J., Wierstra, D.: Draw: a recurrent neural network for image generation. In: International Conference on Machine Learning (2015)
26. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006)
27. Hoffman, M.D., Blei, D.M., Wang, C., Paisley, J.: Stochastic variational inference. J. Mach. Learn. Res. **14**(1), 1303–1347 (2013)
28. Holbrook, A.: Geometric Bayes. Ph.D. thesis, UC Irvine, 2018
29. Jordan, M.I., Ghahramani, Z., Jaakkola, T.S., Saul, L.K.: An introduction to variational methods for graphical models. Mach. Learn. **37**(2), 183–233 (1999)
30. Kim, Y., Wiseman, S., Rush, A.M.: A tutorial on deep latent variable models of natural language (2018). arXiv preprint arXiv:1812.06834
31. Kingma, D.P., Dhariwal, P.: Glow: generative flow with invertible 1×1 convolutions. In: Advances in Neural Information Processing Systems, pp. 10215–10224 (2018)
32. Kingma, D.P., Welling, M.: Auto-encoding variational Bayes. In: International Conference on Learning Representations (2014)
33. Krainski, E.T., Gómez-Rubio, V., Bakka, H., Lenzi, A., Castro-Camilo, D., Simpson, D., Lindgren, F., Rue, H.: Advanced spatial modeling with stochastic partial differential equations using R and INLA. Chapman and Hall/CRC (2018)
34. Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., Blei, D.M.: Automatic differentiation variational inference. J. Mach. Learn. Res. **18**(1), 430–474 (2017)
35. Li, Y., Turner, R.E.: Rényi divergence variational inference. In: Advances in Neural Information Processing Systems, pp. 1073–1081 (2016)
36. Mallasto, A., Hauberg, S., Feragen, A.: Probabilistic Riemannian submanifold learning with wrapped Gaussian process latent variable models. In: International Conference on Artificial Intelligence and Statistics (2019)
37. Normalizing flows on Riemannian manifolds. NeurIPS Bayesian Deep Learning Workshop (2016)
38. O'neill, B: Elementary Differential Geometry. Elsevier, Amsterdam (2006)
39. Opper, M., Saad, D.: Advanced Mean Field Methods: Theory and Practice. MIT Press, Cambridge (2001)
40. Parisi, G.: Statistical Field Theory. Addison-Wesley, Reading (1988)
41. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch (2017)

42. Ranganath, R., Gerrish, S., Blei, D.: Black box variational inference. In: Artificial Intelligence and Statistics, pp. 814–822 (2014)
43. Ravuri, S., Vinyals, O: Classification accuracy score for conditional generative models (2019). arXiv preprint arXiv:1905.10887
44. Razavi, A., van den Oord, A., Vinyals, O.: Generating diverse high-resolution images with VQ-VAE. In: International Conference on Learning Representations Workshop (2019)
45. Rezende, D.J., Mohamed, S.: Variational inference with normalizing flows. In: International Conference on Machine Learning (2015)
46. Rezende, D.J., Mohamed, S., Wierstra, D.: Stochastic backpropagation and approximate inference in deep generative models. In: International Conference on Machine Learning (2014)
47. Robbins, H., Monro, S.: A stochastic approximation method. Ann. Math. Stat. **22**, 400–407 (1951)
48. Roeder, G., Wu, Y., Duvenaud, D.K.: Sticking the landing: simple, lower-variance gradient estimators for variational inference. In: Advances in Neural Information Processing Systems, pp. 6925–6934 (2017)
49. Rue, H., Martino, S., Chopin, N.: Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. J. R. Stat. Soc. Ser. B Stat Methodol. **71**(2), 319–392 (2009)
50. Rumelhart, D.E., Hinton, G.E., Williams, R.J., et al.: Learning representations by back-propagating errors. Cogn. Model. **5**(3), 1 (1988)
51. Saha, A., Bharath, K., Kurtek, S.: A geometric variational approach to Bayesian inference. J. Am. Stat. Assoc., 1–26 (2019)
52. Shukla, A., Uppal, S., Bhagat, S., Anand, S., Turaga, P.: Geometry of deep generative models for disentangled representations (2019). arXiv preprint arXiv:1902.06964
53. Spivak, M.D.: A Comprehensive Introduction to Differential Geometry, 3rd edn. Publish or Perish (1999)
54. Tang, D., Ranganath, R.: The variational predictive natural gradient. In: International Conference on Machine Learning (2019)
55. Tang, D., Liang, D., Jebara, T., Ruozzi, N.: Correlated variational autoencoders. In: International Conference on Machine Learning (2019)
56. Ur Rahman, I., Drori, I., Stodden, V.C., Donoho, D.L., Schröder, P.: Multiscale representations for manifold-valued data. Multiscale Model. Simul. **4**(4), 1201–1232 (2005)
57. van den Oord, A., Vinyals, O., et al. Neural discrete representation learning. In: Advances in Neural Information Processing Systems, pp. 6306–6315 (2017)
58. Wainwright, M.J., Jordan, M.I., et al.: Graphical models, exponential families, and variational inference. Found. Trends® Mach. Learn. **1**(1–2), 1–305 (2008)
59. Wang, P.Z., Wang, W.Y.: Riemannian normalizing flow on variational Wasserstein autoencoder for text modeling (2019). arXiv preprint arXiv:1904.02399
60. Wengert, R.E.: A simple automatic derivative evaluation program. Commun. ACM **7**(8), 463–464 (1964)
61. Zhang, C., Butepage, J., Kjellstrom, H., Mandt, S.: Advances in variational inference. IEEE Trans. Pattern Anal. Mach. Intell. **41**(8), 2008–2026 (2019)

Part IV
Shapes Spaces and the Analysis of Geometric Data

Chapter 13
Shape Analysis of Functional Data

Xiaoyang Guo and Anuj Srivastava

Contents

Abstract Functional data is one of the most common types of data in our digital society. Such data includes scalar or vector time series, Euclidean curves, surfaces, or trajectories on nonlinear manifolds. Rather than applying past statistical techniques developed using standard Hilbert norm, we focus on analyzing functions according to their shapes. We summarize recent developments in the field of elastic shape analysis of functional data, with a perspective on statistical inferences. The key idea is to use metrics, with appropriate invariance properties, to register corresponding parts of functions and to use this registration in quantification of shape differences. Furthermore, one introduces square-root representations of functions to help simplify computations and facilitate efficient algorithms for large-scale data analysis. We will demonstrate these ideas using simple examples from common application domains.

X. Guo (✉) · A. Srivastava
Florida State University, Tallahassee, FL, USA
e-mail: xiaoyang.guo@stat.fsu.edu; anuj@stat.fsu.edu

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_13

13.1 Introduction

Statistical shape analysis aims to study the shapes of given geometric objects by statistical methods. It has wide applications in biology, computer vision, medical images, etc. For instance, in bioinformatics, it is important to associate the shapes of biological objects like RNAs and proteins with their functionality. Given a sample of shapes, one would like to use some statistical tools to summarize the information and make the inference. Some important techniques include an appropriate metric for quantifying shape difference, geodesics to study a natural deformation between two shapes, summary statistics (mean, covariance) of shapes, shape models to characterize shape populations and regression models using shapes as predictors or responses.

The object of interest varies depending on different applications. Examples include scalar functions, planar or 3D curves, surfaces, etc. As the result, shape analysis is naturally related to the subject of differential geometry. A typical framework for shape analysis starts with mathematical representations of objects and removes certain shape-preserving transformations as pre-processing. The remaining transformations that cannot be removed by pre-processing are dealt with equivalent classes defined by group actions. For example, since shape is invariant with respect to different rotations, an equivalent class defined by rotation of a specific shape is a set that contains all the possible rotations of that shape. And one treats this set as a specific observation in shape analysis.

Since shape analysis is an important branch of statistics, numerous methods have been developed in the literature. In the earlier works, shapes are represented by landmarks, a finite set of points [6, 9, 19]. One of the earliest formal mathematical frameworks is introduced in [9] where one removes rigid motions and global scaling from landmarks to reach final shape representations. Translation and scaling are first removed by centering and rescaling the landmarks, as a pre-processing. The space achieved is also called *preshape space*. The remaining transformation, rotation, is removed by forming orbits (equivalent classes) under the group action. A metric is then imposed on the space of orbits, which is also called quotient space, followed with rich methods in statistical analysis. More recently, there is a trend that shapes are more continuously represented other than using the finite, discrete points as landmarks.

One of the important challenges in shape analysis is the *registration* problem, which means finding the correspondence points between different objects. Historically, some shape analysis methods presume that objects have been already registered while others use different methods to register first and use this registration in subsequent own methods for analyzing shapes. However, both approaches are restrictive and questionable. A simultaneous registration and shape analysis called *elastic shape analysis* [20] has achieved significant recognition over the past few years. This is a class of Riemannian metrics based solutions that perform registration along with the process of shape analysis. The key idea is to equip the shape space with an elastic Riemannian metric that is invariant under the action of registration group. Such elastic metrics are often complicated if used directly. However, a square-root transformation can simplify them into the standard Euclidean metric and results in an efficient solution.

In this chapter, we summarize advances in elastic shape analysis. As mentioned earlier, there are different objects of shapes. For example, planar curves come from boundaries or silhouettes of objects in images [22]. 3D curves can be extracted from complex biomolecular structures like proteins or RNAs [13]. A special case of this problem is when the functional data is in $\mathbb{R}$, i.e., real numbers, which is also called *functional data analysis* (FDA) [17], where one analyzes shapes of scalar functions on a fixed interval [23]. The use of elastic Riemannian metrics and square-root transformations for curves were first introduced in [29, 30] although this treatment used complicated arithmetic and was restricted to planar curves. Later on, a family of elastic metrics are presented [16] that allowed for different levels of elasticity in shape comparisons. The works [21, 23] introduced a square-root representation that was applicable to curves in any Euclidean space. Subsequently, several other elastic metrics and square-root representations, each representing a different strength and limitation, have been discussed in the literature [2, 3, 11, 15, 31]. In this paper, we focus on the framework in [21, 23] and demonstrate that approach using a number of examples involving functional and curve data.

In addition to methods summarized in this chapter, we mention that elastic frameworks have also been developed for curves taking values on nonlinear domains also, including unit spheres [32], hyperbolic spaces [5], the space of symmetric positive definite (SPD) matrices [33], and some other manifolds. In the case where the data is a trajectory of functions or curves, for instance, the blood oxygenation level-dependent (BOLD) signal along tracts can be considered as trajectories of functions. A parallel transported square-root transformation in [24] can be used to effectively analyze and summarize the projection pathway. Additionally, elastic metrics and square-root representations have also been used to analyze shapes of surfaces in $\mathbb{R}^3$. These methods provide techniques for registration of points across objects, as well as comparisons of their shapes, in a unified metric-based framework. Applications include modeling parameterized surfaces of endometrial tissues that reconstructed from 2D MRI slices [12], shape changes of brain structures associated with Alzheimer [8], etc. For details, we refer to the textbook [7].

13.2 Registration Problem and Elastic Framework

We provide a comprehensive framework in a similar spirit of Kendall's [9] approach for comparing shapes of functional objects. The essence is to treat them as parameterized objects and use an *elastic metric* to register them. The invariant property with respect to reparameterization of the elastic metric enable us to conduct registration and shape analysis simultaneously.

13.2.1 The Use of the $\mathbb{L}^2$ Norm and Its Limitations

The problem of registration is fundamental for comparing shapes of functional data. To formulate the problem, we consider $\mathcal{F}$ as all the parameterized functional objects, whose elements are $f : D \to \mathbb{R}^n$ where D represents the domain of parameterization. As an example, for open planar curves, $n = 2$ and D is the unit interval [0, 1]. While for analyzing shapes of surfaces, D can be a unit sphere $\mathbb{S}^2$, unit disk, etc. The reparametrization of f is given by the composition with $\gamma : f \circ \gamma$, where $\gamma : D \to D$ is a boundary-preserving diffeomorphism that is an invertible function maps from domain D to itself that both the function and its inverse are smooth. We denote Γ as all the boundary-preserving diffeomorphisms of domain D. One can show that Γ forms a group with action as composition and the identity element is $\gamma_{id}(t) = t, t \in D$. Therefore, for any two $\gamma_1, \gamma_2 \in \Gamma$, $\gamma_1 \circ \gamma_2 \in D$ is also a (cumulative) reparameterization. Reparametrization does not change the shape of $f \in \mathbb{R}^n, n \geq 2$, i.e., f and $f \circ \gamma, \gamma \in \Gamma$ has the exact same shape. For scalar functions $f \in \mathbb{R}$, the reparametrization is usually called *time warping*, and we will discuss details later. An example of reparametrization of 2D curves can be found in Fig. 13.1. The top row shows the sine functions in the plane in different reparametrizations plotted in the bottom row. The middle column shows the original parametrization while left and right columns visualize different reparametrizations. For any $t \in D$ and any two functional object $f_1, f_2 \in \mathcal{F}$, $f_1(t)$ are registered to $f_2(t)$. Therefore, if we reparametrize f_2 to $f_2 \circ \gamma$, we can change the registration between f_1 and f_2, controlled by the diffeomorphism γ.

In order to quantify the problem, one needs an objective function to measure the quality of registration. A seemingly natural choice is using L^2 norm. Let $\|\cdot\|$

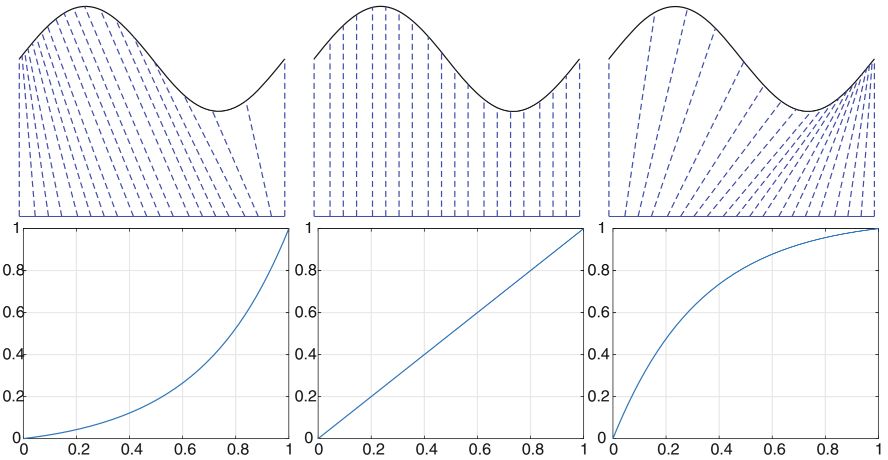


Fig. 13.1 An illustration of reparametrization of a 2D open curve. Top row is the curve in different parametrization. Bottom row shows the corresponding γ (diffeomorphism)

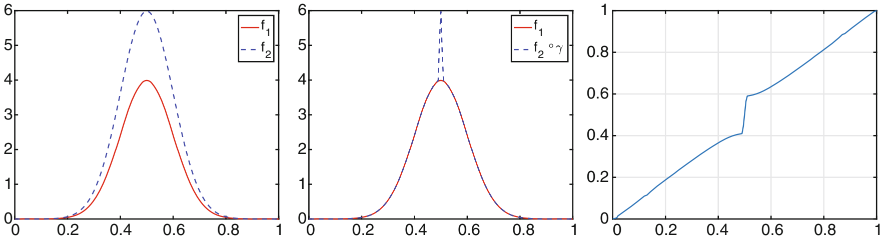


Fig. 13.2 Pinching effect when using $\mathbb{L}^2$ norm to align scalar functions. The rightmost is the time warping function

represents the L^2 norm, i.e., $\|f\| = \sqrt{\int_D |f(t)|^2 dt}$. Therefore, the corresponding objective function becomes $\inf_{\gamma\in\Gamma} \|f_1 - f_2 \circ \gamma\|$. There are several problems related to it. The main issue is that it leads to degeneracy solution. In other words, one can find a γ to reduce the cost to be infinitesimal even if f_1, f_2 are quite different. Such γ minimize the cost by eliminating the part of f_2 that is greatly different from the part of f_1, which is referred to *pinching problem* in the literature [20]. Figure 13.2 shows a simple example to illustrate the idea using two scalar functions. We have two scalar functions on the unit interval [0, 1] showed in the left panel. If we optimize the previous L^2 based objective functions, the obtained time warping function γ is plotted on the right panel while the middle panel visualizes the reparameterization $f_2 \circ \gamma$. As we can see, it kills the height of f_2 to get this degenerate solution. To avoid this, people proposed the modified solution that penalize large time warpings by some roughness penalties:

$$\inf_{\gamma\in\Gamma} (\|f_1 - f_2 \circ \gamma\| + \lambda\mathcal{R}(\gamma)) \,, \tag{13.1}$$

where $\mathcal{R}(\gamma)$ represents the roughness of γ. For example, it can be the norm of the first or the second derivatives.

While this solution prevents the pinching problem, it introduces new issues. For example, the solution is not inverse consistent. That is, the registration of f_2 to f_1 is no longer equivalent with that of f_1 to f_2. We use Fig. 13.3 to explain this issue. The task is to align two scalar functions f_1 and f_2, shown in the top panel of Fig. 13.3. And in this example, we use the first order penalty $\mathcal{R}(\gamma) = \int \dot{\gamma}(t)^2 dt$ in Eq. (13.1). To study the property of symmetry, for each row, we perform registration using different template and target on a fixed λ, i.e., warping f_2 to register to f_1 to get γ_1 and warping f_1 to register to f_2 to get γ_2. Then, we compose the two obtained optimal warping functions. If the solution is symmetric, the composition $\gamma_1 \circ \gamma_2^{-1}$ should be the identity function: $\gamma_{id}(t) = t$. The last column shows the compositions. As we can see, when $\lambda = 0$, the solution is symmetric. However, it suffers the pinching problem. As λ increases, the pinching effect is reducing but the solution is no longer inverse consistent. In the last row, where λ is large, the

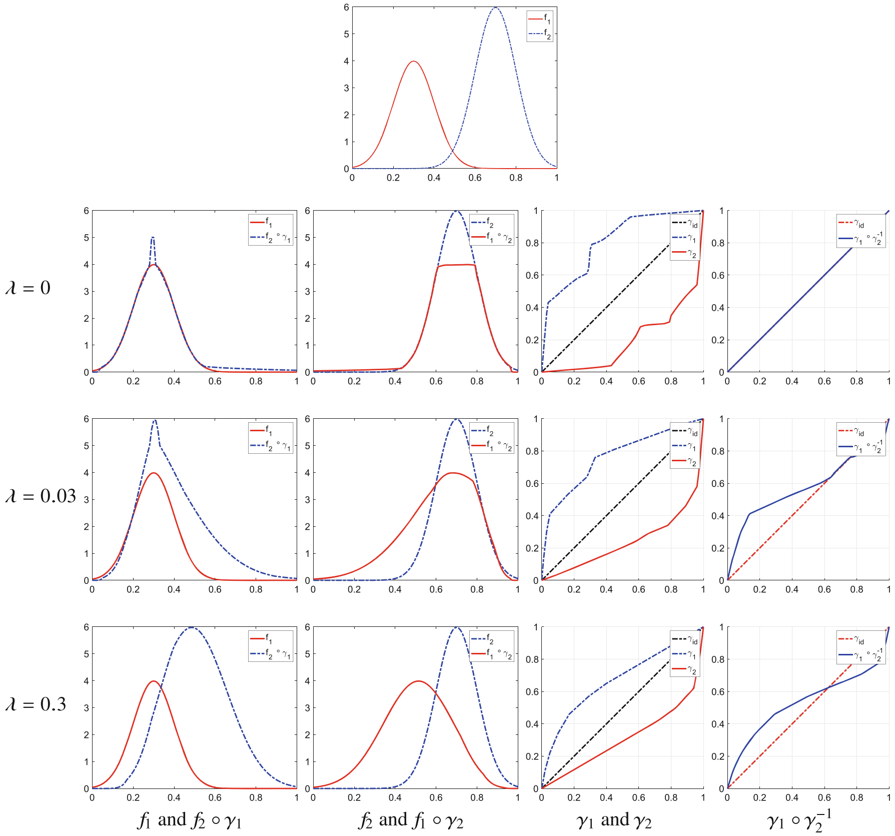


Fig. 13.3 Example of penalized-$\mathbb{L}^2$ based alignment. The top row shows the two original functions. From the second row to the bottom row, the roughness tuning parameter is set to $\lambda = 0, 0.03, 0.3$, respectively. As λ increases, the solution becomes more and more asymmetric

pinching problem disappears. However, the alignment is also largely limited. It is not obvious to select the appropriate λ for this example. In reality, it is even difficult to tune this parameter.

In the following sections, we will go through the shape analysis in elastic framework for scalar functions, parametrized curves.

13.2.2 Elastic Registration of Scalar Functions

Among various functional objects one comes across in shape analysis, the simplest types are real-valued functions on a fixed interval. For simplicity, functional data in this section refers to the scalar functions. Examples include human activities collected by wearable devices, biological growth data, weather data, etc. Shape analysis on scalar functions is reduced to alignment problem in FDA. If one does not account for misalignment in the given data, which happens when functions are contaminated with the variability in their domain, this can inflate variance artificially and can overwhelm any statistical analysis. The task is warping the temporal domain of functions so that geometric features (peaks and valleys) of functions are well-aligned, which is also called *curve registration* or *phase-amplitude separation* [14]. While we have illustrated the limitation of L^2 norm earlier, we will introduce a desirable solution in elastic framework as follows.

Definition 13.1 For a function $f(t) : [0, 1] \to \mathbb{R}$, define the square-root velocity function (SRVF) (or square-root slope function (SRSF)) $q(t)$ as follows:

$$q(t) = sign(\dot{f}(t))\sqrt{|\dot{f}(t)|}\,. \tag{13.2}$$

It can be shown if $f(t)$ is absolutely continues, $q(t)$ is square-integrable, i.e., $q(t) \in L^2$. The representation is invertible given $f(0)$: $f(t) = f(0) + \int_0^t q(s)|q(s)|ds$. If the function f is warped as $f \circ \gamma$, then the SRVF becomes: $(q \circ \gamma)\sqrt{\dot{\gamma}}$, denoted by $(q*\gamma)$. One of the most import properties of the representation is isometry under the action of diffeomorphism: $\|q_1-q_2\| = \|(q_1*\gamma)-(q_2*\gamma)\|, \forall\gamma \in \Gamma$. In other words, L^2 norm of SRVFs is preserved under time warping. One can show that the L^2 metric of SRVF is non-parametric Fisher-Rao metric on f, given f is absolutely continuous and $\dot{f} > 0$, and it can be extended to the larger space $\mathcal{F}_0 = \{f \in \mathcal{F} | f \text{ is absolute continuous}\}$ [20]. Then, in order to register f_1 and f_2, the problem becomes

$$\inf_{\gamma\in\Gamma} \|q_1 - (q_2 * \gamma)\| = \inf_{\gamma\in\Gamma} \|q_2 - (q_1 * \gamma)\|\,. \tag{13.3}$$

One can efficiently solving above objective function using Dynamic Programming [4]. Gradient based algorithm or exact solutions [18] are also available.

For aligning multiple functions, one can easily extend the framework to align every function to their Karcher mean [20] by iteratively updating the following equations:

$$\gamma_i = \arg\inf_{\gamma\in\Gamma} \|\mu - (q_i * \gamma)\|\,,$$

$$\mu = \frac{1}{n}\sum_{i=1}^{n}(q_i * \gamma_i)$$

We demonstrate an application of function alignment using the famous Berkeley growth data [27], where observations are heights of human subjects in the age domain recorded from birth to age 18. In order to understand the growth pattern, we use a smoothed version of the first time derivative (growth velocity) of the height

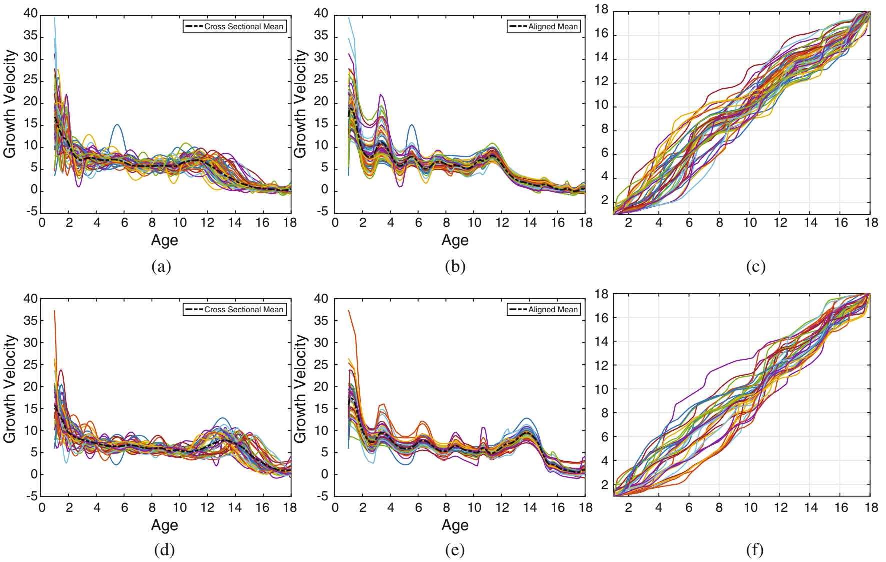


Fig. 13.4 Alignment on Berkeley growth data. (**a**) Original male growth function. (**b**) Aligned male growth function. (**c**) Time warping functions for male. (**d**) Original female growth function. (**e**) Aligned female growth function. (**f**) Time warping functions for female

functions, instead of the height functions themselves, as functional data, plotted in (a) and (d) of Fig. 13.4 for male and female subjects, respectively. The task is to align the growth spurts of different subjects in order to make inferences about the number and placement of such spurts for the underlying population. The aligned functions are presented in (b) and (e) of Fig. 13.4. After alignment, it becomes much easier to estimate the location and size of the growth spurts in observed subjects, and make inferences about the general population.

There are many studies related to the *elastic functional analysis* in the literature. For example, one can construct a generative model for functional data, in terms of both amplitude and phase parts [25]. One can also take account the elastic part into functional principal component analysis [26]. For regression models using elastic functions as predictors, readers can refer to [1].

13.2.3 Elastic Shape Analysis of Curves

13.2.3.1 Registration of Curves

As previously mentioned, square-root transformations were first proposed for planar curves [29, 30]. We can register curves in $\mathbb{R}^2$ and $\mathbb{R}^3$ using SRVFs as defined below.

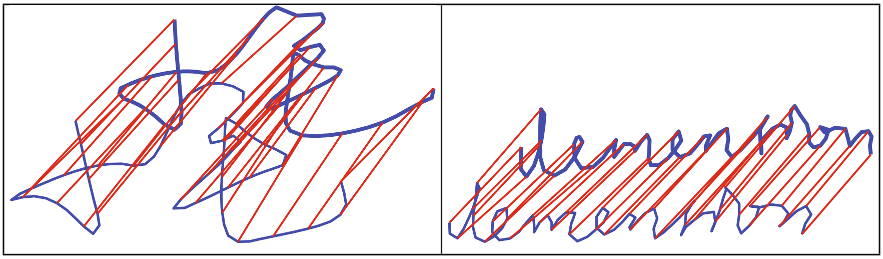


Fig. 13.5 Registration of signatures

Definition 13.2 Define SRVF of 2D or 3D parametrized curves:

$$q(t) = \begin{cases} \frac{\dot{f}(t)}{\sqrt{|\dot{f}(t)|}}, & |\dot{f}(t)| \neq 0 \\ 0\,, & |\dot{f}(t)| = 0 \end{cases}\,. \tag{13.4}$$

Here $f : [0, 1] \to \mathbb{R}^n, n = 2, 3$, is an absolutely continuous parametrized curve. (For closed curves, $\mathbb{S}^1$ is a more appropriate domain.) It is worth noting that this definition is valid for $\mathbb{R}^n$. The L^2 metric in the space of SRVFs is a special elastic metric in the space of curves, which measures the bending and stretching from the curves [20]. To register curves, we again use Eq. (13.3). An example of registering 2D curves is shown in Fig. 13.5, where we are registering signatures.

13.2.3.2 Shape Space

Now we know how to align functions and to register curves using the SRVF framework. And these will now serve as fundamental tools for our ultimate goal: shape analysis. One has to note that shapes are invariant to some nuanced group actions: translation, scaling, rotation, and reparametrization. For example, Fig. 13.6 illustrates that using a bird shape. On the left panel, although the bird contour is shifted, scaled and rotated, the shape is keeping the same as the original one. One the right panel, two shapes have different reparametrizations but they need to be treated as the same shape. Therefore, it is important to identify what is the space that shapes reside in. We represent a curve f by its SRVF q and thus it is invariant to translation (because it is a derivative). The curve can be rescaled into unit length to remove scaling. Since the length of f is $L[f] = \|q\|$, after rescaling, $\|q\| = 1$. As the result, the unit length q is on the unit Hilbert sphere. Let $C = \{q \in L^2 | \|q\| = 1\}$ denote the unit Hilbert sphere inside L^2, which is also called *preshape space*. The geometry of C is simple: the distance between any two point $q_1, q_2 \in C$ is given by the arc length $d_C(q_1, q_2) = \cos^{-1}(\langle q_1, q_2 \rangle)$, where $<, >$ represents L^2 inner

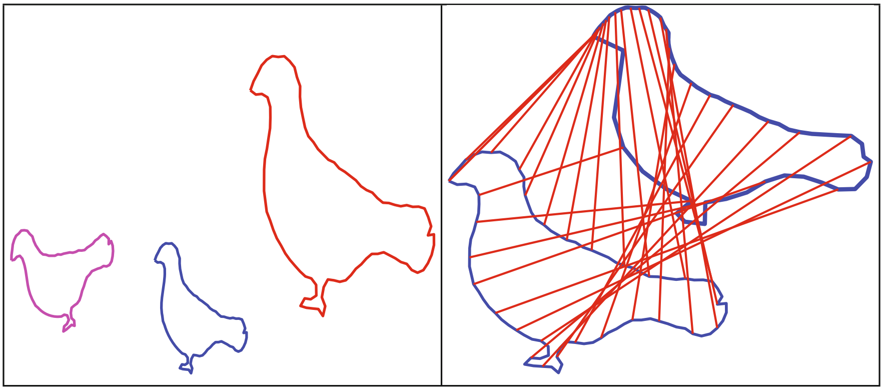


Fig. 13.6 Example of bird shapes

product. The geodesic (shortest path) between q_1 and q_2 is $\alpha : [0, 1] \to C$ is the shortest arc length on the greater circle:

$$\alpha(\tau) = \frac{1}{\sin\theta}(\sin((1-\tau)\theta)q_1 + \sin(\tau\theta)q_2), \tau \in [0, 1]$$

The remaining variability that has not been removed is rotation and reparametrization. We will remove them by using equivalent classes that are defined by group actions. Let $SO(n)$ represent the set of all the rotation matrices in $\mathbb{R}^n$. For any $O \in SO(n)$ and $q \in C$, Oq has exactly same shape with q. (The SRVF of Of is Oq.) The same holds for a reparametrization $(q * \gamma)$, $\forall\gamma \in \Gamma$. We will treat them as the same object in the shape space as follows. Define the action of group $SO(n) \times \Gamma$ on C according to:

$$(SO(n) \times \Gamma) \times C \to C, (O, \gamma) * q = O(q * \gamma) ,$$

which leads to the equivalent classes or orbits:

$$[q] = \{O(q * \gamma) | O \in SO(n), \gamma \in \Gamma\} .$$

Therefore, each orbit $[q]$ represents a unique shape of curves. The shape space (quotient space) $\mathcal{S}$ is the collection of all the orbits:

$$\mathcal{S} = C/(SO(n) \times \Gamma) = \{[q] | q \in C\} .$$

As we mentioned earlier, the inner product or L^2 norm of SRVF is preserved under reparametrization. This is also true for rotation actions: $\langle q_1, q_2\rangle = \langle Oq_1, Oq_2\rangle$. As the result, we can inherit the metric from preshape space into shape space as follows:

Definition 13.3 For two representations of shapes $[q_1]$ and $[q_2]$, define the shape metric as:

$$d_{\mathcal{S}}([q_1], [q_2]) = \inf_{\gamma \in \Gamma, O \in SO(n)} d_C(q_1, O(q_2 * \gamma)) \,. \tag{13.5}$$

The above equation is a proper metric in the shape space [20] and thus can be used for ensuing statistical analysis. The optimization over $SO(n)$ is performed using Procrustes method [10]. For instance, for curves in 2D, the optimal rotation O^* is given by

$$O* = \begin{cases} UV^T & \text{if } det(A) > 0 \\ U \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} V^T & \text{otherwise} \end{cases} \,, \tag{13.6}$$

where $A = \int_0^1 q_1 q_2^T dt$ and $A = U \Sigma V^T$ (singular value decomposition). While the optimization of γ can be implement by Dynamic Programming or gradient based methods [20]. For $[q_1]$ and $[q_2]$ in $\mathcal{S}$, the geodesic path is given by the geodesic between q_1 and $\tilde{q}_2$, while $\tilde{q}_2$ is rotated and reparemetrized w.r.t. q_1. We present an example geodesic in Fig. 13.7. The top row is the geodesic in $\mathcal{S}$. For comparison, we also plot the geodesic path in C in bottom row. It is clear to see that elastic registration makes a more reasonable deformation since it matches the corresponding parts.

Shape Spaces of Closed Curves For parametrized closed curves, there is one more constraint: $f(0) = f(1)$. Therefore, as we mentioned earlier, $\mathbb{S}^1$ is a more natural domain for parametrized closed curves. Let q denote the SRVF of a closed curve f,

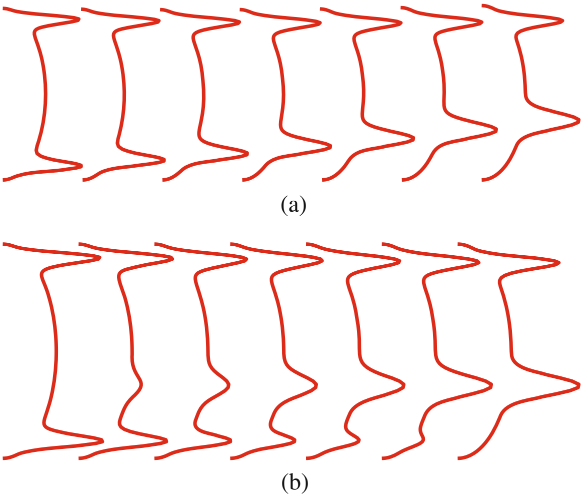


Fig. 13.7 Comparison between geodesic and interpolation for a toy 2D open curve. (**a**) Geodesic in $\mathcal{S}$. (**b**) Geodesic in C

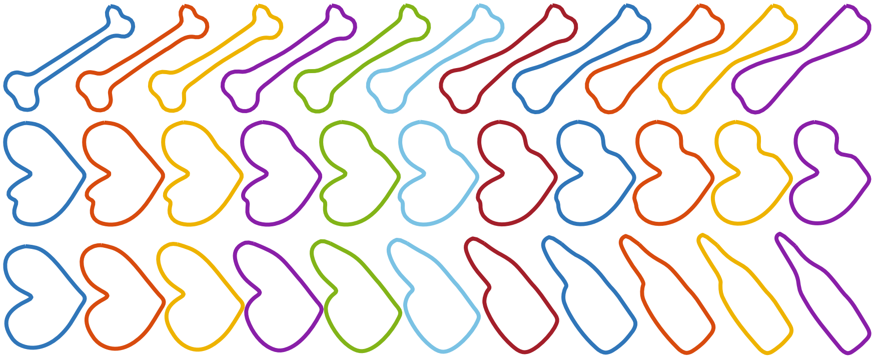


Fig. 13.8 Some geodesic paths between two closed planar curves in $\mathcal{S}^c$

the above condition $f(0) = f(1)$ becomes $\int_{\mathbb{S}^1} q(t)|q(t)|dt = 0$. As the result, the prespace $\mathcal{C}^c$ for unit length closed curve is:

$$\mathcal{C}^c = \left\{ q \in \mathbb{L}^2(\mathbb{S}^1, \mathbb{R}^n) | \int_{\mathbb{S}^1} |q(t)|dt = 1, \int_{\mathbb{S}^1} q(t)|q(t)|dt = 0 \right\} \subset \mathcal{C} .$$

One can still use $d_{\mathcal{C}}$ as the extrinsic metric in $\mathcal{C}^c$ [20]. Unlike the open curves, the geodesics in $\mathcal{C}^c$ have no closed form. A numerical approximation method called path straightening [20] can be used to compute the geodesic. The shape space is $\mathcal{S}^c = \mathcal{C}^c/(SO(n) \times \Gamma)$. whose elements, equivalence classes or orbits, are $[q] = \{O(q * \gamma)|q \in \mathcal{C}^c, O \in SO(n), \gamma \in \Gamma\}$. Some examples of geodesic paths can be found in Fig. 13.8, where we can see natural deformations between two shapes.

13.3 Shape Summary Statistics, Principal Modes and Models

The framework we have developed so far is able to define and compute several statistics for analysis of shapes. For instance, we may want to compute the mean shape from a sample of curves to represent the underlying population. The intrinsic sample mean on a nonlinear manifold is typically defined as the Fréchet mean or Karcher mean, defined as follows.

Definition 13.4 (Mean Shape) Given a set of curves $f_1, f_2, \ldots, f_n \in \mathcal{F}_0$ with corresponding shapes $[q_1], [q_2], \ldots, [q_n]$, we define the mean shape $[\mu]$ as:

$$[\mu] = \arg\min_{[q]} \sum_{i=1}^{n} d_{\mathcal{S}}^2([q], [q_i]) .$$

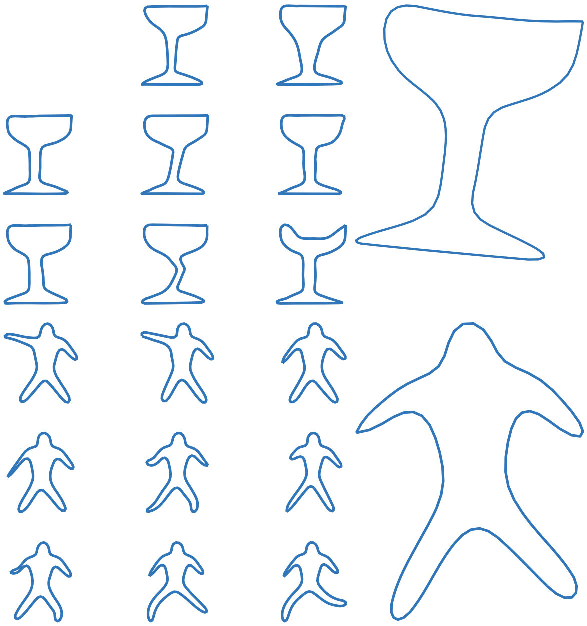
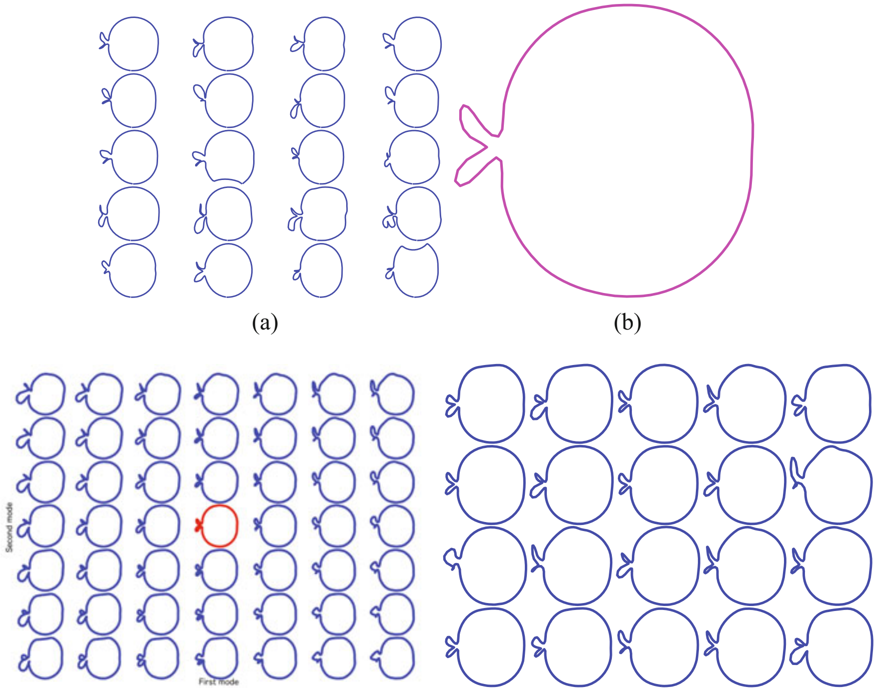


Fig. 13.9 Examples of mean shapes

The algorithm for computing a mean shape [20] is similar to the one described earlier in multiple function alignment. We iteratively find the best one from rotation, registration and average while fixing the other two, in way of coordinate descent [28]. Figure 13.9 illustrates some mean shapes. One the left hand side, there are some sample shapes: glasses and human beings. Their corresponding mean shapes are plotted on the right.

Besides the Karcher mean, the Karcher covariance and modes of variation can be calculated to summarize the given sample shapes. As it is known that the shape space $\mathcal{S}$ is a non-linear manifold, we will use tangent PCA [20] to flatten the space. Let $T_{[\mu]}\mathcal{S}$ denote the tangent space to $\mathcal{S}$ at the mean shape $[\mu]$ and $\log_{[\mu]}([q])$ denote the mapping from the shape space $\mathcal{S}$ to this tangent space using inverse exponential map. Let $v_i = \log_{[\mu]}([q])$, for $i = 1, 2, \ldots, n$ be the shooting vectors from the mean shape to the given shapes in the tangent space. Since these shooting vectors are in the linear space, we are able to compute the covariance matrix $C = \frac{1}{n-1}\sum_{i=1}^{n} v_i v_i^t$. Performing Principal Component Analysis (PCA) of C provides the directions of maximum variability in the given shapes and can be used to visualize (by projecting back to the shape space) the main variability in that set. Besides that, one can impose a Gaussian model on the principal coefficients $s_i, i = 1, 2, \ldots, n$ in the tangent space. To valid the model, one can generate a random vector r_i from the estimated model and project back to the shape space using exponential map $\exp_{[\mu]}(r_i)$, where random vectors become random shapes. We use Figs. 13.10 and 13.11 as illustrations. We have several sample shapes of apples and



(a) (b)

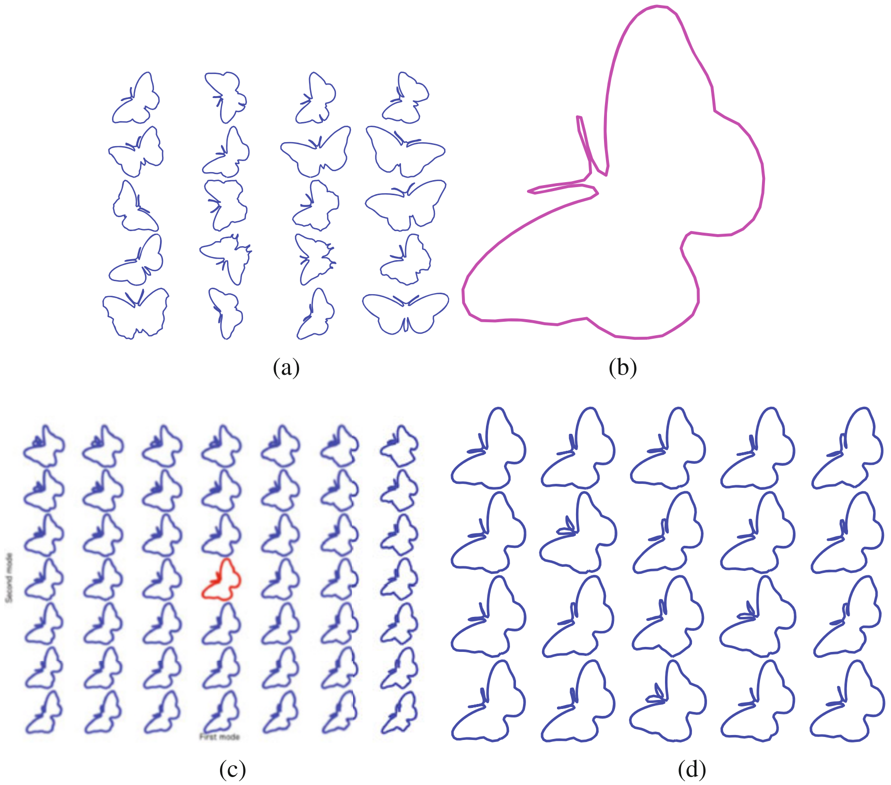
(c) (d)

Fig. 13.10 Principal modes and random samples of apple shapes. (**a**) Sample shapes. (**b**) Mean shapes. (**c**) Principal modes. (**d**) Random samples from Tangent Gaussian model

butterflies in panel (a) and the mean shapes are presented in panel (b). We perform tangent PCA as described above and show the results in panel (c). While the mean shapes are the red shapes in the shape matrix, the modes in first and second principal direction are plotted horizontally and vertically, respectively, which explain the first and second modes of variation in the given sample shapes. Finally, we generate some random shapes from the estimated tangent Gaussian model and show them in panel (d). The similarity between random shapes and given samples validates the fitness of the shape models.

13.4 Conclusion

In this chapter, we describe the elastic framework for shape analysis of scalar functions and curves in Euclidean spaces. The SRVF transformation simplifies the registration and makes the key point for the approach. Combining with L^2 norm, it derives an appropriate shape metric that unifies registration with comparison



(a) (b)

(c) (d)

Fig. 13.11 Principal modes and random samples of butterfly shapes. (**a**) Sample shapes. (**b**) Mean shapes. (**c**) Principal modes. (**d**) Random samples from Tangent Gaussian model

of shapes. As the result, one can compute geodesic paths, summary statistics. Furthermore, these tools can be used in statistical modeling of shapes.

References

1. Ahn, K., Derek Tucker, J., Wu, W., Srivastava, A.: Elastic handling of predictor phase in functional regression models. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (2018)
2. Bauer, M., Bruveris, M., Marsland, S., Michor, P.W.: Constructing reparameterization invariant metrics on spaces of plane curves. Differ. Geom. Appl. **34**, 139–165 (2014)
3. Bauer, M., Bruveris, M., Harms, P., Møller-Andersen, J.: Second order elastic metrics on the shape space of curves (2015). Preprint. arXiv:1507.08816
4. Bellman, R.: Dynamic programming. Science **153**(3731), 34–37 (1966)
5. Brigant, A.L.: Computing distances and geodesics between manifold-valued curves in the SRV framework (2016). Preprint. arXiv:1601.02358
6. Dryden, I., Mardia, K.: Statistical Analysis of Shape. Wiley, London (1998)

7. Jermyn, I.H., Kurtek, S., Laga, H., Srivastava, A.: Elastic shape analysis of three-dimensional objects. Synth. Lect. Comput. Vis. **12**(1), 1–185 (2017)
8. Joshi, S.H., Xie, Q., Kurtek, S., Srivastava, A., Laga, H.: Surface shape morphometry for hippocampal modeling in alzheimer's disease. In: 2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA), pp. 1–8. IEEE, Piscataway (2016)
9. Kendall, D.G.: Shape manifolds, procrustean metrics, and complex projective spaces. Bull. Lond. Math. Soc. **16**(2), 81–121 (1984)
10. Kendall, D.G.: A survey of the statistical theory of shape. Stat. Sci. **4**(2), 87–99 (1989)
11. Kurtek, S., Needham, T.: Simplifying transforms for general elastic metrics on the space of plane curves (2018). Preprint. arXiv:1803.10894
12. Kurtek, S., Xie, Q., Samir, C., Canis, M.: Statistical model for simulation of deformable elastic endometrial tissue shapes. Neurocomputing **173**, 36–41 (2016)
13. Liu, W., Srivastava, A., Zhang, J.: A mathematical framework for protein structure comparison. PLoS Comput. Biol. **7**(2), e1001075 (2011)
14. Marron, J.S., Ramsay, J.O., Sangalli, L.M., Srivastava, A.: Functional data analysis of amplitude and phase variation. Stat. Sci. **30**(4) 468–484 (2015)
15. Michor, P.W., Mumford, D., Shah, J., Younes, L.: A metric on shape space with explicit geodesics (2007). Preprint. arXiv:0706.4299
16. Mio, W., Srivastava, A., Joshi, S.: On shape of plane elastic curves. Int. J. Comput. Vis. **73**(3), 307–324 (2007)
17. Ramsay, J.O.: Functional data analysis. In: Encyclopedia of Statistical Sciences, vol. 4 (2004)
18. Robinson, D., Duncan, A., Srivastava, A., Klassen, E.: Exact function alignment under elastic riemannian metric. In: Graphs in Biomedical Image Analysis, Computational Anatomy and Imaging Genetics, pp. 137–151. Springer, Berlin (2017)
19. Small, C.G.: The Statistical Theory of Shape. Springer, Berlin (2012)
20. Srivastava, A., Klassen, E.P.: Functional and Shape Data Analysis. Springer, Berlin (2016)
21. Srivastava, A., Jermyn, I., Joshi, S.: Riemannian analysis of probability density functions with applications in vision. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8. IEEE, Piscataway (2007)
22. Srivastava, A., Klassen, E., Joshi, S.H., Jermyn, I.H.: Shape analysis of elastic curves in euclidean spaces. IEEE Trans. Pattern Anal. Mach. Intell. **33**(7), 1415–1428 (2011)
23. Srivastava, A., Wu, W., Kurtek, S., Klassen, E., Marron, J.S.: Registration of functional data using Fisher-Rao metric (2011). Preprint. arXiv:1103.3817
24. Su, J., Kurtek, S., Klassen, E., Srivastava, A., et al.: Statistical analysis of trajectories on riemannian manifolds: bird migration, hurricane tracking and video surveillance. Ann. Appl. Stat. **8**(1), 530–552 (2014)
25. Tucker, J.D., Wu, W., Srivastava, A.: Generative models for functional data using phase and amplitude separation. Comput. Stat. Data Anal. **61**, 50–66 (2013)
26. Tucker, J.D., Lewis, J.R., Srivastava, A.: Elastic functional principal component regression. Stat. Anal. Data Min.: ASA Data Sci. J. **12**(2), 101–115 (2019)
27. Tuddenham, R.D., Snyder, M.M.: Physical growth of california boys and girls from birth to eighteen years. Publ. Child Dev. Univ. Calif. **1**(2), 183 (1954)
28. Wright, S.J.: Coordinate descent algorithms. Math. Program. **151**(1), 3–34 (2015)
29. Younes, L.: Computable elastic distances between shapes. SIAM J. Appl. Math. **58**(2), 565–586 (1998)
30. Younes, L.: Optimal matching between shapes via elastic deformations. Image Vis. Comput. **17**(5–6), 381–389 (1999)
31. Younes, L.: Elastic distance between curves under the metamorphosis viewpoint (2018). Preprint. arXiv:1804.10155
32. Zhang, Z., Klassen, E., Srivastava, A.: Phase-amplitude separation and modeling of spherical trajectories. J. Comput. Graph. Stat. **27**(1), 85–97 (2018)
33. Zhang, Z., Su, J., Klassen, E., Le, H., Srivastava, A.: Video-based action recognition using rate-invariant analysis of covariance trajectories (2015). Preprint. arXiv:1503.06699

Chapter 14
Statistical Analysis of Trajectories of Multi-Modality Data

Jingyong Su, Mengmeng Guo, Zhipeng Yang, and Zhaohua Ding

Contents

Abstract We develop a novel comprehensive Riemannian framework for analyzing, summarizing and clustering trajectories of multi-modality data. Our framework relies on using elastic representations of functions, curves and trajectories. The elastic representations not only provide proper distances, but also solve the problem of registration. We propose a proper Riemannian metric, which is a weighted average of distances on product spaces. The metric allows for joint comparison and registration of multi-modality data. Specifically, we apply our framework to detect stimulus-relevant fiber pathways and summarize projection pathways. We evaluate our method on two real data sets. Experimental results show that we can cluster

J. Su (✉)
Harbin Institute of Technology (Shenzhen), Shenzhen, China
e-mail: sujingyong@hit.edu.cn

M. Guo
Texas Tech University, Lubbock, TX, USA
e-mail: mengmeng.guo@ttu.edu

Z. Yang
Sichuan University, Chengdu, China
e-mail: yangzp@cuit.edu.cn

Z. Ding
Vanderbilt University, Nashville, TN, USA
e-mail: zhaohua.ding@vanderbilt.edu

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_14

fiber pathways correctly and compute better summaries of projection pathways. The proposed framework can also be easily generalized to various applications where multi-modality data exist.

14.1 Introduction and Background

This chapter describes geometric ideas for analyzing trajectories of multi-modality data. There are a lot of multi-modality data in medical imaging, computer vision and many other applications. Examples of multi-modality data include white matter fibers associated with functional data (BOLD (blood oxygenation level dependent) signals) and symmetric positive definite (SPD) matrices (diffusion tensors). BOLD signals in white matter fibers encode neural activity related to their functional roles connecting cortical volumes. Functional MRI has proven to be effective in detecting neural activity in brain cortices on the basis of BOLD contrast. A diffusion tensor MRI scan of the brain generates a field of 3×3 SPD matrices that describes the constraints on local Brownian motion of water molecules [19]. At each observation of white matter fiber, we are able to detect and extract BOLD signal and SPD matrix. Therefore, white matter fibers associated with functional data (BOLD signals) and SPD matrices (diffusion tensors) are two examples of multi-modality data. For such data types, we analyze the data by considering them as three dimensional open curves and trajectories of functional data or SPD matrices jointly. There have been a number of studies on quantifying and clustering white matter fibers. The top row of Fig. 14.1 displays three different views of some white matter fibers of one subject. Motivated by the goal of diagnosing different white matter diseases, various techniques have been developed to analyze brain fibers. Previous work on fiber analysis mostly involves diffusion MRI. Figure 14.2 displays the internal capsule structure of a human, 12 representative projection pathways passing through internal capsule and the corresponding trajectories of tensors (SPD matrices).

There are some important reasons for studying multi-modality data using a Riemannian geometric approach. The data incorporates curve features and associated functions or tensors along curves. Since functions, curves and trajectories are observed with domain flexibility, they have some obvious geometric features and local characteristic pattern such as ridges and valleys. An important task lies in matching and finding correspondences between them, quantifying the differences by defining distances and calculating statistical summaries such as means and covariances. We integrate the registration into shape analysis, which involves a matching of points or shapes across objects when the shapes or trajectories are quantified. At the same time, these shape or trajectory objects can preserve their shapes and have invariant properties. For example, shapes are invariant to arbitrary rotation, scaling, translation, and re-parameterization. For the multi-modality data of curves associated with functions or tensors, our goal is to match and find correspondences between them by analyzing these two types of information jointly. We apply a metric-based geometric approach for representing functions, curves and trajectories. Our Riemnnian framework allows for computing their differences and registration

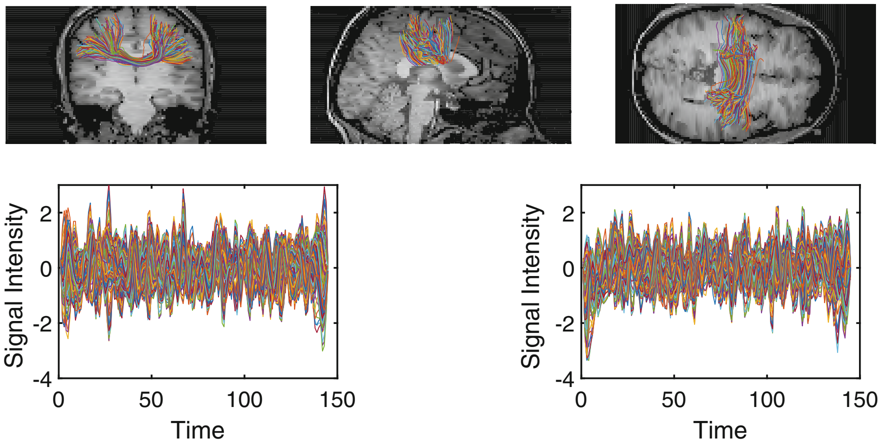


Fig. 14.1 Top: Three different views of white matter fibers of one subject; Bottom: Left: BOLD signals transmitted at each point of one fiber along this white matter fiber. Right: BOLD signals transmitted at each point of another fiber along this white matter fiber

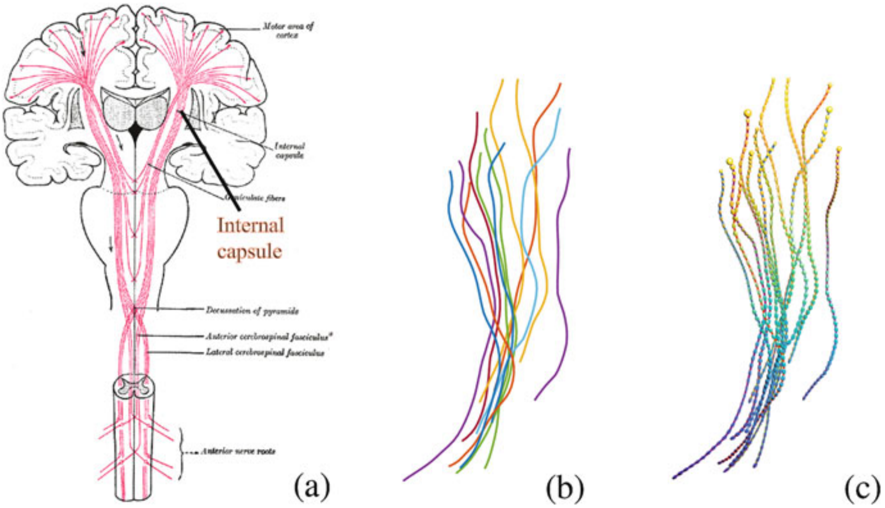


Fig. 14.2 Data acquisition: (**a**) projection pathways; (**b**) and (**c**) 12 fibers and corresponding SPD trajectories

of them at the same time. We develop a comprehensive Riemnnian framework for providing registration and computing statistical summaries by incorporating and analyzing curves associated with functions or tensors jointly.

In this chapter, we apply the framework to two medical imaging studies. We use it to detect stimulus-relevant fiber pathways and summarize projection pathways. There are few different methods to quantify and analyze diffusion tensor and white matter data such as regional basis analysis [1], voxel based analysis [20, 34] and tract based analysis [6, 16, 17, 28, 44]. An important and challenging task in analyzing white matter bundles lies in fiber clustering. There are some previous tract based methods to cluster fibers such as [8, 14, 16, 17, 27, 45, 47]. Another important task in fiber analysis is parcellation based connectome analysis such as [7, 27, 48]. Quantifying the similarity of fiber tracts can be addressed with different shape statistics such as fractional anisotropy (FA), mean diffusivity (MD) and mean squared difference (MSD) [2, 5, 21]. Other works using diffusion tensor MRI include atlas building for group studies [22, 26], statistical methods for quantitative analysis [22, 50] and other measures of white matter integrity along fibers [15].

With the increasing importance of analyzing and clustering white matter tracts for clinical needs, there has been a growing demand for a mathematical framework to perform quantitative analysis of white matter fibers incorporating their shape features and underlying physical significance. Shape analysis has been studied with a variety of mathematical representations of shapes. The representation proposed in [35] has been applied to summarize and cluster white matter fibers. Based on such representation, [23] proposes a comprehensive Riemannian framework for the analysis of whiter matter tracts based on different physical features. Under this framework, shape of curves can be compared, aligned, and deformed in chosen metrics. Zhang et al. [49] avoids discarding the rich functional data on the shape, size and orientation of fibers, developing flexible models for characterizing the population distribution of fibers between brain regions of interest within and across different individuals.

Recently, neuroimaging techniques based on blood oxygenation level-dependent (BOLD) from fMRI are also used for detecting neural activity in human brain. The bottom row of Fig. 14.1 displays BOLD signals from two different fibers. BOLD signal changes are associated with hemodynamic responses to stimuli. Ding et al. [9] suggests that changes are associated to neural activity. With BOLD signals in functional loading, white matter fibers can be studied as trajectories of functions. There has been a lot of work on handling and analyzing functional data. Functional data analysis (FDA) has been studied extensively in [30, 31]. However, none of these papers studies trajectories of functions as a whole. Su et al. [39] proposed a general framework for statistical analysis of trajectories on manifolds. We will apply it to functional data for our purpose. Our goal in this chapter is to develop a comprehensive framework for analyzing and clustering white matter fibers that can incorporate both physical properties of fibers and BOLD signals along fibers. A proper metric on the product space of shapes and functions is proposed to compare, align, and summarize white matter tracts.

The rest of this chapter is organized as follows. We will follow the metric-based Riemannian framework for analyzing functions, curves, trajectories and curves associated with trajectories. The mathematical framework for elastic shape analysis of open curves is presented in Sect. 14.2. In Sect. 14.3, we present a mathematical framework for elastic analysis of trajectories. The joint framework of analyzing shapes and trajectories is displayed in Sect. 14.4. We study two types of trajectories: functions and tensors. In the study of detecting stimulus-relevant fiber pathways, we display clusters of fibers and compute correlation between stimuli and BOLD signals. In the study of summarizing projection pathways, we calculate and compare statistical summaries by considering fibers as 3D open curves and trajectories of tensors. In Sect. 14.5, we conclude with a summary.

14.2 Elastic Shape Analysis of Open Curves

As discussed before, we may consider fiber tracts as 3D open curves with physical features. We adopt the mathematical representation of curves proposed in [36], and now briefly summarize the main idea. The representation of shape data is based on the use of square-root velocity function (SRVF) of curves by [36]. The set of all re-parameterizations is defined as:

$$\Gamma = \{\gamma\colon [0,1] \to [0,1] | \gamma(0) = 0, \gamma(1) = 1, \gamma \text{ is a diffeomorphism}\}, \tag{14.1}$$

Considering an absolutely continuous, parametrized 3D open curve $\beta\colon [0,1] \to \mathbb{R}^3$, a reparametrization of the curve β can be defined as the composition $\beta \circ \gamma$. A major problem lies in defining distance between two curves β_1 and β_2 using standard metrics. Most papers apply the standard Euclidean metric $\mathbb{L}^2$ to define the distance and perform registration by using the distance $inf_{\gamma\in\Gamma}\|\beta_1 - \beta_2 \circ \gamma\|$. However, there are various difficulties with such a distance. The first problem is it is not symmetric. The optimal alignment of β_1 to β_2 is different from the alignment of β_2 to β_1, that is $inf_{\gamma\in\Gamma}\|\beta_1 - \beta_2 \circ \gamma\| \neq inf_{\gamma\in\Gamma}\|\beta_1 \circ \gamma - \beta_2\|$. The second problem is pinching problem in [24], that is, even when β_1 and β_2 are quite different, the cost can be reduced to be close to zero. To address this problem, a penalty term $\mathcal{R}$ can be imposed on this criterion $inf_{\gamma\in\Gamma}\|\beta_1 - \beta_2 \circ \gamma\| + \lambda\mathcal{R}(\gamma)$. $\mathcal{R}$ is a smooth penalty on γ. But how to choose parameter λ still remains open. The third problem is that it is not invariant to re-parameterizations. That is, for $\gamma \in \Gamma$, $\|\beta_1 - \beta_2\| \neq \|\beta_1 \circ \gamma - \beta_2 \circ \gamma\|$. Therefore, the standard $\mathbb{L}^2$ metric is not isometric to the group action of Γ. In order to achieve this property, [36] introduces a representation of curves by the square-root velocity function (SRVF), which is defined as

$$q(t) = \frac{\dot{\beta}(t)}{\sqrt{|\dot{\beta}(t)|}}. \tag{14.2}$$

Due to the SRVF representation, the elastic metric in [25] becomes a standard $\mathbb{L}^2$ metric, thus we have access to more efficient ways of computing geodesics distances with SRVF. If a curve β is re-parameterized by any $\gamma \in \Gamma$, then the SRVF of re-parameterized curve is given by $(q,\gamma)\colon = (q \circ \gamma)\sqrt{\dot{\gamma}}$. If β is rotated by a rotation matrix $O \in SO(3)$, its SRVF also rotates by the same O. It is easy to show that for two SRVFs q_1 and q_2, $\|q_1 - q_2\| = \|O(q_1,\gamma) - O(q_2,\gamma)\|$. That is, the $\mathbb{L}^2$ distance between SRVFs of any two curves is unchanged by simultaneous re-parameterization $\gamma \in \Gamma$ and rotation $O \in SO(3)$ of these curves. Therefore, actions of the group Γ and O are isometric on the space of SRVFs under $\mathbb{L}^2$

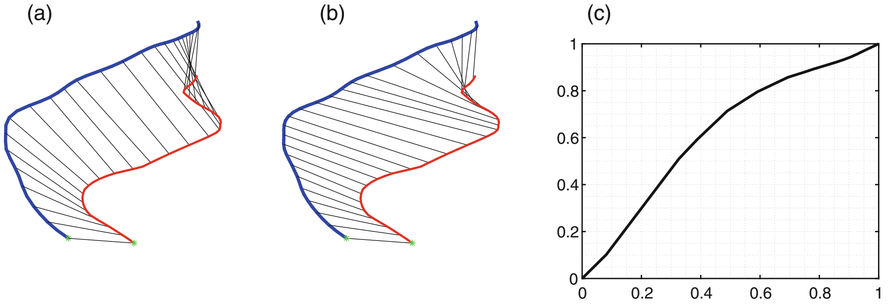


Fig. 14.3 (**a**) Fibers before registration; (**b**) fibers after registration; (**c**) re-parameterization function for these two fibers

metric. This isometry property avoids the pinching problems. Since the fact that re-parameterization is shape preserving, the cost function for pairwise registration of curves can be written as: $inf_{(\gamma,O)\in\Gamma\times SO(3)}\|q_1 - O(q_2\circ\gamma)\sqrt{\dot{\gamma}}\|$, which has the advantage of being symmetric, positive definite and satisfying triangle inequality. We unify all SRVFs from re-parameterizations and rotations of same curve by the notation of orbits under the actions of $SO(3)$ and Γ. That is, each curve is associated with a equivalent class defined as the set: $[q] = \{O(q\circ\gamma)\sqrt{\dot{\gamma}}|(\gamma, O) \in \Gamma\times SO(3)\}$. The set $\mathcal{S}$ of all orbits is considered as our shape space. The distance between β_1 and β_2, represented by their SRVF orbits $[q_1]$ and $[q_2]$ is given by

$$d_{\mathcal{S}}([q_1],[q_2]) = inf_{(\gamma,O)\in\Gamma\times SO(3)}\|q_1 - O(q_2\circ\gamma)\sqrt{\dot{\gamma}}\|. \tag{14.3}$$

The optimal rotation is found by $O^* = UV'$, where $U\sum V' = svd(B)$ and $B = \int_0^1 (q_1(s)q_2(s)^T)\,ds$, and the optimal re-parametrization is computed by dynamic programming in [3] for solving:

$$\gamma^* = \text{argmin}_{\gamma\in\Gamma}(\|q_1 - O^*(q_2\circ\gamma)\sqrt{\dot{\gamma}}\|).$$

An example of registration of 3D curves can be shown in Fig. 14.3. Figure 14.3 shows two fibers before registration, after registration and re-parameterization function for these two fibers. The blue fiber β_1 is registered with the red fiber β_2 through re-parameterization function γ. The black lines are the correspondences in these two fibers.

14.3 Elastic Analysis of Trajectories

In addition to fiber tracts, there are BOLD signals or tensors available along tracts, which can be considered as trajectories of functions or tensors. Su et al. [39] has introduced a general framework of analyzing trajectories on manifolds, including spheres, Lie groups and shape spaces. In this section, we will discuss the mathematical framework in [39] and apply it to functional data or SPD matrices space for our purpose. Let α denote a smooth trajectory on Riemannian manifold M and $\mathcal{P}$ denote the set of all such trajectories, for any two smooth trajectories $\alpha_1, \alpha_2 \in \mathcal{P}$, we desire to register functions along trajectories and calculate a time-warping invariant distance between them. We assume for any points $p, q \in M$, we have an expression for parallel transporting any vector $v \in \mathcal{T}_p(M)$ along the geodesic from p to q, denoted by $(v)_{p \to q}$, the geodesic between them is unique and the parallel transport is well defined.

In order to achieve the property of invariance, we propose a novel representation of trajectories for comparison and registration. Given a trajectory $\alpha \in \mathcal{P}$, we define a representation called the trajectory square-root vector field (TSRVF) according to:

$$h_\alpha(t) = \frac{\dot{\alpha}(t)_{\alpha(t) \to c}}{\sqrt{|\dot{\alpha}(t)|}} \in \mathcal{T}_c.$$

where $|\cdot|$ denotes the norm related to the Riemannian metric and the tangent space at c is denoted by $\mathcal{T}_c(M)$, where c is a reference point on $\mathcal{P}$, the choice of c depends on $\mathcal{P}$. Riemannian metric depends on the point c where it is evaluated. There is a one-to-one correspondence between trajectories and their TSRVFs. Define $\mathcal{H}$ be the set of smooth curves in $\mathcal{T}_c$ as TSRVFs of trajectories, $\mathcal{H} = \{h_\alpha | \alpha \in \mathcal{P}\}$. If M is $\mathbb{R}^d$ with the Euclidean metric then h is exactly the square-root velocity function defined in Sect. 14.2. Since h_α is a vector field, we can use the $\mathbb{L}^2$ norm to compare such trajectories and perform alignment. For calculating the distance between any two trajectories, one can solve for the optimal correspondence between them by:

$$\gamma^* = argmin_{\gamma \in \Gamma}((\int_0^1 |h_{\alpha_1}(t) - h_{\alpha_2}(\gamma(t))\sqrt{\dot{\gamma}(t)}|^2 dt)^{\frac{1}{2}}). \tag{14.4}$$

Again, the norm also depends on c. Every h_α is associated with an equivalent class defined as the set: $[h_\alpha] = \{(h_\alpha \circ \gamma)\sqrt{\dot{\gamma}} \mid \gamma \in \Gamma\}$. The distance between two orbits, is given by

$$d_h([h_{\alpha_1}], [h_{\alpha_2}]) = inf_{\gamma \in \Gamma}(\int_0^1 |h_{\alpha_1}(t) - h_{\alpha_2}(\gamma(t))\sqrt{\dot{\gamma}(t)}|^2 dt)^{\frac{1}{2}}. \tag{14.5}$$

It can be shown that the distance d_h defined in Eq. (14.5) is a proper metric [39], and it will be used for comparing trajectories of functions or tensors.

14.4 Joint Framework of Analyzing Shapes and Trajectories

In Sect. 14.3, we have described the general elastic analysis of trajectory $\alpha \in \mathcal{M}$. For analyzing trajectories of functions or tensors for special case, $\mathcal{M}$ can be the functional space or the space of SPD matrices. We use $\mathcal{P}$ in general. Since we are interested in functions or tensors that are defined along curves, it is only meaningful to find a common correspondence under a unified framework for joint analysis of shapes and trajectories of functions or tensors. To this end, we propose a joint metric for comparison and registration.

The original data is the integration of 3D curves and trajectories of BOLD signals or tensors. Let the pair (β, ω) denote a curve and a trajectory of BOLD signals or tensors associated with it. $\mathcal{M}$ denote the set of all such trajectories, the underlying spaces becomes the product space of $\mathbb{R}^3 \times \mathcal{P}$. Let (q, h_α) denote the representations of SRVF and TSRVF. To compare two pairs (β_1, ω_1) and (β_2, ω_2), we define a metric on the product space, which is a weighted sum of $d_\mathcal{S}$ and d_h on $\mathbb{R}^3 \times \mathbb{L}^2$, given by

$$d((\beta_1, \omega_1), (\beta_2, \omega_2)) = \phi_1 d_\mathcal{S}([q_1], [q_2]) + \phi_2 d_h([h_{\alpha_1}], [h_{\alpha_2}]). \tag{14.6}$$

ϕ_1 and ϕ_2 denote the weights of metrics $d_\mathcal{S}$ in Eq. (14.3) and d_h in Eq. (14.5) respectively. And this is also the new metric for joint parametrization of shapes and trajectories. The choice of different weights gives a great flexibility. Larger values of weights will imply higher importance of the information in comparison. Since $d_\mathcal{S}$ and d_h are proper metrics on each individual space, it can be shown that this distance d is also a proper metric on the product space $\mathbb{R}^3 \times \mathbb{L}^2$. It satisfies symmetry, positive definiteness and triangle inequality. Since the distance is a proper metric, it can be further used to compute statistical summaries, such as mean and covariance, and perform multiple registration.

When two pairs (β_1, ω_1) and (β_2, ω_2) are compared using the metric d, optimal correspondence between them are also solved. Rather than finding the optimal registration under $d_\mathcal{S}$ and d_h separately, this needs to be performed under the joint metric d in Eq. (14.6). The optimal common correspondence, denoted as γ^*, is given by:

$$\begin{aligned}\gamma^* = argmin_{\gamma^* \in \Gamma} (\phi_1 d_\mathcal{S}(q_1, O^*(q_2 \circ \gamma)\sqrt{\dot{\gamma}}) \\ + \phi_2 d_h(h_{\alpha_1}, (h_{\alpha_2} \circ \gamma)\sqrt{\dot{\gamma}})).\end{aligned} \tag{14.7}$$

14.4.1 Trajectories of Functions

In case the multi-modality data includes both curve features and trajectories of functions, we need a comprehensive framework to compare and register such data. Let f be a real-valued function with the domain [0, 1], and $\mathcal{P}$ is the set of all such functions. In a pairwise alignment problem, the goal is to align any two functions f_1 and f_2. A majority of past methods uses the standard $\mathbb{L}^2$ metric. However, this alignment is neither symmetric nor positive definite. To address this and other related problems, we adopt a mathematical expression for representing a function introduced in [37, 41]. This function, $q_f : [0, 1] \rightarrow \mathbb{R}$, is called the square-root

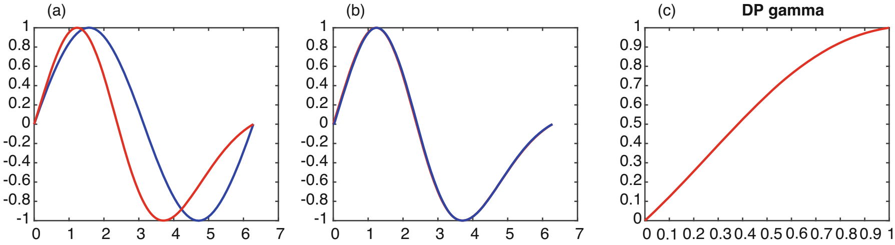


Fig. 14.4 (**a**) Functions before registration; (**b**) functions after registration; (**c**) re-parameterization function for these two functions

slope function or SRSF of f, and is defined as $q_f(t) = sign(\dot{f}(t))\sqrt{|\dot{f}(t)|}$. There is a one-to-one correspondence between functions and their SRSFs. It can be shown that if the function f is absolutely continuous, then the resulting SRSF is square-integrable. Thus we define $\mathbb{L}^2([0, 1], \mathbb{R})$, or simply $\mathbb{L}^2$, to be the set of all SRSFs. That is, for all functions $f \in \mathcal{P}$, we use SRSF to make all functions from functional space $\mathcal{P}$ to $\mathbb{L}^2$. Then we register $q_f(t)$ on different fibers. The benefit of using SRSF is the same as using SRVF for curves. For more details and applications, please refer to [29, 42, 43]. Figure 14.4 shows two simulated functions before registration, after registration and re-parameterization function for these two functions. We can see these two simulated functions have different peaks and valleys. After registration, they are same.

A trajectory of functions now can be expressed as a trajectory of SRSFs. Now, α in Sect. 14.2 becomes a smooth trajectory of SRSFs on $\mathbb{L}^2$. We now apply the joint framework of analyzing shapes and trajectories of functions in this study. First, we take the example of stimulus-relevant pathways to show an application of multi-modality data. The corpus callosum is the largest collection of fiber bundles and it connects the left and the right brain hemisphere. Some regions of corpus callosum can be affected by pathologies such as multiple sclerosis [11] and schizophrenia [13], which motivates the study of segmenting different regions of corpus callosum. In the example we consider full brain MRI data were acquired from three healthy right-handed adult volunteers.

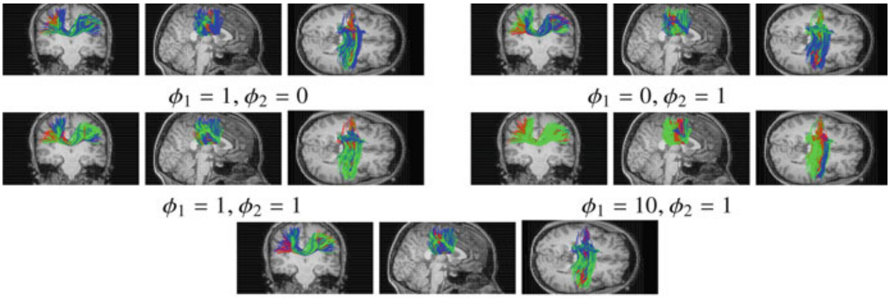
White matter fibers are tracked and extracted across corpus callosum on each subject. The numbers of fibers are 661, 771, and 520 respectively. Every fiber consists of 100 points. We have imposed two different stimuli including motor and touch, denoted as MR and TR, and measured BOLD signals under three scenarios, including resting state (RS), MR and TR. Prior to administration of the stimuli, each subject is in a resting state. TR represents the sensory stimulus, which is imposed in a block format on each subject. It begins with 30 s of right palm stimulations by continuous brushing followed by 30 s of no stimulation, and so on [46]. MR represents the movement of the right hand in the same way as TR. There are 145 points on each BOLD signal.

Methodology Given fiber tracts along with BOLD signals transmitted along them, we will compare results of clustering in three cases:

- Consider white matter fibers as 3D continuous open curves, and cluster them based on the metric introduced in Eq. (14.3).
- Consider BOLD signals as trajectory on $\mathbb{L}^2$, and cluster them based on the metric introduced in Eq. (14.5).
- Consider fiber tracts along with BOLD signals together, and cluster them based on the joint metric in Eq. (14.6). We have chosen optimal weights to achieve the best result.

In addition, we also compare with the method of functional principal component analysis (FPCA). Principal component analysis (PCA) plays a very important role in analyzing and modeling functional data. There are some other approaches using FPCA to estimate regularized or sparse empirical basis functions and compute their corresponding scores [4, 18, 38]. Here, we reduce the dimension of functions to 10, and augment 10 principal scores to fiber positions. Now we can consider fibers as open curves in $\mathbb{R}^{13}$ and apply elastic shape analysis to cluster them using Eq. (14.3).

Choice of Weights The choice of weights ϕ_1 and ϕ_2 in Eq. (14.6) will lead to different clustering results. Figure 14.5 displays outcomes under TR for one subject. For each combination of weights, we display results from 3 different views. Three clusters are obtained by structure of corpus callosum, denoted by green, red and blue colors. When $\phi_1 = 1, \phi_2 = 0$, it's equivalent as clustering fibers by considering them as 3D continuous open curves based on the distance in Eq. (14.3). When $\phi_1 = 0, \phi_2 = 1$, it's equivalent as clustering fibers according to BOLD signals using the distance in Eq. (14.5). It is clear that clustering results are quite different for different combinations of weights. The precentral gyrus (also known as the motor strip) controls the voluntary movements of skeletal muscles and the postcentral gyrus is a prominent gyrus in the lateral parietal lobe of the human brain. It is the location of the primary somatosensory cortex, the main sensory receptive area for the sense of touch. Here, we apply the K-means algorithm to cluster them based on



$\phi_1 = 1, \phi_2 = 0$

$\phi_1 = 0, \phi_2 = 1$

$\phi_1 = 1, \phi_2 = 1$

$\phi_1 = 10, \phi_2 = 1$

$\phi_1 = 1, \phi_2 = 10$

Fig. 14.5 Clustering results of different combinations of weights ϕ_1 and ϕ_2 under TR for one subject

the distance matrix computed by Eq. (14.6). By the anatomical structure of brain, we take $K = 3$ in the K-means clustering method. According to the anatomical structure of brain and distribution of regions, the underlying fact is that clusters with green and red colors should be active to touch stimuli (TR), while clusters with red and blue colors should be active to motor stimuli (MR). The fact is verified once we find proper values of weights in the following.

By evaluating these clustering results, we want to seek the optimal combination of weight coefficients ϕ_1 and ϕ_2 that can yield more meaningful clusters based on the structure of corpus callosum. BOLD signals are characterized in both the time and frequency domains. In the time domain, temporal correlations between the sensory stimuli and stimuli evoked BOLD signals from identified white matter tracts are analyzed. In the frequency domain, power spectra of the BOLD signals are computed for mean function of each cluster, and the magnitudes of frequency corresponding to the fundamental frequency of the periodic stimuli are determined, yielding three magnitude maps of stimulus frequency respectively for the acquired three clusters. We select the optimal weights of ϕ_1 and ϕ_2 by comparing correlations, temporal variations and power spectra between stimuli and BOLD signals. Since the BOLD signals are influenced by many factors during transmission along brain fibers, it is hard to measure such influences in the transmission process. Thus, for each point on each fiber, we calculate correlation between the BOLD signal and the stimulus, compute the mean correlation on each point of all fibers. Finally, for each cluster, we select the maximum of these mean correlations for comparison. Taking subject 1 for instance, a summary of these maximum correlations for each stimulus state for different combinations of weight coefficients ϕ_1 and ϕ_2, is shown in Table 14.1. According to the anatomical structure of corpus callosum, stimulus

Table 14.1 Comparison of correlations between BOLD signals and stimuli under different combinations of ϕ_1 and ϕ_2

Different states	ϕ_1	ϕ_2	Green cluster	Red cluster	Blue cluster
MR	1	1	0.348	0.377	0.454
MR	10	1	0.339	0.361	0.431
MR	1	10	0.355	**0.490**	**0.437**
MR	1	0	0.467	0.369	0.296
MR	0	1	0.355	0.490	0.437
TR	1	1	0.472	0.255	0.281
TR	10	1	0.375	0.212	0.271
TR	1	10	**0.474**	**0.467**	0.276
TR	1	0	0.465	0.253	0.274
TR	0	1	0.276	0.467	0.474
RS	1	1	0.171	0.195	0.194
RS	10	1	0.189	0.192	0.171
RS	1	10	0.193	0.180	0.168
RS	1	0	0.109	0.124	0.176
RS	0	1	0.008	0.040	0.040

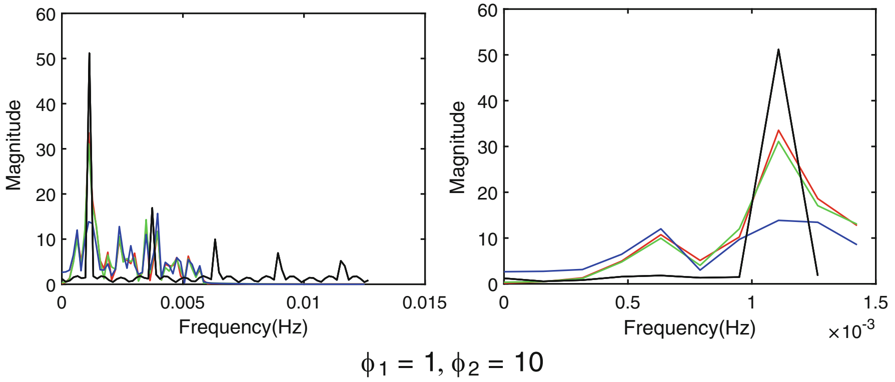
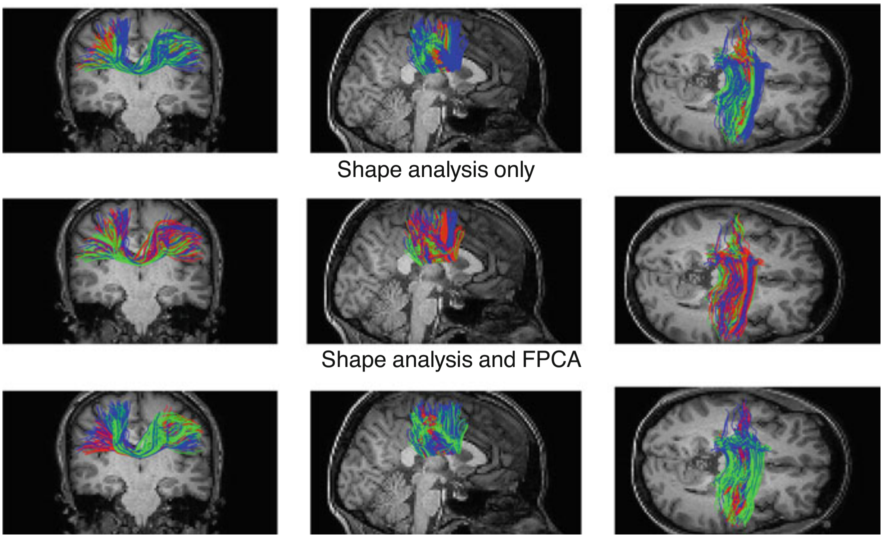


Fig. 14.6 Left: power spectra of TR stimulus and BOLD signals in each cluster for $\phi_1 = 1, \phi_2 = 10$; Right: Enlarged view near the first peak

MR should activate red and blue clusters, stimulus TR should activate green and red clusters. Through this table, we can see correlations for stimuli MR and TR are much larger then that under the resting state, which indicates that BOLD signals exhibit greater correlations with stimuli than in the resting state. When $\phi_1 = 1$, $\phi_2 = 10$, stimulus MR exhibits higher correlations in red and blue clusters, and stimulus TR exhibits higher correlations in green and red clusters. Such higher correlations are highlighted in bold values for the activated clusters in stimuli MR and TR of Table 14.1. These cluster results are consistent with the anatomical structure of corpus callosum. Results of clusters on another two subjects lead to the same conclusion as expected.

Also, the performance of clustering can be evaluated by checking periodicity of BOLD signals in each cluster. These periodic variations are reflected by their much greater magnitudes at the stimulus frequency. We use the fast Fourier transform (FFT) to sample a signal over a period of time (or space) and divides it into its frequency components. Figure 14.6 shows power spectra of stimuli and BOLD signals in clusters using FFT for $\phi_1 = 1, \phi_2 = 10$. When $\phi_1 = 1, \phi_2 = 10$, the magnitudes of clusters with red and green color are much higher than others. This agrees with the fact that the stimulus TR evokes cluster with red and green colors. We choose $\phi_1 = 1, \phi_2 = 10$ for our problem.

Comparison with Different Methods We compare our method with two other methods. One is shape analysis of fibers alone by considering them as 3D open curves as in [48, 49]. The second one is joint analysis of shapes and functions by FPCA, as discussed before. Taking TR for one subject as an example, the clusters based on three different methods are shown in Fig. 14.7. The clusters obtained by our method are better in terms of consistency with the structure of corpus callosum than clustering results by shape analysis and shape analysis with PCA.



Shape analysis only

Shape analysis and FPCA

Our method

Fig. 14.7 Comparison of clusters based on different methods

Table 14.2 Comparison of different methods under MR and TR

	Green cluster	Red cluster	Blue cluster
Different clusters under MR			
Shape analysis [48, 49]	0.280	0.370	0.264
Shape analysis and FPCA	0.281	0.309	0.276
Our method	0.256	**0.308**	**0.389**
Different clusters under TR			
Shape analysis [48, 49]	0.271	0.405	0.283
Shape analysis and FPCA	0.361	0.251	0.391
Our method	**0.422**	**0.309**	0.235

We compute correlations for all 3 subjects under MR, TR and the resting state. Table 14.2 reports the averages of correlations for MR and TR. Based on the anatomical structure of corpus callosum, stimulus MR evokes red and blue clusters, stimulus TR evokes red and green clusters. Therefore, stimulus MR is expected to have higher correlations in red and blue clusters than green cluster, stimulus TR is expected to have higher correlations in red and green clusters than blue cluster.

For stimulus MR, red and blue clusters have higher correlations than green cluster by our method. For stimulus TR, green and red clusters have higher correlations than blue cluster by our method. These higher correlations are highlighted in bold values in Table 14.2, which match the ground truth only with our method. Based on the correlations, we have clustered fibers correctly by our method as expected according to the structure of corpus callosum.

Finally, the mean correlations are converted to Fisher's Z-scores for statistical testing. The two-tailed, paired students' t-tests are used to evaluate differences between green cluster versus red and blue clusters for MR, differences between green and red clusters versus blue cluster for TR. In both cases, the p-values are less than 0.0001, which indicates the clustering results by our method are very significant.

14.4.2 Trajectories of Tensors

In case the multi-modality data includes both the curve features and trajectories of tensors, we also need a comprehensive framework to compare and register them jointly. There has been a great interest in statistical analysis of SPD matrices using the Riemannian metric in tensor space (like [10, 12, 32, 33]). Let $\mathcal{P}(n)$ be the space of $n \times n$ SPD matrices. And, $\tilde{\mathcal{P}}(n) = \{P|P \in \mathcal{P}(n)\ and\ det(P) = 1\}$. For the diffusion tensor, we take $n = 3$. The identity matrix $I_{3\times3}$ is chosen as reference point c in 2.2. $\mathcal{M} = \{\alpha : [0, 1] \to \tilde{\mathcal{P}}(n)|\alpha$ is smooth$\}$. Because of $det(\tilde{P}) > 0$, for any $\tilde{P} \in \tilde{\mathcal{P}}(n)$, we can write $\tilde{P} = (P, \frac{1}{n}log(det(\tilde{P})))$, in which $\tilde{\mathcal{P}}(n)$ can be regarded as product space of $\mathcal{P}(n)$ and $\mathbb{R}$. Let V be a tangent vector to $\mathcal{P}(n)$ at $\tilde{P}$, we can denote $\tilde{V}$ as $\tilde{V} = (V, v)$, where V is a tangent vector of $\tilde{\mathcal{P}}(n)$ at P. The parallel transport of V from P to $I_{3\times3}$ is $P^{-1}V$, the parallel transport of v is still itself. We visualize each SPD matrix as an ellipsoid. The left figure of Fig. 14.8 shows two simulated trajectories of tensors (1st and 2nd row) and registration of the first trajectory with second trajectory (3rd row). The right figure of Fig. 14.8 shows re-parameterization function for these two trajectories of tensors. We can see the two simulated trajectories have very different pattern. After registration, the registered trajectory have similar pattern with second trajectory.

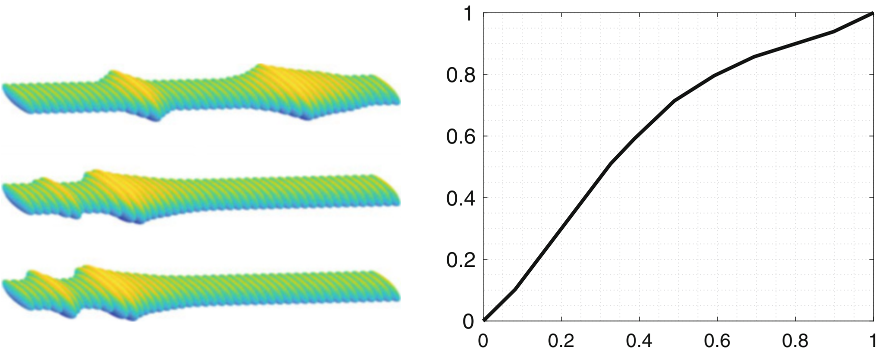


Fig. 14.8 Left: Registration of simulated trajectories of tensors; Right: Re-parameterization function

Statistical Summary of Trajectories One of advantage of this comprehensive framework is that one can calculate an average of several fibers both incorporating fibers' physical features and diffusion tensors. More specifically, we consider fibers as 3D open curves and trajectories of tensors. Then, we use this template for registering multiple trajectories. We use Karcher mean to define and calculate average trajectories. Given a set of sample trajectories $\alpha_1, \ldots, \alpha_n$ on $\mathcal{M}$, their Karcher mean is defined by: $\mu_h = argmin_\alpha \sum_1^n d_s([\alpha], [\alpha_i])^2$. The algorithm to compute the Karcher mean both incorporating fibers' physical features and trajectories of tensors is given as follows:

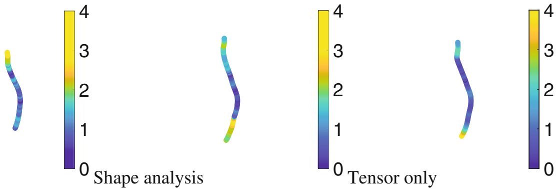
Algorithm: Karcher Mean of Multiple Trajectories

```
1. Initialization step: Select $\mu_f$ and $\mu_h$ to be the initialized mean of fibers and mean of trajectories of tensors.
2. Align each fiber $\beta_i, i = 1, \ldots, n$ to $\mu_f$ and align each trajectory $h_i, i = 1, \ldots, n$ to $\mu_h$ by finding an optimal $\gamma^*$. That is solve for $\gamma^*$ using the DP algorithm in Eq. (14.7) and set $\tilde{\beta_i} = \beta_i \circ \gamma_i^*, \tilde{\alpha}_i = \alpha_i \circ \gamma_i^*$.
3. Compute TSRVFs of the warped trajectories, $h_{\tilde{\alpha}_i}$, and calculate the average of them according to: $\bar{h}(t) = \frac{1}{n} \sum_1^n h_{\tilde{\alpha}_i}(t)$. Compute SRVFs of the warped fibers, $q_{\tilde{\beta}_i}$, and calculate the average of them according to $\bar{q}(t) = \frac{1}{n} \sum_1^n q_{\tilde{\beta}_i}(t)$.
4. Compute $E = \phi_1 d_{\mathcal{S}}(\mu_f, q_{\tilde{\beta}_i}) + \phi_2 d_h(\mu_h, h_{\tilde{\alpha}_i})$ and check for convergence. If not converged, update $\mu_f$ with $\bar{q}(t)$ and $\mu_h$ with $\bar{h}(t)$, then return to step 2.
```

Besides the Karcher mean, one would also like to quantify the amount by which each curve deviates from the mean and to analyze the most variation of curves from mean. This can be done by computing the covariance matrix and modes of variation. Since the trajectory space is a nonlinear manifold, we use the tangent space $T_c(\mathcal{M})$ of $\mathcal{M}$ at the mean μ_h, which is a vector space, to perform statistical analysis and find statistical summaries. We find $h_{\tilde{\alpha}_i}$, the element of $[h_i]$ that has the shortest geodesic distance to μ_h. Then, we find the shooting vector from μ_h to $h_{\tilde{\alpha}_i}$ using the inverse exponential map, denoted as $v_i \in T_c(\mathcal{M})$. One can compute a covariance matrix of all shooting vectors from μ_h to $h_{\tilde{\alpha}_i}$ by $K = \frac{1}{n-1} \sum_{i=1}^n v_i v_i^T$.

We apply our method on projection pathway data. The data includes 12 fibers, which are denoted as $X_i, i = 1, \ldots, 12$. It is shown in the second figure of Fig. 14.2. The corresponding trajectories of SPD matrices, denoted as $\alpha_i, i = 1, \ldots, 12$, shown in the third figure of Fig. 14.2. We extract these 12 fibers from 12 subjects, they are the representative projection pathways passing through internal capsule. In this study, we try three methods to calculate mean of fibers with different colors. The variances of fibers are also shown along the mean fibers. First, we consider 12 fibers as open curves, we compute a mean fiber using method in [48, 49]. In this method, we consider fibers as 3D open curves. The mean is shown in first figure of Fig. 14.9. Second, we compute a mean tensor-based path using method in [40], denoted as μ_h, then for each trajectory α, find the registration γ_i^* as in Eq. (14.4). In this method, we consider fibers as trajectories of tensors only. The

Fig. 14.9 Comparison of mean and variance based on different methods



Shape analysis Tensor only Our method

Table 14.3 Comparison of variance by different methods

Different methods	Variance
Shape analysis [48, 49]	115.45
Tensor only [40]	124.77
Our method	**108.71**

mean and variance are shown in second figure of Fig. 14.9. Third, we compute a mean by considering fibers as 3D open curves and trajectories of tensors. We use the algorithm in Sect. 14.4.2 to compute the mean fiber. We choose $\phi_1 = 1, \phi_2 = 5$ by trying different pairs of weight coefficients and comparing their variances. The mean and variance are shown in third figure of Fig. 14.9. There are not much difference of mean fibers by these three different methods. However, the mean fiber has the smallest variance by analyzing fibers as open curves and trajectories of tensors jointly, see Table 14.3. This means our comprehensive Riemannian framework is more effective in summarizing multi-modality data.

14.5 Conclusion

In this chapter, we have presented a comprehensive Riemannian framework for analyzing, clustering and summarizing multi-modality data. The elastic representation on a product space provides a proper metric for registration, clustering and summarization. We use our framework in two applications, which consider white matter fibers associated with BOLD signals or tensors. We have presented a proper metric, which is a weighted average of distances on the product space of shapes and trajectories of functions or tensors. The metric allows for joint comparison and registration of fibers associated with BOLD signals and tensors. Our framework has correctly identified clusters with different stimuli, which agree with the underlying truth from the brain structure. Also, our framework is very effective in summarizing projection pathways. The proposed framework is flexible such that it can be generalized to other applications where multi-modality data exist.

References

1. Alexander, A.L., Lee, J.E., Lazar, M., Boudos, R., DuBray, M.B., Oakes, T.R., Miller, J.N., Lu, J., Jeong, E.K., McMahon, W.M., Bigler, E.D., Lainhart, J.E.: Diffusion tensor imaging of the corpus callosum in autism. NeuroImage **34**(1), 61–73 (2007)
2. Batchelor, P.G., Calamante, F., Tournier, J.-D., Atkinson, D., Hill, D.L.G., Connelly, A.: Quantification of the shape of fiber tracts. Magn. Reson. Med. **55**(4), 894–903 (2006)
3. Bertsekas, D.P.: Dynamic Programming and Optimal Control. 3rd and 4th edn. Athena Scientific, Nashua (2011)
4. Besse, P.C., Cardot, H.: Approximation spline de la prévision d'un processus fonctionnel autorégressif d'ordre 1. Can. J. Stat. **24**(4), 467–487 (1996)
5. Corouge, I., Fletcher, P.T., Joshi, S., Gouttard, S., Gerig, G.: Fiber tract-oriented statistics for quantitative diffusion tensor MRI analysis. Med. Image Anal. **10**(5), 786–798 (2006)
6. Cousineau, M., Jodoin, P.M., Garyfallidis, E., Côté, M.A., Morency, F.C., Rozanski, V., Grand'Maison, M., Bedell, B.J., Descoteaux, M.: A test-retest study on Parkinson's PPMI dataset yields statistically significant white matter fascicles. Neuroimage Clin. **16**, 222–233 (2017)
7. de Reus, M.A., van den Heuvel, M.P.: The parcellation-based connectome: limitations and extensions. NeuroImage **80**, 397–404 (2013). Mapping the Connectome
8. Ding, Z., Gore, J.C., Anderson, A.W.: Classification and quantification of neuronal fiber pathways using diffusion tensor MRI. Magn. Reson. Med. **49**(4), 716–721 (2003)
9. Ding, Z., Huang, Y., Bailey, S.K., Gao, Y., Cutting, L.E., Rogers, B.P., Newton, A.T., Gore, J.C.: Detection of synchronous brain activity in white matter tracts at rest and under functional loading. In: Proceedings of the National Academy of Sciences (2018)
10. Dryden, I.L., Koloydenko, A., Zhou, D.: Non-euclidean statistics for covariance matrices, with applications to diffusion tensor imaging. Ann. Appl. Stat. **3**(3), 1102–1123 (2009)
11. Evangelou, N., Konz, D., Esiri, M.M., Smith, S., Palace, J., Matthews, P.M.: Regional axonal loss in the corpus callosum correlates with cerebral white matter lesion volume and distribution in multiple sclerosis. Brain **123**, 1845–1849 (2000)
12. Fletcher, P.T., Joshi, S.: Riemannian geometry for the statistical analysis of diffusion tensor data. Signal Proc. **87**(2), 250–262 (2007). Tensor Signal Processing
13. Foong, J., Maier, M., Clark, C.A., Barker, G.J., Miller, D.H., Ron, M.A.: Neuropathological abnormalities of the corpus callosum in schizophrenia: a diffusion tensor imaging study. J. Neurol. Neurosurg. Psychiatry **68**(2), 242–244 (2000)
14. Garyfallidis, E., Côté, M.A., Rheault, F., Sidhu, J., Hau, J., Petit, L., Fortin, D., Cunanne, S., Descoteaux, M.: Recognition of white matter bundles using local and global streamline-based registration and clustering. NeuroImage **170**, 283–295 (2018). Segmenting the Brain
15. Goodlett, C.B., Fletcher, P.T., Gilmore, J.H., Gerig, G.: Group analysis of DTI fiber tract statistics with application to neurodevelopment. NeuroImage **45**, S133–S142 (2008)
16. Guevara, M., Román, C., Houenou, J., Duclap, D., Poupon, C., Mangin, J.F., Guevara, P.: Reproducibility of superficial white matter tracts using diffusion-weighted imaging tractography. NeuroImage **147**, 703–725 (2017)
17. Jin, Y., Shi, Y., Zhan, L., Gutman, B.A., de Zubicaray, G.I., McMahon, K.L., Wright, M.J., Toga, A.W., Thompson, P.M.: Automatic clustering of white matter fibers in brain diffusion MRI with an application to genetics. NeuroImage **100**, 75–90 (2014)
18. Johnstone, I.M., Lu, A.Y., Nadler, B., Witten, D.M., Hastie, T., Tibshirani, R., Ramsay, J.O.: On consistency and sparsity for principal components analysis in high dimensions. J. Am. Stat. Assoc. **104**(486), 682–703 (2009)
19. Kurtek, S., Srivastava, A., Klassen, E., Ding, Z.: Statistical modeling of curves using shapes and related features. J. Am. Stat. Assoc. **107**(499), 1152–1165 (2012)
20. Lee, J.E., Chung, M.K., Lazar, M., DuBray, M.B., Kim, J., Bigler, E.D., Lainhart, J.E., Alexander, A.L.: A study of diffusion tensor imaging by tissue-specific, smoothing-compensated voxel-based analysis. NeuroImage **44**(3), 870–883 (2009)

21. Leemans, A., Sijbers, J., De Backer, S., Vandervliet, E., Parizel, P.: Multiscale white matter fiber tract coregistration: a new feature-based approach to align diffusion tensor data. Magn. Reson. Med. **55**(6), 1414–1423 (2006)
22. Maddah, M., Grimson, W.E.L., Warfield, S.K., Wells, W.M.: A unified framework for clustering and quantitative analysis of white matter fiber tracts. J. Med. Image Anal. **12**(2), 191–202 (2008)
23. Mani, M., Kurtek, S., Barillot, C., Srivastava, A.: A comprehensive riemannian framework for the analysis of white matter fiber tracts. In: 2010 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, pp. 1101–1104 (2010)
24. Marron, J.S., Ramsay, J.O., Sangalli, L.M., Srivastava, A.: Functional data analysis of amplitude and phase variation. Stat. Sci. **30**(4), 468–484 (2015)
25. Mio, W., Srivastava, A., Joshi, S.: On shape of plane elastic curves. Int. J. Comput. Vis. **73**(3), 307–324 (2007)
26. O'Donnell, L.J., Westin, C.F.: Automatic tractography segmentation using a high-dimensional white matter atlas. IEEE Trans. Med. Imaging **26**, 1562–1575 (2007)
27. O'Donnell, L.J., Golby, A.J., Westin, C.-F.: Fiber clustering versus the parcellation-based connectome. NeuroImage **80**, 283–289 (2013). Mapping the Connectome
28. Olivetti, E., Berto, G., Gori, P., Sharmin, N., Avesani, P.: Comparison of distances for supervised segmentation of white matter tractography. In: 2017 International Workshop on Pattern Recognition in Neuroimaging (PRNI), pp. 1–4 (2017)
29. Ramírez, J., Orini, M., Tucker, J.D., Pueyo, E., Laguna, P.: Variability of ventricular repolarization dispersion quantified by time-warping the morphology of the T-waves. IEEE Trans. Biomed. Eng. **64**(7), 1619–1630 (2017)
30. Ramsay, J., Silverman, B.W.: Functional Data Analysis. Springer in Statistics. Springer, Berlin (2010)
31. Ramsay, J.O., Hooker, G., Graves, S.: Functional Data Analysis with R and MATLAB, 1st edn. Springer, Berlin (2009)
32. Schwartzman, A.: Random ellipsoids and false discovery rates: statistics for diffusion tensor imaging data. Ph.D. Thesis, Stanford University (2006)
33. Schwartzman, A., Mascarenhas, W.F., Taylor, J.E., Inference for eigenvalues and eigenvectors of gaussian symmetric matrices. Ann. Statist. **36**(6), 2886–2919 (2008)
34. Schwarz, C.G., Reid, R.I., Gunter, J.L., Senjem, M.L., Przybelski, S.A., Zuk, S.M., Whitwell, J.L., Vemuri, P., Josephs, K.A., Kantarci, K., Thompson, P.M., Petersen, R.C., Jack, C.R.: Improved DTI registration allows voxel-based analysis that outperforms tract-based spatial statistics. NeuroImage **94**, 65–78 (2014)
35. Srivastava, A., Joshi, S.H., Mio, W., Liu, X.: Statistical shape analysis: clustering, learning, and testing. IEEE Trans. Pattern Anal. Mach. Intell. **27**(4), 590–602 (2005)
36. Srivastava, A., Klassen, E., Joshi, S.H., Jermyn, I.H.: Shape analysis of elastic curves in euclidean spaces. IEEE Trans. Pattern Anal. Mach. Intell. **33**(7), 1415–1428 (2011)
37. Srivastava, A., Kurtek, S., Wu, W., Klassen, E., Marron, J.S.: Registration of functional data using fisher-rao metric. arXiv:1103.3817 (2011)
38. Staniswalis, J.G., Lee, J.J.: Nonparametric regression analysis of longitudinal data. J. Am. Stat. Assoc. **93**(444), 1403–1418 (1998)
39. Su, J., Kurtek, S., Klassen, E., Srivastava, A.: Statistical analysis of trajectories on riemannian manifolds: bird migration, hurricane tracking and video surveillance. Ann. Appl. Stat. **8**(1), 530–552 (2014)
40. Su, J., Srivastava, A., Souza, F.D.M.D., Sarkar, S.: Rate-invariant analysis of trajectories on riemannian manifolds with application in visual speech recognition. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 620–627 (2014)
41. Tucker, J.D., Wu, W., Srivastava, A.: Generative models for functional data using phase and amplitude separation. Comput. Stat. Data Anal. **61**, 50–66 (2013)
42. Tucker, J.D., Wu, W., Srivastava, A.: Analysis of proteomics data: phase amplitude separation using an extended fisher-rao metric. Electron. J. Statist. **8**(2), 1724–1733 (2014)

43. Tucker, J.D., Lewis, J.R., Srivastava, A.: Elastic functional principal component regression. Stat. Anal. Data Mining ASA Data Sci. J. **12**(2), 101–115 (2019)
44. Wang, D., Luo, Y., Mok, V.C.T., Chu, W.C.W., Shi, L.: Tractography atlas-based spatial statistics: statistical analysis of diffusion tensor image along fiber pathways. NeuroImage **125**, 301–310 (2016)
45. Wassermann, D., Deriche, R.: Simultaneous manifold learning and clustering: grouping white matter fiber tracts using a volumetric white matter atlas. In: MICCAI 2008 Workshop – Manifolds in Medical Imaging: Metrics, Learning and Beyond (2008)
46. Wu, X., Yang, Z., Bailey, S.K., Zhou, J., Cutting, L.E., Gore, J.C., Ding, Z.: Functional connectivity and activity of white matter in somatosensory pathways under tactile stimulations. NeuroImage **152**, 371–380 (2017)
47. Zhang, S., Correia, S., Laidlaw, D.: Identifying white-matter fiber bundles in DTI data using an automated proximity-based fiber-clustering method. IEEE Trans. Vis. Comput. Graph. **14**, 1044–1053 (2008)
48. Zhang, Z., Descoteaux, M., Zhang, J., Girard, G., Chamberland, M., Dunson, D., Srivastava, A., Zhu, H.: Mapping population-based structural connectomes. NeuroImage **172**, 130–145 (2018)
49. Zhang, Z., Descoteaux, M., Dunson, D.: Nonparametric bayes models of fiber curves connecting brain regions. J. Am. Stat. Assoc. **12**, 1–23 (2019)
50. Zhu, H., Kong, L., Li, R., Styner, M., Gerig, G., Lin, W., Gilmore, J.H.: FADTTS: functional analysis of diffusion tensor tract statistics. NeuroImage **56**(3), 1412–1425 (2011)

Chapter 15
Geometric Metrics for Topological Representations

Anirudh Som, Karthikeyan Natesan Ramamurthy, and Pavan Turaga

Contents

Abstract In this chapter, we present an overview of recent techniques from the emerging area of topological data analysis (TDA), with a focus on machine-learning applications. TDA methods are concerned with measuring shape-related properties of point-clouds and functions, in a manner that is invariant to topological transformations. With a careful design of topological descriptors, these methods can result in a variety of limited, yet practically useful, invariant representations. The generality of this approach results in a flexible design choice for practitioners interested in developing invariant representations from diverse data sources such as image, shapes, and time-series data. We present a survey of topological representations and metrics on those representations, discuss their relative pros and cons, and illustrate their impact on a few application areas of recent interest.

A. Som (✉) · P. Turaga
School of Electrical, Computer and Energy Engineering, School of Arts, Media and Engineering,
Arizona State University, Tempe, AZ, USA
e-mail: asom2@asu.edu; pturaga@asu.edu

K. N. Ramamurthy
IBM Research, Yorktown Heights, NY, USA
e-mail: knatesa@us.ibm.com

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_15

15.1 Introduction

Questions surrounding geometry have evoked interest for over several millennia. Topology on the other hand, is relatively new and has been studied for just a few centuries in mathematics. As said by Galileo Galilei, *"To understand the universe one must first learn its language, which is mathematics, with its characters being different geometric shapes like lines, triangles and circles"*. Topological Data Analysis (TDA) is a collection of mathematical tools that aim to study the invariant properties of shape or underlying structure in data. Real-world data can often be represented as a point cloud, i.e., a discrete set of points lying in a high-dimensional space. Identification and use of suitable feature representations that can both preserve intrinsic information and reduce complexity of handling high-dimensional data is key to several applications including machine-learning, and other data-driven applications. In this chapter, we survey recent developments in the field of topological data analysis, with a specific focus towards applications in machine-learning (ML). ML applications require the generation of descriptors, or representations, that encode relevant invariant properties of a given point-cloud. This is then followed by choices of metrics over the representations, which leads to downstream fusion with standard machine-learning tools.

One of the core unsolved problems in ML algorithms is the characterization of invariance. Invariance refers to the ability of a representation to be unaffected by nuisance factors, such as illumination variations for face recognition, pose variations for object recognition, or rate-variations for video analysis. Provably invariant representations have been very hard to find, especially in a manner that also results in discriminative capabilities. One category of approaches involves ad-hoc choices of features or metrics between features that offer some invariance to specific factors (c.f. [14]). However, this approach suffers from a lack of generalizable solutions. The second approach involves increasing the size of training data by collecting samples that capture different possible variations in the data, allowing the learning algorithm to implicitly marginalize out the different variations. This can be achieved by simple data augmentation [106]. Yet, the latter approach does not offer any theoretical insight, and it is known that contemporary deep-learning methods are quite brittle to unexpected changes in factors like illumination and pose. Based on recent work in the field, and including our own, we feel that topological data analysis methods may help in creating a third category of approaches for enforcing practically useful invariances, while being fusible with existing ML approaches.

Rooted in algebraic topology, Persistent Homology (PH) offers a simple way to characterize the intrinsic structure and shape of data [29, 48, 61]. It does so by finding the number of k-dimensional holes when we connect nearby discrete data points. An easier way to describe PH is by comparing it to humans trying to identify constellation patterns by connecting neighboring stars in the sky [60]. PH employs a multi-scale filtration process and produces a series of nested simplicial complexes. We will describe what a simplicial complex is in the next section. By sweeping the scale parameter over a range, one can encode the structural information of the data by capturing the duration of existence of different topological invariants such as connected components, cycles, voids, higher-dimensional holes, level-sets and monotonic regions of functions defined on the data [29, 49]. Often topological invariants of interest live longer in these multi-scale simplicial complexes. The lifespan of these invariants is directly related to the geometric properties of interest.

Although the formal beginnings of topology is already a few centuries old, dating back to Leonhard Euler, algebraic topology has seen a revival in the past two decades with the advent of computational tools and software like JavaPlex [4], Perseus [94], DIPHA [12], jHoles [18], GUDHI [90], Ripser [11], PHAT [13], R-TDA [52], Scikit-TDA [114], *etc*. This has caused a spike in interest to use TDA as a complementary analysis tool to traditional data analysis and machine learning algorithms. TDA has been successfully implemented in various applications like general data analysis [29, 86, 96, 109, 128], image analysis [8, 37, 45, 54, 62, 67, 88, 97, 104], shape analysis [19, 66, 83, 119, 137, 139], time-series analysis [7, 89, 115, 119, 124, 129, 138], computer vision [7, 53, 119, 126], computational biology [27, 44, 98], bioinformatics [74], materials science [91], medical imaging [38, 79, 80], sphere packing [111], language and text analysis [65, 140], drug design [24–26, 95, 133], deep-learning model selection and analysis [23, 25, 45, 55, 56, 70, 107, 110], sensor networks [3, 41, 42, 57, 131], financial econometrics [63, 64] and invariance learning [119]. However, three main challenges exist for effectively combining PH and ML, namely—(1) topological representations of data; (2) TDA-based distance metrics; (3) TDA-based feature representations. A lot of progress has been made on all three fronts, with the literature scattered across different research areas [59, 93, 103, 130]. In this chapter we will briefly go over the various topological feature representations and their associated distance metrics. The rest of the chapter is outlined as follows: In Sect. 15.2 we will go over necessary theoretical background and definitions. Section 15.3 provides details of various topological feature representations. Section 15.4 describes the different metrics defined to compare topological features. Section 15.5 goes over some of the application areas mentioned earlier in more detail and Sect. 15.6 concludes the chapter.

15.2 Background and Definitions

In this section we will briefly go over some of the history and a few important definitions that will help us both appreciate and better understand the underlying complexities involved in topology. A convex polyhedron is the intersection of finitely many closed half-spaces. A half-space is either of the two parts when a hyperplane divides an affine space. The 5 convex regular polyhedrons known to exist in three dimensional spaces are the *tetrahedron, cube, octahedron, dodecahedron* and *icosahedron*, also known as the *5 Platonic solids*, named after the Greek philosopher Plato. He theorized that the natural elements were constructed from them. In addition, Euclid gave a complete description of the Platonic solids in the XIII Books of the Elements [69]. An interesting fact to note is that the face vector (#vertices, #edges, #faces) of the octahedron (6, 12, 8) is reverse of that of the cube (8, 12, 6). Similarly, the face vector of the dodecahedron (20, 30, 12) is reverse of the icosahedron (12, 30, 20). However, the face vector of the tetrahedron is a palindrome (4, 6, 4). A pattern that can be observed for all 5 Platonic solids is the alternating sum of the face numbers is always equal to 2, i.e., #vertices − #edges + #faces = 2. Leonhard Euler discovered this relationship and is widely considered as the starting point in the field of topology. The relation is referred to as the Euler characteristic of the polyhedron and is a global statement, without depending on the precise geometric shape. It has taken more than a century to show Euler's original observation as a special case and to prove when the relation holds [78]. This generalization is due to Henri Poincaré, which is why the more general result is referred to as the Euler-Poincaré formula. It relates the alternating sums of face numbers and Betti numbers, where f_i is the number of i-dimensional faces, and β_i is the ith Betti number. β_i is defined as the rank of the ith homology group.

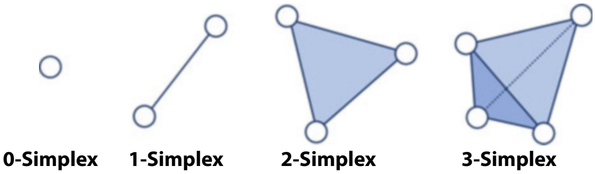
$$\sum_{i\geq 0}(-1)^i f_i = \sum_{i\geq 0}(-1)^i \beta_i \tag{15.1}$$

Despite having existed for a few hundred years, the recent revival and gain in popularity of algebraic topology is greatly attributed to the development of various software packages [4, 11–13, 18, 52, 90, 94, 114]. Most of these packages are well documented and offer simple tutorials making it easy for beginners to try out the software. However, it is important to know the definitions of some of the underlying steps that go into capturing different topological invariants from the data being analyzed. Many definitions and examples below are inspired by and adapted from [140]. We discuss only geometric realizations, but simplicial persistent homology discussed below is applicable to abstract settings also [68] is an excellent reference for further reading.

Definition 15.1 ([140]) A ***p-simplex*** is the convex hull of $p + 1$ affinely independent points $x_0, x_1, \ldots, x_p \in \mathbb{R}^d$. It can be denoted as $\sigma = \text{conv}\{x_0, \ldots, x_p\}$.

The $p + 1$ points are said to be affinely independent if the p vectors $x_i - x_0$, with $i = 1, \ldots, p$ are linearly independent. Simplices can be treated as the building blocks of discrete spaces. The convex hull formed by these points is simply the solid polyhedron. In the point cloud space, the points or vertices represent 0-simplices, edges represent 1-simplices, triangles represent 2-simplices and a tetrahedron represents a 3-simplex. These are illustrated in Fig. 15.1. A p-simplex is also referred to as a pth order hole or as a topological feature in the H_p homology group.

Fig. 15.1 Illustration of p-simplices, with $p = 0, 1, 2, 3$



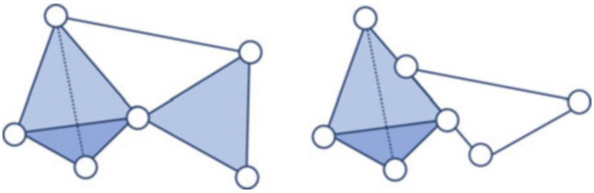


Fig. 15.2 Example of simplicial complex (left) and not a simplicial complex (right)

Definition 15.2 ([140]) A ***face*** of a p-simplex σ is the convex hull of a subset of the $p + 1$ vertices.

For example, the tetrahedron shown in Fig. 15.1 has 4 triangle faces, 6 edge faces and 4 vertex faces. Similarly, a triangle has 3 edge faces and 3 vertex faces. Finally, an edge has just 2 vertex faces.

Definition 15.3 ([140]) Given a set of points $x \in X$, the ***simplicial complex*** of this point set can be denoted by $K = (X, \Sigma)$, where Σ is a family of non-empty subsets of X, and each subset $\sigma \in \Sigma$ is a simplex.

In a simplicial complex K, if τ is a face of σ, then $\tau \in \Sigma$. It is also important to note that both $\sigma, \sigma' \in \Sigma$, which implies that their intersection is either empty or a face of both σ and σ'. This forces the simplices to be either glued together along whole faces or be separate. An example of what constitutes a simplicial complex is shown in Fig. 15.2. In TDA we use simplicial complexes to construct and study shapes from point cloud data.

Definition 15.4 ([140]) A ***p-chain*** is a subset of p-simplices in a simplicial complex.

As an example, let us consider a tetrahedron as the target simplicial complex. It has four triangle faces. A 2-chain for a tetrahedron is a subset of these four triangles, bringing the total number of distinct 2-chains to 2^4. Similarly, we can construct 2^6 distinct 1-chains using the six edges of a tetrahedron. A p-chain does not have to be connected, in spite of having the term *chain* in it.

Definition 15.5 ([140]) A ***p-chain group*** C_p is a set of p-chains in a simplicial complex along with a group operation (addition).

The addition of p-chains gives us another p-chain with the duplicate p-simplices cancelling out. See Fig. 15.3 for an example.

Definition 15.6 ([140]) The ***boundary*** ∂_p of a p-simplex is the set of $(p - 1)$-simplices faces. For example, a tetrahedron's boundary consists of the set of 4 triangle faces. A triangle's boundary is its three edges, and finally the boundary of an edge is its two vertices. The boundary of a p-chain is the XOR or mod-2 addition of the boundaries of its simplices.

Definition 15.7 ([140]) A ***p-cycle*** is a p-chain with empty boundary. Figure 15.3 illustrates both the boundary operator and the notion of cycle.

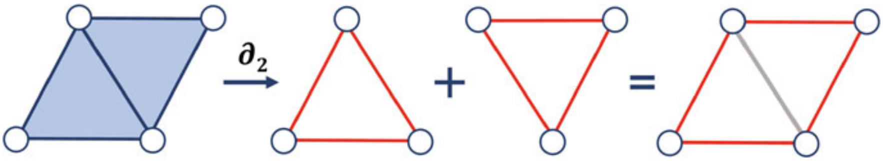


Fig. 15.3 Example of the boundary operator ∂_2 acting on the 2-chain (collection of two triangles) to create a 1-chain (collection of 4 edges). The addition operation of the two 1-chains leads to cancellation of the common 1-simplex. Applying ∂_1 on the resulting 1-chain results in 0, and hence this 1-chain is also a 1-cycle

With the definitions out of the way, let us now look at the process of constructing simplicial complexes and summarizing the topology of data. Consider a point cloud $x_0, x_1, \ldots, x_n \in \mathbb{R}^d$. We can construct a simplicial complex by identifying any subset of $p+1$ points that are close enough, such that we add a p-simplex σ, where the points serve as vertices to the complex. An easy way to do this is by constructing a ***Vietoris-Rips complex*** [143]. At scale ϵ, the Vietoris-Rips complex can be defined as $\mathrm{VR}(\epsilon) = \{\sigma \mid \mathrm{diam}(\sigma) \leq \epsilon\}$, with $\mathrm{diam}(\sigma)$ being the largest distance between any two points in the simplex σ. Increasing the scale ϵ produces a sequence of increasing simplicial complexes, i.e., $\mathrm{VR}(\epsilon_1) \subseteq \mathrm{VR}(\epsilon_2) \subseteq \cdots \subseteq \mathrm{VR}(\epsilon_m)$. This process is referred to as ***filtration***. Persistent homology keeps a track of how the pth homology holes change as ϵ changes and summarizes this information using a persistence diagram or persistence barcode plot. In this section we briefly discuss the barcode representation and will explain both persistence diagrams and persistence barcodes in more detail in Sect. 15.3. We provide an example adapted from [140]. Consider six points positioned at (0, 0), (0, 1), (2, 0), (2, 1), (5, 0), (5, 1) in a two-dimensional (2D) Cartesian coordinate axis as shown in Fig. 15.4. Varying scale ϵ causes the appearance and disappearance of H_0 and H_1 homology holes. For instance, at $\epsilon = 0$ there are six disconnected vertices, making $\beta_0 = 6$. Three edges are formed at $\epsilon = 1$, reducing β_0 to 3. Two more edges are formed at $\epsilon = 2$, which sets $\beta_0 = 2$. The points become fully connected at $\epsilon = 3$, and β_0 becomes 1. With respect to H_1 homology, we observe the first hole form at $\epsilon = 2$. However, this hole is short-lived as it disappears at $\epsilon = \sqrt{5} = 2.236$. A second hole is formed at $\epsilon = 3$ and disappears at $\epsilon = \sqrt{10} = 3.162$. The above information can be best summaried using a persistence barcode. The persistence barcodes for H_0 and H_1 homology groups is also shown in Fig. 15.4. Each bar in the barcode represents the birth-death of each hole in the H_p homology group. Just like the Vietoris-Rips complex, other types of complexes also exist, such as Čech complex, Alpha complex, Clique complex, Cubical complex, and Morse-Smale complex [103]. In the next section we will look at persistence diagrams, persistence barcodes and other topological representations in more detail.

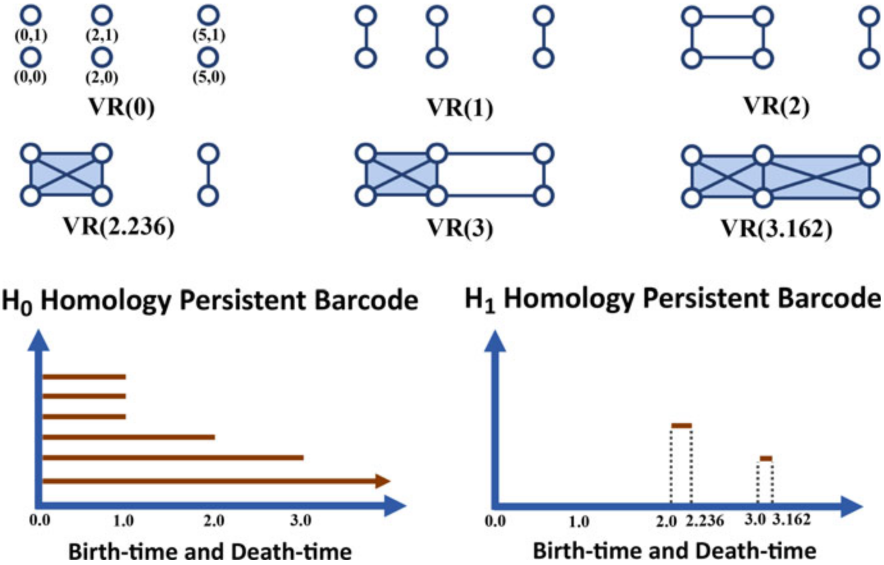

(0,1)
(2,1)
(5,1)
(0,0)
(2,0)
(5,0)
VR(0)
VR(1)
VR(2)
VR(2.236)
VR(3)
VR(3.162)
H₀ Homology Persistent Barcode
H₁ Homology Persistent Barcode
0.0
1.0
2.0
3.0
Birth-time and Death-time
0.0
1.0
2.0
2.236
3.0
3.162
Birth-time and Death-time

Fig. 15.4 An example of the filtration process of a Vietoris-Rips simplicial complex

15.3 Topological Feature Representations

From an application perspective, persistent homology (PH) is the most popular tool in topological data analysis (TDA). It offers a useful multi-scale summary of different topological invariants that exist in the data space. This information is represented using a ***Persistence Diagram*** (PD) [39] which is a set of points on a two-dimensional (2D) Cartesian plane. The two axis in this plane represent the *birth-time* (BT), i.e., the filtration value or scale at which a topological invariant is formed; and *death-time* (DT), the scale at which the topological invariant ceases to exist. The DT is always greater than the BT. This results in utilizing just the top half plane of the PD. The persistence or life-time (LT) of a point is the absolute difference between the DT and BT. For point j in the PD we will refer to the BT, DT, LT as b_j, d_j, l_j respectively.

Points in a PD can also be represented using a set of bars, with the length of each bar reflecting the LT of the point, i.e., $[l_j] = [b_j, d_j]$. This representation is called a ***Persistence Barcode*** (PB) [61]. An example of a PD and its PB is shown in Fig. 15.5. In a PD only half of the 2D plane is utilized. To fully use the entire 2D surface one can employ a rotation function $\mathcal{R}(b_j, d_j) = (b_j, d_j - b_j) = (b_j, l_j)$. Now, the new set of axis represent BT and LT respectively. Since it is a multi-set of points, it is not possible to directly use this representation in ML algorithms that use fixed-size features and operate in the Euclidean space. This has resulted in various topological representations being proposed to better understand the information captured using

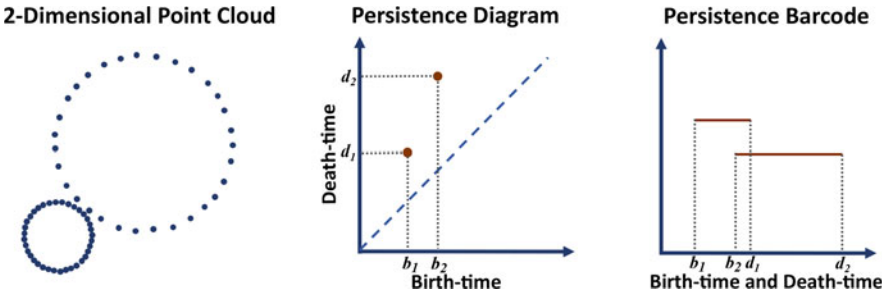


Fig. 15.5 H_1-Homology persistence diagram and persistence barcode for a 2D point cloud. In the persistence diagram, point (b_1, d_1) represents the smaller circle and point (b_2, d_2) represents the larger circle

PDs/PBs and that can be used along with ML tools [5–7, 21, 25, 27, 28, 41, 73, 119, 120, 134]. In the remainder of this section we will briefly go over the different topological representations.

The ***Persistent Betti Number*** (PBN) is defined as the summation of all k-dimensional holes in the PD and is defined in Eq. (15.2) [50]. It transforms the 2D points in the PD to a 1D function that is not continuous. Here, $\mathcal{X}_{[b_j,d_j]}$ is a step function, i.e., it equals 1 where there is a point and 0 otherwise.

$$f_{\mathrm{PBN}}(x) = \sum_j \mathcal{X}_{[b_j,d_j]}(x) \tag{15.2}$$

Kelin Xia proposed the ***Persistent Betti Function*** (PBF) defined in Eq. (15.3) [134]. It is a 1D continuous function and there is a strict one-to-one correlation between PDs and PBFs. The weight variable w_j needs to be suitable set and σ is the resolution parameter.

$$f_{\mathrm{PBF}}(x) = \sum_j w_j \exp\left(- \frac{(x - (\frac{b_j+d_j}{2}))}{\sigma(d_j - b_j)} \right)^2 \tag{15.3}$$

Peter Bubenik proposed the ***Persistence Landscape*** (PL) feature in [21]. PLs are stable, invertible functional representations of PDs. A PL lies in the Banach space and is a sequence of envelope functions defined on the points in the PD. These functions are ordered based on their importance. PLs were primarily motivated to derive a unique mean representation for a set of PDs which is comparatively difficult to do using other techniques such as Fréchet means [92]. However, their practical utility has been limited since they provide decreasing importance to secondary and tertiary features in PDs that are usually useful in terms of discriminating between data from different classes. For a PL, a piece-wise linear function can be defined on each point in the PD as shown below. The PL can be defined using a sequence of functions $\lambda_m : \mathbb{R} \to [0, \infty],\ m = 1, 2, 3, \ldots$ where $\lambda_m(x)$ is the mth largest value of $f_{\mathrm{PL}}(x)$. It is set to zero if the mth largest value does not exist.

$$f_{\mathrm{PL}}(x) = \begin{cases} 0 & \text{if } x \notin (b_j, d_j); \\ x - b_j & \text{if } x \in (b_j, \frac{b_j+d_j}{2}]; \\ -x + d_j & \text{if } x \in [\frac{b_j+d_j}{2}, d_j). \end{cases} \tag{15.4}$$

The ***Persistence Surface*** (PS) is defined in Eq. (15.5) [5]. It is a weighted sum of Gaussian functions, with each function centered at each point in the PD.

$$\rho(x, y) = \sum_j w(x, y, t_1, t_2)\, \phi(x, y, b_j, d_j) \tag{15.5}$$

Here, $\phi(.)$ is a differentiable probability distribution function and is defined as $\phi(x, y, b_j, d_j) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x-b_j)^2+(y-(d_j-b_j))^2}{2\sigma^2}\right)$. A simple choice of weighting function depends on the death-time. To weight the points of higher persistence more heavily, non-decreasing functions like sigmoid functions are a natural choice. The weight function with constants t_1, t_2 is defined as

$$w(x, y, t_1, t_2) = \begin{cases} 0 & \text{if } y \le t_1; \\ \frac{y-t_1}{t_2-t_1} & \text{if } t_1 < y < t_2; \\ 1 & \text{if } y \ge t_2. \end{cases} \tag{15.6}$$

We can discretize the continuous PS function by fitting a Cartesian grid on top of it. Integrating the PS over each grid gives us a ***Persistence Image*** (PI) [5]. The ***Persistent Entropy*** (PE) function is proposed to quantify the disorder of a system and is defined as $f_{\mathrm{PE}} = \sum_j -p_j \ln(p_j)$, where $p_j = \frac{d_j - b_j}{\sum_j (d_j - b_j)}$ [112].

One can also collect different statistical measurements from a PD and use it as a feature representation. Examples of such measurements include maxima, minima, variance, and summation, of the BT, DT, and LT. Cang et al. used 13 such different measurements to characterize the topological structural information [27]. One can also consider doing algebraic combinations or using tropical functions of BT, DT and LT [6, 73]. ***Binning*** approaches have gained more popularity as one can construct well-structured features that can easily be used as input to ML algorithms. For instance, binning with respect to PBN and PBF can be done by collecting values at grid points, i.e., the set $\{f(x_i) \mid i = 0, 1, \ldots, n;\ k = 0, 1, 2\}$, n corresponds to the grid number and k is the homology group [25, 28]. The same binning approach can be adopted for PLs as well. However, it needs to be repeated m times and thus the m set of $\{\lambda_m(x_i) \mid i = 0, 1, \ldots, n\}$ values are used as features [22]. In the case of PIs, we first rotate the PD so that the 2D Cartesian coordinate axis are BT and LT respectively. Next, we compute the PI for the rotated PD and discretize it into a $n \times n$ grid. We can evaluate the values along each grid which will result in feature vector containing total of n^2 elements. The distribution functions of BTs, DTs and PLs can be discretized and used as feature vectors. For each interval $[x_i, x_{i+1}]$, we can count the numbers of the k dimensional BTs, DTs, PLs located in this range and denote them as $N^i_{\text{BT}}, N^i_{\text{DT}}, N^i_{\text{PL}}$, respectively. These sets of counts $\{(N^i_{\text{BT}}, N^i_{\text{DT}}, N^i_{\text{PL}}) \mid i = 0, 1, \ldots, n; k = 0, 1, 2\}$ can be assembled as a feature vector. It should be noticed that normally for β_0 (rank of H_0 homology group), the BTs are 0, thus DTs are usually equal to PLs. So only the set of $\{N^i_{\text{PL}} \mid i = 0, 1, \ldots, n\}$ is used instead [25, 28].

Just like bagging, ***Persistent Codebooks*** are also popular for getting fixed-size feature representations by using different clustering or bagging methods over the points in the PD [19, 142]. For instance the ***Persistent Bag-of-Words*** (P-BoW) method matches each point in a rotated PD $\mathcal{R}(b_{i,m}, d_{i,m})$ to a precomputed dictionary/codebook to get a feature vector representation [142]. k-means clustering is used to cluster the points into c clusters $\text{NN}(\mathcal{R}(b_{i,m}, d_{i,m})) = i$, with $i = 1, 2, \ldots, c$ and $m = 1, 2, \ldots, s_i$. $\text{NN}(x, y) = i$ means that point (x, y) belongs to cluster i and s_i is the total number of points present in cluster i. The center of each cluster is represented as $z_i = (x_i, y_i)$. Thus the P-BoW is denoted by $f_{\text{P-BoW}} = (z_i)_{i=1,2,\ldots,c}$. One could also take the persistence information into account during clustering. This would result in a more adaptive codebook. The ***Persistent Vector of Locally Aggregated Descriptors*** (P-VLAD) captures more information than P-BoW [142]. It also employs k-means clustering. The aggregated distance between each rotated point $\mathcal{R}(b_{i,m}, d_{i,m})$ and its closest codeword z_i is defined as follows $f_{\text{P-VLAD}} = \sum\limits_{m=1,2,\ldots,s_i} (\mathcal{R}(b_{i,m}, d_{i,m}) - z_i)$. The c vectors are concatenated into a $2c$ dimensional vector.

The ***Persistent Fisher Vector*** (PFV) captures the rotated PD with a gradient vector from a probability model [142]. Let the set of Gaussian Mixture Model (GMM) parameters be represented using $\lambda_{\text{GMM}} = \{w_i, \mu_i, \Sigma_i\}$. Here, w_i, μ_i, Σ_i denote the weight, Gaussian center and covariance matrix of the ith Gaussian respectively. The likelihood that the rotated point $\mathcal{R}(b_j, d_j)$ is generated by the ith Gaussian is shown in Eq. (15.7) and the function is defined in Eq. (15.8).

$$p_i(\mathcal{R}(b_j, d_j) \mid \lambda_{\text{GMM}}) = \frac{\exp(-\frac{1}{2}(\mathcal{R}(b_j, d_j) - \mu_i)'\Sigma_i^{-1}(\mathcal{R}(b_j, d_j) - \mu_i))}{2\pi|\Sigma_i|^{\frac{1}{2}}} \tag{15.7}$$

$$f(\mathcal{R}(\text{PD}_{b,d}) \mid \lambda_{\text{GMM}}) = \sum_j \log\Big(\sum_i w_i p_i(\mathcal{R}(b_j, d_j) \mid \lambda_{\text{GMM}})\Big) \tag{15.8}$$

Another feature representation was proposed by Chevyrev et al., where the PB is first represented as a persistent path which in turn is represented as a tensor series [35].

Anirudh et al., proposed a feature representation denoted by f_{HS}, that is based on Riemannian geometry [7]. This feature is obtained by modeling PDs as 2D probability density functions (PDF) that are represented using the square-root framework on the Hilbert Sphere. The resulting space is more intuitive with closed form expressions for common operations and statistical analysis like computing geodesics, exponential maps, inverse-exponential maps and computing means. Assuming that the supports for each 2D PDF p is in $[0, 1]^2$, the Hilbert Sphere feature representation of the PD is shown in Eq. (15.9). Here, $\psi = \sqrt{p}$.

$$f_{\mathrm{HS}} = \{\psi : [0,1] \times [0,1] \to \mathbb{R}\ \forall x, y \mid \psi \geq 0;\ \text{with} \int_0^1 \int_0^1 \psi^2(x,y)\partial x \partial y = 1\} \tag{15.9}$$

Motivated from the successful use of Riemannian geometry to encode PDs, Som et al. proposed ***Perturbed Topological Signatures*** (PTS), a more robust topological descriptor where a set of PDs can be projected to a point on the Grassmann manifold [119]. We refer our readers to the following papers that provide a good introduction to the geometry, statistical analysis, and techniques for solving optimization problems on the Grassmann manifold [1, 2, 36, 47, 132]. Instead of creating more variations in the data space and then computing PDs, the authors induce variations by directly augmenting in the space of PDs. They do so by creating a set of randomly perturbed PDs from the original PD. Each perturbed PD in this set has its points randomly shifted but within a certain defined radius about the original position of the points. The extent of perturbation is constrained to ensure that the topological structure of data being analyzed is not abruptly changed. A perturbed PD is analogous to extracting the PD from data that is subjected to topological noise. Next, 2D PDFs are computed for each of the PDs in this set. Finally, the set of 2D PDFs are vectorized, stacked and then mapped to a point on the Grassmannian. Mapping to a point on the Grassmann manifold is done by applying singular value decomposition (SVD) on the stacked matrix of perturbed PDFs. Once in this space, we can use the various metrics defined for the Grassmann manifold to do basic operations and statistical analysis, just like in [7].

There is also recent interest bringing the areas of topological representation learning and deep learning closer, and explore how they can help each other. In [70], the authors propose to use PDs in deep neural network architectures using a novel input layer that performs necessary parametrizations. Som et al. also explored the use of deep learning models for computing topological representations directly from data [120]. They proposed simple convolutional architectures called ***PI-Net*** to directly learn mappings between time-series or image data and their corresponding PIs, thereby reducing the amount of time taken to compute PIs from large datasets significantly. Given a new dataset, they also discuss ways of using transfer learning to fine-tune a pre-trained model on a subset of the new dataset. This opens doors to exploring deep learning methods for computing topological features from the target datasets. In the next section we will go over the different metrics and kernel methods defined for the various topological representations described earlier.

15.4 Geometric Metrics for Representations

As mentioned earlier, a persistence diagram (PD) is a multi-set of points that lies on a 2D Cartesian plane. Its unique format poses a challenge for a finding a suitable metric to compare PDs. However, several metrics have been proposed that are suited specifically for PDs. Metrics have also been formulated for other topological representations that are functional approximations of PDs [7, 30, 34, 92, 119, 123]. In addition, various kernel functions for topological data have been proposed to replace the role of features. In this section, we will briefly go over these geometric metrics and topological kernels.

The two classical metrics used to measure the dissimilarity between PDs are the ***Bottleneck*** and ***p-Wasserstein distances*** [92, 123]. Both of these are transport metrics, and are computed by matching corresponding points in PDs. For H_k homology group, the bottleneck distance between a pair of PDs D and D' is shown in Eq. (15.10), with γ ranging over all bijections from D to D', and x_j representing the jth point.

$$d_{\mathrm{B}}(D, D') = \inf_{\gamma} \sup_{j} \|x_j - \gamma(x_j)\|_\infty \tag{15.10}$$

Here, $\|x_j - x_{j'}\|_\infty = \max\{|b_j - b_{j'}|, |d_j - d_{j'}|\}$, with $(b,\ d)$ corresponding to BT and DT respectively. The p-Wasserstein distance between two PDs D and D' is shown in Eq. (15.11), with $p > 0$.

$$d_{\mathrm{W,p}}(D, D') = \inf_{\gamma} \left[\sum_{j} ||x_j - \gamma(x_j)||_\infty^p \right]^{\frac{1}{p}} \tag{15.11}$$

Despite being principled metrics that can quantify the changes between the PDs, these metrics are computationally expensive. For example, to compare two PDs with n points each, the worst-case computational complexity is of the order of $O(n^3)$ [15]. This and the fact that the PDs are not vector space representations makes the computation of statistics in the space of PDs challenging. This has led to the emergence of other topological representations with their corresponding metrics.

The ***Sliced Wasserstein distance*** [105] between two PDs is defined as

$$d_{\mathrm{SW}}(D, D') = \frac{1}{2\pi} \int \mathcal{W}(\mu(\theta, D) + \mu_\Delta(\theta, D'), \mu(\theta, D') + \mu_\Delta(\theta, D))\, d\theta. \tag{15.12}$$

Since the points in a PD live in a restriction of the 2D Euclidean space, we can define a line $f(\theta) = \{\lambda(\cos(\theta), \sin(\theta)) \mid \lambda \in \mathbb{R}\}$ for $\theta \in [-\pi/2, \pi/2]$ in this space. Further $\pi_\theta : \mathbb{R}^2 \rightarrow f(\theta)$ is defined as the orthogonal projection of a point onto this line and the π_Δ is the orthogonal projection onto the diagonal line (i.e., $\theta = \pi/4$). We denote $\mu(\theta, D) = \sum_j \delta_{\pi_\theta(x_j)}$ and $\mu_\Delta(\theta, D) = \sum_j \delta_{\pi_\theta \circ \pi_\Delta(x_j)}$ and $\mathcal{W}$ is the generic

Kantorovich formulation of optimal transport. The main idea behind this metric is to slice the plane with lines passing through the origin, to project the measures onto these lines where $\mathcal{W}$ is computed, and to integrate those distances over all possible lines. Based on this metric, Carriere et al. proposed the ***Sliced Wasserstein kernel*** [31].

Reininghaus et al., proposed the ***Persistence Scale Space Kernel*** (PSSK) [108] that is defined as

$$\mathcal{K}_{\text{PSSK}}(D, D', \sigma) = \frac{1}{8\pi\sigma} \sum_{x_j, x_{j'}} \exp\left(\frac{-\|x_j - x_{j'}\|^2}{8\sigma}\right) - \exp\left(\frac{-\|x_j - \overline{x_{j'}}\|^2}{8\sigma}\right). \tag{15.13}$$

Here, $\overline{x_{j'}}$ is $x_{j'}$ mirrored at the diagonal. The proposed kernel is positive definite and is defined via an L_2-valued feature map, based on ideas from scale space theory [71]. The authors also show that the proposed kernel is Lipschitz continuous with respect to the 1-Wasserstein distance and apply it to different shape classification/retrieval and texture recognition experiments. Kwitt et al. proposed the ***Universal Persistence Scale Space Kernel*** (u-PSSK) [77], which is a modification of PSSK and is defined as,

$$\mathcal{K}_{\text{u-PSSK}}(D, D', \sigma) = \exp(\mathcal{K}_{\text{PSSK}}(D, D', \sigma)). \tag{15.14}$$

The ***Persistence Weighted Gaussian Kernel*** (PWGK) is also a positive definite kernel, proposed by Kusano et al. [76] and defined as

$$\mathcal{K}_{\text{PWGK}}(D, D', \sigma) = \sum_{x_j, x_{j'}} w_{arc}(x_j) w_{arc}(x_{j'}) \exp\left(\frac{-\|x_j - x_{j'}\|^2}{2\sigma^2}\right). \tag{15.15}$$

Here, $w_{arc}(x_j) = \arctan(C(d_j - b_j)^p)$, with parameters p and C being positive values. PWGK has the following 3 advantages over PSSK: (1) PWGK can better control the effect of persistence using parameters p, C in w_{arc}, which are independent of the bandwidth parameter σ in the Gaussian factor, while PSSK has just σ; (2) approximation by random Fourier features is applicable only in PWGK, since PSSK is not shift-invariant in total; (3) PWGK is a non-linear kernel on the reproducing kernel Hilbert space (RKHS), where as PSSK is a linear kernel.

The ***Geodesic Topological Kernel*** (GTK) is proposed by Padellini and Brutti [99] and is defined as

$$\mathcal{K}_{\text{GTK}}(D, D', \sigma) = \exp\left(\frac{1}{h} d_{\text{W},2}(D, D')^2\right) \tag{15.16}$$

where $d_{\text{W},2}$ is the 2-Wasserstein distance and $h > 0$. Similarly the ***Geodesic Laplacian Kernel*** (GLK) is defined as

$$\mathcal{K}_{\text{GLK}}(D, D', \sigma) = \exp\left(\frac{1}{h} d_{\text{W},2}(D, D')\right). \tag{15.17}$$

Unlike PSSK and PWGK, both GTK and GLK are not positive definite kernels. However, the authors show that this does not affect the performance of the kernel in a supervised learning setting, as they exploit the predictive power of the negative part of the kernels and can do better in a narrowed class of problems.

The ***Persistence Fisher Kernel*** (PFK) proposed by Le and Yamada is a positive definite kernel that preserves the geometry of the Riemannian manifold as it is built upon the Fisher information metric for PDs without approximation [81]. The PFK is defined in Eq. (15.18). Here, t_0 is a positive scalar value; d_{FIM} is the Fisher information metric; $\rho(x, y, D) = \frac{1}{Z} \sum\limits_{j} N(x, y|l_j, \sigma)$ with j ranging over all points in the PD; $Z = \int \sum\limits_{j} N(x, y|l_j, \sigma) \partial x \partial y$ and $N(x, y|l_j, \sigma)$ is the normal distribution.

$$\mathcal{K}_{\text{PFK}}(D, D') = \exp(-t_0 d_{\text{FIM}}(\rho(x, y, D), \rho(x, y, D'))) \tag{15.18}$$

Zhu et al. proposed three persistent landscape-based kernels namely: ***Global Persistent Homology Kernel*** (GPHK), ***Multi-resolution Persistent Homology Kernel*** (MPHK) and ***Stochastic Multi-resolution Persistent Homology Kernel*** (SMURPHK) [141]. However, both GPHK and MPHK do not scale well to point clouds with large number of points. SMURPHK solves the scalability issue with Monte Carlo sampling.

The PTS representation by Som et al. is a point on the Grassmann manifold [119]. This allows one to utilize the different distance metrics and Mercer kernels defined for the Grassmannian. The minimal ***geodesic distance*** ($d_{\mathbb{G}}$) between two points $\mathcal{Y}_1$ and $\mathcal{Y}_2$ on the Grassmann manifold is the length of the shortest constant speed curve that connects these points. To do this, the velocity matrix $A_{\mathcal{Y}_1, \mathcal{Y}_2}$ or the inverse exponential map needs to be calculated, with the geodesic path starting at $\mathcal{Y}_1$ and ending at $\mathcal{Y}_2$. $A_{\mathcal{Y}_1, \mathcal{Y}_2}$ can be computed using the numerical approximation method described in [85]. The geodesic distance between $\mathcal{Y}_1$ and $\mathcal{Y}_2$ is represented in Eq. (15.19). Here θ is the principal angle matrix between $\mathcal{Y}_1, \mathcal{Y}_2$ and can be computed as $\theta = \arccos(S)$, where $USV^T = \text{svd}(\mathcal{Y}_1^T \mathcal{Y}_2)$. The authors show the stability of the proposed PTS representation using the normalized geodesic distance represented by $d_{\mathbb{NG}}(\mathcal{Y}_1, \mathcal{Y}_2) = \frac{1}{D} d_{\mathbb{G}}(\mathcal{Y}_1, \mathcal{Y}_2)$, where D is the maximum possible geodesic distance on $\mathbb{G}_{p,n}$ [72, 82].

$$d_{\mathbb{G}}(\mathcal{Y}_1, \mathcal{Y}_2) = trace(A_{\mathcal{Y}_1, \mathcal{Y}_2} A_{\mathcal{Y}_1, \mathcal{Y}_2}{}^{\text{T}}) = \sqrt{trace(\theta^T \theta)} \tag{15.19}$$

The ***symmetric directional distance*** (d_{Δ}) is another popular metric to compute distances between Grassmann representations with different subspace dimension p [121, 127]. Its been used areas like computer vision [9, 10, 40, 87, 135], communications [116], and applied mathematics [46]. It is equivalent to the chordal metric [136] and is defined in Eq. (15.20). Here, k and l are subspace dimensions for the orthonormal matrices $\mathcal{Y}_1$ and $\mathcal{Y}_2$ respectively. The following papers propose methods to compute distances between subspaces of different dimensions [121, 127, 136].

$$d_{\Delta}(\mathcal{Y}_1, \mathcal{Y}_2) = \left(\max(k,l) - \sum_{i,j=1}^{k,l} ({y_{1,i}}^{\mathrm{T}} y_{2,j})^2\right)^{\frac{1}{2}} \tag{15.20}$$

15.5 Applications

In this section, we describe three application areas that have benefited from topological methods, including time-series modeling, image and shape analysis. We also compare the performance of some of the topological representations and metrics described in Sects. 15.3 and 15.4.

15.5.1 Time-Series Analysis

A lot of work has gone into modeling dynamical systems. A popular approach involves reconstructing the phase space of the dynamical system by implementing Takens' embedding theorem on a 1D time-series signal [122]. For a discrete dynamical system with a multi-dimensional phase space, the embedding or time-delay vectors are obtained by stacking time-delayed versions of the 1D signal. This can be easily expressed through Eq. (15.21).

$$\mathbf{x}(n) = [x(n), x(n+\tau), \ldots, x(n+(m-1)\tau)]^T \tag{15.21}$$

Here, x is the 1D time-series signal, m is the embedding dimension and τ is the embedding delay or delay factor. An example of reconstructing the phase space of the Lorenz attractor is shown in Fig. 15.6. Takens' embedding theorem has

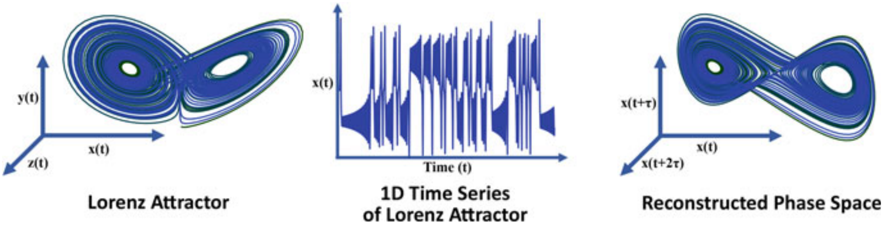


Fig. 15.6 Phase space reconstruction of the Lorenz attractor using Takens' embedding theorem. The Lorenz attractor (left) is obtained using a system of three ordinary differential equations: $x(t)$, $y(t)$, $z(t)$, with control parameters $\rho = 28$, $\sigma = 10$, $\beta = 2.667$. Takens' embedding theorem is applied on $x(t)$ (middle), with $m = 3$, $\tau = 10$ to get the reconstructed phase space (right)

been successfully employed in various applications [117–119, 125]. Skraba et al. proposed a framework that analyzes dynamical systems using persistent homology, that requires almost no prior information of the underlying structure [43]. The authors observed that the reconstructed phase space can reveal the recurrent nature of the system in the form of loops and returning paths. Persistent homology can be used to quantify these recurrent structures using persistent diagrams or Betti numbers. It is important to note that these loops need not necessarily exist in the 1D signal, thereby making the reconstructed phase space even more attractive. A periodic dynamical system exhibits Betti numbers equivalent to that of a circle, a quasi-periodic system with p periods will have Betti numbers equal to that of a p-dimensional torus. Apart from counting the number of loops in the reconstructed phase space, persistent homology also allows one to measure the periodicity of a signal which is represented by the size of the loops or holes.

Perea and Harer also used persistent homology to discover periodicity in sliding windows of noisy periodic signals [101]. Perea later extended the same idea to quasi-periodic signals and also provide details for finding the optimal time-delay, window size for sliding window embedding [100]. Berwald and Gidea used persistence diagrams constructed using Vietoris-Rips filtration to discover important transitions in genetic regulatory systems by identifying significant topological difference in consecutive time-series windows [16, 17]. Garland et al. constructed persistence diagrams using Witness complex filtrations to model the underlying topology of noisy samples in dynamical systems [58]. Chazal et al. proposed the idea of persistence-based clustering, where they showed that stable peaks possessed longer life-times [33]. The life-time of points in the persistence diagram can reflect the hierarchy of importance of the cluster centroids. Based on this other persistence-based clustering ideas have also come up in recent years [32, 102]. Emrani et al. used Betti-1 persistence barcodes for wheeze detection [51]. Sanderson et al. used TDA to capture differences between same musical notes played on different instruments [113]. They used persistent rank functions as features from a persistent diagram and observed better results than a classifier trained on fast fourier transform (FFT) features [111]. Ideas from [100, 101] were also used for identifying early signs of important transitions in financial time-series trends using persistent homology [63, 64].

15.5.2 Image Analysis

Topological methods have played a crucial role for different image-based applications, including image analysis [8, 37, 45, 54, 62, 67, 88, 97, 104], computer vision [7, 53, 119, 126], medical imaging [38, 79, 80] and so on. Li et al. used persistence diagrams together with bag-of-features representation for texture classification. They theorized that while bag-of-features capture the distribution of functional values defined over the data, PDs on the other hand capture structural properties that reflect spatial information in an invariant way. Similarly, Som et al. used Perturbed Topological Signature (PTS) features along with self-similarity matrix based representations for multi-view human activity recognition task. Lawson et al. used persistent homology for classification and quantitative evaluation of architectural features in prostate cancer histology [79, 80].

TDA methods are also being investigated as complementary representations to those afforded by deep-learning representations for image classification problems [45, 120]. Given an image, one can consider it as a point-cloud of pixels with additional information associated to each pixel. In [45], the authors suggest a mapping from each pixel—$f : I \mapsto \mathbb{R}^5$ mapping the RGB values of a given pixel at location (x, y) to a point $(\frac{r-\mu_r}{\sigma_r}, \frac{g-\mu_g}{\sigma_g}, \frac{b-\mu_b}{\sigma_b}, x-\overline{x}, y-\overline{y})$, where μ and σ represent the mean and standard deviations of individual R,G,B channels in the image, and $\overline{x}, \overline{y}$ represent the mean spatial co-ordinates. Under this mapping, it can be shown with some simple analysis that the topological properties of the resulting point cloud will be invariant to spatial transforms like affine transforms, or simple monotonic intensity transforms like gamma correction. The author proposed using a persistence barcode features as a way to extract these invariant representations. The study shows that fusion with features from deep-nets is possible, using methods like Fisher vector encoding. The performance improvements shown are significant across many different datasets like CIFAR-10, Caltech-256, and MNIST. Another interesting approach involves directly computing topological representations from data using deep learning. For example, Som et al. build simple convolution neural networks to learn mappings between images and their corresponding persistence images [120]. However, this would require us to first compute the ground-truth persistence images using conventional TDA methods. Nevertheless, the trained network offers a speed up in the computation time by about two orders of magnitude. We feel that the above approaches open a new class of image representations, with many possible design choices, beginning with how an image can be converted to a topological representation, all the way to fusion approaches.

15.5.3 Shape Analysis

Point cloud shape analysis is a topic of major current interest due to emergence of Light Detection and Ranging (LIDAR) based vision systems in autonomous vehicles. The different invariances one tries to seek include shape articulation, i.e., stretching, skewing, rotation of shape that does not alter the fundamental object class. These invariances are optimally defined in terms of topological invariants.

For 3D shape analysis we conduct an experiment on 10 random shapes selected from the SHREC 2010 dataset [84]. The dataset consists of 200 near-isometric watertight 3D shapes with articulating parts, equally divided into 10 classes. Each 3D mesh is simplified to 2000 faces. The 10 shapes used in the experiment are denoted as $\mathcal{S}_i, i = 1, 2, \ldots, 10$ and are shown in Fig. 15.7. The minimum bounding

Fig. 15.7 Sample shapes from SHREC 2010 dataset

sphere for each of these shapes has a mean radius of 54.4 with standard deviation of 3.7 centered at $(64.4, 63.4, 66.0)$ with coordinate-wise standard deviations of $(3.9, 4.1, 4.9)$ respectively. Next, we generate 100 sets of shapes, infused with topological noise. Topological noise is applied by changing the position of the vertices of the triangular mesh face, which results in changing its normal. We do this by applying a zero-mean Gaussian noise to the vertices of the original shape, with the standard deviation σ varied from 0.1 to 1 in steps of 0.1. For each shape $\mathcal{S}_i$, its 10 noisy shapes with different levels of topological noise are denoted by $\mathcal{N}_{i,1}, \ldots, \mathcal{N}_{i,10}$.

A 17-dimensional scale-invariant heat kernel signature (SIHKS) spectral descriptor function is calculated on each shape [75], and PDs are extracted for each dimension of this function resulting in 17 PDs per shape. To know more about the process of extracting PDs from the SIHKS descriptor, we refer our readers to the paper by Li et al. [83]. The 3D mesh and PD representation for 5 of the 10 shapes (shown in Fig. 15.7) and their respective noisy-variants (Gaussian noise with standard deviation 1.0) is shown in Fig. 15.8. Now we evaluate the robustness of each topological representation by trying to correctly classify shapes with different levels of topological noise. Displacement of vertices by adding varying levels of topological noise, interclass similarities and intraclass variations of the shapes make this a challenging task. A simple unbiased one-nearest-neighbor (1-NN) classifier is used to classify the topological representations of the noisy shapes in each set. The classification results are averaged over the 100 sets and tabulated in Table 15.1. We compare the performance of the following topological representations: PI [5], PL [20], PSSK [108], PWGK [76] and PTS [119]. For PTS, we set the discretization of the grid $k = 50$ and use $\sigma = 0.0004$. For PIs we chose the linear ramp weighting function, set k and σ for the Gaussian kernel function, same as the PTS feature. For PLs we use the first landscape function with 500 elements. A linear SVM classifier is used instead of the 1-NN classifier for the PSSK and PWGK methods. The classification results and the average time taken to compare topological representations is shown in Table 15.1.

We observe that PIs, PLs and PWGK take the least amount of time to compare topological features. PDs on the other hand take the most amount of time. At lower levels of topological noise, there is little difference in the overall performance of each topological feature. However, with the increase in topological noise the classification performance deteriorates drastically for PIs, PLs, PSSK and PWGK. Even PDs with the Bottleneck distance and 2-Wasserstein metric show poor results as the noise level increases. The PTS representation shows the most stability with

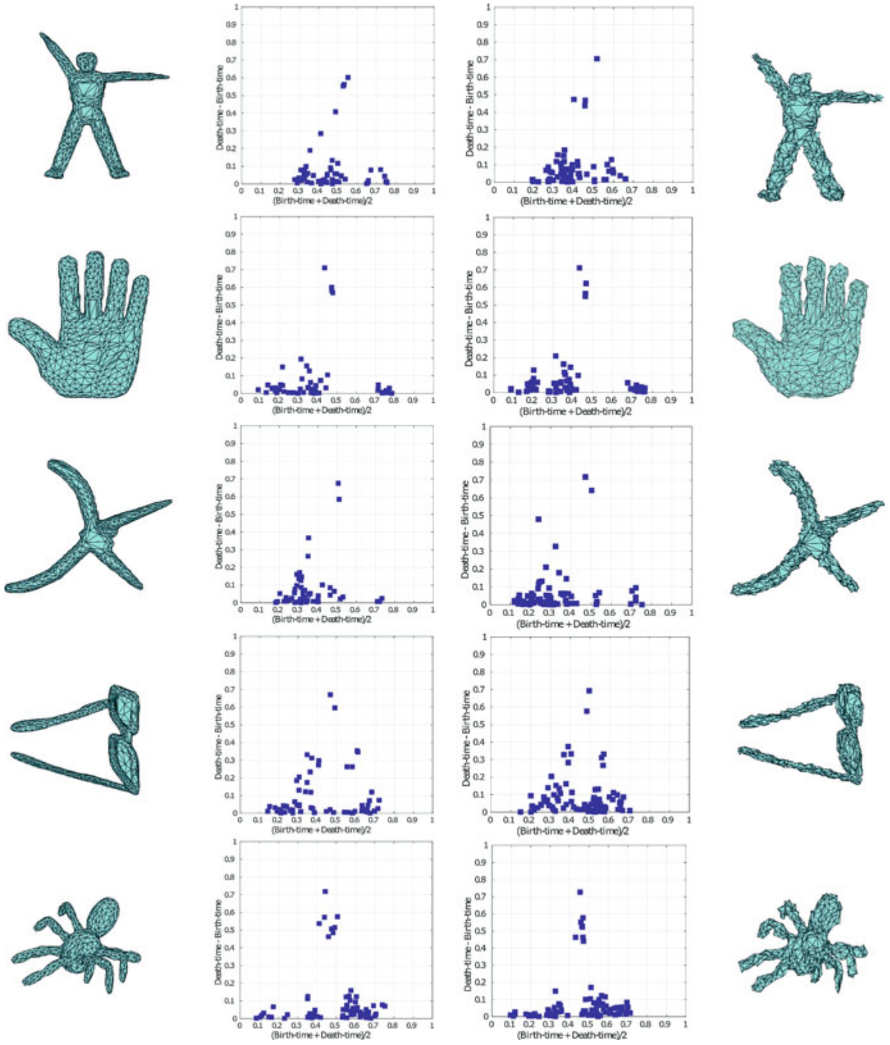


Fig. 15.8 PD representations for 5 shapes and their noisy variants. Columns 1 and 4 represent the 3D shape with triangular mesh faces; columns 2 and 3 show the corresponding ninth dimension SIHKS function-based PDs. A zero mean Gaussian noise with standard deviation 1.0 is applied on the original shapes in column 1 to get the corresponding noisy variant in column 4

respect to the applied topological noise. This is attributed to the fact that the PTS representation takes into account different possible perturbations that are artificially induced in the PD space before being mapped to a point on the Grassmann manifold. Also, both Grassmannian metrics $\boldsymbol{d}_{\mathbb{G}}$, $\boldsymbol{d}_{\boldsymbol{\Delta}}$ still observe about two orders of magnitude faster times to compare PTS representations.

Table 15.1 Average time taken to compare two topological representations and the 1-NN classification performance of different topological representations in correctly classifying 3D shapes induced with topological noise. The results are reported after averaging over 100 sets. The difficulty in classifying the shapes is proportional to the amount of topological noise added to the shape

Method	$\mathcal{N}_{i,1}$	$\mathcal{N}_{i,2}$	$\mathcal{N}_{i,3}$	$\mathcal{N}_{i,4}$	$\mathcal{N}_{i,5}$	$\mathcal{N}_{i,6}$	$\mathcal{N}_{i,7}$	$\mathcal{N}_{i,8}$	$\mathcal{N}_{i,9}$	$\mathcal{N}_{i,10}$	Average Average	Average time taken $(\mathbf{10^{-4}}$ s)
PD (**1**-*Wasserstein*)	100.00	100.00	100.00	99.90	100.00	99.80	99.60	99.00	96.60	94.40	98.93	256.00
PD (**2**-*Wasserstein*)	97.50	98.00	98.10	97.20	97.20	96.00	94.40	92.80	90.30	88.50	95.00	450.00
PD (*Bottleneck*)	99.90	99.90	99.90	99.20	99.40	98.60	97.10	96.90	94.30	92.70	97.79	36.00
PI ($\boldsymbol{L_1}$)	100.00	100.00	100.00	99.70	98.10	93.70	83.20	68.30	56.00	44.90	84.39	0.31
PI ($\boldsymbol{L_2}$)	99.90	99.50	98.60	97.40	93.10	88.50	82.90	69.70	59.40	49.90	83.89	0.26
PI ($\boldsymbol{L_\infty}$)	89.10	83.00	80.20	78.90	78.40	69.90	68.60	64.00	61.90	56.80	73.08	0.12
PL ($\boldsymbol{L_1}$)	99.20	99.70	99.00	98.50	98.50	97.30	95.90	92.30	89.10	84.50	95.40	0.74
PL ($\boldsymbol{L_2}$)	99.10	99.70	98.90	98.50	98.30	96.90	95.60	92.10	89.00	84.30	95.24	0.76
PL ($\boldsymbol{L_\infty}$)	98.90	99.60	98.80	98.40	98.30	96.50	94.80	91.70	88.70	83.80	94.95	0.09
PSSK–SVM	100.00	100.00	100.00	100.00	100.00	100.00	91.60	90.00	89.80	89.00	96.04	4.55
PWGK–SVM	100.00	100.00	100.00	100.00	100.00	99.90	99.40	95.90	87.50	73.30	95.60	0.17
PTS ($\boldsymbol{d}$)	100.00	100.00	100.00	100.00	100.00	99.90	99.80	98.80	96.80	93.60	98.89	2.30
PTS ($\boldsymbol{d_\Delta}$)	100.00	100.00	100.00	100.00	100.00	99.90	99.90	99.30	97.10	94.10	99.03	1.60

15.6 Conclusion

TDA methods are continuing to find more applications in diverse domains, as a new lens to encode *'shape'*-related information. The theoretical work of the past two decades has resulted in a variety of tools, which are being actively transitioned to many different applications. We feel that TDA methods will continue to attract interest from machine-learning practitioners, as we need newer methods to address outstanding issues in standard ML approaches. Our conjecture is that the problem of enforcing invariances in ML architectures will be one of the significant points of transitions of TDA tools to the ML field. TDA methods are also being used to throw light on how deep-learning methods learn, and generalize to other tasks. In conclusion, we feel that TDA methods are poised to advance ML techniques as well as many different applications where analysis of the shape of underlying data is important.

Acknowledgements This work was supported in part by NSF CAREER grant 1452163 and ARO grant number W911NF-17-1-0293 during the preparation of this chapter.

References

1. Absil, P.A., Mahony, R., Sepulchre, R.: Riemannian geometry of grassmann manifolds with a view on algorithmic computation. Acta Appl. Math. **80**(2), 199–220 (2004)
2. Absil, P.A., Mahony, R., Sepulchre, R.: Optimization algorithms on matrix manifolds. Princeton University Press, Princeton (2009)
3. Adams, H., Carlsson, G.: Evasion paths in mobile sensor networks. Int. J. Robot. Res. **34**(1), 90–104 (2015)
4. Adams, H., Tausz, A., Vejdemo-Johansson, M.: Javaplex: a research software package for persistent (co) homology. In: International Congress on Mathematical Software, pp. 129–136. Springer, Berlin (2014)
5. Adams, H., Emerson, T., Kirby, M., Neville, R., Peterson, C., Shipman, P., Chepushtanova, S., Hanson, E., Motta, F., Ziegelmeier, L.: Persistence images: a stable vector representation of persistent homology. J. Mach. Learn. Res. **18**(8), 1–35 (2017)
6. Adcock, A., Carlsson, E., Carlsson, G.: The ring of algebraic functions on persistence bar codes (2013). Preprint. arXiv:1304.0530
7. Anirudh, R., Venkataraman, V., Natesan Ramamurthy, K., Turaga, P.: A riemannian framework for statistical analysis of topological persistence diagrams. In: The IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 68–76 (2016)
8. Bae, W., Yoo, J., Chul Ye, J.: Beyond deep residual learning for image restoration: persistent homology-guided manifold simplification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 145–153 (2017)
9. Bagherinia, H., Manduchi, R.: A theory of color barcodes. In: IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 806–813 (2011)
10. Basri, R., Hassner, T., Zelnik-Manor, L.: Approximate nearest subspace search. IEEE Trans. Pattern Anal. Mach. Intell. **33**(2), 266–278 (2011)
11. Bauer, U.: Ripser: a lean c++ code for the computation of vietoris–rips persistence barcodes. Software available at https://github.com/Ripser/ripser (2017)

12. Bauer, U., Kerber, M., Reininghaus, J.: Distributed computation of persistent homology. In: 2014 Proceedings of the Sixteenth Workshop on Algorithm Engineering and Experiments (ALENEX), pp. 31–38. SIAM, Philadelphia (2014)
13. Bauer, U., Kerber, M., Reininghaus, J., Wagner, H.: Phat–persistent homology algorithms toolbox. In: International Congress on Mathematical Software, pp. 137–143. Springer, Berlin (2014)
14. Begelfor, E., Werman, M.: Affine invariance revisited. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 2087–2094. IEEE, Piscataway (2006)
15. Bertsekas, D.P.: A new algorithm for the assignment problem. Math. Program. **21**(1), 152–171 (1981)
16. Berwald, J., Gidea, M.: Critical transitions in a model of a genetic regulatory system (2013). Preprint. arXiv:1309.7919
17. Berwald, J., Gidea, M., Vejdemo-Johansson, M.: Automatic recognition and tagging of topologically different regimes in dynamical systems (2013). Preprint. arXiv:1312.2482
18. Binchi, J., Merelli, E., Rucco, M., Petri, G., Vaccarino, F.: jHoles: a tool for understanding biological complex networks via clique weight rank persistent homology. Electron. Notes Theor. Comput. Sci. **306**, 5–18 (2014)
19. Bonis, T., Ovsjanikov, M., Oudot, S., Chazal, F.: Persistence-based pooling for shape pose recognition. In: International Workshop on Computational Topology in Image Context, pp. 19–29. Springer, Berlin (2016)
20. Bubenik, P.: Statistical topological data analysis using persistence landscapes. J. Mach. Learn. Res. **16**(1), 77–102 (2015)
21. Bubenik, P.: The persistence landscape and some of its properties (2018). Preprint. arXiv:1810.04963
22. Bubenik, P., Dłotko, P.: A persistence landscapes toolbox for topological statistics. J. Symb. Comput. **78**, 91–114 (2017)
23. Bubenik, P., Holcomb, J.: Statistical inferences from the topology of complex networks. Technical Report, Cleveland State University Cleveland United States (2016)
24. Cang, Z., Wei, G.W.: Analysis and prediction of protein folding energy changes upon mutation by element specific persistent homology. Bioinformatics **33**(22), 3549–3557 (2017)
25. Cang, Z., Wei, G.W.: Topologynet: topology based deep convolutional and multi-task neural networks for biomolecular property predictions. PLoS Comput. Biol. **13**(7), e1005690 (2017)
26. Cang, Z., Wei, G.W.: Integration of element specific persistent homology and machine learning for protein-ligand binding affinity prediction. Int. J. Numer. Methods Biomed. Eng. **34**(2), e2914 (2018)
27. Cang, Z., Mu, L., Wu, K., Opron, K., Xia, K., Wei, G.W.: A topological approach for protein classification. Comput. Math. Biophys. **3**(1), 140–162 (2015)
28. Cang, Z., Mu, L., Wei, G.W.: Representability of algebraic topology for biomolecules in machine learning based scoring and virtual screening. PLoS Comput. Biol. **14**(1), e1005929 (2018)
29. Carlsson, G.: Topology and data. Bull. Am. Math. Soc. **46**(2), 255–308 (2009)
30. Carriere, M., Bauer, U.: On the metric distortion of embedding persistence diagrams into reproducing kernel hilbert spaces (2018). Preprint. arXiv:1806.06924
31. Carriere, M., Cuturi, M., Oudot, S.: Sliced wasserstein kernel for persistence diagrams. In: Proceedings of the 34th International Conference on Machine Learning, vol. 70, pp. 664–673. JMLR.org (2017)
32. Chang, H.W., Bacallado, S., Pande, V.S., Carlsson, G.E.: Persistent topology and metastable state in conformational dynamics. PLoS One **8**(4), e58699 (2013)
33. Chazal, F., Guibas, L.J., Oudot, S.Y., Skraba, P.: Persistence-based clustering in riemannian manifolds. J. ACM **60**(6), 41 (2013)
34. Chazal, F., Fasy, B., Lecci, F., Michel, B., Rinaldo, A., Rinaldo, A., Wasserman, L.: Robust topological inference: distance to a measure and kernel distance. J. Mach. Learn. Res. **18**(1), 5845–5884 (2017)

35. Chevyrev, I., Nanda, V., Oberhauser, H.: Persistence paths and signature features in topological data analysis. IEEE Trans. Pattern Anal. Mach. Intell. (2018)
36. Chikuse, Y.: Statistics on Special Manifolds, vol. 174. Springer, New York (2012)
37. Chung, Y.M., Lawson, A.: Persistence curves: a canonical framework for summarizing persistence diagrams (2019). Preprint. arXiv:1904.07768
38. Chung, M.K., Bubenik, P., Kim, P.T.: Persistence diagrams of cortical surface data. In: International Conference on Information Processing in Medical Imaging, pp. 386–397. Springer, Berlin (2009)
39. Cohen-Steiner, D., Edelsbrunner, H., Harer, J.: Stability of persistence diagrams. Discret. Comput. Geom. **37**(1), 103–120 (2007)
40. da Silva, N.P., Costeira, J.P.: The normalized subspace inclusion: robust clustering of motion subspaces. In: IEEE International Conference on Computer Vision (ICCV), pp. 1444–1450. IEEE, Piscataway (2009)
41. De Silva, V., Ghrist, R.: Coverage in sensor networks via persistent homology. Algebr. Geom. Topol. **7**(1), 339–358 (2007)
42. De Silva, V., Ghrist, R.: Homological sensor networks. Not. Am. Math. Soc. **54**(1), 10–17 (2007)
43. de Silva, V., Skraba, P., Vejdemo-Johansson, M.: Topological analysis of recurrent systems. In: Workshop on Algebraic Topology and Machine Learning, NIPS (2012)
44. Dey, T.K., Mandal, S.: Protein classification with improved topological data analysis. In: 18th International Workshop on Algorithms in Bioinformatics (WABI 2018). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2018)
45. Dey, T.K., Mandal, S., Varcho, W.: Improved image classification using topological persistence. In: Hullin, M., Klein, R., Schultz, T., Yao, A. (eds.) Vision, Modeling & Visualization. The Eurographics Association (2017)
46. Draper, B., Kirby, M., Marks, J., Marrinan, T., Peterson, C.: A flag representation for finite collections of subspaces of mixed dimensions. Linear Algebra Appl. **451**, 15–32 (2014)
47. Edelman, A., Arias, T.A., Smith, S.T.: The geometry of algorithms with orthogonality constraints. SIAM J. Matrix Anal. Appl. **20**(2), 303–353 (1998)
48. Edelsbrunner, H.: A Short Course in Computational Geometry and Topology. Mathematical methods. Springer, Berlin (2014)
49. Edelsbrunner, H., Harer, J.: Computational Topology: An Introduction. American Mathematical Society, Providence (2010)
50. Edelsbrunner, H., Letscher, D., Zomorodian, A.: Topological persistence and simplification. Discret. Comput. Geom. **28**(4), 511–533 (2002)
51. Emrani, S., Gentimis, T., Krim, H.: Persistent homology of delay embeddings and its application to wheeze detection. IEEE Signal Process Lett. **21**(4), 459–463 (2014)
52. Fasy, B.T., Kim, J., Lecci, F., Maria, C.: Introduction to the **R** package **TDA** (2014). Preprint. arXiv:1411.1830
53. Freedman, D., Chen, C.: Algebraic topology for computer vision. Comput. Vis. 239–268 (2009)
54. Frosini, P., Landi, C.: Persistent betti numbers for a noise tolerant shape-based approach to image retrieval. Pattern Recogn. Lett. **34**(8), 863–872 (2013)
55. Gabella, M., Afambo, N., Ebli, S., Spreemann, G.: Topology of learning in artificial neural networks (2019). Preprint. arXiv: 1902.08160
56. Gabrielsson, R.B., Carlsson, G.: Exposition and interpretation of the topology of neural networks (2018). Preprint. arXiv: 1810.03234
57. Gamble, J., Chintakunta, H., Krim, H.: Applied topology in static and dynamic sensor networks. In: 2012 International Conference on Signal Processing and Communications (SPCOM), pp. 1–5. IEEE, Piscataway (2012)
58. Garland, J., Bradley, E., Meiss, J.D.: Exploring the topology of dynamical reconstructions. Phys. D Nonlinear Phenomena **334**, 49–59 (2016)
59. Gholizadeh, S., Zadrozny, W.: A short survey of topological data analysis in time series and systems analysis (2018). Preprint. arXiv: 1809.10745

60. Gholizadeh, S., Seyeditabari, A., Zadrozny, W.: Topological signature of 19th century novelists: persistent homology in text mining. Big Data Cogn. Comput. **2**(4), 33 (2018)
61. Ghrist, R.: Barcodes: the persistent topology of data. Bull. Am. Math. Soc. **45**(1), 61–75 (2008)
62. Giansiracusa, N., Giansiracusa, R., Moon, C.: Persistent homology machine learning for fingerprint classification (2017). Preprint. arXiv: 1711.09158
63. Gidea, M.: Topological data analysis of critical transitions in financial networks. In: International Conference and School on Network Science, pp. 47–59. Springer, Berlin (2017)
64. Gidea, M., Katz, Y.: Topological data analysis of financial time series: landscapes of crashes. Phys. A Stat. Mech. Appl. **491**, 820–834 (2018)
65. Guan, H., Tang, W., Krim, H., Keiser, J., Rindos, A., Sazdanovic, R.: A topological collapse for document summarization. In: 2016 IEEE 17th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), pp. 1–5. IEEE, Piscataway (2016)
66. Guo, W., Manohar, K., Brunton, S.L., Banerjee, A.G.: Sparse-tda: sparse realization of topological data analysis for multi-way classification. IEEE Transactions on Knowledge and Data Engineering **30**(7), 1403–1408 (2018)
67. Han, Y.S., Yoo, J., Ye, J.C.: Deep residual learning for compressed sensing CT reconstruction via persistent homology analysis (2016). Preprint. arXiv: 1611.06391
68. Hatcher, A.: Algebraic Topology. Cambridge University Press, Cambridge (2005)
69. Heath, T.L., et al.: The Thirteen Books of Euclid's Elements. Courier Corporation, North Chelmsford (1956)
70. Hofer, C., Kwitt, R., Niethammer, M., Uhl, A.: Deep learning with topological signatures (2017). Preprint. arXiv: 1707.04041
71. Iijima, T.: Basic theory on the normalization of pattern (in case of typical one-dimensional pattern). Bull. Electro. Tech. Lab. **26**, 368–388 (1962)
72. Ji-guang, S.: Perturbation of angles between linear subspaces. Int. J. Comput. Math. 58–61 (1987)
73. Kališnik, S.: Tropical coordinates on the space of persistence barcodes. Found. Comput. Math. **19**(1), 101–129 (2019)
74. Kasson, P.M., Zomorodian, A., Park, S., Singhal, N., Guibas, L.J., Pande, V.S.: Persistent voids: a new structural metric for membrane fusion. Bioinformatics **23**(14), 1753–1759 (2007)
75. Kokkinos, I., Bronstein, M., Yuille, A.: Dense scale invariant descriptors for images and surfaces. Ph.D. Thesis, INRIA (2012)
76. Kusano, G., Hiraoka, Y., Fukumizu, K.: Persistence weighted gaussian kernel for topological data analysis. In: International Conference on Machine Learning (ICML), pp. 2004–2013 (2016)
77. Kwitt, R., Huber, S., Niethammer, M., Lin, W., Bauer, U.: Statistical topological data analysis – a kernel perspective. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems 28, pp. 3070–3078. Curran Associates, Inc., Red Hook (2015). http://papers.nips.cc/paper/5887-statistical-topological-data-analysis-a-kernel-perspective.pdf
78. Lakatos, I.: Proofs and Refutations: The Logic of Mathematical Discovery. Cambridge University Press, Cambridge (1976)
79. Lawson, P., Sholl, A.B., Brown, J.Q., Fasy, B.T., Wenk, C.: Persistent homology for the quantitative evaluation of architectural features in prostate cancer histology. Sci. Rep. **9**, 1–15 (2019)
80. Lawson, P., Schupbach, J., Fasy, B.T., Sheppard, J.W.: Persistent homology for the automatic classification of prostate cancer aggressiveness in histopathology images. In: Medical Imaging 2019: Digital Pathology, vol. 10956, p. 109560G. International Society for Optics and Photonics, Bellingham (2019)
81. Le, T., Yamada, M.: Persistence fisher kernel: a riemannian manifold kernel for persistence diagrams. In: Advances in Neural Information Processing Systems, pp. 10007–10018 (2018)

82. Li, C., Shi, Z., Liu, Y., Xu, B.: Grassmann manifold based shape matching and retrieval under partial occlusions. In: International Symposium on Optoelectronic Technology and Application: Image Processing and Pattern Recognition (2014)
83. Li, C., Ovsjanikov, M., Chazal, F.: Persistence-based structural recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1995–2002 (2014)
84. Lian, Z., Godil, A., Fabry, T., Furuya, T., Hermans, J., Ohbuchi, R., Shu, C., Smeets, D., Suetens, P., Vandermeulen, D., et al.: Shrec'10 track: non-rigid 3d shape retrieval. Eurographics Workshop on 3D Object Retrieval (3DOR) **10**, 101–108 (2010)
85. Liu, X., Srivastava, A., Gallivan, K.: Optimal linear representations of images for object recognition. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) (2003)
86. Liu, X., Xie, Z., Yi, D., et al.: A fast algorithm for constructing topological structure in large data. Homology Homotopy Appl. **14**(1), 221–238 (2012)
87. Luo, D., Huang, H.: Video motion segmentation using new adaptive manifold denoising model. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
88. Makarenko, N., Kalimoldayev, M., Pak, I., Yessenaliyeva, A.: Texture recognition by the methods of topological data analysis. Open Eng. **6**(1) (2016)
89. Marchese, A., Maroulas, V.: Signal classification with a point process distance on the space of persistence diagrams. Adv. Data Anal. Classif. **12**(3), 657–682 (2018)
90. Maria, C., Boissonnat, J.D., Glisse, M., Yvinec, M.: The gudhi library: Simplicial complexes and persistent homology. In: International Congress on Mathematical Software, pp. 167–174. Springer, Berlin (2014)
91. Maroulas, V., Nasrin, F., Oballe, C.: Bayesian inference for persistent homology (2019). Preprint. arXiv: 1901.02034
92. Mileyko, Y., Mukherjee, S., Harer, J.: Probability measures on the space of persistence diagrams. Inverse Prob. **27**(12), 124007 (2011)
93. Munch, E.: A user's guide to topological data analysis. J. Learn. Anal. **4**(2), 47–61 (2017)
94. Nanda, V.: Perseus: the persistent homology software. Software available at http://www.sas.upenn.edu/~vnanda/perseus (2012)
95. Nguyen, D.D., Xiao, T., Wang, M., Wei, G.W.: Rigidity strengthening: a mechanism for protein–ligand binding. J. Chem. Inf. Model. **57**(7), 1715–1721 (2017)
96. Niyogi, P., Smale, S., Weinberger, S.: A topological view of unsupervised learning from noisy data. SIAM J. Comput. **40**(3), 646–663 (2011)
97. Obayashi, I., Hiraoka, Y., Kimura, M.: Persistence diagrams with linear machine learning models. J. Appl. Comput. Topol. **1**(3–4), 421–449 (2018)
98. Pachauri, D., Hinrichs, C., Chung, M.K., Johnson, S.C., Singh, V.: Topology-based kernels with application to inference problems in alzheimer's disease. IEEE Trans. Med. Imaging **30**(10), 1760–1770 (2011)
99. Padellini, T., Brutti, P.: Supervised learning with indefinite topological kernels (2017). Preprint. arXiv: 1709.07100
100. Perea, J.A.: Persistent homology of toroidal sliding window embeddings. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6435–6439. IEEE, Piscataway (2016)
101. Perea, J.A., Harer, J.: Sliding windows and persistence: an application of topological methods to signal analysis. Found. Comput. Math. **15**(3), 799–838 (2015)
102. Pereira, C.M., de Mello, R.F.: Persistent homology for time series and spatial data clustering. Expert Syst. Appl. **42**(15–16), 6026–6038 (2015)
103. Pun, C.S., Xia, K., Lee, S.X.: Persistent-homology-based machine learning and its applications–a survey (2018). Preprint. arXiv: 1811.00252
104. Qaiser, T., Tsang, Y.W., Taniyama, D., Sakamoto, N., Nakane, K., Epstein, D., Rajpoot, N.: Fast and accurate tumor segmentation of histology images using persistent homology and deep convolutional features. Med. Image Anal. **55**, 1–14 (2019)

105. Rabin, J., Peyré, G., Delon, J., Bernot, M.: Wasserstein barycenter and its application to texture mixing. In: International Conference on Scale Space and Variational Methods in Computer Vision, pp. 435–446. Springer, Berlin (2011)
106. Rahmani, H., Mian, A., Shah, M.: Learning a deep model for human action recognition from novel viewpoints. IEEE Trans. Pattern Anal. Mach. Intell. **40**(3), 667–681 (2017)
107. Ramamurthy, K.N., Varshney, K.R., Mody, K.: Topological data analysis of decision boundaries with application to model selection (2018). Preprint. arXiv: 1805.09949
108. Reininghaus, J., Huber, S., Bauer, U., Kwitt, R.: A stable multi-scale kernel for topological machine learning. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
109. Rieck, B., Mara, H., Leitte, H.: Multivariate data analysis using persistence-based filtering and topological signatures. IEEE Trans. Vis. Comput. Graph. **18**(12), 2382–2391 (2012)
110. Rieck, B., Togninalli, M., Bock, C., Moor, M., Horn, M., Gumbsch, T., Borgwardt, K.: Neural persistence: a complexity measure for deep neural networks using algebraic topology (2018). Preprint. arXiv: 1812.09764
111. Robins, V., Turner, K.: Principal component analysis of persistent homology rank functions with case studies of spatial point patterns, sphere packing and colloids. Phys. D Nonlinear Phenomena **334**, 99–117 (2016)
112. Rucco, M., Castiglione, F., Merelli, E., Pettini, M.: Characterisation of the idiotypic immune network through persistent entropy. In: Proceedings of ECCS 2014, pp. 117–128. Springer, Berlin (2016)
113. Sanderson, N., Shugerman, E., Molnar, S., Meiss, J.D., Bradley, E.: Computational topology techniques for characterizing time-series data. In: International Symposium on Intelligent Data Analysis, pp. 284–296. Springer, Berlin (2017)
114. Saul, N., Tralie, C.: Scikit-TDA: Topological data analysis for python (2019). https://doi.org/10.5281/zenodo.2533369
115. Seversky, L.M., Davis, S., Berger, M.: On time-series topological data analysis: New data and opportunities. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 59–67 (2016)
116. Sharafuddin, E., Jiang, N., Jin, Y., Zhang, Z.L.: Know your enemy, know yourself: Block-level network behavior profiling and tracking. In: IEEE Global Telecommunications Conference (GLOBECOM 2010), pp. 1–6 (2010)
117. Som, A., Krishnamurthi, N., Venkataraman, V., Turaga, P.: Attractor-shape descriptors for balance impairment assessment in parkinson's disease. In: IEEE Conference on Engineering in Medicine and Biology Society (EMBC), pp. 3096–3100 (2016)
118. Som, A., Krishnamurthi, N., Venkataraman, V., Ramamurthy, K.N., Turaga, P.: Multiscale evolution of attractor-shape descriptors for assessing parkinson's disease severity. In: IEEE Global Conference on Signal and Information Processing (GlobalSIP) (2017)
119. Som, A., Thopalli, K., Natesan Ramamurthy, K., Venkataraman, V., Shukla, A., Turaga, P.: Perturbation robust representations of topological persistence diagrams. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 617–635 (2018)
120. Som, A., Choi, H., Ramamurthy, K.N., Buman, M., Turaga, P.: PI-net: a deep learning approach to extract topological persistence images (2019). Preprint. arXiv: 1906.01769
121. Sun, X., Wang, L., Feng, J.: Further results on the subspace distance. Pattern Recogn. **40**(1), 328–329 (2007)
122. Takens, F.: Detecting strange attractors in turbulence. In: Dynamical Systems and Turbulence, Warwick 1980, pp. 366–381. Springer, Berlin (1981)
123. Turner, K., Mileyko, Y., Mukherjee, S., Harer, J.: Fréchet means for distributions of persistence diagrams. Discret. Comput. Geom. **52**(1), 44–70 (2014)
124. Umeda, Y.: Time series classification via topological data analysis. Inf. Media Technol. **12**, 228–239 (2017)
125. Venkataraman, V., Turaga, P.: Shape distributions of nonlinear dynamical systems for video-based inference. IEEE Trans. Pattern Anal. Mach. Intell. **38**(12), 2531–2543 (2016)

126. Venkataraman, V., Ramamurthy, K.N., Turaga, P.: Persistent homology of attractors for action recognition. In: IEEE International Conference on Image Processing (ICIP), pp. 4150–4154. IEEE, Piscataway (2016)
127. Wang, L., Wang, X., Feng, J.: Subspace distance analysis with application to adaptive bayesian algorithm for face recognition. Pattern Recogn. **39**(3), 456–464 (2006)
128. Wang, B., Summa, B., Pascucci, V., Vejdemo-Johansson, M.: Branching and circular features in high dimensional data. IEEE Trans. Vis. Comput. Graph. **17**(12), 1902–1911 (2011)
129. Wang, Y., Ombao, H., Chung, M.K., et al.: Persistence landscape of functional signal and its application to epileptic electroencaphalogram data. ENAR Distinguished Student Paper Award (2014)
130. Wasserman, L.: Topological data analysis. Annual Rev. Stat. Appl. **5**, 501–532 (2018)
131. Wilkerson, A.C., Moore, T.J., Swami, A., Krim, H.: Simplifying the homology of networks via strong collapses. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 5258–5262. IEEE, Piscataway (2013)
132. Wong, Y.C.: Differential geometry of grassmann manifolds. Proc. Natl. Acad. Sci. **57**(3), 589–594 (1967)
133. Wu, C., Ren, S., Wu, J., Xia, K.: Weighted (co) homology and weighted laplacian (2018). Preprint. arXiv: 1804.06990
134. Xia, K.: A quantitative structure comparison with persistent similarity (2017). Preprint. arXiv: 1707.03572
135. Yan, J., Pollefeys, M.: A general framework for motion segmentation: independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In: European Conference on Computer Vision (ECCV), pp. 94–106. Springer, Berlin (2006)
136. Ye, K., Lim, L.H.: Schubert varieties and distances between subspaces of different dimensions. SIAM J. Matrix Anal. Appl. **37**(3), 1176–1197 (2016)
137. Zeppelzauer, M., Zieliński, B., Juda, M., Seidl, M.: Topological descriptors for 3d surface analysis. In: International Workshop on Computational Topology in Image Context, pp. 77–87. Springer, Berlin (2016)
138. Zhang, Z., Song, Y., Cui, H., Wu, J., Schwartz, F., Qi, H.: Early mastitis diagnosis through topological analysis of biosignals from low-voltage alternate current electrokinetics. In: 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 542–545. IEEE, Piscataway (2015)
139. Zhou, Z., Huang, Y., Wang, L., Tan, T.: Exploring generalized shape analysis by topological representations. Pattern Recogn. Lett. **87**, 177–185 (2017)
140. Zhu, X.: Persistent homology: an introduction and a new text representation for natural language processing. In: International Joint Conference on Artificial Intelligence (2013)
141. Zhu, X., Vartanian, A., Bansal, M., Nguyen, D., Brandl, L.: Stochastic multiresolution persistent homology kernel. In: International Joint Conference on Artificial Intelligence, pp. 2449–2457 (2016)
142. Zielinski, B., Juda, M., Zeppelzauer, M.: Persistence codebooks for topological data analysis (2018). Preprint. arXiv: 1802.04852
143. Zomorodian, A.: Fast construction of the vietoris-rips complex. Comput. Graph. **34**(3), 263–271 (2010)

Chapter 16
On Geometric Invariants, Learning, and Recognition of Shapes and Forms

Gautam Pai, Mor Joseph-Rivlin, Ron Kimmel, and Nir Sochen

Contents

Abstract Extracting meaningful representations from geometric data has prime importance in the areas of computer vision, computer graphics, and image processing. Classical approaches use tools from differential geometry for modeling the problem and employed efficient and robust numerical techniques to engineer them for a particular application. Recent advances in learning methods, particularly in the areas of deep-learning and neural networks provide an alternative mechanism of extracting meaningful features and doing data-engineering. These techniques have

G. Pai (✉) · R. Kimmel
Faculty of Computer Science, Technion, Haifa, Israel
e-mail: paigautam@cs.technion.ac.il; ron@cs.technion.ac.il

M. Joseph-Rivlin
Faculty of Electrical Engineering, Technion, Haifa, Israel
e-mail: mor1joseph@campus.technion.ac.il

N. Sochen
Department of Applied Mathematics, Tel-Aviv University, Tel Aviv-Yafo, Israel
e-mail: sochen@post.tau.ac.il

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_16

proven very successful for various kinds of visual and semantic cognition tasks achieving state-of-the art results. In this chapter we explore the synergy between these two seemingly disparate computational methodologies. First, we provide a short treatise on geometric invariants of planar curves and a scheme to discover them from data in a learning framework, where the invariants are modelled using neural networks. Secondly, we also demonstrate the reverse, that is, imputing principled geometric invariants like geometric moments into standard learning architectures enables a significant boost in performance. Our goal would not only be to achieve better performance, but also to provide a geometric insight into the learning process thereby establishing strong links between the two fields.

16.1 Introduction

Geometric invariance is an important issue in various problems of geometry processing, computer vision and image processing, having both a theoretical and a computational aspect to it. In order to solve problems like shape correspondence and retrieval, it is essential to develop representations which do not change under the action of various types of transformations. The difficulty of estimating invariants is usually related to the complexity of the transform for which the invariance is desired and the numerical construction of invariant functions from discrete sampled data is typically non-trivial.

Learning methods involve the study and design of algorithms that develop functional architectures which can be trained to perform a specific task when they are endowed with an objective cost function and a sufficient amount of data. Recent advances in convolutional neural networks and deep learning have catapulted their use in solving a multitude of problems in computer vision, speech processing, pattern recognition and many other data-sciences. Their success has been largely attributed to the meaningful semantic features they generate from the data which enables them to perform tasks like recognition, classification, discrimination etc. with state-of-the-art superiority.

This chapter provides a synergy between the principled computation of numerical geometry and the learning based schemes of neural networks and deep learning. In this chapter, we aim to explore and highlight possible benefits of using learning to enable better estimation of geometric invariants as well as using principled geometric constructions to enable superior learning. By bringing these two disparate themes together we hope to advocate the research for discovery of novel methods that can combine the valuable insight and understanding provided by principled axiomatic methods in geometry with the black-box-yet superior data dependant models of deep learning. To this aim, we make two such demonstrations in this chapter. First, in Sect. 16.2 we design a setup to *learn* the curvature (specifically an invariant representation) of a curve from examples. A first course on differential geometry shows that the arc-length and curvature are fundamental invariants of a curve and provably invariant to specific class of transformations [1, 6, 7, 35]. Hence, the Euclidean arc-length and curvature are the simplest invariants to Euclidean transformations, Affine arc-length and affine curvature to affine transformations [29], and so on. We demonstrate experimentally that, by synthetically generating examples of these transformations, we can train a neural network to *learn* these invariants [24] from data. Importantly, we demonstrate that such invariants have superior numerical properties in terms of a much improved robustness to noise and sampling as compared to their axiomatic counterparts.

Second, in Sect. 16.3 we demonstrate that the input of geometric moments into neural network architectures well known for point cloud processing result in an improved shape recognition performance. Classically, geometric moments have been used to define invariant signatures for classification of rigid objects [19]. Comparing two point clouds by correlating their shape moments is a well known method in the geometry processing literature. At the other end, geometric moments are composed of high order polynomials of the points coordinates, and approximation those polynomial is not trivial to a neural networks [40]. Here, we propose to add polynomial functions of the coordinates to the network inputs, which will allow the network to learn the elements of the moments and should better capture the geometric structure of the data.

16.2 Learning Geometric Invariant Signatures for Planar Curves

16.2.1 Geometric Invariants of Curves

Invariant representations or invariants are functions defined over geometries which do not change despite the action of a certain type of transformation. These invariants are essential to facilitate our understanding of the idea of a shape. Kendall [18] provides a very apt definition: shape is what remains after translation, rotation and scale have been factored out. In general invariant theory, this can be extended to any general Lie group of transformation. Hence, we can state that shape is what remains after the effects of a certain type of deformation is factored out. Importantly, geometric invariants provide a means to characterize the shape and hence their efficient and robust computation is essential to the success of a given geometric processing framework.

In the next two subsections, we enumerate on two different types of invariants for the simplest geometric structures: planar curves. We discuss their theoretical properties and also numerical considerations for their efficient implementation. We show that the computational schematic involving these invariants correlates very aptly to convolutional neural networks. Therefore, by following the same overall computational pipeline, and replacing the key structural components with learnable options like convolutional kernels, and rectified linear units, we can learn the invariants from examples in a metric learning framework.

16.2.1.1 Differential Invariants

A classic theorem from E. Cartan and Sophus Lie [1, 9] characterize the space of invariant signatures for two and three dimensional geometric entities. For planar curves, it begins with the concept of invariant arc-length, which is a generalized notion of length of the curve. Given a particular Lie-group transformation, one can define an invariant arc-length and an invariant curvature with respect to this arc-length. The set of differential invariant signatures comprises of this invariant curvature and its successive derivatives with respect to the invariant arc-length [6, 7].

The basic idea is that, one can formulate a differential notion of length and a local signature, based on identifying invariance properties of a particular group of transformations. Consider the simple case of euclidean invariants as demonstrated in Fig. 16.1. Consider $C(p) = \begin{bmatrix} x(p) \\ y(p) \end{bmatrix}$: a planar curve with coordinates x and y parameterized by some parameter p. Locally at any given point p on the curve, we have the euclidean arclength given by:

$$ds = \sqrt{dx^2 + dy^2} = \sqrt{\left(\frac{dx}{dp}\right)^2 + \left(\frac{dy}{dp}\right)^2}\ dp \tag{16.1}$$

It is straightforward to observe that expression 16.1 is provably invariant to any euclidean transformations of the curve. Therefore, the length of a curve from point p_0 to p_1 is given by integrating the differential lengths:

$$s = \int_{p_0}^{p_1} |C_p|\, dp = \int_{p_0}^{p_1} \sqrt{x_p^2 + y_p^2}\, dp, \tag{16.2}$$

where $x_p = \frac{dx}{dp}$, and $y_p = \frac{dy}{dp}$. Similarly, a local signature at a point on the curve can be obtained by observing that the rate of change of the unit tangent vector with respect to the arc-length is invariant to euclidean transformations. This gives us the principal invariant signature, that is the Euclidean curvature, given by

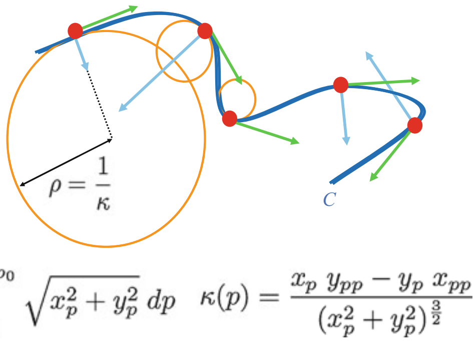

$\rho = \frac{1}{\kappa}$
C
$s = \int_{p=0}^{p=p_0} \sqrt{x_p^2 + y_p^2}\, dp$
$\kappa(p) = \frac{x_p\, y_{pp} - y_p\, x_{pp}}{(x_p^2 + y_p^2)^{\frac{3}{2}}}$

Fig. 16.1 Schematic for differential invariants: Euclidean arclength s and Euclidean curvature κ which is the inverse of the radius of curvature of the osculating circle at a given point

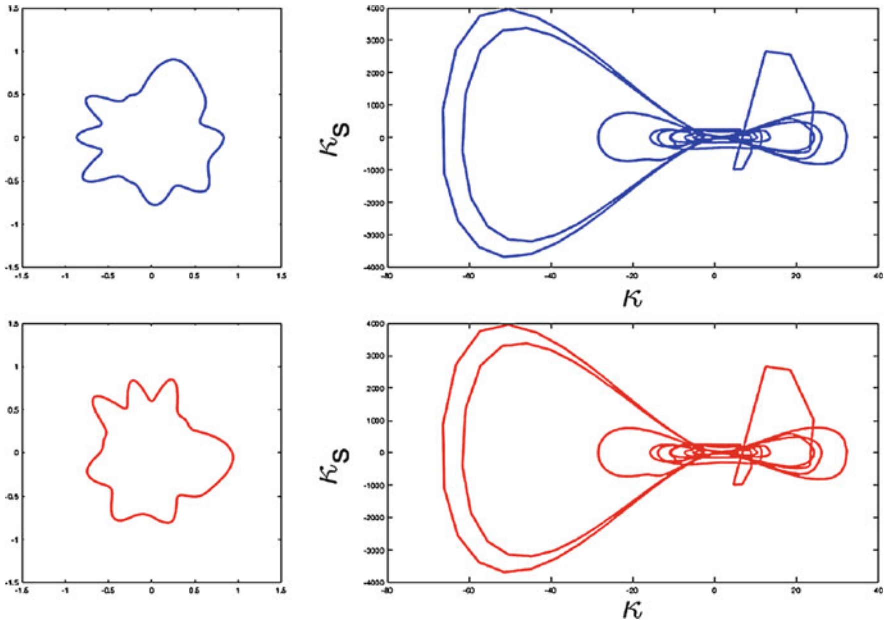


Fig. 16.2 Cartans theorem demonstrating differential invariant signatures. The curves (red and blue) are related by a Euclidean transformation. The plot of the first two differential signatures: $\{\kappa, \kappa_s\}$ are identical

$$\kappa(p) = \frac{\det(C_p, C_{pp})}{|C_p|^3} = \frac{x_p y_{pp} - y_p x_{pp}}{(x_p^2 + y_p^2)^{\frac{3}{2}}}. \tag{16.3}$$

In addition, the euclidean curvature of Eq. (16.3) is also the inverse of radius of the osculating circle at the point as shown in Fig. 16.2.

A similar philosophy can used for the derivation of arc-length and curvature for the equiffine case: $\tilde{C} = A\,C + t$ with $\det(A) = 1$. We refer the interested reader to [29] for a comprehensive and detailed treatise on affine differential geometry. We have the differential equi-affine arc-length given by

$$dv = |\kappa|^{\frac{1}{3}}\ |C_p|\ dp \tag{16.4}$$

and hence the equi-affine length between points p_0 and p_1:

$$v = \int_{p_0}^{p_1} |\kappa|^{\frac{1}{3}}\ |C_p|\ dp, \tag{16.5}$$

and the equi-affine curvature,

$$\mu = (C_{vv},\ C_{vvv})$$

$$\mu = \frac{4(x_{pp}y_{ppp} - y_{pp}x_{ppp}) + (x_p y_{pppp} - y_p x_{pppp})}{3(x_p y_{pp} - y_p x_{pp})^{\frac{5}{3}}} - \frac{5(x_p y_{ppp} - y_p x_{ppp})^2}{9(x_p y_{pp} - y_p x_{pp})^{\frac{8}{3}}}. \tag{16.6}$$

These differential invariant signatures are unique, that is, two curves are related by the corresponding transformation *if and only if* their differential invariant signatures are identical. Moreover, every invariant of the curve is a function of these fundamental differential invariants and its successive derivatives with respect to the invariant arc-length. Thus, we have the set of euclidean differential invariants: $S_{euclidean}$, and the set of equi-affine differential invariants: $S_{equi-affine}$, for every point on the curve:

$$S_{euclidean} = \{\kappa,\ \kappa_s,\ \kappa_{ss},\ \ldots\} \tag{16.7}$$

$$S_{equi-affine} = \{\mu,\ \mu_v,\ \mu_{vv},\ \ldots\} \tag{16.8}$$

Cartan's theorem provides a strong characterization of the invariant signatures and establishes invariant curvature as the principal signature of a curve. For planar curves, any contour can be recovered up to the transformation given only the first two sets of invariants, that is $\{\kappa, \kappa_s\}$, for euclidean, and $\{\mu, \mu_v\}$ for the equi-affine. Figure 16.2 demonstrates this phenomenon.

An important observation is that differential invariants are, as the name implies, *differential* in nature, that is, the formulas involving their computation involve derivatives of the curve. From a signal processing perspective, this is equivalent to *high-pass filtering* over the coordinate functions of the curve combined with further non-linearities. Therefore, numerical estimation of these invariants is a challenge in low signal-to-noise ratio settings.

16.2.1.2 Integral Invariants

Integral invariants are functions computed using integral measures along the curve using parameterized kernels $h_r(p, x) : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$

$$I_r(p) = \int_C h_r(p, x) d\mu(x) \tag{16.9}$$

In contrast to differential invariants, integral invariants are constructed by designing invariant functions that are a result of local integral computations on the curve. Typically, integral invariants are associated with a parameter r that characterizes the extent of the integration locally at each point. Typical examples are the integral distance invariant for which $h_r(p, x)$ is the distance function of the point p to all points on the curve within a ball of radius r. Another more prominent example is the

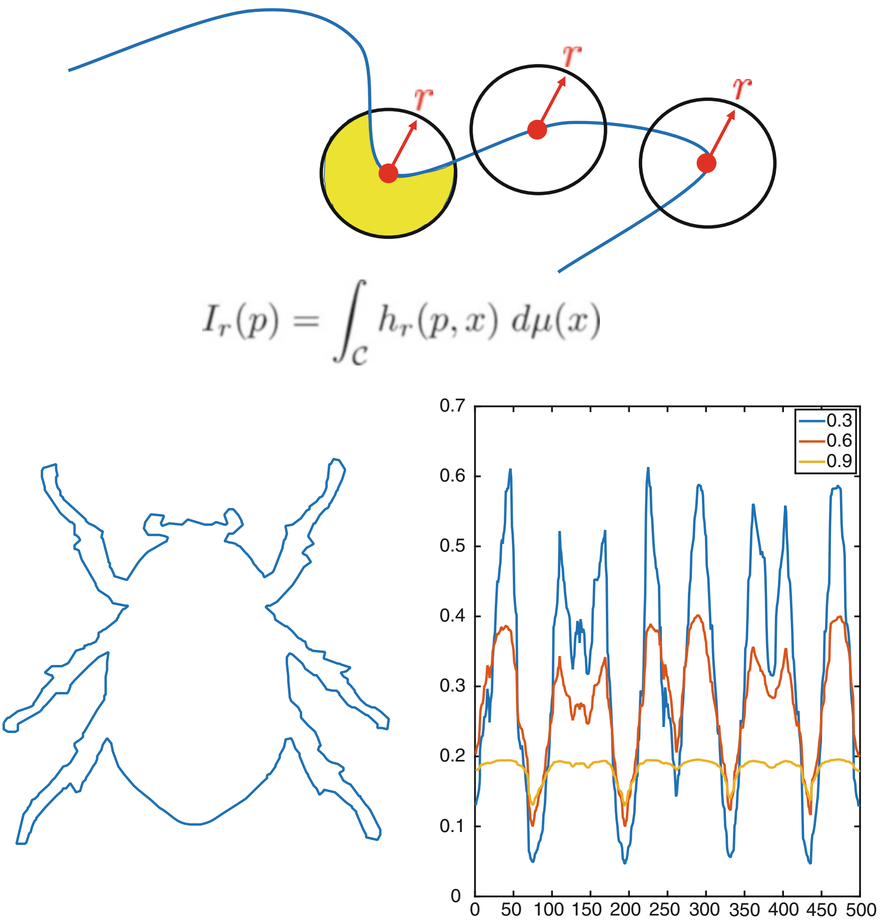


Fig. 16.3 (Top) Schematic for integral invariants: the integral area invariant. At each point, for a predefined radius r, the signature is evaluated as the area of the yellow region. (Bottom) Integral invariant signatures plotted for a closed contour for different raddii of the ball. Each signature captures structure at a different scale as defined by the radius of the ball

integral area invariant which, for each point on the contour, computes the area of the intersection between a ball of radius r and the interior of the curve. [11, 13, 21, 25] provide a detailed treatise on various kinds of integral invariants and their theoretical and numerical properties.

In contrast to differential invariants, integral invariants are more robust to noise due to the integral nature of their computation. Moreover, The radius r provides a scale-space like property, where a big enough r can be set to compensate for all noise within that magnitude. Figure 16.3 shows an example of Integral Invariants.

16.2.2 Learning Geometric Invariant Signatures of Planar Curves

16.2.2.1 Motivation

It is straightforward to observe that differential and integral invariants can be thought of as being obtained from non-linear operations of convolution filters. The construction of differential invariants employ filters for which the action is equivalent to numerical differentiation (high pass filtering) whereas integral invariants use filters which act like numerical integrators (low pass filtering) for stabilizing the invariant. This provides a motivation to adopt a learning based approach and we demonstrate that the process of estimating these filters and functions can be outsourced to a learning framework.

We use the Siamese configuration (see Fig. 16.4) for implementing this idea. Siamese configurations have been used in signature verification [4], face-verification and recognition [14, 31, 32], metric learning [10], image descriptors [8], dimensionality reduction [12] and also for generating 3D shape descriptors for correspondence and retrieval [22, 38]. In these papers, the goal was to learn the descriptor and hence the similarity metric from data using notions of only positive and negative examples. We use the same framework for estimation of geometric invariants. However, in contrast to these methods, we also discuss and analyze the output descriptor and provide a geometric context to the learning process. The contrastive loss function (Eq. (16.10)) driving the training ensures that the network chooses filters which push and pull different features of the curve into the invariant by balancing a mix of robustness and fidelity.

16.2.2.2 Training for Invariance

A planar curve can be represented either explicitly by sampling points on the curve or implicitly using a representation like level sets [19]. We work with an explicit representation of simple curves (open or closed) with random variable sampling of the points along the curve. Thus, every curve is a $N \times 2$ array denoting the 2D coordinates of the N points. We build a convolutional neural network which inputs a curve and outputs a representation or signature for every point on the curve (see (bottom) in Fig. 16.4). We can interpret this architecture as an algorithmic scheme of representing a function over the curve.

A Siamese configuration (see Fig. 16.4) for training neural networks comprises of stacking two identical copies of a network (hence shared parameters as shown in Fig. 16.4), that process two separate inputs of data. A training example comprises of a pair of data points ($\mathbf{X_1}$ and $\mathbf{X_2}$ in Fig. 16.4) and a label: λ, which deems them as a positive ($\lambda = 1$) or a negative ($\lambda = 1$) example. During training, for each pair of positive examples ($\lambda = 1$), the loss forces the network to adjust its parameters so as to reduce the difference in signatures for this pair. For each pair of negative

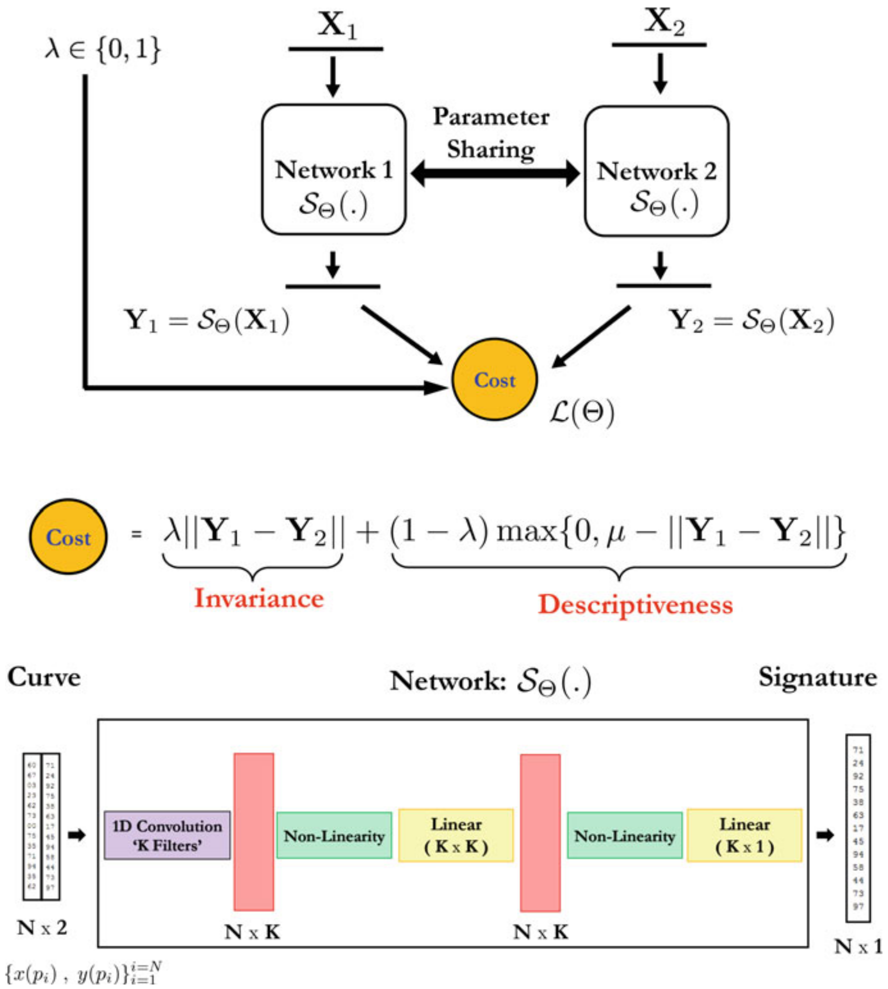

$\lambda \in \{0,1\}$
$\mathbf{X}_1$
$\mathbf{X}_2$
Parameter Sharing
Network 1 $\mathcal{S}_\Theta(.)$
Network 2 $\mathcal{S}_\Theta(.)$
$\mathbf{Y}_1 = \mathcal{S}_\Theta(\mathbf{X}_1)$
$\mathbf{Y}_2 = \mathcal{S}_\Theta(\mathbf{X}_2)$
Cost
$\mathcal{L}(\Theta)$
Cost = $\underbrace{\lambda\|\mathbf{Y}_1 - \mathbf{Y}_2\|}_{\text{Invariance}} + \underbrace{(1-\lambda)\max\{0, \mu - \|\mathbf{Y}_1 - \mathbf{Y}_2\|\}}_{\text{Descriptiveness}}$
Curve
Network: $\mathcal{S}_\Theta(.)$
Signature
1D Convolution 'K Filters'
Non-Linearity
Linear (K x K)
Non-Linearity
Linear (K x 1)
N x 2
N x K
N x K
N x 1
$\{x(p_i)\ ,\ y(p_i)\}_{i=1}^{i=N}$

Fig. 16.4 (Top) Siamese configuration for learning an invariant signature of the curve. The cost function comprises of two components: positive examples to train for invariance and negative examples to train for descriptiveness. (Bottom) An example of a straightforward one dimensional convolutional neural network architecture that inputs a curve and outputs a point-wise signature of the curve

examples ($\lambda = 0$), the loss tries to maximise the difference in signature outputs of the network. We extract geometric invariance by requiring that the two arms of the Siamese configuration minimize the distance between the outputs for curves which are related by a transformations and maximize for carefully constructed negative examples. For a detailed discussion on the choice of negative examples, [24] provides an explanation that connects invariant curve-evolution and multiscale curve descriptors to the proposed learned signatures.

The intuition behind this positive-negative example based contrastive training is as follows. Training with only positive examples will yield a trivial signature like an all-ones for every curve irrespective of its content. Hence training with negative examples $\lambda = 0$ enables the injection of a sense of descriptiveness into the signature. Minimizing the contrastive cost function (Eq. (16.10)) of the Siamese configuration directs the network architecture to model a function over the curve which is invariant to the transformation and yet descriptive enough to encapsulate the salient features of the curve.

We build a sufficiently large dataset comprising of such positive and negative examples of the transformation from a database of curves. Let $\mathbf{X_1}^{(i)}, \mathbf{X_2}^{(i)} \in \mathbb{R}^{2\times n}$ be two n-point curves inputted to the first and the second arm of the configuration for the ith example of the data with label λ_i. Let $\mathcal{S}_\Theta(\mathbf{X})$ denote the output of the network for a given set of weights Θ for input curve $\mathbf{X}$. The contrastive loss function is given by,

$$\mathcal{L}(\Theta) = \sum_i \lambda_i \; || \, \mathcal{S}_\Theta(\mathbf{X_1}^{(i)}) - \mathcal{S}_\Theta(\mathbf{X_2}^{(i)}) \, || \\ + \; (1-\lambda_i) \; \max\Big(0, \; \mu \; - \; || \, \mathcal{S}_\Theta(\mathbf{X_1}^{(i)}) - \mathcal{S}_\Theta(\mathbf{X_2}^{(i)}) \, ||\Big) \quad (16.10)$$

where μ is a cross validated hyper-parameter, known as margin, which defines the metric threshold beyond which negative examples are penalized. Minimizing the contrastive cost function of the Siamese configuration directs the network architecture to model a function over the curve which is invariant to the prevailing transformation.

16.2.2.3 Results

Figures 16.5 and 16.6 visualize the output of the learned invariant. To test the numerical stability and robustness of the invariant signatures we add increasing levels of zero-mean Gaussian noise to the curve and compare the three types of signatures: differential (euclidean curvature), integral (integral area invariant) and the output of our network (henceforth termed as network invariant) as shown in Fig. 16.5. The network invariant shows a much improved robustness to noise in comparison to its axiomatic counterparts. Apart from adding noise, we also transform the curve to obtain a better assessment of the invariance property.

The results demonstrate that deep learning architectures can indeed be used for learning geometric invariants of planar curves from examples. We show that the learned invariants are more robust to noise in comparison to their axiomatic counterparts. More importantly, our experiments lay out a discussion that provides two different viewpoints to the very important topic of geometric invariance: that of classical differential geometry and deep learning.

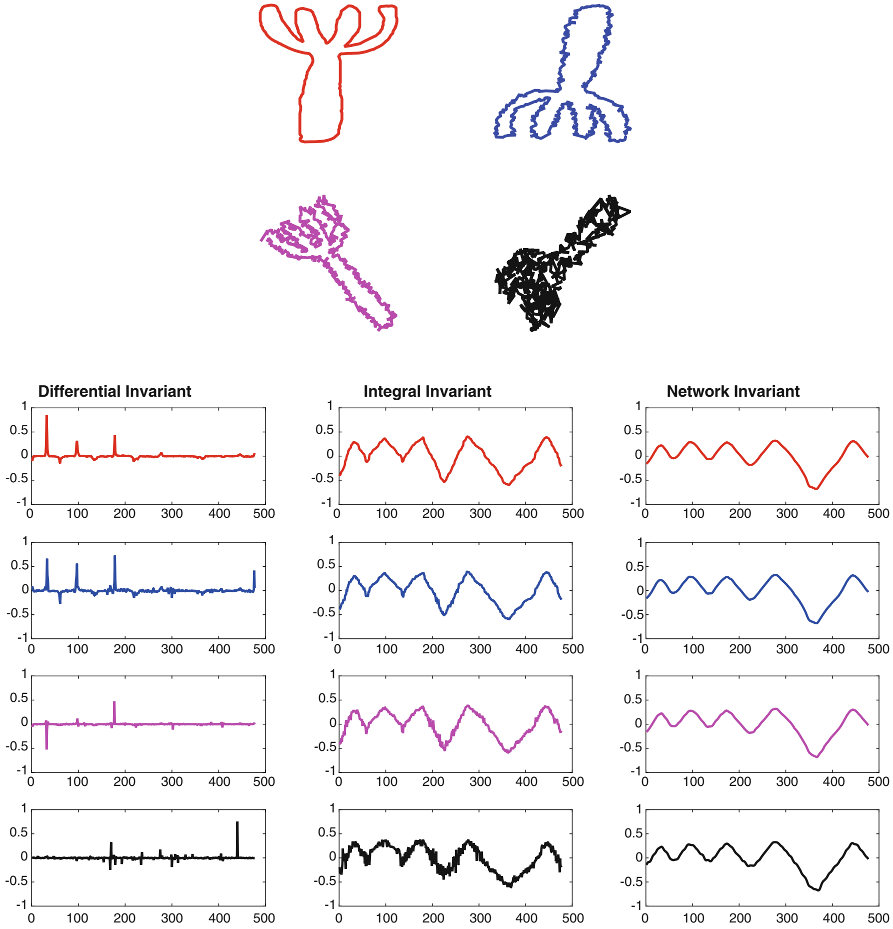
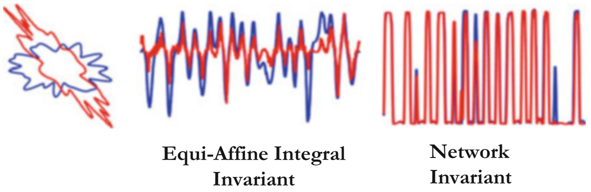


Fig. 16.5 Stability of different signatures in varying levels of noise and Euclidean transformations. The correspondence for the shape and the signature is the color. All signatures are normalized

Fig. 16.6 Equi-affine Invariants: A simple smooth curve, equi-affine transformed along with a small addition of noise. The network is able to learn a much more robust invariant



16.3 Geometric Moments for Advanced Deep Learning on Point Clouds

In the previous section we use a learning mechanism to train for a geometric invariant. We argue that the computational mechanism of differential and integral invariants caters well to a convolutional neural network architecture and demonstrate a framework to learn these invariants from examples. In the this section, we demonstrate the reverse, that is, we input geometric moments into a learning architecture and discuss the benefits of this approach.

In recent years the popularity and demand of 3D sensors has vastly increased. Applications using 3D sensors include robot navigation [34], stereo vision, and advanced driver assistance systems [39] to name just a few. Recent studies attempt to adjust deep neural networks to operate on 3D data representations for diverse geometric tasks. Motivated mostly by memory efficiency, our choice of 3D data representation is to process raw point clouds. An interesting concept is applying geometric features as input to neural networks acting on point clouds for the geometric task of rigid object classification. This is a fundamental problem in analyzing rigid bodies which their isometry group includes only rigid transformations.

16.3.1 Geometric Moments as Class Identifiers

Historically, *geometric moments have been used to define invariant signatures for classification of rigid objects* [5]. Geometric moments of increasing order represent distinct spatial characteristics of the object, implying a strong support for construction of global shape descriptors. By definition, first order moments represent the extrinsic centroid; second order moments measure the covariance and can also be thought of as moments of inertia. We restrict ourselves to point clouds that can be seen as a set of unstructured 3D points that approximate the geometry of the surface. A point cloud is a set of points $X \in \mathbb{R}^3$, where each point $\boldsymbol{x}_j$ is given by its coordinates $(x_j, y_j, z_j)^T$, the moments in that discrete case are defined in Eq. (16.11). Note that in Eq. (16.11b) the sum is over matrices, while in Eq. (16.11c) the sum is over the point coordinates and only one $(p+q+r)^{th}$ moment is computed.

$$1^{st}\ Order\ Moment: \quad \boldsymbol{x}_0 = \sum_j \boldsymbol{x}_j \tag{16.11a}$$

$$2^{nd}\ Order\ Moment: \quad \Sigma = \sum_j \boldsymbol{x}_j \boldsymbol{x}_j^T \tag{16.11b}$$

$$(p+q+r)^{th}\ Order\ Moment: \quad m_{pqr} = \sum_j (x_j)^p (y_j)^q (z_j)^r \tag{16.11c}$$

Higher-order geometric moments provide us descriptors of the surface: taking more orders improves our ability to identify our object. Sampled surfaces, such as point clouds can be identified by their estimated geometric moments, see Eq. (16.11). Moreover, a finite set of moments is often sufficient as a compact signature that defines the surface. This idea was classically used for classification of rigid objects and estimation of surface similarity.

Our goal is to allow a neural network to simply lock onto variations of geometric moments. One of the main challenges of this approach is that training a neural network to approximate polynomial functions requires the network depth and complexity to be logarithmically inversely proportional to the approximation error [40]. In practice, in order to approximate polynomial functions of the coordinates for the calculation of geometric moments the network requires a large number of weights and layers.

In the recent years an extended research has been done on processing point clouds [2, 3, 15, 16, 20, 27, 33] for object classification. Qi et al. [26] were the pioneers proposing a network architecture for directly processing point sets. The framework they suggested includes lifting the coordinates of each point into a high dimensional learned space, while ignoring the geometric structure. At the other end, networks that attempt to process other representations of low dimensional geometric structures such as meshes, voxels (volumetric grids) [23, 28, 36, 37] and multi-view projections [17, 30, 41] are often less efficient when considering both computational and memory complexities (Fig. 16.7).

16.3.2 Raw Point Cloud Classification Based on Moments

Here, we propose a new layer that favors geometric moments for point cloud object classification. The most prominent element of the network is supplementing the given point cloud coordinates together with polynomial functions of the coordinates. This simple operation allows the network to account for higher order moments of a given shape. Explicitly, we simply add polynomial functions of the coordinates to the network inputs that will allow the network to learn the elements of Σ (see

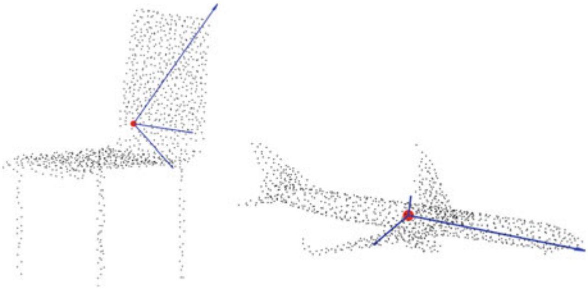


Fig. 16.7 The first and second geometric moments displayed on a point cloud. Using the first order moments (red disc), the translation ambiguity can be removed. The principal directions d_1, d_2, d_3 (blue arrows) are defined by the second order geometric moments

Eq. (16.11)) and, assuming consistent sampling, should better capture the geometric structure of the data.

First, we present a toy example to illustrate that extra polynomial expansions as an input can capture geometric structures well. Figure 16.8 (up) shows the data and the network predictions for 1000 noisy 2D points, each point belongs to one of two spirals. We achieved 98% success with extra polynomial multiplications, i.e x^2, y^2, xy, as input to a one layer network with only 8 hidden ReLU nodes, contrary to the same network without the polynomial multiplications (only 53% success). Figure 16.8 (down) shows the decision boundaries that were formed from the eight hidden nodes of the network in Fig. 16.8 (up left). As expected, radial boundaries can be formed from the additional polynomial extensions and are crucial to forming the entire decision surface.

Second, we present our approach that consumes directly point clouds with their second order polynomial expansions for object classification. An early attempt to

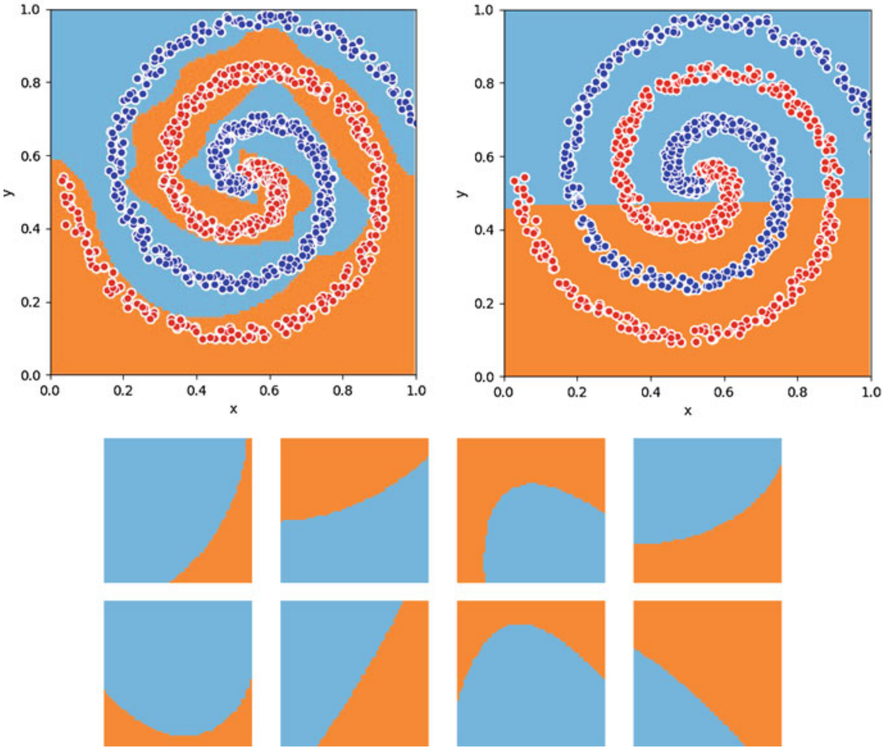


Fig. 16.8 Output response surfaces of a network comprised of 8 hidden nodes with polynomial expansions (up left) and without (up right). The output responses of each hidden node of (up left) is presented in (down). The training set (data points) are colored in red/blue, which indicate positive/negative class respectively while orange/light blue areas are the corresponding network predictions

treat point clouds as non-structured data sets from which a neural network is trained to extract discriminative properties is the PointNet [26] architecture proposed by Qi et al. The classification architecture of PointNet is based on fully connected layers and symmetry functions, like max pooling, to establish invariance to potential permutations of the points. The architecture pipeline commences with Multi-Layer Perceptrons (MLPs) to generate a per point feature vector, then, applies max pooling to generate global features that serve as a signature of the point cloud. Finally, fully connected layers produce output scores for each class. Note that all MLPs operations are performed per point, thus, interrelations between points are accomplished only by weight sharing.

The idea of integrating the coordinates of points in space into ordered vectors that serve as class identifiers motivated us to extend this line of thought in that of classification by comparing shape moments [5]. We claim that PointNet is an implicit way for approximating moments. With this fundamental understanding in mind, the next question is how could we assist the network by lifting it into a more convenient space. This observation allows us to use a classical simple lifting of the feature coordinates. It improves significantly both complexity and accuracy as validated in the experimental results.

The baseline architecture of the suggested network, see Fig. 16.9, is based on the PointNet architecture. Our main contribution is the addition of polynomial functions as part of the input domain. The addition of point coordinate powers to the input is a simple procedure that improves accuracy and decreases the run time during training and inference.

Performance Evaluation

Evaluation is performed on the ModelNet40 benchmark [36]. ModelNet40 is a synthetic dataset composed of Computer-Aided Design (CAD) models. ModelNet40 contains 12,311 CAD models given as triangular meshes. Preprocessing of each triangular mesh as proposed in [26] yields 1024 points sampled from each triangular mesh using the farthest point sampling (FPS) algorithm. Rotation by a random angle

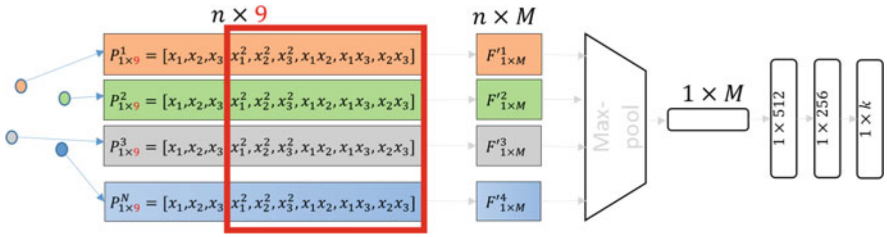

$n \times 9$
$P^1_{1\times 9} = [x_1,x_2,x_3\ x_1^2,x_2^2,x_3^2,x_1x_2,x_1x_3,x_2x_3]$
$P^2_{1\times 9} = [x_1,x_2,x_3\ x_1^2,x_2^2,x_3^2,x_1x_2,x_1x_3,x_2x_3]$
$P^3_{1\times 9} = [x_1,x_2,x_3\ x_1^2,x_2^2,x_3^2,x_1x_2,x_1x_3,x_2x_3]$
$P^N_{1\times 9} = [x_1,x_2,x_3\ x_1^2,x_2^2,x_3^2,x_1x_2,x_1x_3,x_2x_3]$
$n \times M$
$F'^1_{1\times M}$
$F'^2_{1\times M}$
$F'^3_{1\times M}$
$F'^4_{1\times M}$
Max-pool
$1 \times M$
1×512
1×256
$1 \times k$

Fig. 16.9 Illustration of the proposed object classification architecture. The input of the network includes the point cloud coordinates as well as second order polynomial functions of these coordinates. It enables the network to efficiently learn the shape moments. The output is a probability distribution over the k classes, M is the number of features per point and n denotes the number of points

and additive noise are used for better data augmentation. The database contains samples from similar categories, like the flower-pot, plant and vase, for which separation is impossible even for a human observer.

We explore our implicit way of generating second order moments with respect to ModelNet40 database accuracy as a function of the number of nodes in the feature layer, see Table 16.1. The results show that there is a strong relation between the accuracy and the additional inputs that simplify the realization of geometric moments by the network. Our network has better accuracy percentage than the network without second order polynomials for all tested number of neurons in the feature layer. The only difference between the networks is the polynomial functions in the input domain.

A direct comparison with PointNet was also conducted and the experimental results confirm the superiority of such design. Our design is similar to that of PointNet in term of number of feature layers, however we consider also the polynomial expansions of the point coordinates. Table 16.2 presents a comparison with PointNet in term of accuracy, inference time and memory. The computational requirements are with respect to the number of the network's parameters (memory) and with respect to the run-time required by the models for computing the predictions. Our results show that adding polynomial functions to input improve the classification performance, and in the same time lead to computational and memory efficiency.

In conclusion, we proposed to feed the input domain with polynomial functions that represent geometric moments for 3D object classification. We combine an established method from the geometry processing literature with a neural network model to reinforce the model's capacity with a geometric concept. It is vital to

Table 16.1 Comparison of accuracy on ModelNet40 [36] (in %) between implicit 1st order input to our suggested network with implicit 1st + 2nd order input for different number of neurons in the feature layer

M—Number of features per point	16	32	64	128	256	512
1st order	84.0	84.8	85.5	85.7	85.6	85.7
1st + 2nd order	**84.6**	**85.6**	**86.5**	**86.5**	**86.7**	**87.2**

1st order denotes only the point coordinates while 1st + 2nd order refer to the point coordinates concatenate with their polynomial functions as can been see in Fig. 16.9. The bold values are made to highlight the improved performance of our proposed solution

Table 16.2 Comparison of time and space complexity

	Memory (MB)	Inference time (msec)	Overall accuracy (%)
PointNet (baseline)	20	5.1	87.9
PointNet	40	5.6	89.2
Our method	**20**	**5.1**	**89.6**

Memory is the size of the model in mega-byte (MB) and the inference run-time was measured in milli-seconds (ms). Overall Accuracy stands for the suggested network accuracy on ModelNet40. The bold values are made to highlight the improved performance of our proposed solution develop methods which combine proven geometrical methods with deep neural networks to leverage their ability to generalize and produce more accurate models.

16.4 Conclusion

This chapter discusses the synergy between the principled axiomatic computations of geometric invariants and modern learning mechanisms like deep learning. Our attempt is to discover the space of computational methods that smoothly integrate principled axiomatic computation (geometric invariants) and data dependant models of deep learning. To this aim we demonstrated both ends of the coin: superior numerical geometry from learning and also improved deep learning using geometry. First, we have demonstrated a method to learn geometric invariants of planar curves. Using just positive and negative examples of transformations, we show that a convolutional neural network is able to train for an invariant which is numerically more robust as compared to differential and integral invariants. Second, we combined a geometric understanding about the ingredients required to construct compact shape signatures with neural networks that operate on clouds of points to leverage the network's abilities to cope with the problem of rigid objects classification. We demonstrated that lifting the input coordinates of points allowed the network to classify the objects with better efficiency and accuracy compared to previous methods that operate in the same domain. We believe that the ideas introduced in this chapter could be applied in other fields where geometric analysis is involved and could improve the ability of neural networks to efficiently and accurately handle geometric structures.

References

1. Ackerman, M.: Sophus Lie's 1884 Differential Invariant Paper. Math Sci Press, Berkeley (1976)
2. Atzmon, M., Maron, H., Lipman, Y.: Point convolutional neural networks by extension operators. ACM Trans. Graph. **37**(4), 71:1–71:12 (2018)
3. Ben-Shabat, Y., Lindenbaum, M., Fischer, A.: 3DmFV: three-dimensional point cloud classification in real-time using convolutional neural networks. IEEE Robotics Autom. Lett. **3**(4), 3145–3152 (2018)
4. Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., Shah, R.: Signature verification using a "siamese" time delay neural network. In: Advances in Neural Information Processing Systems, pp. 737–744 (1994)
5. Bronstein, A.M., Bronstein, M.M., Kimmel, R.: Numerical Geometry of Non-rigid Shapes. Springer, New York (2008)
6. Bruckstein, A.M., Netravali, A.N.: On differential invariants of planar curves and recognizing partially occluded planar shapes. Ann. Math. Artif. Intell. **13**(3–4), 227–250 (1995)
7. Calabi, E., Olver, P.J., Shakiban, C., Tannenbaum, A., Haker, S.: Differential and numerically invariant signature curves applied to object recognition. Int. J. Comput. Vis. **26**(2), 107–135 (1998)

8. Carlevaris-Bianco, N., Eustice, R.M.: Learning visual feature descriptors for dynamic lighting conditions. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2769–2776. IEEE, Piscataway (2014)
9. Cartan, E.: Geometry of Riemannian Spaces: Lie Groups; History, Frontiers and Applications Series, vol. 13. Math Science Press, , Berkeley (1983)
10. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, pp. 539–546. IEEE, Piscataway (2005)
11. Fidler, T., Grasmair, M., Scherzer, O.: Identifiability and reconstruction of shapes from integral invariants. Inverse Probl. Imag. **2**(3), 341–354 (2008)
12. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), vol. 2, pp. 1735–1742. IEEE, Piscataway (2006)
13. Hong, B.W., Soatto, S.: Shape matching using multiscale integral invariants. IEEE Trans. Pattern Anal. Mach. Intell. **37**(1), 151–160 (2014)
14. Hu, J., Lu, J., Tan, Y.P.: Discriminative deep metric learning for face verification in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1875–1882 (2014)
15. Hua, B.S., Tran, M.K., Yeung, S.K.: Pointwise convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 984–993 (2018)
16. Huang, Q., Wang, W., Neumann, U.: Recurrent slice networks for 3d segmentation of point clouds. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2626–2635 (2018)
17. Kanezaki, A., Matsushita, Y., Nishida, Y.: Rotationnet: joint object categorization and pose estimation using multiviews from unsupervised viewpoints. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5010–5019 (2018)
18. Kendall, D.G.: Shape manifolds, procrustean metrics, and complex projective spaces. Bull. Lond. Math. Soc. **16**(2), 81–121 (1984)
19. Kimmel, R.: Numerical Geometry of Images: Theory, Algorithms, and Applications. Springer, New York (2012)
20. Klokov, R., Lempitsky, V.: Escape from cells: deep kd-networks for the recognition of 3D point cloud models. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 863–872 (2017)
21. Manay, S., Hong, B.W., Yezzi, A.J., Soatto, S.: Integral invariant signatures. In: European Conference on Computer Vision, pp. 87–99. Springer, Berlin (2004)
22. Masci, J., Boscaini, D., Bronstein, M., Vandergheynst, P.: Geodesic convolutional neural networks on riemannian manifolds. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 37–45 (2015)
23. Maturana, D., Scherer, S.: Voxnet: a 3D convolutional neural network for real-time object recognition. In: Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on, pp. 922–928. IEEE, Piscataway (2015)
24. Pai, G., Wetzler, A., Kimmel, R.: Learning invariant representations of planar curves. In: International Conference on Learning Representations (2017)
25. Pottmann, H., Wallner, J., Huang, Q.X., Yang, Y.L.: Integral invariants for robust geometry processing. Comput. Aided Geom. Des. **26**(1), 37–60 (2009)
26. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: deep learning on point sets for 3D classification and segmentation. Proc. Comput. Vision Pattern Recogn. IEEE **1**(2), 4 (2017)
27. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: Advances in Neural Information Processing Systems, pp. 5105–5114 (2017)
28. Riegler, G., Osman Ulusoy, A., Geiger, A.: Octnet: learning deep 3D representations at high resolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3577–3586 (2017)
29. Su, B.: Affine Differential Geometry. CRC Press, Boca Raton (1983)

30. Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E.: Multi-view convolutional neural networks for 3D shape recognition. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 945–953 (2015)
31. Sun, Y., Wang, X., Tang, X.: Deep learning face representation from predicting 10,000 classes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1891–1898 (2014)
32. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: Deepface: closing the gap to human-level performance in face verification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1701–1708 (2014)
33. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph CNN for learning on point clouds (2018). Preprint. arXiv: 1801.07829
34. Weingarten, J.W., Gruener, G., Siegwart, R.: A state-of-the-art 3D sensor for robot navigation. In: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No. 04CH37566), vol. 3, pp. 2155–2160. IEEE, Piscataway (2004)
35. Weiss, I.: Noise-resistant invariants of curves. IEEE Trans. Pattern Anal. Mach. Intell. **15**(9), 943–948 (1993)
36. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3D shapenets: a deep representation for volumetric shapes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1912–1920 (2015)
37. Xiang, Y., Choi, W., Lin, Y., Savarese, S.: Data-driven 3D voxel patterns for object category recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1903–1911 (2015)
38. Xie, J., Fang, Y., Zhu, F., Wong, E.: Deepshape: deep learned shape descriptor for 3D shape matching and retrieval. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1275–1283 (2015)
39. Yan, C., Xie, H., Yang, D., Yin, J., Zhang, Y., Dai, Q.: Supervised hash coding with deep neural network for environment perception of intelligent vehicles. IEEE Trans. Intell. Transp. Syst. **19**(1), 284–295 (2017)
40. Yarotsky, D.: Error bounds for approximations with deep relu networks. Neural Netw. **94**, 103–114 (2017)
41. Zhao, J., Xie, X., Xu, X., Sun, S.: Multi-view learning overview: recent progress and new challenges. Inf. Fusion **38**, 43–54 (2017)

Chapter 17
Sub-Riemannian Methods in Shape Analysis

Laurent Younes, Barbara Gris, and Alain Trouvé

Contents

Abstract Because they reduce the search domains for geodesic paths in manifolds, sub-Riemannian methods can be used both as computational and modeling tools in designing distances between complex objects. This chapter provides a review of the methods that have been recently introduced in this context to study shapes, with a special focus on shape spaces defined as homogeneous spaces under the action of

L. Younes (✉)
Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, MD, USA
e-mail: laurent.younes@jhu.edu

B. Gris
Laboratoire Jacques-Louis Lions, Sorbonne Université, Paris, France

A. Trouvé
Centre de Mathématiques et leurs Applications, ENS Paris-Saclay, Cachan, France
e-mail: Alain.Trouve@cmla.ens-cachan.fr

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_17

diffeomorphisms. It describes sub-Riemannian methods that are based on control points, possibly enhanced with geometric information, and their generalization to deformation modules. It also discusses the introduction of implicit constraints on geodesic evolution, and the associated computational challenges. Several examples and numerical results are provided as illustrations.

17.1 Introduction

One of the main focuses in shape analysis is the study, modeling and quantification of shape variation, as observed in various datasets encountered in computer vision, medical research, archaeology etc. It is essential, in this analysis, to properly define relevant shape spaces and equip them with suitable metrics allowing for comparison and defining plausible modes of shape transformation. A significant number of approaches have been introduced in the literature with this goal in mind. They include methods using Riemannian metrics on point sets [16, 17, 31, 32], curves [29, 33, 42, 44, 52, 62, 66] and surfaces [10, 11, 39, 40, 43], conformal and pseudo-conformal approaches [50, 61, 67], or techniques based on medial axes [51] or spherical harmonics [54], among many others.

Our focus in this chapter will be on methods that are inspired by D'Arcy-Thompson's theory of transformation [55], or by the more modern work of Grenander on deformable templates [25, 26] in which motion in shape spaces is obtained through a time-dependent diffeomorphic deformation from the initial shape. When working in this framework, it becomes necessary to control the transformation that is acting on the shape, or preferably its time derivative, or velocity, by minimizing a cost that accumulates during the transformation. This leads to natural definitions of path distances in shape spaces, which, in our discussion, will be Riemannian or sub-Riemannian. The cost can be defined in multiple ways, for example as measures of the smoothness of the velocity field, and how much it differs from zero. One can, in addition, represent the velocity in a small-dimensional parametric form, or impose some constraints that condition admissible shape motions.

We will discuss these approaches from situations in which a minimal prior information is used on the velocity fields to situations where the design and modeling efforts become central. The paper is organized as follows. Section 17.2 discusses the general framework of shape spaces, with an emphasis on spaces of curves as illustration. It will be followed (Sect. 17.3) by a discussion of sub-Riemannian methods that can be introduced for computational efficiency, as approximations of the general framework. Section 17.4 introduces the general framework of deformation modules, while Sect. 17.5 provides a short overview of sub-Riemannian methods with implicit constraints. Finally, we conclude the chapter in Sect. 17.6.

17.2 Shape Spaces, Groups of Diffeomorphisms and Shape Motion

17.2.1 Spaces of Plane Curves

The simplest example of infinite-dimensional shape space is arguably the space of plane curves. We will focus on this example to describe the main components of the construction of metrics induced by the action of diffeomorphisms, before describing how the approach is extended to other spaces in Sect. 17.2.3.

We define a parametrized shape (sometimes called pre-shape) to be a C^r embedding of the unit circle (denoted S^1) in $\mathbb{R}^2$, as defined below.

Definition 17.1 A C^r embedding of S^1 in $\mathbb{R}^2$ is a one-to-one r times continuously differentiable function $m : S^1 \to \mathbb{R}^2$ such that $m^{-1} : m(S^1) \to S^1$ is continuous (where $m(S^1)$ inherits the topology of $\mathbb{R}^2$) and such that the derivative ∂m never vanishes. We let $\mathrm{Emb}^r(S^1, \mathbb{R}^2)$, or simply Emb^r, denote the space of C^r embeddings of S^1 in $\mathbb{R}^2$.

In simpler terms we define a parametrized shape as a sufficiently differentiable, non-intersecting, closed curve. In Definition 17.1, and in the rest of the chapter, we use the notation ∂f to denote the derivative of a function f, or $\partial_x f$ when the variable with respect to which the derivative is taken needs to be specified. Higher derivatives will be denoted, e.g., $\partial_x^2 f$, or $\partial_x \partial_y f$.

We note that Emb^r is an open subset of the Banach space $C^r(S^1, \mathbb{R}^2)$ of C^r functions from S^1 to $\mathbb{R}^2$, equipped with

$$\|m\|_{r,\infty} = \max_{k \le r} \max_{u \in S^1} |\partial^k m(u)|,$$

where the single bars in the right-hand side refer to the standard Euclidean norm. Emb^r therefore has a trivial Banach manifold structure, with tangent space at $m \in \mathrm{Emb}^r$ given by $T_m\mathrm{Emb}^r = C^r(S^1, \mathbb{R}^2)$.

We let $C^p(\mathbb{R}^2, \mathbb{R}^2)$ denote the space of p times continuously differentiable vector fields $v : \mathbb{R}^2 \to \mathbb{R}^2$ and $C_0^p(\mathbb{R}^2, \mathbb{R}^2)$ the space of $v \in C_0^p(\mathbb{R}^2, \mathbb{R}^2)$ that converge, together with their its first p derivatives, to 0 at infinity. Both form Banach spaces when equipped with $\|m\|_{p,\infty}$. We will consider the group

$$\mathrm{Diff}_0^p(\mathbb{R}^2) = \left\{\varphi = \mathrm{id} + v : \varphi \text{ diffeomorphism of } \mathbb{R}^2, v \in C_0^p(\mathbb{R}^2, \mathbb{R}^2)\right\} \qquad (17.1)$$

(Here, id denotes the identity mapping.) This space is a group for the composition of functions, and has a simple differential structure because $\mathrm{Diff}_0^p(\mathbb{R}^2) - \mathrm{id}$ is an open subset of $C_0^p(\mathbb{R}^2, \mathbb{R}^2)$.

If $p \ge r$, the mapping $(\varphi, m) \mapsto \varphi \cdot m = \varphi \circ m$ forms a group action of $\mathrm{Diff}_0^p(\mathbb{R}^2)$ on Emb^r. If $v \in C_0^p(\mathbb{R}^2, \mathbb{R}^2)$ and $m \in \mathrm{Emb}^r$, we will also use the notation

$v \cdot m \overset{\Delta}{=} v \circ m \in C^r(S^1, \mathbb{R}^2)$, which provides the so-called *infinitesimal action* of $C_0^p(\mathbb{R}^2, \mathbb{R}^2)$ on Emb^r, i.e., the derivative of the mapping $\varphi \mapsto \varphi \cdot m$ at $\varphi = \mathrm{id}$, given in our case by

$$v \cdot m \overset{\Delta}{=} \partial_t \left((\mathrm{id} + tv) \cdot m\right)_{|t=0} = v \circ m\,.$$

17.2.2 Basic Sub-Riemannian Structure

Recall that a Riemannian metric on a differential manifold Q specifies an inner product, denoted $\langle \cdot\,,\,\cdot \rangle_q$, on each tangent space $T_q Q$. The length of a differentiable curve $q : t \in [a, b] \to q(t) \in Q$ is

$$\int_a^b \|\partial_t q(t)\|_{q(t)}\, dt,$$

and the geodesic distance between two points q_0 and q_1 is the minimum length among all differentiable curves q that satisfy $q(0) = q_0$ and $q(1) = q_1$. It is equivalently given by the square root of the minimal energy

$$\int_a^b \|\partial_t q(t)\|_{q(t)}^2\, dt,$$

among all such curves. Curves that achieve the minimal energy are called geodesics.

We now introduce some notation and definitions relative to sub-Riemannian metrics.

Definition 17.2 (Sub-Riemannian Metric) Let Q is a differential manifold of dimension N and class C^r. Let H be a fixed vector space. A distribution on Q is a collection $(W_q, q \in Q)$ with $W_q = \boldsymbol{w}(q, H)$, where $\boldsymbol{w} : Q \times H \to TQ$ is a C^r mapping such that, for all $q \in Q$, the mapping $\boldsymbol{w}_q : h \mapsto \boldsymbol{w}(q, h)$ is linear from H to $T_q Q$.

A sub-Riemannian metric on Q is provided by a distribution $(W_q, q \in Q)$ with each W_q equipped with an inner product, denoted $\langle \cdot\,,\,\cdot \rangle_q$, which is C^r as a function of Q. The norm associated with the inner product is denoted $\|\cdot\|_q$, $q \in Q$.

A differential manifold equipped with a sub-Riemannian metric is called a sub-Riemannian manifold.

In the finite-dimensional setting, with $H = \mathbb{R}^K$ for some K, one often assumes that $\boldsymbol{w}_q$ is one-to-one, so that all spaces W_q have dimension K. We do not enforce this condition here, since it is harder to satisfy in infinite dimensions.

Definition 17.3 (Geodesics) Let Q be a C^r sub-Riemannian manifold with distribution $(W_q, q \in Q)$. One says that a differentiable curve $t \mapsto q(t)$ from $[0, 1]$ to Q

is admissible if $\partial_t q(t) \in W_{q(t)}$ for all t. A point $q_1 \in Q$ is attainable from another point q_0 is there exists an admissible curve with q_0 and q_1 as extremities.

The length of an admissible curve $t \mapsto q(t)$ is defined by $\int_0^1 \|\partial_t q(t)\|_q \, dt$ and its energy by $\int_0^1 \|\partial_t q(t)\|_q^2 \, dt$. The sub-Riemannian distance between two points q_0 and q_1 is defined as the square root of the minimal energy among all curves that connect them, and curves achieving this minimum, when they exists, are called sub-Riemannian geodesics.

One can show that the sub-Riemannian distance between q_0 and q_1 is also the minimal length among all curves that connect them.

We now describe how these concepts apply to shape spaces, keeping, however, the discussion at a formal level in this infinite-dimensional context. In the deformable template approach, the focus is placed on curves that result from diffeomorphic transformations, therefore taking the form $m(t) = \varphi(t) \circ m(0)$, where $t \mapsto \varphi(t) \in \mathrm{Diff}_0^p(\mathbb{R}^2)$ is differentiable. This implies that $\partial_t m(t) = v(t) \circ m(t)$ for $v(t) = \partial_t \varphi(t) \circ \varphi(t)^{-1}$. Thus, $\partial_t m(t) \in \hat{W}_{m(t)}$, where

$$\hat{W}_m \triangleq \{v \circ m : v \in C_0^p(\mathbb{R}^2, \mathbb{R}^2)\}.$$

This space is generally a strict subset of $C^r(S^1, \mathbb{R}^2)$, and the family $(\hat{W}_m, m \in \mathrm{Emb}^r)$ is a distribution in Emb^r, with $\boldsymbol{w}(m, v) = v \circ m$ in Definition 17.2. Define $\hat{\xi}_m : C_0^p(\mathbb{R}^2, \mathbb{R}^2) \to C^r(S^1, \mathbb{R}^2)$ by $\hat{\xi}_m v = v \circ m$, so that $\hat{\xi}_m$ is the infinitesimal action considered for fixed m. Here, we have $\hat{W}_m = \hat{\xi}_m(C_0^p(\mathbb{R}^2, \mathbb{R}^2))$, and many of the concepts below will naturally be expressed in terms of this operator.

To define a sub-Riemannian metric on Emb^r, we need to equip the distribution with an inner product. Still in line with the deformable template paradigm, this inner product will be defined by transporting an inner product on $C_0^p(\mathbb{R}^2, \mathbb{R}^2)$ to $C^r(S^1, \mathbb{R}^2)$. Interesting inner products will however be defined only on subspaces of $C_0^p(\mathbb{R}^2, \mathbb{R}^2)$ rather than on the full space, and we introduce such a subspace V, with inner product denoted as $\langle \cdot\,,\, \cdot \rangle_V$. In order to specify well-posed and computationally friendly variational problems, we require that $(V, \|\cdot\|_V)$ form a Hilbert space, which implies that V must be a strict subset of $C_0^p(\mathbb{R}^2, \mathbb{R}^2)$. (More precisely, we assume that the inclusion of $(V, \|\cdot\|_V)$ in $(C_0^p(\mathbb{R}^2, \mathbb{R}^2), \|\cdot\|_{p,\infty})$ is a continuous linear map, i.e., an embedding.) Given this, we let

$$W_m \triangleq \hat{\xi}_m(V) = \{v \circ m : v \in V\}. \tag{17.2}$$

We will let ξ_m denote the restriction of the infinitesimal action map, $\hat{\xi}_m$, to V.

Given $\zeta \in W_m$, we now define

$$\|\zeta\|_m = \min\{\|v\|_V : \zeta = v \circ m\}, \tag{17.3}$$

which represents the minimum cost (as measured by the norm in V) needed to represent ζ as an infinitesimal deformation of m. Equivalently, if we let $\mathcal{N}_m$ denote the null space of ξ_m, containing all $v \in V$ such that $v \circ m = 0$, then one can check that

$$\|v \circ m\|_m = \|\pi_{\mathcal{N}_m^\perp}(v)\|_V, \tag{17.4}$$

where $\pi_{\mathcal{N}_m^\perp}$ denotes the orthogonal projection (for the inner-product in V) on the space perpendicular to $\mathcal{N}_m$. Because $v \circ m = v' \circ m$ if $v - v' \in \mathcal{N}_m$, one has

$$W_m = \xi_m(\mathcal{N}_m^\perp) \tag{17.5}$$

with $\|v \circ m\|_m = \|v\|_V$ if $v \in \mathcal{N}_m^\perp$. This provides our basic construction, upon which we will build in the rest of the chapter, and which is summarized in the following definition.

Definition 17.4 (Basic Sub-Riemannian Metric on Shape Space) Let V be a Hilbert space continuously embedded in $C_0^p(\mathbb{R}^2, \mathbb{R}^2)$. Let $\xi_m v = v \circ m$ and

$$\mathcal{N}_m = \{v \in V : \xi_m v = 0\}.$$

One defines a sub-Riemannian structure on Emb^r by letting $W_m = \xi_m(V) = \xi_m(\mathcal{N}_m^\perp)$, with norm defined by $\|\xi_m v\|_m = \|v\|_V$ for $v \in \mathcal{N}_m^\perp$.

By definition, a geodesic between two shapes m_0 and m_1 must minimize (with respect to $t \mapsto m(t)$)

$$\int_0^1 \|\partial_t m\|^2_{m(t)}\, dt$$

subject to $m(0) = m_0$, $m(1) = m_1$ and $\partial_t m(t) \in W_{m(t)}$. From the definition of $\|\cdot\|_m$, one may also rephrase this optimization problem as minimizing (with respect to $t \mapsto v(t)$)

$$\int_0^1 \|v(t)\|_V^2\, dt$$

subject to $m(0) = m_0$, $m(1) = m_1$ and $\partial_t m(t) = v \circ m(t)$.

We have so far only considered sub-Riemannian geodesics in Emb$^r(S^1, \mathbb{R}^2)$, which is a space of parametrized shapes. When comparing geometric objects, one needs to identify two curves m and $\tilde{m}$ such that $m = \tilde{m} \circ \psi$, where ψ is a C^r diffeomorphism of S^1, i.e., $\psi \in$ Diff$^r(S^1)$, leading to the minimization (with respect to $t \mapsto v(t)$ and ψ) of

$$\int_0^1 \|v(t)\|_V^2\, dt$$

subject to $m(0) = m_0$, $m(1) = m_1 \circ \psi$ and $\partial_t m(t) = v \circ m(t)$.

This results in a very challenging optimization problem, both numerically and theoretically. It is, for example, difficult to determine whether a shape m_1 is attainable from a given m_0 through a curve with finite energy. Such issues disappear, however, if one relaxes the fixed end-point constraint and replaces it with a penalty. Introducing an end-point cost $(m, \tilde{m}) \mapsto U(m, \tilde{m})$ that measures the difference between m and $\tilde{m}$, the new problem becomes minimizing (with respect to $t \mapsto v(t)$ and ψ)

$$\frac{1}{2}\int_0^1 \|v(t)\|_V^2\, dt + U(m(1), m_1 \circ \psi)$$

subject to $m(0) = m_0$ and $\partial_t m(t) = v \circ m(t)$. If, in addition, the function U only depends on the geometry of the compared curves, ensuring that $U(m, \tilde{m}) = U(m, \tilde{m} \circ \psi)$ for $\psi \in \mathrm{Diff}^r(S^1)$, then the problem simplifies to a minimization in v of

$$\frac{1}{2}\int_0^1 \|v(t)\|_V^2\, dt + U(m(1), m_1)$$

subject to $m(0) = m_0$ and $\partial_t m(t) = v \circ m(t)$. This is the basic optimal control problem leading to the large deformation diffeomorphic metric mapping (LDDMM) framework for curves, discussed, in particular, in [24].

17.2.3 Generalization

A very similar discussion can be made for parametrized sub-manifolds in a d-dimensional space, i.e., the space $\mathrm{Emb}(\mathcal{S}, \mathbb{R}^d)$ of embeddings $m : \mathcal{S} \to \mathbb{R}^d$ where $\mathcal{S}$ (which replaces S^1) is a k-dimensional manifold providing the parameter space. The space V in this case is a Hilbert space of vector fields in $\mathbb{R}^d$ embedded in $C_0^p(\mathbb{R}^d, \mathbb{R}^d)$ for $p \geq 1$. This leads to instances of the LDDMM algorithm for manifolds. The important case of $\mathcal{S} = S^2$ (the two-dimensional unit sphere) and $d = 3$ provides closed surfaces of genus 0 in $\mathbb{R}^3$, and is the framework of most applications in medical imaging.

Several geometric end-point costs that are amenable to computations have been proposed for such manifolds, many of them using shape representations as bounded linear forms on suitably chosen reproducing kernel Hilbert spaces (and defining U using the associated norm on their dual spaces). They include representations as measures [23], currents [58], varifolds [14, 30] and more recently normal cycles [47]. We refer the reader to the cited papers for more details.

The existence of optimal geodesics in such frameworks has been discussed in several places (see [5, 19, 56, 65], for example) and is essentially ensured by the embedding assumption of the Hilbert space V.

The special case when $\mathcal{S} = \{s_1, \ldots, s_N\}$ is a finite set is especially interesting, because it applies to most numerical implementations of the previous family of problems (and also to situations in which the compared structures are simply point sets, or landmarks). In this case, the set $\mathcal{N}_m^\perp$ in Definition 17.4 can be explicitly described in terms of the reproducing kernel of the space V, requiring a few additional definitions.

One says that a Hilbert space H of functions from a domain Ω in $\mathbb{R}^d$ to $\mathbb{R}^k$ is a reproducing Kernel Hilbert space (RKHS) if the evaluation maps $\delta_x : H \to \mathbb{R}^k$ defined by $\delta_x(v) = v(x)$ are continuous. For such spaces, the Riesz representation theorem implies that, for any $x \in \Omega$ and any $a \in \mathbb{R}^k$, the linear form $v \mapsto a^T v(x)$ can be represented using the inner product in H as a mapping $v \mapsto \left\langle w_{x,a}\,,\ v\right\rangle_H$ for some $w_{x,a} \in H$. Because, for any $y \in \Omega$, the transformation $a \mapsto w_{x,a}(y)$ is linear from $\mathbb{R}^k$ to itself, this representation can be expressed in the form $w_{x,a}(y) = K_H(y, x)a$ where $K_H : \Omega^2 \to \mathbf{M}_k(\mathbb{R})$ takes values in the space of $k \times k$ real matrices and is called the reproducing kernel of H. It satisfies the equation

$$\langle K_H(\cdot, x)a\,,\ K_H(\cdot, y)b\rangle_H = a^T K_H(x, y)b \tag{17.6}$$

which will be used several times in this paper. In practice, one prefers RKHS's for which the kernel is known explicitly, and they are used extensively in approximation theory, machine learning or random field modeling. The theory of reproducing kernels Hilbert spaces was introduced in [7], and has been described since then in many books and papers, such as [12, 15, 18, 38, 48, 60]. The reader can also refer to [41, 65] for a discussion including vector-valued functions.

Returning to V, the assumption that this space is continuously embedded in $C_0^p(\mathbb{R}^d, \mathbb{R}^d)$ obviously implies that the evaluation mapping is continuous, so that V is an RKHS, and we will let $K = K_V$ denote its kernel, therefore a function defined on $\mathbb{R}^d \times \mathbb{R}^d$ and taking values in $\mathbf{M}_d(\mathbb{R})$. The space $\mathcal{N}_m$ is, by definition, the space of all functions $v \in V$ such that $v(m(s)) = 0$ for all $s \in \mathcal{S}$, which, using the kernel representation, makes it the space perpendicular to the vector space

$$V_m^{(r)} \triangleq \operatorname{span}(K(\cdot, m(s))\alpha : s \in \mathcal{S},\ \alpha \in \mathbb{R}^d).$$

If $\mathcal{S}$ is finite, then $V_m^{(r)}$ is finite dimensional, hence closed in V, and $\mathcal{N}_m^\perp = \left(\left(V_m^{(r)}\right)^\perp\right)^\perp = V_m^{(r)}$. (In the general case, $\mathcal{N}_m^\perp$ is the closure of $V_m^{(r)}$ in V, which is seldom known explicitly.)

As a consequence, the optimal solution $t \mapsto v(t)$ of the LDDMM problem must, when $\mathcal{S}$ is finite, take the form

$$v(t) = \sum_{s\in\mathcal{S}} K(\cdot, m(t, s))\alpha(t, s)$$

where $(\alpha(t, s), s \in \mathcal{S})$ is a collection of vectors in $\mathbb{R}^d$. One can use this property to reparametrize the problem in terms of a time-dependent family of vectors

$(\alpha(t,s), s \in \mathcal{S})$, minimizing (using (17.6))

$$\frac{1}{2}\int_0^1 \sum_{s,s'\in\mathcal{S}} \alpha(t,s)^T K(m(t,s), m(t,s'))\alpha(t,s')\, dt + U(m(1), m_1)$$

subject to $m(0) = m_0$ and

$$\partial_t m(t,s) = \sum_{s'\in\mathcal{S}} K(m(t,s), m(t,s'))\alpha(t,s')$$

for all $s \in \mathcal{S}$.

We note that, in this case, we have obtained an explicit representation of the distribution $W_m = \{v \circ m : v \in V\}$ in the form $W_m = \{v \circ m : v \in V_m^{(r)}\}$ where

$$V_m^{(r)} = \left\{ \sum_{s\in\mathcal{S}} K(\cdot, m(s))\alpha(s) : \alpha(s) \in \mathbb{R}^d, s \in \mathcal{S} \right\}. \tag{17.7}$$

This fully describes the basic reduction of the LDDMM sub-Riemannian framework in the case of point sets. The same reduction is not always explicitly available for infinite-dimensional structures, or, at least, not in full generality. However, under some assumptions on the data attachment term U (namely that it is differentiable, in an appropriate space, with respect to its first variable), which is satisfied in most practical cases, one can show that, at all times t, the optimal vector field $v(t)$ belongs to the space $V_{m(t)}^{(r)}$ where $V_m^{(r)}$ is defined by

$$V_m^{(r)} = \left\{ v = \int_{\mathcal{S}} K(\cdot, m(s))\alpha(s) d\mu_0(s) : \ \alpha : \mathcal{S} \to \mathbb{R}^d \right\}. \tag{17.8}$$

Here, μ_0 is the volume measure on $\mathcal{S}$ and the functions α in the definition of $V_m^{(r)}$ are assumed to be measurable and to satisfy

$$\int_{\mathcal{S}\times\mathcal{S}} \alpha(m(s))^T K(m(s), m(s'))\alpha(s') d\mu_0(s) d\mu_0(s') < \infty.$$

Note that, unlike the case of finite point sets, $V_m^{(r)}$ here is not equal to $\mathcal{N}_m^{\perp}$, but turns out to be the space that contain the optimal solution for "good" data attachment terms. This however shows that, in such contexts, there is no loss of generality in restricting oneself to the distribution on $\mathrm{Emb}^r(\mathcal{S}, \mathbb{R}^d)$ given by

$$W_m^{(r)} = \{v \circ m : v \in V_m^{(r)}\} \subset W_m$$

(where the inclusion is strict in general) when considering the solutions of the LDDMM problem.

In the sections that follow, we will consider various sub-Riemannian frameworks that differ from LDDMM in at least one of the following features.

1. Letting M denote the (pre-)shape space, e.g., $\mathrm{Emb}(\mathcal{S}, \mathbb{R}^d)$, introduce an extended space $\hat{M}$, on which diffeomorphisms also act, with an action denoted $(\varphi, \hat{m}) \mapsto \varphi \cdot \hat{m}$ and infinitesimal action $(v, \hat{m}) \mapsto v \cdot \hat{m}$. We will also ensure the existence of a surjection $\pi : \hat{M} \to M$ satisfying $\pi(\varphi \cdot \hat{m}) = \varphi \circ \pi(\hat{m})$. Typically, $\hat{m} \in \hat{M}$ will take the form $\hat{m} = (m, \theta)$, where θ is a "geometric attribute," with $\pi(m, \theta) = m$ and the first component of $\varphi \cdot (m, \theta)$ is $\varphi \circ m$.
2. Associate to each extended shape a space $V_{\hat{m}}^{(s)} \subset V$, typically of finite dimensions, which will be designed with the dual role of reducing the complexity of the original problem and of allowing for fine and interpretable modeling of the deformations that are physically or biologically relevant for a given application. The distribution in the extended shape space will then be
$$W_{\hat{m}}^{(s)} = \{v \cdot \hat{m} \; : \; v \in V_{\hat{m}}^{(s)}\}.$$
3. Replace $\|v\|_V$ in (17.3) (properly generalized to $\hat{M}$) by another norm $\|v\|_{\hat{m}}$ defined for all $v \in V_{\hat{m}}^{(s)}$, which is shape dependent and allows for increased modeling flexibility for the deformation process. The shape-dependent norms are generally assumed to control the V norm (i.e., $\|v\|_V \leq c\|v\|_{\hat{m}}$ for some $c > 0$), an assumption which is usually sufficient to ensure the existence of sub-Riemannian geodesics.
4. Given two shapes m_0 and m_1, and an initial extended shape $\hat{m}_0$ such that $\pi(\hat{m}_0) = m_0$, one then minimizes
$$\frac{1}{2}\int_0^1 \|v(t)\|_{\hat{m}(t)}^2 \, dt + U(\pi(\hat{m}(1)), m_1)$$
subject to $\hat{m}(0) = \hat{m}_0$ and $\partial_t \hat{m}(t) = v(t) \cdot m(t)$. (Alternatively, instead of assuming that $\hat{m}_0$ is given, one can treat it as a partially free variable with the constraint $\pi(\hat{m}(0)) = m_0$.)

As an immediate example, the basic reduction of LDDMM falls in this framework, with $\hat{M} = M$, $\hat{m} = m$, $V_{\hat{m}}^{(s)} = V_m^{(r)}$ and $\|v\|_{\hat{m}} = \|v\|_V$.

Remark 17.5 A general framework for infinite-dimensional shape spaces with precise regularity assumptions on the considered objects was described in [3, 5]. In particular it builds the LDDMM sub-Riemannian metric and develops the basic LDDMM reduction in the general case of a generic shape space. We refer to this paper for more technical details on the relevant theory, while we remain in this chapter at a semi-informal level in terms of theoretical accuracy.

17.2.4 Pontryagin's Maximum Principle

The minimization problems that were sketched in the previous section are special cases of optimal control problems, which generically take the form: Minimize

$$\int_0^1 g(q(t), u(t))\, dt + G(q(1)) \tag{17.9}$$

subject to $q(0) = q_0$ and $\partial_t q(t) = f(q(t), u(t))$. Here, q is the state, belonging to a state space Q (assumed to simplify to form an open subset of a finite-dimensional space $\boldsymbol{Q}$) and u is the control, belonging to a space $\mathcal{U}$. The function $g(q, u)$, which takes values in $[0, +\infty)$, is the state-dependent cost associated to the control, and $f(q, u)$, which takes values in $\mathbf{Q}$, determines the state evolution equation for a given control. More precisely, the control $u(t)$ specifies at time t a direction $f(q(t), u(t))$ for the evolution of $q(t)$ with $f(q(t), \mathcal{U}) = W_{q(t)}$. Then, under mild assumptions, since (17.9) is a constrained minimization problem under dynamic equality constraints, there exists a time-dependent $p : [0, 1] \to \boldsymbol{Q}$ (the co-state), such that the problem boils down to an unconstrained minimization problem for the Lagrangian

$$\begin{aligned}\mathcal{L}(q(.), p(.), u(.)) &= \int_0^1 (g(q, u) + p^T(\partial_t q - f(q, u)))dt + G(q(1))\\ &= [p^T q]_0^1 - \int_0^1 (\partial_t p^T q + H_u(q, p))dt + G(q(1))\end{aligned}$$

where for $u \in \mathcal{U}$ the function H_u on $\boldsymbol{Q} \times Q$ (the Hamiltonian) is defined by

$$H_u(p, q) = p^T f(q, u) - g(q, u)\,.$$

Now, the optimality conditions $\partial_{p(.)}\mathcal{L} = \partial_{u(.)}\mathcal{L} = \partial_{q(.)}\mathcal{L} = 0$ induce the following equations

$$\begin{cases}\partial_t q(t) = \partial_p H_{u(t)}(p(t), q(t)) = f(q(t), u(t))\\ \partial_t p(t) = -\partial_q H_{u(t)}(p(t), q(t)) = \partial_q g(q(t), u(t)) - \partial_q f(q(t), u(t))^T p(t)\\ \partial_u H_{u'}(p(t), q(t)) = 0\end{cases}$$

with the boundary conditions $q(0) = q_0$ and $p(1) = -\partial_q G(q(1))$. In our case, $u \mapsto g(q(t), u(t))$ is convex so that $\partial_h H(q, u) = 0$ corresponds to a maximum of the Hamiltonian (i.e. $u(t) = \operatorname{argmax}_{u'} H_{u'}(p(t), q(t)))$ which is a particular case of the Pontryagin's maximum principle [1, 36, 37, 59].

The maximum principle also provides equations for the computation of the gradient of the optimal control problem's objective function. More precisely, let

$$F(u(\cdot)) = \int_0^1 g(q(t), u(t))\, dt + G(q(1)) \tag{17.10}$$

in which q is considered as a function of $u(\cdot)$ through the equations $\partial_t q = f(q, u)$, $q(0) = q_0$. Then

$$\partial_{u(\cdot)} F(u(\cdot))(t) = \partial_u g(q(t), u(t)) - \partial_u f(q(t), u(q))^T p(t) \tag{17.11}$$

where $p(\cdot)$ and $q(\cdot)$ are obtained by solving $\partial_t q = f(q, u)$, with $q(0) = 0$, followed by $\partial_t p = \partial_q g(q(t), u(t)) - \partial_q f(q(t), u(t))^T p(t)$, with $p(1) = -\partial_q G(q(1))$. (This computation is often referred to as the "adjoint method" in optimal control.)

Returning to LDDMM, and to the case of finite $\mathcal{S}$, the state is $q(t) = m(t) = (m(t, s), s \in \mathcal{S})$. Writing $v(t) \in V^{(r)}_{m(t)}$ in the form

$$v(t, \cdot) = \sum_{s \in \mathcal{S}} K(\cdot, m(t, s))\alpha(t, s),$$

one can use $u(t) = (\alpha(t, s), s \in \mathcal{S})$ as control, with

$$\|v(t)\|_V^2 = \sum_{s, s' \in \mathcal{S}} \alpha(t, s)^T K(m(t, s), m(t, s'))\alpha(t, s').$$

Using this identity, and introducing the co-state p, one has

$$\begin{aligned} H_{u(t)}(p(t), m(t)) = &\sum_{s, s' \in \mathcal{S}} p(t, s)^T K(m(t, s), m(t, s'))\alpha(t, s') \\ &- \frac{1}{2} \sum_{s, s' \in \mathcal{S}} \alpha(t, s)^T K(m(t, s), m(t, s'))\alpha(t, s')\,. \end{aligned}$$

The third condition of the PMP implies that optimal controls must satisfy $p = \alpha$, and that m and α should be such that

$$\begin{cases} \partial_t m(t, s) = \displaystyle\sum_{s' \in \mathcal{S}} K(m(t, s), m(t, s'))\alpha(t, s'), \ \ s \in \mathcal{S} \\ \partial_t \alpha(t, s) = -\partial_q \left(\displaystyle\sum_{s' \in \mathcal{S}} \alpha(t, s)^T K(q, m(t, s'))\alpha(t, s') \right)_{q = m(t, s)}, \ \ s \in \mathcal{S} \end{cases} \tag{17.12}$$

Notice that, as a result, each evaluation of the right-hand side of the system of ODE's involved in the adjoint method requires an order of $|\mathcal{S}|^2$ computations.

17.3 Approximating Distributions

17.3.1 Control Points

We now start discussing approaches that apply the program detailed at the end of Sect. 17.2.3 and modify the basic LDDMM framework by specifically designing distributions and/or metrics in this sub-Riemannian context. The first methods we address are motivated by computational efficiency. Indeed, when one works with fine discretizations of curves or surfaces, the basic reduction may still involve too many points to allow for efficient implementations on sequential computers (recent advances in GPU parallelization have however pushed the size limit significantly further; see [13, 34, 57]).

In [63, 64], it is proposed to use small-dimensional approximations of $V_m^{(r)}$ in (17.8) when the approximation obtained after discretization (given by (17.7)) is still too high dimensional. In this framework, one introduces a space Θ of geometric attributes, with an action of diffeomorphisms $(\varphi, \theta) \mapsto \varphi \cdot \theta$. To each $\theta \in \Theta$ is associated a family of vector fields $\zeta_\theta^1(\cdot), \dots, \zeta_\theta^n(\cdot) \in V$, and one lets, for $\hat{m} = (m, \theta)$:

$$V_{\hat{m}}^{(s)} = \left\{ \sum_{i=1}^{n} h_i \zeta_\theta^i \ : \ h \in \mathbb{R}^n \right\} \tag{17.13}$$

Assuming that $\mathcal{S}$ is finite, the basic reduction corresponds to the case $\theta = m$ (so that $\hat{M}$ is the diagonal of $M \times M$), and the vector fields generating $V_{\hat{m}}^{(s)}$ are provided by all the columns of the matrices $K(\cdot, m(s))$, $s \in \mathcal{S}$ (so that $n = d|\mathcal{S}|$ in this case). To reduce the dimension, one can set $\theta = (c(1), \dots, c(k))$, a family of k "control points" in $\mathbb{R}^d$ and use the kd-dimensional space provided by the columns of the matrices $K(\cdot, c(j))$, $j = 1, \dots, k$. Let us make explicit the registration problem in this special case, using $\|v\|_{\hat{m}} = \|v\|_V$. A generic element $v \in V_{\hat{m}}^{(s)}$ takes the form $v(\cdot) = \sum_{j=1}^k K(\cdot, c(j))\alpha(j)$ for $\alpha(1), \dots, \alpha(k) \in \mathbb{R}^d$. Because K is the reproducing kernel of V, one has, using (17.6)

$$\|v\|_V^2 = \sum_{i,j=1}^{k} \alpha(i)^T K(c(i), c(j))\alpha(j)\,.$$

Given two shapes m_0 and m_1 and an initial family of control points c_0, one then needs to minimize

$$\frac{1}{2}\int_0^1 \sum_{i,j=1}^{k} \alpha(t,i)^T K(c(t,i), c(t,j))\alpha(t,j)\, dt + U(m(1), m_1)$$

subject to $m(0) = m_0$, and

$$\begin{cases} \partial_t m(t,s) = \sum_{j=1}^{k} K(m(t,s), c(t,j))\alpha(t,j), & s \in \mathcal{S} \\ \partial_t c(t,i) = \sum_{j=1}^{k} K(c(t,i), c(t,j))\alpha(t,j), & i \in \{1,\dots,k\} \end{cases} \tag{17.14}$$

Note that the right-hand side of these equations now involves an order of $k|\mathcal{S}|$ computations (with k typically much smaller than $|\mathcal{S}|$) instead of $|\mathcal{S}|^2$.

To apply the PMP to this problem, one must introduce a co-state variable p that has the same dimension as the state, decomposing as $\hat{p} = (p_m, p_c)$ for the two components of $\hat{m} = (m, c)$. The Hamiltonian in this case takes the form

$$\begin{aligned} H_\alpha(\hat{p}, \hat{m}) = &\sum_{s\in\mathcal{S}} \sum_{j=1}^{k} p_m(s)^T K(m(s), c(j))\alpha(j) + \sum_{i,j=1}^{k} p_c(i)^T K(c(i), c(j))\alpha(j) \\ &- \frac{1}{2} \sum_{i,j=1}^{k} \alpha(i)^T K(c(i), c(j))\alpha(j). \end{aligned}$$

The PMP implies that, in addition to (17.14) the following equations are satisfied by optimal controls:

$$\partial_t p_m(t) = -\sum_{s\in\mathcal{S}} \sum_{j=1}^{k} \partial_q (p_m(t,s)^T K(q, c(t,j))\alpha(t,j))_{q=m(t,s)}, \tag{17.15}$$

$$\begin{aligned} \partial_t p_c(t,i) = &-\sum_{s\in\mathcal{S}} \sum_{j=1}^{k} \partial_q (p_m(t,s)^T K(m(t,s), q)\alpha(t,j))_{q=c(t,j)} \\ &- \sum_{i,j=1}^{k} \partial_q (p_c(t,i)^T K(q, c(t,j))\alpha(t,j))_{q=c(t,i)} \\ &- \sum_{i,j=1}^{k} \partial_q (\alpha(t,i)^T K(q, c(t,j)) p_c(t,j))_{q=c(t,i)} \\ &+ \sum_{i,j=1}^{k} \partial_q (\alpha(t,i)^T K(q, c(t,j))\alpha(t,j))_{q=c(t,i)}, \end{aligned} \tag{17.16}$$

and $\sum_{s\in\mathcal{S}} K(c(t,j),m(t,s))p_m(t,s)+\sum_{i=1}^{k} K(c(t,i),c(t,j))(p_c(t,j)-\alpha(t,j)) = 0$

with $p_m(1) = -\partial_q U(q,m_1)_{q=m(1)}$ and $p_c(1) = 0$. Equations (17.15) and (17.16) also describe the computation required to evaluate the derivative of the objective function, with (17.11) becoming

$$\begin{aligned}\partial_{u(\cdot)}F(u(\cdot))(t) = &\sum_{i=1}^{k} K(c(t,i),c(t,j))(\alpha(t,j)-p_c(t,j))\\ &-\sum_{s\in\mathcal{S}} K(c(t,j),m(t,s))p_m(t,s).\end{aligned}$$

Here also, the complexity has order $k|\mathcal{S}|$.

Control points have also been introduced in [20–22], where the authors took additional steps in order to reduce the computational complexity. In these references, the authors use (17.12) applied to control points only as a constraint in the optimization problem, i.e., they enforce the equations

$$\begin{cases}\partial_t c(t,i) = \sum_{j=1}^{k} K(c(t,i),c(t,j))\alpha(t,j)\\ \partial_t\alpha(t,i) = -\partial_q\left(\sum_{j=1}^{k}\alpha(t,i)^T K(q,c(t,j))\alpha(t,j)\right)_{q=c(t,i)}\end{cases}\tag{17.17}$$

These constraints imply that $\sum_{i,j=1}^{k}\alpha(t,i)^T K(c(t,i),c(t,j))\alpha(t,j)$ is constant over time, and the objective function becomes

$$\frac{1}{2}\sum_{i,j=1}^{k}\alpha(0,i)^T K(c(0,i),c(0,j))\alpha(0,j) + U(m(1),m_1)$$

which is minimized (with respect to $\alpha(0)$) subject to (17.17), $m(0) = m_0$ and $\partial_t m(t,s) = \sum_{j=1}^{k} K(m(t,s),c(t,j))\alpha(t,j)$. We refer to the cited references for a full description of this problem (which is not sub-Riemannian anymore) and its solution. Note that, in these references, the authors also address the issue of optimizing for the best positioning of the initial control points given the initial m_0 and a collection of target shapes m_1.

17.3.2 Scale Attributes

In most applications, the reproducing kernel K is a radial basis function, i.e., it takes the form

$$K(x, y) = \Gamma\left(\frac{|x-y|}{\sigma}\right)$$

for some function Γ. For example, the function $\Gamma(u) = e^{-u^2/2}\mathrm{Id}_{\mathbb{R}^d}$ (where $\mathrm{Id}_{\mathbb{R}^d}$ is the identity matrix in $\mathbb{R}^d$) provides a Gaussian kernel. Alternatively, Matérn kernels of order ℓ are such that

$$\Gamma(u) = P_\ell(u)e^{-|u|}\mathrm{Id}_{\mathbb{R}^d}$$

where P_ℓ is a reverse Bessel polynomial of order ℓ.

In such cases, the function $x \mapsto K(x, c)$ quickly vanishes when x is far from c. In order to be able to use sparse sets of control points while making sure that no gap is created in the velocity field, one needs to use generating functions in $V_{\hat{m}}^{(s)}$ who decay more slowly than the original kernel K. A simple choice is to let $V_{\hat{m}}^{(s)}$ be generated by the columns of the matrices $\tilde{K}(\cdot, c_j)$, $j = 1, \ldots, k$, with

$$\tilde{K}(x, y) = \Gamma\left(\frac{|x-y|}{\tilde{\sigma}}\right)$$

and $\tilde{\sigma} > \sigma$. The previously considered minimization problem generalizes immediately, with one difficulty, which is that, in order to compute $\|v\|_V^2$ for some $v \in V_{\hat{m}}^{(s)}$, one needs to have a closed form expression of inner products of the form

$$\left\langle \tilde{K}(\cdot, c)\alpha\,,\ \tilde{K}(\cdot, c')\alpha' \right\rangle_V$$

for c, c', α, α' in $\mathbb{R}^d$ to ensure that the computation is tractable. (This product is simply equal to $\alpha^T K(c, c')\alpha'$ when $\tilde{K} = K$.) Such a closed form can be obtained for Gaussian kernels, as described in [8, 53].

Returning to the basic reduction for curves or surfaces, for which the representation in (17.7) can be seen as a discretization of (17.8), it is natural to expect that a basis of $V_{\hat{m}}^{(s)}$ associated with a sparse discretization should scale anisotropically, with a preferred direction tangent to the deforming manifold. In [63, 64], it is suggested to use basis functions provided by columns of the matrices

$$\chi(x; S_j, c_j) = \Gamma\left(\frac{\left|(\sigma^2\mathrm{Id}_{\mathbb{R}^d} + S_k)^{-1/2}(x-c_j)\right|}{2}\right), j = 1, \ldots, k$$

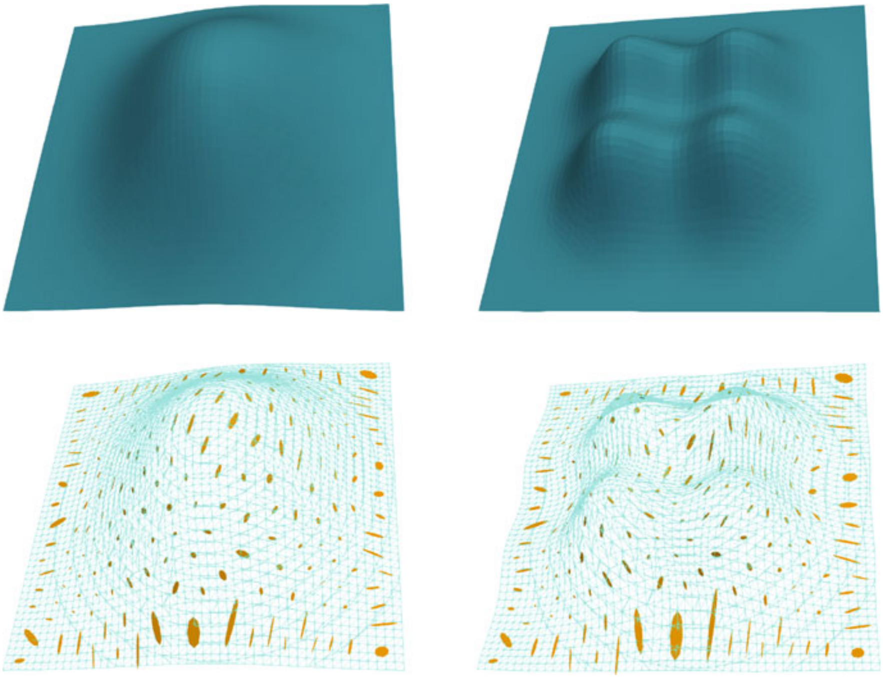


Fig. 17.1 Surface registration using anisotropic diffeons. First row: Initial and target surfaces. Second row: Initial and final surfaces with superimposed glyphs representing the rank-two diffeon matrices $S_1, \ldots, S_k$ visualized as ellipses at positions $c_2, \ldots, c_k$

with parameters $\theta = (S_1, c_1, \ldots, S_k, c_k)$, $S_1, \ldots, S_k$ being non-negative symmetric matrices that vanish when applied to vectors normal to the manifold. The action of a diffeomorphism on θ is $\varphi \cdot \theta = (S_1', c_1', \ldots, S_k', c_k')$ with $S_j' = d\varphi(c_j)S_j d\varphi(c_j)^T$ and $c_j' = \varphi(c_j)$. The resulting minimization problem is detailed in the cited references. Here again, the inner product between these basis functions has a closed form when V has a Gaussian kernel. An example of surface registration using this approach is presented in Fig. 17.1.

Remark 17.6 As suggested by the complexity of Eq. (17.16), the analytical evaluation of the differential of the objective function for these control problems becomes increasingly involved when the geometric attributes that specify the basis functions are refined. (The reader may want to refer to [64] where such differentials are made explicit.) In this regard, it is important to point out recent developments of automatic differentiation software specially adapted to kernel computations [13], which significantly simplify the implementation of such methods.

17.4 Deformation Modules

In the previous sections, the sub-Riemannian point of view emerged naturally for its links with low-dimensional approximations of the tangent bundle that are important for numerically efficient algorithms to compute geodesic paths. This point of view can, however, also be used for modeling purposes, for which it provides a powerful set of tools.

Indeed, a geodesic path computed between two shapes results from the action of a path $\varphi(t)$ of diffeomorphisms on the starting shape inducing a complex global transformation process of the shape and its environment (interior, exterior, subvolumes, etc.). This process is obviously of strong interest: for instance local change of volume, distortions or global large scale changes are potentially important markers of underlying transformation processes and connected to interpretable variables. In this regard, the design of good and interpretable sub-Riemannian structures is an important issue connecting geometry to modeling and analysis. We now describe recent attempts in this direction that emphasize modularity in the design of sub-Riemannian structures, introducing *deformation modules* that capture desirable and interpretable elementary local or global behavior of the shapes. This framework (that can be sourced again in Grenander's Pattern Theory [26]) can furthermore be organised in a hierarchical way.

This point of view follows the previously introduced idea of augmenting a observable shape $m \in M$ with new geometric attributes $\theta \in \Theta$ to produce an augmented shape $\hat{m} \in \hat{M} \overset{\Delta}{=} M \times \Theta$ where Θ is itself a new shape.

17.4.1 Definition

An abstract deformation module $\mathcal{M}$ involves a space of geometric descriptors Θ, a finite-dimensional control space H and a *field generator* ζ associating to any geometric descriptor $\theta \in \Theta$ and control parameter $h \in H$ a vector field $\zeta_\theta(h) \in C_0^p(\mathbb{R}^d, \mathbb{R}^d)$ on $\mathbb{R}^d$ (so that ζ can be seen as a function $\zeta : \Theta \times H \to C_0^p(\mathbb{R}^d, \mathbb{R}^d)$). The field generator should be seen as an elementary interpretable source of transformation of the shape itself and several deformation modules can be combined and act simultaneously to produce a global vector field as we will see later.

During the transformation process, the geometric descriptor will be affected by the resulting vector field v, so that we assume that $\mathrm{Diff}_0^p(\mathbb{R}^r)$ acts on Θ and induces an infinitesimal action $\xi : \Theta \times C_0^p(\mathbb{R}^d, \mathbb{R}^d) \to T\Theta$. Given a geometric descriptor, the control h will be weighted by an energy function or a cost $c_\theta(h) \geq 0$. Assuming that $c_\theta(h)$ is a positive-definite quadratic form on H, we end up with a sub-Riemannian framework on Θ with distribution $W_\theta^\Theta = \zeta_\theta(H)$ and metric $\langle h, h \rangle_\theta = c_\theta(h)$.

To summarize, a deformation module $\mathcal{M}$ is a tuple $\mathcal{M} = (\Theta, H, \zeta, \xi, c)$, as schematically described in Fig. 17.2.

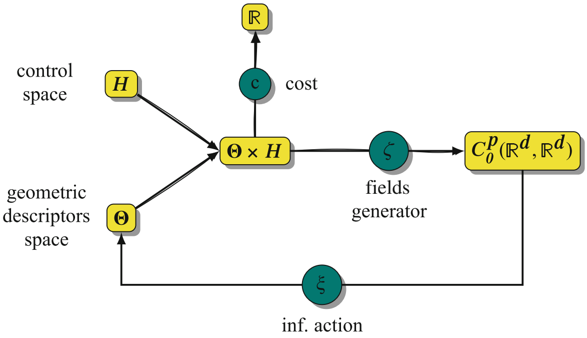

ℝ
c
cost
control space
H
Θ × H
ζ
$C_0^p(\mathbb{R}^d, \mathbb{R}^d)$
fields generator
geometric descriptors space
Θ
ξ
inf. action

Fig. 17.2 Deformation module overview. H is the control space, and $h \to \zeta_\theta(h)$ gives the contribution of the module to the instantaneous deformation fields when the module is geometrically instantiated by the geometric descriptor $\theta \in \Theta$. The cost for generating $\zeta_\theta(h)$ is $c_\theta(h)$

A key point in the construction is the possibility to combine simple modules into more complex ones using a direct sum. More specifically, if $\mathcal{M}^l = (\Theta^l, H^l, \zeta^l, \xi^l, c^l)$, $l = 1, \dots, L$, are L deformation modules of order p, their *compound module* is $C(\mathcal{M}^l, l = 1 \cdots L) = (\Theta, H, \zeta, \xi, c)$ where $\Theta \triangleq \prod_l \Theta^l$, $H \triangleq \prod_l H^l$ and for $\theta = (\theta^1, \dots, \theta^L) \in \Theta$, $h = (h^1, \dots, h^L) \in H$, $v \in C_0^\ell(\mathbb{R}^d)$,

$$\zeta_\theta(h) = \sum_{l=1}^{L} \zeta^l_{\theta^l}(h^l) \in C_0^\ell(\mathbb{R}^d),$$

$$\xi_\theta(v) = (\xi^1_{\theta^1}(v), \dots, \xi^L_{\theta^L}(v)) \in T_\theta\Theta,$$

$$c_\theta(h) = \sum_{l=1}^{L} \alpha_l c^l_{\theta^l}(h^l).$$

where the coefficients α_l are positive scalars that weight the costs of the different modules. By lowering a weight α_l, we favor the use of the corresponding module $\mathcal{M}^l$ in geodesics (trajectories minimizing the total cost).

Remark 17.7 The geometric descriptors will be in general the augmented part θ in the total augmented shape $\hat{m} = (m, \theta)$. But in order to have a unified framework, we will often interpret the input shape m as geometric descriptors of a special module, called a silent module, which is deformed according to its infinitesimal action, but does not contribute to the global velocity field (i.e., $H = \{0\}$). The combination of deformation modules defined above allows these silent shapes to deform according to the action of the vector fields generated by other modules.

The optimal control model associated with a deformation module requires the minimization of

$$\frac{1}{2}\int_0^1 c_{\theta(t)}(h(t))\, dt + D(\theta(1))$$

subject to an initial condition $\theta(0) = \theta_0$, and to the evolution equation $\partial_t\theta = \xi_\theta(\zeta_\theta(h))$. Here, the function D is an end-point cost, typically depending on the transported shape represented as a silent component in the module θ, as described in Remark 17.7. The initial condition θ_0 is partially fixed (e.g., the part that represents the initial shape), but several of its components can, and should, be optimized, as illustrated in Sect. 17.4.3.

The existence of solutions to this control problem can be guaranteed if one enforces a *uniform embedding condition* that requires that $\|\zeta_\theta(h)\|^2_{p,\infty} \leq Cc_\theta(h)$ uniformly on $\Theta \times H$ for some large enough p (see [27]). Note that one can always take $c_\theta(h) = \|\zeta_\theta(h)\|^2_V$, where V is defined as in Sect. 17.2, and the resulting cost satisfies the condition. However, this choice is not stable under module combination and would not allow for the use of the convenient module algebra that was defined above. In contrast the uniform embedding condition is stable under combination, in the sense that it is satisfies by a compound module as soon as it is satisfied by each of its components.

17.4.2 Basic Deformation Modules

The discussion of Sect. 17.3 immediately provides instances of deformation modules, with, using the notation already introduced in that section (e.g., Eq. (17.13)), $H = \mathbb{R}^n$, $\zeta_\theta(h) = \sum_{l=1}^n h^l \zeta_\theta^l$, $c_\theta(h) = \|\zeta_\theta(h)\|^2_V$, and the representation $\hat{m} = (m, \theta)$ is equivalent to a combination of θ with the silent module associated with m. Indeed, a simple example of modules is provided by what we refer to as "sums of local free translations." Here, given a finite set I, and a control $h = (h_i)_{i\in I} \in H = (\mathbb{R}^d)^I$, the module generates a vector field

$$v = \zeta_\theta(h) \text{ with } v(x) = \sum_{i\in I} K_\sigma(x, \theta_i)h_i$$

where $\theta = (\theta_i)_{i\in I} \in \Theta = (\mathbb{R}^d)^I$ is the list of the centers of the individual translations, and the kernel (e.g. a Gaussian kernel) with scale σ determines the scope of each local translation. One can take $c_\theta(h) = \sum_{i,j\in I} K_\sigma(\theta_i, \theta_j)h_i^T h_j$, which coincides with $\|v\|_{V_\sigma}$ for the RKHS V_σ associated with K_σ. The infinitesimal action is here straightforward: $\xi_\theta(w) = (w(\theta_i))_{i\in I}$. Note that, in this case, these modules are exactly those associated with control points in Sect. 17.3.

The flexibility of deformation modules, and their usefulness as modeling tools, is however provided by the ability to start with simple modules and to combine them into complex infinitesimal transformations.

A first illustration of such basic modules is provided by a fixed translation at some location c, with direction $u \in S^{d-1}$ (the unit sphere in $\mathbb{R}^d$), at a given scale σ. These modules are defined by $\theta = (c, u)$, $\Theta = \mathbb{R}^d \times S^{d-1}$, $H = \mathbb{R}$, and the vector field $v = \zeta_\theta(h)$ is given by $v(x) = hK_\sigma(x, c)u$ where K_σ is a kernel (e.g. a Gaussian kernel) at scale σ. In contrast to free translations described above, the

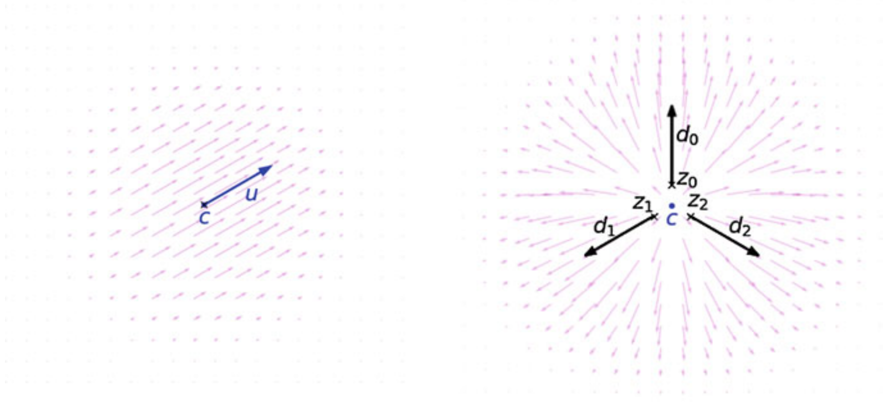


Fig. 17.3 Examples of simple deformation modules. Left: fixed translation where $\theta = (c, u)$; Right: local scaling where $\theta = c$. In both cases $\dim(H) = 1$

control is one-dimensional, constraining strongly the possible vector fields, but the geometric descriptor is $(d^2 - d)$-dimensional (see Fig. 17.2). The cost $c_\theta(h)$ can simply be chosen as ch^2 for a fixed constant c.

The choice of the infinitesimal action in this simple example is interesting. Indeed, c and u can be affected by the action of a vector field w in three different ways, providing three modeling options: (1) $\xi_\theta(w) = (w(c), 0)$ (no action on the direction), (2) $\xi_\theta(w) = (w(c), dw(c)u - (u^T dw(c)u)u)$ (advection of the direction u), or (3) $\xi_\theta(w) = (w(c), -dw^T(c)u + (u^T dw(c)u)u)$ (co-advection of the direction u considered as a normal direction to a hyperplane advected by the flow).

One can build basic deformation modules as local scaling or rotation in a similar way (see [27]), providing natural steps towards combinations of local affine deformations [9, 49, 68]. We present here the case of local scaling, which will be used in the example described in the next section. For such modules, one takes $\Theta = \mathbb{R}^d$, $H = \mathbb{R}$, $\zeta_\theta(h) \overset{\Delta}{=} h \sum_{j=1}^{3} K_\sigma(\cdot, z_j(\theta))d_j$, where K_σ is again possibly a Gaussian kernel at scale σ and, for $j \in \{0, 1, 2\}$, $d_j = (\sin(2j\pi/3), \cos(2j\pi/3))$ and $z_j(\theta) = \theta + \frac{\sigma}{3}d_j$. (See Fig. 17.3 for an illustration of this construction.) The infinitesimal action is given by $\xi_\theta : w \in C_0^p(\mathbb{R}^2) \mapsto w(\theta)$, the velocity field at the scaling center, and the cost by $c_\theta(h) = h^2 \sum_{j,j'} K_\sigma(z_j, z_{j'})d_j^T d_{j'}$. Different choices for vectors z_j lead to other types of local deformations.

17.4.3 Simple Matching Example

As an illustration, we consider a very simple but meaningful example with shapes evocating "peanut pods," displaying random variations of a simple template in the

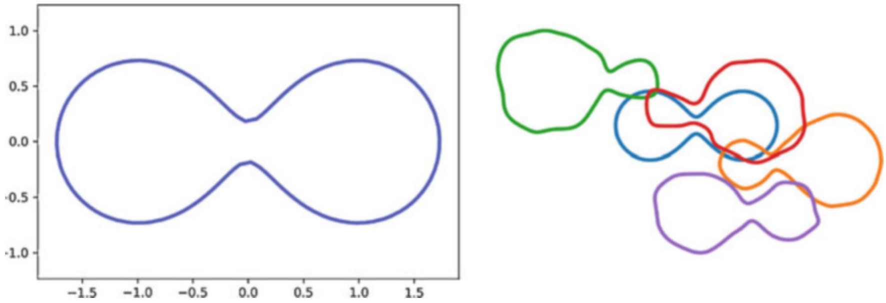


Fig. 17.4 Left: Peanut-pod shaped template. Right: Typical shape variations (with the template (cyan) in at the center)

size of left and right rounded ends (generated as level curves of the sum of two Gaussian functions) as well as a global translation misalignment and more random local perturbations of the boundary (see Fig. 17.4).

Here we see that the major effect is the differential dilations of the left and right parts that we model as a first deformation module $\mathcal{M}^{ls}$ given as the sum of two local scaling with geometric descriptors located at the center on the each rounded part and with scale $\sigma = 1$ associated with a control space H of dimension 2. Moreover, the global translation misalignment can be handled by a module $\mathcal{M}^{gt}$ defined by a free translation at a large scale centered at the center of the object and associated with a another control space of dimension 2. The two modules define the core parametric part of the modelling. On top of that, a third module $\mathcal{M}^{lt}$ given by a sum a free local translations at a small scale $\sigma = 0.2$ supported by geometric descriptors distributed along the boundary that can accommodate more arbitrary small scale variations. The cost for this latter deformation modules is assigned a larger weight than those of the first two so that deformations are preferably (when possible) generated by the scaling modules and the large translation.

In Fig. 17.5 we present the results of registration between the template and two different targets. For each target we consider two deformation models, using or not the local scaling module $\mathcal{M}^{ls}$. Without this module, the situation boils down to a classical LDMMM registration (modified by the global translation module $\mathcal{M}^{gt}$, which here is mandatory when target and template are not pre-aligned by rigid motion). Comparing the left (LDDMM) and right (parametric model with scaling modules) columns, one can clearly see that, even if both models yield a perfect registration, the underlying optimal diffeomorphisms acting on the template are quite different, in particular in the interior regions of the rounded ends. The module $\mathcal{M}^{ls}$ gives a preference, as expected, to shape changes due to local scaling within theses regions. Note also that all the modules interact to produce the final quasi-perfect matching. Of particular modeling interest is the possibility to decompose the overall deformation into its different components by following the flow generated by the corresponding module and its geodesic controls (see Fig. 17.6). This clarifies the

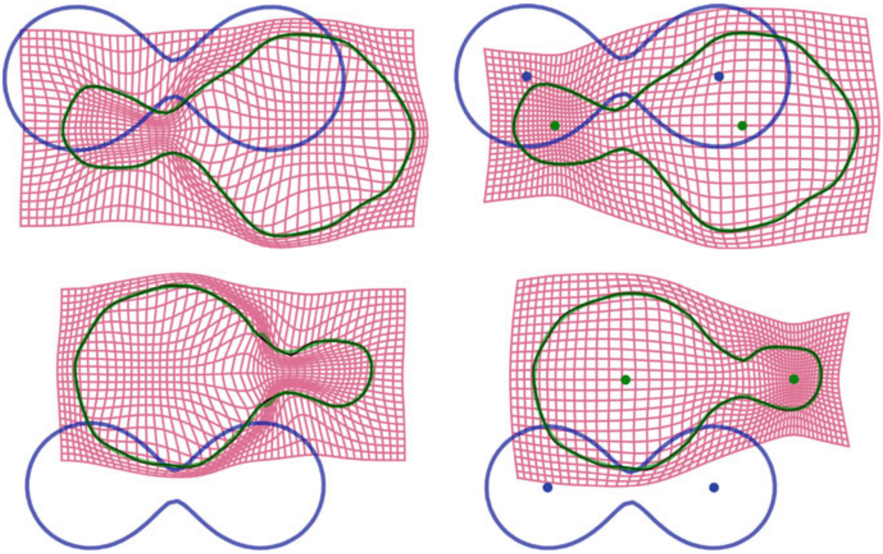


Fig. 17.5 Geodesic matching between template (in blue) and two different targets (top and bottom) in black. The final registered curve is in green. Left: LDDMM-like registration using only a combination of $\mathcal{M}^{gt}$ and $\mathcal{M}^{lt}$. Right: parametric registration with a combination of $\mathcal{M}^{ls}$ (scalings), $\mathcal{M}^{gt}$ and $\mathcal{M}^{lt}$. The initial and final centers of the scaling modules are represented as blue and green points, respectively. In both cases, a varifold data attachment term is considered [14]

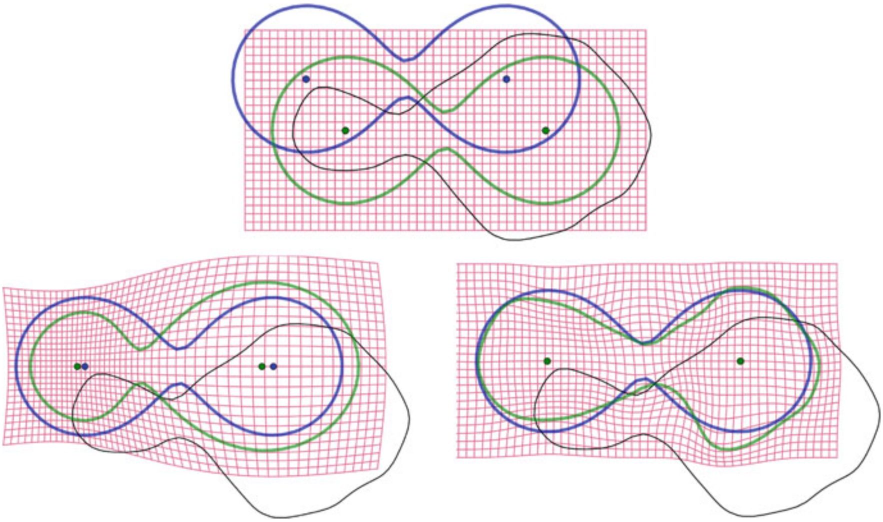


Fig. 17.6 Decomposition along submodules: the initial shape (and centers of the scaling modules) are in blue, the target is in black and the final curve (and centers) obtained following a single module are in green in green. First row: following the global translation $\mathcal{M}^{gt}$. Second row, Left: following the local scalings $\mathcal{M}^{ls}$; Right: following the local translations $\mathcal{M}^{lt}$ (see text for comments)

contribution of each module in the total deformation. In particular, the deformation generated by local translations illustrates how the data differs from the template in addition to the global translation and the standard variation of the sizes of the rounded parts.

17.4.4 Population Analysis

Given a population of shapes $(m_i^T)_{1\leq N}$, let us define the cost

$$J(\widehat{m}_0, h_1, \ldots, h_N) \triangleq \sum_{i=1}^{N} \left(\int_0^1 c_{\theta_i(t)}(h_i(t))dt + U(m_i(1), m_i^T) \right),$$

where $\hat{m}_i(t=0) = \hat{m}_0 = (m_0, \theta_0)$ for all i and $\partial_t \hat{m}_i(t) = \xi_{\hat{m}_i(t)} \zeta_{\theta_i(t)}(h_i(t))$. In the following, the shape m_0, called the template, is assumed to be fixed.

Minimizing this cost J with respect to the time-varying controls h_i and the common initial geometric descriptors θ_0 defines the *atlas* of the population of shapes $(m_i^t)_{1\leq N}$ for the chosen deformation model. The optimal geometric descriptor θ_0 corresponds to the best element, within the proposed vocabulary (defined by the deformation module), enabling one to describe the variation of the population with respect to the initial template m_0.

In addition, the variables parametrizing the geodesics transforming m_0 into each shape belong to a common vector space, and can be used to represent the population when performing a statistical study.

We present here the atlas of the population in Fig. 17.7, obtained with the same deformation module as in the previous section, i.e. combination of 3 modules: $\mathcal{M}^{gt}$ (global translations), $\mathcal{M}^{lt}$ (sum of local translations at scale 0.2) and $\mathcal{M}^{ls}$ (sum of two scalings at scale 1.). In Fig. 17.7 we show the initial positions of the centers of scaling, before and after optimization. We also present in Fig. 17.8 the final shapes

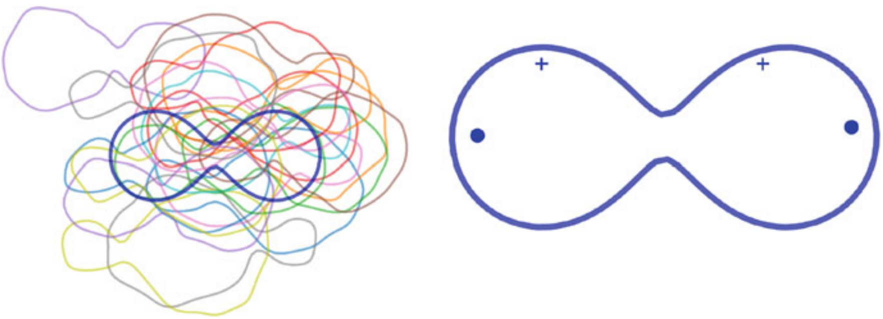


Fig. 17.7 Atlas with scalings of scale 1. Left: Population of shapes and the initial template (in blue). Right: Initial template and centres of scalings before (+) and after (o) optimisation

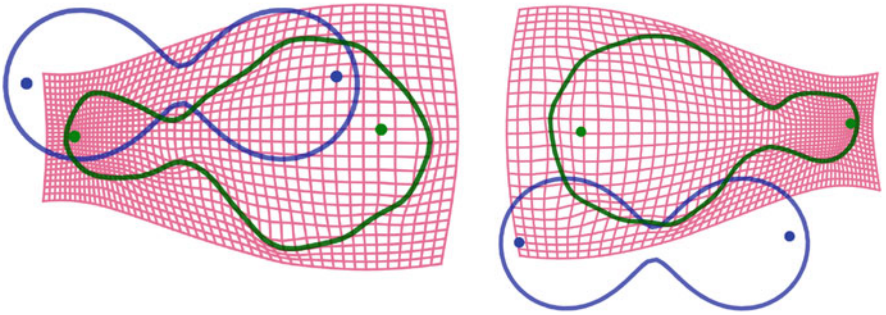


Fig. 17.8 Atlas with scalings of scale 1. Initial (blue) and final (green) shapes m_i and centres of scalings for two elements of the population. The targets are in black

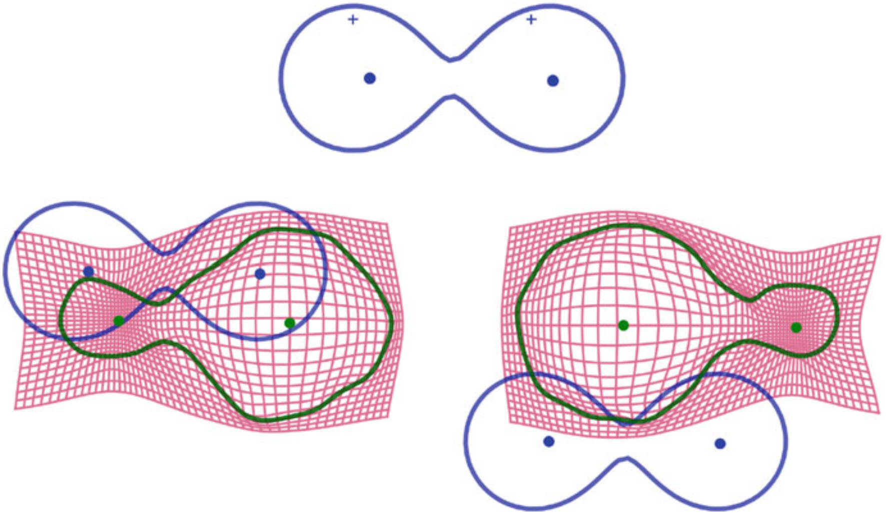


Fig. 17.9 Atlas with scalings of scale 0.6. Top: initial template and centres of scalings before (+) and after (o) optimisation. Bottom: initial (blue) and final (green) shapes m_i and centres of scalings for two elements of the population. The targets are in black

$m_i(1)$ for the same elements as in Fig. 17.5. Finally, in Fig. 17.9, we present results using the same atlas, but with $\mathcal{M}^{ls}$ generating scalings that are now of scale 0.6. We show the optimized positions of the geometric descriptors as well as the final shapes $m_i(1)$ as before.

We see here that even though the registrations are satisfactory with both deformation models, the optimal initial centers of the scalings are not at the same positions. The optimal position in each case is the best one, given the constraints of the chosen vocabulary (deformation module).

17.5 Constrained Evolution

In the previous sections, the spaces W_m (or W_θ) were described using a parametrized basis and were in most cases small dimensional. Here, we switch to a situation in which this space is described using linear constraints on the vector fields, without an explicit description of a generative family. Returning to the notation of Sect. 17.2, we will assume that, for all $m \in M$, a linear operator $C(m)$ is defined from V into a Banach space $\mathcal{Y}$, and let

$$V_m = \{v \in V : C(m)v = 0\}, \tag{17.18}$$

defining the space $W_m = \{v \cdot m : m \in M\}$.

On both theoretical and numerical levels, such formulations can be significantly more complex than those we have previously considered, especially when the constraints are also infinite dimensional. We refer the reader to [2, 3, 5] in which these issues are addressed, with, in particular, a version of the PMP in the constrained case that is obtained under the assumption that the operator $C(m)$ is onto for all m.

When both state and constraints have been discretized, a reduction similar to that made for LDDMM can be derived and the control reduced to a finite-dimensional setting too. Among possible constrained optimization algorithms that can be used for this problem, the one that leads to the simplest formulation is the augmented Lagrangian method, which was used in [4, 6, 46], and we refer to these references for more details on its implementation. We now describe a few examples of applications of this framework.

Normal Streamlines

As we have remarked in (17.8), LDDMM applied to manifold matching results, under mild conditions on the end-point cost U, in solutions for which the vector field takes the form

$$v(t, x) = \int_S K(x, m(t, s))\alpha(t, s)d\mu_0(s)\,. \tag{17.19}$$

When the end-point cost is parametrization invariant, the optimal function α can furthermore be shown to be perpendicular to the evolving manifold, so that $\partial_s m(t, s)^T \alpha(t, s) = 0$ for all $t \in [0, 1]$ and $s \in \mathcal{S}$. This normality property is not conserved after applying the kernel in (17.19), i.e., $\partial_s m(t, s)^T v(t, m(t, s))$ does not vanish in general. In [46] the problem of building intermediate layers between two open surfaces (with parametrizations m_0 and m_1) was considered and formulated as a search for a coordinate system $\psi : [0, 1] \times \mathcal{S} \to \mathbb{R}^3$ such that $\psi(0, m_0(s)) = m_0(s)$ for all $s \in \mathcal{S}$ and $\psi(1, \cdot)$ maps $\mathcal{S}$ onto $m_1(\mathcal{S})$ (i.e., it provides a reparametrization of m_1). The surfaces $S_t = \psi(t, \mathcal{S})$ are then interpreted as "layers" associated with the resulting foliation. The curves $\gamma_s(t) = \psi(t, s)$, $t \in [0, 1]$ provide "streamlines" and ψ is build with the requirement that the streamlines are perpendicular to the layers. This construction provides (among other

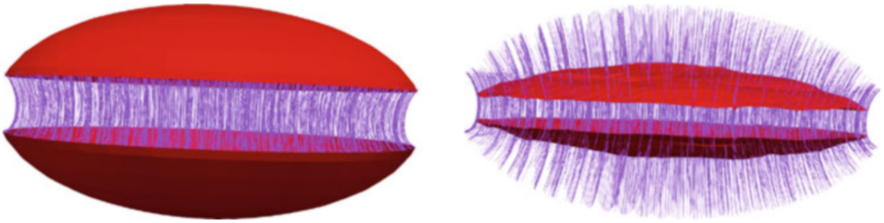


Fig. 17.10 Normal streamlines estimated between two inverted spherical caps are depicted in the left panel. Intermediate layers at times $t = 0.3$ and $t = 0.7$ are visualized in the right panel

things) a suitable way to define the "thickness" of the space separating m_0 and m_1, as the length of the streamlines, i.e.,

$$\theta(s) = \int_0^1 |\partial_t \psi(t, s)|\, dt.$$

(This definition of thickness was used, in particular, in [35] within a study of the impact of Alzheimer's disease on the trans-entorhinal cortex.)

The approach that was used to build such a coordinate system uses LDDMM to compute a flow of diffeomorphisms $\varphi(t, \cdot)$ such that $\varphi(1, m_0(\cdot))$ provides a reparametrization of $m_1(\cdot)$ (up to the error measured by the end-point cost), and defines $\psi(t, s) = m(t, s) = \varphi(t, m_0(s))$. Letting v denote the velocity field associated with φ, the normality condition requires that $\partial_s m(t, s)^T v(t, m(t, s)) = 0$, so that $C(m)v = 0$ in the definition of V_m is $\partial_s m^T v \circ m = 0$. We refer to [46] for implementation details (see Fig. 17.10 for an illustration).

Multi-Shapes

In [6], the problem of mapping complexes formed by multiple shapes is considered. In this context, we assume the one starts with several non-intersecting shapes, $m_0^{(1)}, \ldots, m_0^{(k)}$ that need to be aligned with another collection of shapes, say $m_1^{(1)}, \ldots, m_1^{(l)}$ (depending on the choice of end-point function, one may not need to require $k = l$, or to assume that the shapes are labelled). In the multi-shape model, each $m_0^{(j)}$ is deformed via its own diffeomorphism $\varphi^{(j)}(t, \cdot)$ and associated vector field $v^{(j)}(t, \cdot)$, providing a time-dependent shape $m^{(j)}(t, \cdot)$. To ensure that the deformation process is consistent, so that, for example, shapes are prevented from intersecting along the deformation path, an additional shape, denoted $m^{(0)}$, is created to represent the background and simply defined as the union of the k shapes forming the complex. More precisely, if all shapes are represented as functions from $\mathcal{S}$ to $\mathbb{R}^d$, then $m^{(0)}$ is parametrized by $\bigcup_{j=1}^k \mathcal{S} \times \{j\}$ with $m^{(0)}(t, (s, j)) = m^{(j)}(t, s)$. This latter identity forms a set of constraints that is imposed on the deformation process, which, assuming that the identity holds at $t = 0$ (defining $m_0^{(0)}$), can also be written as

$$v^{(0)}(t, m^{(0)}(t, (s, j))) = v^{(j)}(t, m^{(j)}(t, s)). \tag{17.20}$$

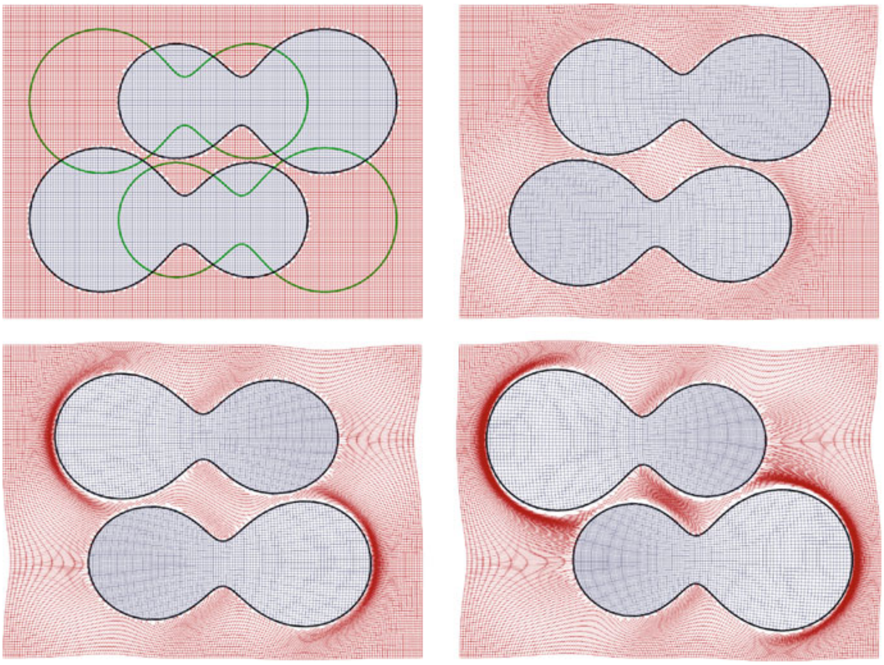


Fig. 17.11 Multi-shape geodesic with sliding constraints. The pair of black curves for a multi-shape that evolves towards the targets drawn in green in the upper-left panel. The red grid represents the background deformation along the process, and the blue ones provide the deformation that is attached to each curve, with the former significantly more pronounced than the latter due to different choices of kernel widths for both processes. Times $t = 0, 0.3, 0.7$ and 1 are provided from top to bottom and left to right

This results in a sub-Riemannian framework on the shape space formed by complexes $m = (m^{(0)}, m^{(1)}, \ldots, m^{(k)})$, in which these consistency constraints are enforced. This framework offers increased modeling flexibility, with, in particular, the possibility of specifying different Hilbert space $V^{(j)}$ for each vector field $v^{(j)}$, allowing each shape to have its own deformation properties. This is especially useful for the background ($j = 0$) for which one typically wants to allow large deformations with low cost.

Still in [6], a variant of (17.20) is introduced in order to allow each shape to "slide" along the background, in contrast to being "stitched" to it as enforced by (17.20). For this purpose, one still assumes that $m^{(0)}(0, (s, j)) = m^{(j)}(0, s)$, but it is only assumed that the normal components of $v^{(0)}$ and $v^{(j)}$ coincide over times. Here again, we refer to the cited reference for implementation details, and in particular for the way the constraints are discretized in the case of curves and surfaces. An example of curve registration using these constraints is provided in Fig. 17.11.

Atrophy Constraints

As a last example, we note the approach developed in [4] in which evolving shapes are assumed to evolve with a pointwise velocity making an acute angle with the inward normal. This results in a problem that deviates from that considered in (17.18) in that in involves inequality constraints. Indeed, Letting $N_m(s)$ denote the inward unit normal to a surface $m : \mathcal{S} \to \mathbb{R}^3$, the constraint is simply $N_m(s)^T v(m(s)) \geq 0$ for all $s \in \mathcal{S}$. Once the constraint is discretized on triangulated surfaces, the implementation uses augmented Lagrangian (adapted for inequality constraints, see [45]) and is fully described in [4].

17.6 Conclusion

As we have seen, sub-Riemannian geometry is a natural ingredient in the shape space framework where the infinitesimal evolutions around a given shape are driven by the action of smooth vector fields. However, the beauty of the Riemannian setting in the context of shapes spaces brings with it an embarrassment of riches, since one needs to design the metric on every tangent spaces in a high- (in fact infinite-) dimensional situation. The extension to a sub-Riemannian setting makes the situation even worse.

Of course, one can opt for relatively “simple” models, such as the standard LDDMM construction, which, as we have seen, requires two key parameters, namely a Hilbert space V on vector fields, itself determined by the choice of its kernel and an infinitesimal action (which is usually quite straightforward). Given these, we get from any shape exemplar an induced Riemannian shape space that comes with a computational framework for the analysis of the variations of shapes, including the computation of geodesics, linear approximations in tangent spaces leading to tangent principal components analysis, etc. However, the success of a Riemannian point of view is connected with its ability to generate realistic geodesics for interpolation and extrapolation. Usual translation and rotation invariance on the metric in V induces corresponding natural invariance under rigid motions. Smoothness requirements on V at a given scale induce corresponding smoothness requirement on the shape space. But stronger dependencies of metrics on the shape are usually necessary to incorporate prior knowledge on the differential structural properties of computed optimal deformations, which can be useful in several important situations:

1. when prior knowledge is available and needs to be incorporated, as material differences between foreground and background in the situation of multishapes, homogeneous behavior inside a shape, or known deformation patterns.
2. when one wants to test various hypothetical patterns inside a structured framework.

The modeling framework provided by deformation modules delineates an interesting route to produce a structured analysis of shape ensembles with different levels of complexity, addressing the points (1) and (2) and managing in an organized way the otherwise overwhelming expressiveness of the sub-Riemannian setting.

This framework also provides opportunities for future advances. As a final example, recent developments and work in progress in this context include the introduction of possibly infinite-dimensional spaces V_m based on physically motivated priors. In [28], for example, admissible vector fields are defined as regularized equilibrium solutions of a linear elasticity model subject to external forces that serve as alternative controls. These forces can themselves be parametrized, leading to finite- or infinite-dimensional distributions. When these forces are associated with events happening in the modeled shapes (such as disease propagation), it is natural to have them follow the shape evolution, and therefore be advected by the motion, leading to a model very similar to deformations module that we discussed in this chapter. These forces may also have their own evolution dynamics, which, combined to the advection, lead to more complex models that are currently investigated.

References

1. Agrachev, A.A., Sachkov, Y.: Control Theory from the Geometric Viewpoint, vol. 87. Springer, New York (2013)
2. Arguillère, S.: Sub-Riemannian geometry and geodesics in Banach manifolds. J. Geom. Anal. 1–42 (2019)
3. Arguillère, S., Trélat, E.: Sub-riemannian structures on groups of diffeomorphisms. J. Inst. Math. Jussieu **16**(4), 745–785 (2017)
4. Arguillère, S., Miller, M., Younes, L.: Lddmm surface registration with atrophy constraints. SIAM J. Imag. Sci. **9**(3) (2015)
5. Arguillère, S., Trélat, E., Trouvé, A., Younes, L.: Shape deformation analysis from the optimal control viewpoint. Journal des Mathématiques Pures et Appliquées **104**(1), 139–178 (2015)
6. Arguillère, S., Trélat, E., Trouvé, A., Younes, L.: Registration of multiple shapes using constrained optimal control. SIAM J. Imag. Sci. **9**(1), 344–385 (2016)
7. Aronszajn, N.: Theory of reproducing kernels. Trans. Am. Math. Soc. **68**, 337–404 (1950)
8. Arrate, F., Ratnanather, J.T., Younes, L.: Diffeomorphic active contours. SIAM J. Imag. Sci. **3**(2), 176–198 (2010)
9. Arsigny, V., Commowick, O., Ayache, N., Pennec, X.: A fast and log-euclidean polyaffine framework for locally linear registration. J. Math. Imaging Vision **33**(2), 222–238 (2009)
10. Bauer, M., Harms, P., Michor, P.: Sobolev metrics on shape space of surfaces. J. Geom. Mech. **3**(4), 389–438 (2011)
11. Bauer, M., Bruveris, M., Michor, P.W.: Overview of the geometries of shape spaces and diffeomorphism groups. J. Math. Imaging Vision **50**(1–2), 60–97 (2014)
12. Buhmann, M.: Radial basis functions: theory and implementations. In: Cambridge Monographs on Applied and Computational Mathematics, vol. 12. Cambridge University Press, Cambridge (2003)
13. Charlier, B., Feydy, J., Glaunès, J.: Keops: calcul rapide sur GPU dans les espaces à noyaux. In: Proceedings of Journées de Statistique de la SFdS, Paris (2018)
14. Charon, N., Trouvé, A.: The varifold representation of nonoriented shapes for diffeomorphic registration. SIAM J. Imag. Sci. **6**(4), 2547–2580 (2013)
15. Cheney, E.W., Light, W.A.: A Course in Approximation Theory, vol. 101. American Mathematical Society, Providence (2009)
16. Dryden, I.L., Mardia, K.V.: Statistical Shape Analysis, vol. 4. Wiley, New York (1998)

17. Dryden, I.L., Mardia, K.V.: Statistical Shape Analysis: With Applications in R. John Wiley & Sons, Hoboken (2016)
18. Duchon, J.: Interpolation des fonctions de deux variables suivant le principe de la flexion des plaques minces. R.A.I.R.O. Analyse Numerique **10**, 5–12 (1977)
19. Dupuis, P., Grenander, U., Miller, M.: Variational problems on flows of diffeomorphisms for image matching. Q. Appl. Math. **56**(3), 587 (1998)
20. Durrleman, S., Prastawa, M., Gerig, G., Joshi, S.: Optimal data-driven sparse parameterization of diffeomorphisms for population analysis. In: Székely, G.H.H. (ed.), IPMI 2011: Information Processing in Medical Imaging, pp. 123–134. Springer, Berlin (2011)
21. Durrleman, S., Allassonnière, S., Joshi, S.: Sparse adaptive parameterization of variability in image ensembles. Int. J. Comput. Vis. **101**(1), 161–183 (2013)
22. Durrleman, S., Prastawa, M., Charon, N., Korenberg, J.R., Joshi, S., Gerig, G., Trouvé, A.: Morphometry of anatomical shape complexes with dense deformations and sparse parameters. NeuroImage **101**, 35–49 (2014)
23. Glaunès, J., Trouvé, A., Younes, L.: Diffeomorphic matching of distributions: a new approach for unlabelled point-sets and sub-manifolds matching. In: CVPR 2004, vol. 2, pp. II-712–II-718. IEEE, Piscataway (2004)
24. Glaunès, J., Qiu, A., Miller, M.I., Younes, L.: Large deformation diffeomorphic metric curve mapping. Int. J. Comput. Vis. **80**(3), 317–336 (2008)
25. Grenander, U.: Elements of Pattern Theory. Johns Hopkins Univ Press, Baltimore (1996)
26. Grenander, U., Miller, M.M.I.: Pattern Theory: From Representation to Knowledge. Oxford University Press, Oxford (2007)
27. Gris, B., Durrleman, S., Trouvé, A.: A sub-Riemannian modular framework for diffeomorphism-based analysis of shape ensembles. SIAM J. Imag. Sci. **11**(1), 802–833 (2018)
28. Hsieh, D.-N., Arguillère, S., Charon, N., Miller, M.I., Younes, L.: A model for elastic evolution on foliated shapes. In: Proceedings of IPMI'19 (2019)
29. Joshi, S.H., Klassen, E., Srivastava, A., Jermyn, I.: A novel representation for Riemannian analysis of elastic curves in R^2. In: Computer Vision and Pattern Recognition, 2007, vol. 2007, pp. 1–7 (2007)
30. Kaltenmark, I., Charlier, B., Charon, N.: A general framework for curve and surface comparison and registration with oriented varifolds. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 3346–3355 (2017)
31. Kendall, D.G.: Shape manifolds, Procrustean metrics and complex projective spaces. Bull. London Math. Soc. **16**, 81–121 (1984)
32. Kendall, D., Barden, D., Carne, T., Le, H.: Shape and Shape Theory, vol. 11. Wiley, Hoboken (1999)
33. Klassen, E., Srivastava, A.: Geodesics between 3D closed curves using path-straightening. In: European Conference on Computer Vision 2006, pp. 95–106. Springer, Berlin (2006)
34. Kühnel, L., Sommer, S.: Computational anatomy in Theano. In: Graphs in Biomedical Image Analysis, Computational Anatomy and Imaging Genetics, pp. 164–176. Springer, Berlin (2017)
35. Kulason, S., Tward, D.J., Brown, T., Sicat, C.S., Liu, C.-F., Ratnanather, J.T., Younes, L., Bakker, A., Gallagher, M., Albert, M., Miller, M.I.: Cortical thickness atrophy in the transentorhinal cortex in mild cognitive impairment. NeuroImage Clin. **21**, 101617 (2019)
36. Luenberger, D.G.: Optimization by Vector Space Methods. J. Wiley and Sons, Hoboken (1969)
37. Macki, J., Strauss, A.: Introduction to Optimal Control Theory. Springer, Berlin (2012)
38. Meinguet, J.: Multivariate interpolation at arbitrary points made simple. J. Appl. Math. Phys. **30**, 292–304 (1979)
39. Memoli, F.: On the use of Gromov-Hausdorff distances for shape comparison. In: Botsch, M., Pajarola, R., Chen, B., Zwicker, M., (eds.), Eurographics Symposium on Point-Based Graphics. The Eurographics Association, Aire-la-Ville (2007)
40. Memoli, F.: Gromov-Hausdorff distances in Euclidean spaces. In: CVPR Workshop on Nonrigid Shape Analysis (2008)

41. Micheli, M., Glaunès, J.A.: Matrix-valued kernels for shape deformation analysis. Geom. Imag. Comput. **1**(1), 57–139 (2014)
42. Michor, P., Mumford, D.: Riemannian geometries on spaces of plane curves. J. Eur. Math. Soc. **1**, 1–48 (2006)
43. Michor, P., Mumford, D.: An overview of the Riemannian metrics on spaces of curves using the Hamiltonian approach. Appl. Comput. Harmon. Anal. **23**(1), 74–113 (2007)
44. Mio, W., Srivastava, A., Joshi, S.: On shape of plane elastic curves. Int. J. Comput. Vis. **73**(3), 307–324 (2007)
45. Nocedal, J., Wright, S.J.: Numerical Optimization. Springer, Berlin (1999)
46. Ratnanather, J.T., Arguillère, S., Kutten, K.S., Hubka, P., Kral, A., Younes, L.: 3D normal coordinate systems for cortical areas. In: Kushnarev, S., Qiu, A., Younes, L. (eds.), Mathematics of Shapes and Applications, abs/1806.11169 (2019)
47. Roussillon, P., Glaunès, J.: Representation of surfaces with normal cycles. Application to surface registration. Technical Report, University Paris-Descartes (2019)
48. Schölkopf, B., Smola, A.J., Bach, F., et al.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge (2002)
49. Seiler, C., Pennec, X., Reyes, M.: Capturing the multiscale anatomical shape variability with polyaffine transformation trees. Med. Image Anal. **16**(7), 1371–1384 (2012)
50. Sharon, E., Mumford, D.: 2D shape analysis using conformal mapping. Int. J. Comput. Vis. **70**(1), 55–75 (2006)
51. Siddiqi, K., Pizer, S.: Medial Representations: Mathematics, Algorithms and Applications, vol. 37. Springer, New York (2008)
52. Srivastava, A., Klassen, E.P.: Functional and Shape Data Analysis. Springer, Berlin (2016)
53. Staneva, V., Younes, L.: Modeling and estimation of shape deformation for topology-preserving object tracking. SIAM J. Imag. Sci. **7**(1), 427–455 (2014)
54. Styner, M., Oguz, I., Xu, S., Brechbühler, C., Pantazis, D., Levitt, J.J., Shenton, M.E., Gerig, G.: Framework for the statistical shape analysis of brain structures using SPHARM-PDM. Insight J. **2006**(1071), 242–250 (2006)
55. Thompson, D.W.: On Growth and Form. Revised Edition. Cambridge University Press (1961), Cambridge (1917)
56. Trouvé, A.: Infinite Dimensional Group Action and Pattern Recognition. Technical Report, DMI, Ecole Normale Supérieure (1995)
57. Tward, D.J., Brown, T., Patel, J., Kageyama, Y., Mori, S., Troncoso, J.C., Miller, M.: Quantification of 3D tangle distribution in medial temporal lobe using multimodal image registration and convolutional neural networks. Alzheimers Dement. **14**(7), P1291 (2018)
58. Vaillant, M., Glaunes, J.: Surface matching via currents. In: Information Processing in Medical Imaging 2005, pp. 381–392. Springer, New York (2005)
59. Vincent, T.L., Grantham, W.J.: Nonlinear and Optimal Control Systems. Wiley, Hoboken (1997)
60. Wahba, G.: Spline Models for Observational Data. SIAM, Philadelphia (1990)
61. Wang, S., Wang, Y., Jin, M., Gu, X.D., Samaras, D.: Conformal geometry and its applications on 3D shape matching, recognition, and stitching. IEEE Trans. Pattern Anal. Mach. Intell. **29**(7), 1209–1220 (2007)
62. Younes, L.: Computable elastic distances between shapes. SIAM J. Appl. Math. **58**(2), 565–586 (1998)
63. Younes, L.: Constrained diffeomorphic shape evolution. In: Foundations of Computational Mathematics (2012)
64. Younes, L.: Gaussian diffeons for surface and image matching within a Lagrangian framework. Geom. Imaging and Comput. **1**(1), 141–171 (2014)
65. Younes, L.: Shapes and Diffeomorphisms, 2nd edn. Springer, Berlin (2019)
66. Younes, L., Michor, P., Shah, J., Mumford, D.: A metric on shape spaces with explicit geodesics. Rend. Lincei Mat. Appl. **9**, 25–57 (2008)

67. Zeng, W., Gu, X.D.: Registration for 3D surfaces with large deformations using quasi-conformal curvature flow. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2457–2464. IEEE, Piscataway (2011)
68. Zhang, W., Noble, J.A., Brady, J.M.: Adaptive non-rigid registration of real time 3D ultrasound to cardiovascular mr images. In: Information Processing in Medical Imaging, pp. 50–61. Springer, Berlin (2007)

Part V
Optimization Algorithms and Numerical Methods

Chapter 18
First Order Methods for Optimization on Riemannian Manifolds

Orizon P. Ferreira, Maurício S. Louzeiro, and Leandro F. Prudente

Contents

Abstract This chapter considers optimization problems on Riemannian manifolds and presents asymptotic and iteration-complexity analysis for gradient and subgradient methods on manifolds with sectional curvatures bounded from below. It also establishes asymptotic and iteration-complexity analysis for the proximal point method on Hadamard manifolds.

O. P. Ferreira (✉) · L. F. Prudente
Universidade Federal de Goiás, Institute of Mathematics Statistics, Goiânia, GO, Brazil
e-mail: orizon@ufg.br; lfprudente@ufg.br

M. S. Louzeiro
TU Chemnitz, Fakultät für Mathematik, Chemnitz, Germany
e-mail: mauricio.silva-louzeiro@mathematik.tu-chemnitz.de

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_18

18.1 Introduction

Over the years, the interest in using Riemannian geometry tools to solve constrained optimization problems has been increasing. This is largely due to the fact that a wide range of optimization problems can be naturally posed in a Riemannian setting, which can then be exploited to greatly reduce the cost of obtaining the solutions. For this reason, the number of papers dealing with extensions of concepts and techniques of nonlinear programming from the Euclidean context to the Riemannian scenario are also increasing. Such extensions are necessary, usually natural and not trivial. They are intended to provide theoretical support for efficient computational implementations of algorithms. Works on this subject include, but are not limited to [1, 2, 32, 42, 45, 48, 55, 57, 59, 60, 73, 81, 87]. The Riemannian machinery from the theoretical point of view allows, by the introduction of a suitable metric, a nonconvex Euclidean problem to be seen as a Riemannian convex problem. From an algorithmic point of view, it enables modifications of numerical methods in order to find global minimizer of the considered problems; see [30, 36, 37, 67, 75, 76] and references therein. Furthermore, in order to take advantage of the inherent geometric structure in the problems, it is preferable to treat constrained optimization problems as intrinsic Riemannian problems (i.e., as unconstrained Riemannian problems), rather than using Lagrange multipliers, penalty methods, or projection methods; see [2, 35, 53, 54, 78]. In addition to the theoretical and algorithmic issues addressed in Riemannian optimization—which have an interest of their own—it is worth mentioning that several interesting practical applications in this context have appeared over the last few years. Although we are not concerned with practical issues at this time, we emphasize that practical applications appear whenever the natural structure of the data is modeled as an optimization problem on a Riemannian manifold. For example, several problems in image processing, computational vision and signal processing can be modeled as problems in this setting, papers dealing with this subject include [6, 16, 17, 22, 43, 85, 86], and problems in medical imaging modeled in this context are addressed in [8, 33]. Problems of tracking, robotics and scene motion analysis are also posed in Riemannian manifolds, see [39, 64]. We also mention that there are many papers on statistics in Riemannian context, see for example [18, 38]. Finally, it is worth mentioning that many other applications can be found through this book. In this chapter, we are interested in the following convex optimization problem:

$$\min\{f(p) \colon p \in \mathcal{M}\}, \tag{18.1}$$

where the constraint set $\mathcal{M}$ is endowed with a *Riemannian structure* and $f : \mathcal{M} \to \mathbb{R}$ is a *convex function*. For future reference we denote the optimal value of (18.1) by $f^* := \inf_{p\in\mathcal{M}} f(p)$ and its solution set by Ω^*, which *will be assumed to be nonempty only when explicitly stated*. We will present asymptotic and iteration-complexity analysis for gradient, subgradient and proximal point methods to solve Problem (18.1). In the presented analysis of gradient and subgradient methods we assume that the curvature of $\mathcal{M}$ is bounded from below and in the analysis of proximal point method that $\mathcal{M}$ is a Hadamard manifold. Despite the long history of these methods, it seems that their simplicity still attracts the attention of the optimization researchers. In fact, this simplicity has given renewed emphasis to recent applications in large-scale problems. In addition, these methods are the starting point for designing many more sophisticated and efficient methods. All these questions motivated this review.

This chapter is organized as follows. Section 18.2 presents definitions and auxiliary results. In Sect. 18.3 we present some examples of convex functions. In Sect. 18.4, we study properties of the gradient method. In Sect. 18.5 we study the main properties of the subgradient method. Section 18.6 is devoted to the analysis of the proximal point method and, in Sect. 18.7 we end the chapter with some perspectives.

18.2 Notations and Basic Results

In this section, we recall some concepts, notations, and basics results about Riemannian manifolds and optimization. For more details see, for example, [31, 67, 71, 76]. Let us begin with concepts about Riemannian manifolds. We denote by $\mathcal{M}$ a finite dimensional Riemannian manifold and by $T_p\mathcal{M}$ the *tangent plane* of $\mathcal{M}$ at p. The corresponding norm associated to the Riemannian metric $\langle \cdot\ ,\ \cdot \rangle$ is denoted by $\| \ \cdot \ \|$. We use $\ell(\gamma)$ to denote the length of a piecewise smooth curve $\gamma : [a, b] \to \mathcal{M}$. The Riemannian distance between p and q in $\mathcal{M}$ is denoted by $d(p, q)$, which induces the original topology on $\mathcal{M}$, namely, $(\mathcal{M}, d)$, which is a complete metric space. Let $(\mathcal{N}, \langle \cdot\ ,\ \cdot \rangle)$ and $(\mathcal{M}, \langle \cdot\ ,\ \cdot \rangle)$ be Riemannian manifolds, a mapping $\Phi : \mathcal{N} \to \mathcal{M}$ is called an isometry, if Φ is C^∞, and for all $q \in \mathcal{N}$ and $u, v \in T_q\mathcal{N}$, we have $\langle u, v\rangle = \langle d\Phi_q u, d\Phi_q v\rangle$, where $d\Phi_q : T_q\mathcal{N} \to T_{\Phi(q)}\mathcal{M}$ is the differential of Φ at $q \in \mathcal{N}$. One can verify that Φ preserves geodesics, that is, β is a geodesic in $\mathcal{N}$ iff $\Phi \circ \beta$ is a geodesic in $\mathcal{M}$. Denote by $\mathcal{X}(\mathcal{M})$, the space of smooth vector fields on $\mathcal{M}$. Let ∇ be the Levi-Civita connection associated to $(\mathcal{M}, \langle \cdot\ ,\ \cdot \rangle)$. For $f : \mathcal{M} \to \mathbb{R}$ a differentiable function, the Riemannian metric induces the mapping $f \mapsto \operatorname{grad} f$ which associates its *gradient* via the following rule $\langle \operatorname{grad} f(p), X(p)\rangle := df(p)X(p)$, for all $p \in \mathcal{M}$. For a twice-differentiable function, the mapping $f \mapsto \operatorname{Hess} f$ associates its *Hessian* via the rule $\langle \operatorname{Hess} f X, X\rangle := d^2 f(X, X)$, for all $X \in \mathcal{X}(\mathcal{M})$, where the last equalities imply that $\operatorname{Hess} f X = \nabla_X \operatorname{grad} f$, for all $X \in \mathcal{X}(\mathcal{M})$. A vector field V along γ is said to be *parallel* iff $\nabla_{\gamma'} V = 0$. If γ' is itself parallel, we say that γ is a *geodesic*. Given that the geodesic equation $\nabla_{\gamma'}\gamma' = 0$ is a second order nonlinear ordinary differential equation, then the geodesic $\gamma := \gamma_v(\cdot, p)$ is determined by its position p and velocity v at p. The restriction of a geodesic to a closed bounded interval is called a *geodesic segment*. A geodesic segment joining p to q in $\mathcal{M}$ is said to be *minimal* if its length is equal to $d(p, q)$. For each $t \in [a, b]$, ∇ induces an isometry, relative to $\langle \cdot, \cdot \rangle$, $P_{\gamma,a,t} \colon T_{\gamma(a)}\mathcal{M} \to T_{\gamma(t)}\mathcal{M}$ defined by $P_{\gamma,a,t}\, v = V(t)$, where V is the unique vector field on γ such that $\nabla_{\gamma'(t)}V(t) = 0$ and $V(a) = v$, the so-called *parallel transport* along the geodesic segment γ joining $\gamma(a)$ to $\gamma(t)$. When there is no confusion, we consider the notation $P_{\gamma,p,q}$ for the parallel transport along the geodesic segment γ joining p to q. A Riemannian manifold is *complete* if the geodesics are defined for any values of $t \in \mathbb{R}$. Hopf-Rinow's theorem asserts that any pair of points in a complete Riemannian manifold $\mathcal{M}$ can be joined by a (not necessarily unique) minimal geodesic segment. *In this chapter, all manifolds are assumed to be connected, finite dimensional, and complete.* Owing to the completeness of the Riemannian manifold $\mathcal{M}$, the *exponential map* $\exp_p : T_p\mathcal{M} \to \mathcal{M}$ can be given by $\exp_p v = \gamma_v(1, p)$, for each $p \in \mathcal{M}$. A complete, simply connected Riemannian manifold of non-positive sectional curvature is called a *Hadamard manifold.* For $\mathcal{M}$ a Hadamard manifold and $p \in \mathcal{M}$, the exponential map $\exp_p : T_p\mathcal{M} \to \mathcal{M}$ is a diffeomorphism and $\exp_p^{-1} : \mathcal{M} \to T_p\mathcal{M}$ denotes its inverse. In this case, $d(q\ , \ p) = ||\exp_p^{-1} q||$ and the function $d_q^2 : \mathcal{M} \to \mathbb{R}$ defined by $d_q^2(p) := d^2(q, p)$ is C^∞ and grad $d_q^2(p) := -2\exp_p^{-1} q$.

Now, we recall some concepts and basic properties about optimization in the Riemannian context. For that, given two points $p, q \in \mathcal{M}$, Γ_{pq} denotes the set of all geodesic segments $\gamma : [0, 1] \to \mathcal{M}$ with $\gamma(0) = p$ and $\gamma(1) = q$. A function $f : \mathcal{M} \to \mathbb{R}$ is said to be *convex* if, for any $p, q \in \mathcal{M}$ and $\gamma \in \Gamma_{pq}$, the composition $f \circ \gamma : [0, 1] \to \mathbb{R}$ is convex, i.e., $(f \circ \gamma)(t) \leq (1 - t)f(p) + tf(q)$, for all $t \in [0, 1]$. A vector $s \in T_p\mathcal{M}$ is said to be a *subgradient* of a convex function f at p, iff $f(\exp_p v) \geq f(p) + \langle s, \ v\rangle$, for all $v \in T_p\mathcal{M}$. Let $\partial f(p)$ be the *subdifferential* of f at p, namely, the set of all subgradients of f at p. Then, f is convex iff there holds $f(\exp_p v) \geq f(p) + \langle s, \ v\rangle$, for all $p \in \mathcal{M}$ and $s \in \partial f(p)$ and $v \in T_p\mathcal{M}$. If $f : \mathcal{M} \to \mathbb{R}$ is convex, then $\partial f(p)$ is nonempty, for all $p \in \mathcal{M}$ and, in particular, for a differentiable function we have $\partial f(p) = \{\text{grad} f(p)\}$.

Definition 18.1 A function $f : \mathcal{M} \to \mathbb{R}$ is said to be Lipschitz continuous with constant $L \geq 0$ on $\Omega \subset \mathcal{M}$ if, for any $p, q \in \Omega$ and $\gamma \in \Gamma_{pq}$, there holds $|f(p) - f(q)| \leq L\ell(\gamma)$. Given $p \in \mathcal{M}$, if there exists $\delta > 0$ such that f is Lipschitz continuous on $B_\delta(p)$, then f is said to be Lipschitz continuous at p. Moreover, if for all $p \in \mathcal{M}$, f is Lipschitz continuous at p, then f is said to be locally Lipschitz continuous on $\mathcal{M}$.

All convex functions f are locally Lipschitz continuous. Consequently, the map ∂f is bound on a bounded set. In particular we have the following result, see [81, Proposition 2.5].

Proposition 18.2 *Let $f : \mathcal{M} \to \mathbb{R}$ be convex and $\{p_k\} \subset \mathcal{M}$ be a bounded sequence. If $\{s_k\}$ is such that $s_k \in \partial f(p_k)$, for each $k = 0, 1, \ldots,$, then $\{s_k\}$ is also bounded.*

The concept of Lipschitz continuity of gradient vector fields, which was introduced in [28], is stated as follows.

Definition 18.3 Let $f : \mathcal{M} \to \mathbb{R}$ be a differentiable function. The gradient vector field of f is said to be Lipschitz continuous with constant $L > 0$ if, for any $p, q \in \mathcal{M}$ and $\gamma \in \Gamma_{pq}$, it holds that $\|P_{\gamma,p,q} \operatorname{grad} f(p) - \operatorname{grad} f(q)\| \leq L\ell(\gamma)$.

For $f : \mathcal{M} \to \mathbb{R}$ twice differentiable, the *norm of the Hessian* Hess f at $p \in \mathcal{M}$ is defined by $\|\text{Hess}\, f(p)\| := \sup\left\{\|\text{Hess}\, f(p)v\| \ : \ v \in T_p\mathcal{M}, \ \|v\| = 1\right\}$. The following result presents a characterization for twice continuously differentiable functions with Lipschitz continuous gradient, its proof is similar to the Euclidean counterpart.

Lemma 18.4 *Let $f : \mathcal{M} \to \mathbb{R}$ be a twice continuously differentiable function. The gradient vector field of f is Lipschitz continuous with constant $L > 0$ iff there exists $L > 0$ such that $\|Hess\ f(p)\| \leq L$, for all $p \in \mathcal{M}$.*

Next we present the concept of quasi-Fejér convergence, which play an important role in the asymptotic convergence analysis of the methods studied in this chapter.

Definition 18.5 A sequence $\{y_k\}$ in the complete metric space $(\mathcal{M}, d)$ is quasi-Fejér convergent to a set $W \subset \mathcal{M}$ if, for every $w \in W$, there exist a sequence $\{\epsilon_k\} \subset \mathbb{R}$ such that $\epsilon_k \geq 0$, $\sum_{k=1}^{\infty} \epsilon_k < +\infty$, and $d^2(y_{k+1}, w) \leq d^2(y_k, w) + \epsilon_k$, for all $k = 0, 1, \ldots$.

In the following we state the main property of the quasi-Fejér concept, its proof is similar to its Euclidean counterpart proved in [23].

Theorem 18.6 *Let $\{y_k\}$ be a sequence in the complete metric space $(\mathcal{M}, d)$. If $\{y_k\}$ is quasi-Fejér convergent to a nonempty set $W \subset \mathcal{M}$, then $\{y_k\}$ is bounded. If furthermore, a cluster point $\bar{y}$ of $\{y_k\}$ belongs to W, then* $\lim_{k\to\infty} y_k = \bar{y}$.

18.3 Examples of Convex Functions on Riemannian Manifolds

In this section our purpose is to present some examples of convex functions on Riemannian manifolds. In Sect. 18.3.1 we recall usual examples in general Riemannian manifolds. In Sects. 18.3.2–18.3.4 we present some examples in particular Riemannian manifolds. In all examples presented, the functions are not convex and their gradients are not continuous Lipschitz with respect to the Euclidean metric. However, by changing the metric, all that functions become convex with Lipschitz continuous gradient with respect to the new metric. As we will show in next sections, this property is particularly interesting in the complexity analysis of the optimization methods to solve the Problem (18.1). Moreover, despite the Euclidean nonconvexity of the objective function, its Riemannian convexity ensures that a local solution is globally optimal. In particular, these examples show how to take advantage of using Riemaninan geometry concept to study the Problem (18.1); see [3, 36, 37, 67, 75, 76].

18.3.1 General Examples

It is well known that new convex functions can be designed from other convex functions through operations that preserve convexity. For instance, letting $f_1, \ldots, f_m$ be convex functions on the Riemannian manifolds $\mathcal{M}$, the following functions are also convex $f(p) = \sum_{i=1}^{m} \mu_i f_i(p)$, where $\mu_i \geq 0$ for all $i = 1, \ldots, m$ and $f(p) = \max\{f_1(p), \ldots, f_m(p)\}$. Let $\mathcal{N}$ and $\mathcal{M}$ be Riemannian manifolds and $\Phi : \mathcal{N} \to \mathcal{M}$ an isometry. The function $f : \mathcal{M} \to \mathbb{R}$ is convex if only if $f \circ \Phi : \mathcal{N} \to \mathbb{R}$ is convex. One of the most important functions in a Riemannian manifolds is the distance function. Let us present some examples of convex function that arise from the distance function. Let $\mathcal{M}$ be a Hadamard manifold and d the Riemannian distance. The function $h(p) = d(p, \Phi(p))$ is convex. Moreover, for any $\bar{p} \in \mathcal{M}$ the function $\mathcal{M} \ni p \mapsto d(p, \bar{p})$ is a convex function. We end this section with an important family of convex functions which has applications in different areas,

$$f_a(p) := \begin{cases} \frac{1}{a} \sum_{i=1}^{m} w_i d^a(p, p_i) & 1 \leq a < \infty \\ \max_i d(p, p_i), & a = \infty. \end{cases} \tag{18.2}$$

where $\{p_i : i = 1, \ldots, m\} \subset \mathcal{M}$ and $\{w_i : i = 1, \ldots m\} \subset \mathbb{R}$ such that $\sum_{i=1}^{m} w_i = 1$ and $0 \leq w_i \leq 1$, for all $i = 1, \ldots, m$. We recall that a local minimizer of f_a is called *local center of mass* of the data set $\{p_i : i = 1, \ldots, m\}$ with respect to the weights $\{w_i : i = 1, \ldots m\}$; see [3]. When $\mathcal{M}$ is the cone of positive definite matrices, (18.2) is called *Karcher (Fréchet) mean*, see for example [3, 4, 46, 75, 90].

18.3.2 Example in the Euclidean Space with a New Riemannian Metric

In this section, we use the concept of an isometry between two Riemannian manifolds to study convex functions. In particular, we change the metric of $\mathbb{R}^n$ so that the *extended Rosenbrock's banana function* becomes convex with Lipschitz gradient in this new manifold. Let $f : \mathbb{R}^{2n} \to \mathbb{R}$ be the Rosenbrock's banana function, defined by $f(x_1, \ldots, x_{2n}) = \sum_{i=1}^{n} a_i \left(x_{2i-1}^2 - x_{2i}\right)^2 + (x_{2i-1} - b_i)^2$, $a_i \in \mathbb{R}_{++}$, and $b_i \in \mathbb{R}$. Denote by $\bar{\mathcal{M}}$ the Euclidean space $\mathbb{R}^{2n}$ with the usual metric. It is well known that f is non-convex and its gradient is non-Lipschitz continuous in $\bar{\mathcal{M}}$. Endowing $\mathbb{R}^{2n}$ with the new Riemannian metric $\langle u, v \rangle := u^T G(x) v$, where $u, v \in \mathbb{R}^{2n}$ and $G(x)$ is the $2n \times 2n$ block diagonal matrix $G(x) = \operatorname{diag}(G_1(x), \ldots, G_n(x))$ with

$$G_i(x) := \begin{pmatrix} 1 + 4x_{2i-1}^2 & -2x_{2i-1} \\ -2x_{2i-1} & 1 \end{pmatrix}, \qquad i = 1, \ldots, n,$$

and $x = (x_1, \dots, x_{2n})$, we obtain a Hadamard manifold $\mathcal{M} := (\mathbb{R}^{2n}, G)$. Let $\Phi : \bar{\mathcal{M}} \to \mathcal{M}$ be defined by $\Phi(z_1, \dots, z_{2n}) = \big(z_1, z_1^2 - z_2, \dots, z_{2n-1}, z_{2n-1}^2 - z_{2n}\big)$. We can prove that the function Φ is an isometry. Now, let $g : \bar{\mathcal{M}} \to \mathbb{R}$ be defined by $g(z_1, \dots, z_{2n}) = (f \circ \Phi)(z_1, \dots, z_{2n}) = \sum_{i=1}^{n} a_i z_{2i}^2 + (z_{2i-1} - b_i)^2$. Since g is a convex quadratic function and Φ is a isometry, the last equality implies that f is convex in $\mathcal{M}$. Moreover, due to the gradient of g be Lipschitz continuous with constant $L := \max\{2, 2a_1, \dots, 2a_n\}$, we also have that f has Lipschitz gradient with constant L. On the other hand, note that $\Phi^{-1}(x) = (x_1, x_1^2 - x_2, \dots, x_{2n-1}, x_{2n-1}^2 - x_{2n})$. Therefore, taking any convex function with Lipschitz continuous gradient g in $\bar{\mathcal{M}}$, we can define a new convex function with Lipschitz continuous gradient in $\mathcal{M}$ by setting $f = g \circ \Phi^{-1}$. This idea leads to a way to obtain new convex functions from others, whenever an isometry is known. The convexity of Rosenbrock's banana function in two dimension was first presented in [76, p. 83].

18.3.3 Examples in the Positive Orthant with a New Riemannian Metric

In this section, we present examples in the positive orthant. For that we denote the set of $n \times n$ matrices with real entries by $\mathbb{R}^{n\times n}$, the n-dimensional Euclidean space by $\mathbb{R}^n \equiv \mathbb{R}^{n\times 1}$, the positive orthant by $\mathbb{R}^n_+ = \big\{x = (x_1, \dots, x_n)^T \in \mathbb{R}^{n\times 1} : x_i \geq 0,\ i = 1, \dots, n\}$, and its interior by $\mathbb{R}^n_{++}$. Set $\operatorname{diag}(y) = \operatorname{diag}(y_1, \dots, y_n) \in \mathbb{R}^{n\times n}$ the diagonal matrix with (i, i)-th entry equal to y_i, $i = 1, \dots, n$. Let $\mathcal{M} := (\mathbb{R}^n_{++}, G)$ be the Hadamard manifold obtained by endowing $\mathbb{R}^n_{++}$ with the new Riemannian metric $\langle u, v\rangle := u^T G(x) v$, for all $x \in \mathbb{R}^n_{++}$ and $u, v \in T_x\mathbb{R}^n_{++} \equiv \mathbb{R}^n$, where

$$G(x) := \operatorname{diag}(x_1^{-2}, \dots, x_n^{-2}) \in \mathbb{R}^{n\times n}. \tag{18.3}$$

Let $f : \mathcal{M} \to \mathbb{R}$ be a twice differentiable function, $f'(x)$ and $f''(x)$ be the Euclidean gradient and Hessian of f at x, respectively. Thus, (18.3) implies that the Riemannian *gradient* and *Hessian* of f at $x \in \mathcal{M}$ are given, respectively, by

$$\operatorname{grad} f(x) = \operatorname{diag}(x)^2 f'(x), \qquad f(x) = \operatorname{diag}(x)^2 f''(x) + \operatorname{diag}(x)\operatorname{diag}\big(f'(x)\big), \tag{18.4}$$

where $\operatorname{diag}(x)^2 := \operatorname{diag}(x_1^2, \dots, x_n^2)$. Below we present three functions that are nonconvex with non-Lipschitz continuous gradient on $\mathbb{R}^n_{++}$. However, by using (18.4) and Lemma 18.4, we can prove that these functions are convex with their gradients being Lipschitz continuous on $\mathcal{M}$; for more details see [36].

1. Let $a_i, c_i, b_i, d_i \in \mathbb{R}_+$ satisfy $c_i > a_i$, for all $i = 1, \dots n$. The function $f : \mathbb{R}^n_{++} \to \mathbb{R}$ defined by $f(x) = -\sum_{i=1}^{n} a_i e^{-b_i x_i} + \sum_{i=1}^{n} c_i \ln(x_i)^2 + \sum_{i=1}^{n} d_i \ln(x_i)$, is convex with Lipschitz gradient with constant $L < \sum_{i=1}^{n}(a_i + 2c_i)^2$ on $\mathcal{M}$.

2. Let $a_i, b_i, c_i, d_i \in \mathbb{R}_{++}$ satisfy $c_i < a_i d_i$ and $d_i \geq 2$, for all $i = 1, \ldots n$. The function $f : \mathbb{R}^n_{++} \to \mathbb{R}$ defined by $f(x) = \sum_{i=1}^n a_i \ln(x_i^{d_i} + b_i) - \sum_{i=1}^n c_i \ln(x_i)$, is convex with Lipschitz gradient with constant $L < \sum_{i=1}^n a_i^2 d_i^4$ on $\mathcal{M}$.
3. Let $u := (u_1, \ldots, u_n), w := (w_1, \ldots, w_n) \in \mathbb{R}^n_{++}$ and $a, b, c \in \mathbb{R}_{++}$. The function $f : \mathbb{R}^n_{++} \to \mathbb{R}$ defined by $f(x) = a \ln\left(\prod_{i=1}^n x_i^{u_i} + b_i\right) - \sum_{i=1}^n w_i \ln(x_i) + c \sum_{i=1}^n \ln^2(x_i)$, is convex with Lipschitz gradient with constant $L \leq a u^T u / b + 2c$ on $\mathcal{M}$.

18.3.4 Examples in the Cone of SPD Matrices with a New Riemannian Metric

In this section, our examples are on symmetric positive definite (SPD) matrices cone $\mathbb{P}^n_{++}$. Following Rothaus [69], let $\mathcal{M} := (\mathbb{P}^n_{++}, \langle \cdot, \cdot \rangle)$ be the Hadamard manifold endowed with the new Riemannian metric given by $\langle U, V \rangle := \operatorname{tr}(V X^{-1} U X^{-1})$, for $X \in \mathcal{M}$ and $U, V \in T_X\mathcal{M}$, where $\operatorname{tr}(X)$ denotes the trace of the matrix X, $T_X\mathcal{M} \approx \mathbb{P}^n$ is the tangent plane of $\mathcal{M}$ at X and $\mathbb{P}^n$ denotes the set of symmetric matrices of order $n \times n$. We recall that $\mathcal{M}$ is a Hadamard manifold, see for example [50, Theorem 1.2. p. 325]. The *gradient* and *Hessian* of $f : \mathbb{P}^n_{++} \to \mathbb{R}$ are given by

$$\operatorname{grad} f(X) = X f'(X) X, \quad \operatorname{Hess} f(X) V = X f''(X) V X + \frac{1}{2}\left[V f'(X) X + X f'(X) V\right],$$

respectively, where $V \in T_X\mathcal{M}$, $f'(X)$ and $f''(X)$ are the Euclidean gradient and Hessian of f at X, respectively. In the following, we present two functions that are non-convex with non-Lipschitz continuous gradient on $\mathbb{P}^n_{++}$ endowed with the Euclidean metric. However, by using Lemma 18.4, we can prove that these functions are convex with Lipschitz continuous gradient on $\mathcal{M}$; for more details see [36].

1. Let $a, b \in \mathbb{R}_{++}$ and $f : \mathbb{P}^n_{++} \to \mathbb{R}$ defined by $f(X) = a \ln(\det(X))^2 - b \ln(\det(X))$. The function f is convex with Lipschitz continuous gradient with constant $L \leq 2a\sqrt{n}$ on $\mathcal{M}$.
2. Let $a, b_1, b_2, c \in \mathbb{R}_{++}$ with $c < a b_1$. The function $f : \mathbb{P}^n_{++} \to \mathbb{R}$ defined by $f(X) = a \ln(\det(X)^{b_1} + b_2) - c \ln(\det X)$, is convex with Lipschitz continuous gradient with constant $L < a b_1^2 n$ on $\mathcal{M}$.

Bibliographic Notes and Remarks

A comprehensive study of convex functions in Riemannian manifolds can be found in [76]. Interesting examples of convex function on the SPD matrices cone are presented in [75]. As far as we know, the first studies on geometric properties of SPD matrices cone with new metric have appeared in [69]. Applications and related studies about convex function on Riemannian manifolds include [4, 12, 49, 60, 67, 70, 75, 90].

18.4 Gradient Method for Optimization

In this section we study properties of the gradient method for solving Problem (18.1) with the two most commonly used strategies for choosing the step-size. For that we *assume that* $f : \mathcal{M} \to \mathbb{R}$ *is a continuously differentiable function and that the solution set* Ω^* *of the Problem* (18.1) *is nonempty.*

Algorithm: Gradient method in the Riemanian manifold $\mathcal{M}$

```
Step 0.  Let $p_0 \in \mathcal{M}$. Set $k = 0$.
Step 1.  If grad $f(p_k) = 0$, then stop; otherwise, choose $t_k > 0$ and compute
             $p_{k+1} := \exp_{p_k}(-t_k \operatorname{grad} f(p_k))$.
Step 2.  Set $k \leftarrow k + 1$ and proceed to Step 1.
```

In the following we present the two most commonly used strategies for choosing the step-size $t_k > 0$ in the gradient method.

Strategy 18.7 (Lipschitz Step-Size) Assume that grad f is Lipschitz continuous with constant $L > 0$. Let $\varepsilon > 0$ and take

$$\varepsilon < t_k \leq \frac{1}{L}. \tag{18.5}$$

Strategy 18.8 (Armijo's step-size) Choose $\delta \in (0, 1)$ and take

$$t_k := \max\left\{2^{-i} : \ f\left(\gamma_k(2^{-i})\right) \leq f(p_k) - \delta 2^{-i} \left\|\operatorname{grad} f(p_k)\right\|^2, \ i = 0, 1, \ldots\right\}, \tag{18.6}$$

where $\gamma_k(2^{-i}) := \exp_{p_k}\left(-2^{-i} \ \operatorname{grad} f(p_k)\right)$. □

The next lemma can be found in [14, Corollary 2.1]. Its proof is a straight forward application of the fundamental theorem of calculus.

Lemma 18.9 *Let* $f : \mathcal{M} \to \mathbb{R}$ *be a differentiable function. Assume that* grad f *is Lipschitz continuous. Then there holds*

$$f\left(\exp_p(-t \operatorname{grad} f(p))\right) \leq f(p) - \left(1 - \frac{L}{2}t\right) t \left\|\operatorname{grad} f(p)\right\|^2, \qquad \forall\, t \in \mathbb{R}.$$

Remark 18.10 Asume that grad f is Lipschitz continuous on $\mathcal{M}$ with constant $L > 0$. Thus, Lemma 18.9 implies that

$$f(\exp_{p_k}(-t \text{ grad } f(p_k))) \leq f(p_k) + \left(\frac{Lt}{2} - 1\right) t \, \|\text{grad } f(p_k)\|^2, \qquad \forall \, t \in \mathbb{R}.$$

Hence, by taking any $t \in (0, 2[1-\delta]/L]$ we conclude from the last inequality that

$$f(\exp_{p_k}(-t \text{ grad } f(p_k))) \leq f(p_k) - \delta t \| \text{ grad } f(p_k)\|^2.$$

Therefore, t_k in Strategy 18.8 satisfies the inequality $t_k > [1-\delta]/L$, for all $k = 0, 1, \ldots$.

The proof of the well-definedness of Strategy 18.8 follows the same usual arguments as in the Euclidean setting. Hence, we assume that the sequence $\{p_k\}$ generated by the gradient method with any of the Strategies 18.7 or 18.8 is well-defined. Finally we remark that grad $f(p) = 0$ if only if $p \in \Omega^*$. Therefore, *from now on we also assume that* grad $f(p_k) \neq 0$, *or equivalently,* $p_k \notin \Omega^*$, *for all* $k = 0, 1, \ldots$.

18.4.1 Asymptotic Convergence Analysis

In this section our goal is to present some convergence results for the gradient method to solve the Problem (18.1) with Strategy 18.7 or 18.8.

Lemma 18.11 *Let* $\{p_k\}$ *be generated by the gradient method with Strategies 18.7 or 18.8. Then,*

$$f(p_{k+1}) \leq f(p_k) - \nu t_k \, \|\text{grad } f(p_k)\|^2, \qquad k = 0, 1, \ldots, \tag{18.7}$$

where $\nu = 1/2$ *for Strategy* 18.7 *and* $\nu = \delta$ *for Strategy* 18.8. *Consequently,* $\{f(p_k)\}$ *is a non-increasing sequence and* $\lim_{k \to +\infty} t_k \| \text{ grad } f(p_k)\|^2 = 0$.

Proof For Strategy *18.8*, inequality (18.7) follows directly from (18.6). Now, we assume that $\{p_k\}$ is generated by using Strategy 18.7. In this case, Lemma 18.9 implies that

$$f(p_{k+1}) = f(\exp_{p_k}(-t_k \text{ grad } f(p_k))) \leq f(p_k) - \left(1 - \frac{L}{2} t_k\right) t_k \, \|\text{grad } f(p_k)\|^2,$$

for all $k = 0, 1, \ldots$. Hence, taking into account (18.5) we have $1/2 \leq (1 - Lt_k/2)$ and then (18.7) follows for Strategy 18.7. It is immediate from (18.7) that $\{f(p_k)\}$ is non-increasing. Moreover, (18.7) implies that

$$\sum_{k=0}^{\ell} t_k \left\|\operatorname{grad} f(p_k)\right\|^2 \leq \frac{1}{\nu} \sum_{k=0}^{\ell} f(p_k) - f(p_{k+1}) \leq \frac{1}{\nu}\left(f(p_0) - f^*\right),$$

for each nonnegative integer ℓ. As a consequence, the sequence $\{t_k \left\|\operatorname{grad} f(p_k)\right\|^2\}$ converges to zero, completing the proof. □

Theorem 18.12 *Let $\{p_k\}$ be generated by the gradient method with Strategies 18.7 or 18.8. Then any cluster point of $\{p_k\}$ is a solution of the Problem* (18.1).

Proof Let $\bar{p}$ be a cluster point of $\{p_k\}$ and $\bar{t} \in [0, \max\{1/L, 1\}]$ a cluster point of $\{t_k\}$. Take a subsequence $\{(t_{k_j}, p_{k_j})\}$ of $\{(t_k, p_k)\}$ such that $\lim_{j\to\infty}\{(t_{k_j}, p_{k_j})\} = \{(\bar{t}, \bar{p})\}$. Since Lemma 18.11 implies that $\lim_{k\to\infty} t_k \left\|\operatorname{grad} f(p_k)\right\|^2 = 0$ and considering that grad f is continuous, we have $0 = \lim_{j\to\infty} t_{k_j} \|\operatorname{grad} f(p_{k_j})\| = \bar{t}\left\|\operatorname{grad} f(\bar{p})\right\|$. If Strategy 18.7 is used we have $\bar{t} > 0$, consequently, grad $f(\bar{p}) = 0$ and then $\bar{p} \in \Omega^*$. If Strategy 18.8 is used and $\bar{t} > 0$, then we also have $\bar{p} \in \Omega^*$. Now, consider the case $\bar{t} = 0$ for Strategy 18.8. Since $\{t_{k_j}\}$ converges to $\bar{t} = 0$, we take $r \in \mathbb{N}$ such that $t_{k_j} < 2^{-r}$ for j sufficiently large. Thus the Armijo's condition (18.6) is not satisfied for 2^{-r}, i.e.,

$$f(\exp_{p_{k_j}}(2^{-r}[-\operatorname{grad} f(p_{k_j})])) > f(p_{k_j}) - 2^{-r}\delta\| \operatorname{grad} f(p_{k_j})\|^2.$$

Letting j go to $+\infty$ in the above inequality and taking into account that grad f and the exponential mapping are continuous, we have

$$-\frac{1}{2^{-r}}\left(f(\exp_{\bar{p}}(2^{-r}[-\operatorname{grad} f(\bar{p})])) - f(\bar{p})\right) \leq \delta \left\|\operatorname{grad} f(\bar{p})\right\|^2.$$

Thus, letting r go to $+\infty$ we obtain $\left\|\operatorname{grad} f(\bar{p})\right\|^2 \leq \delta \left\|\operatorname{grad} f(\bar{p})\right\|^2$, which implies grad $f(\bar{p}) = 0$, i.e., $\bar{p} \in \Omega^*$. □

In order to simplify the notations we define two auxiliary constants. Let $p_0 \in \mathcal{M}$. By using (18.7) together with (18.5) and (18.6) we define the first constant $\rho > 0$ as follows

$$\sum_{k=0}^{\infty} t_k^2 \| \operatorname{grad} f(p_k)\|^2 \leq \rho := \begin{cases} 2[f(p_0) - f^*]/L, & \text{for Strategy 18.7;} \\ [f(p_0) - f^*]/\delta, & \text{for Strategy 18.8.} \end{cases} \tag{18.8}$$

Let $\kappa \in \mathbb{R}$ and $q \in \mathcal{M}$. The second auxiliary constant $\mathcal{K}^q_{\rho,\kappa} > 0$ is defined by

$$\mathcal{K}^q_{\rho,\kappa} := \frac{\sinh\left(\sqrt{|\kappa|\rho}\right)}{\sqrt{|\kappa|\rho}} \, \frac{\cosh^{-1}\left(\cosh(\sqrt{|\kappa|}d(p_0, q))e^{\frac{1}{2}\sqrt{|\kappa|\rho}\sinh(\sqrt{|\kappa|\rho})}\right)}{\tanh\cosh^{-1}\left(\cosh(\sqrt{|\kappa|}d(p_0, q))e^{\frac{1}{2}\sqrt{|\kappa|\rho}\sinh(\sqrt{|\kappa|\rho})}\right)}, \tag{18.9}$$

where ρ is given in (18.8) and $\kappa < 0$. Since $\lim_{\kappa\to 0}\mathcal{K}^q_{\rho,\kappa} = 1$, we define $\mathcal{K}^q_{\rho,0} := 1$.

The proof of the next lemma follows the same arguments used to prove [36, Lemma 6]. Since its proof is quite technical it will be omitted here. It is worth mentioning that this result is a generalization of the corresponding classical one that has appeared in, see for example, [26, Lemma 1.1]. *Finally, to establish the full convergence and iteration-complexity results for the gradient method, we need to assume that the Riemannian manifolds $\mathcal{M}$ under consideration has sectional curvature bounded from below.*

Lemma 18.13 *Let $\mathcal{M}$ be a Riemannian manifold with sectional curvature $K \geq \kappa$, and $\{p_k\}$ be generated by the gradient method with Strategies 18.7 or 18.8. Then, for each $q \in \Omega^*$, there holds*

$$d^2(p_{k+1}, q) \leq d^2(p_k, q) + \mathcal{K}^q_{\rho,\kappa} t_k^2 \| \operatorname{grad} f(p_k)\|^2 + 2t_k[f^* - f(p_k)], \qquad (18.10)$$

for all $k = 0, 1, \ldots$, where ρ is defined in (18.8).

Finally, we are ready to prove the full convergence of $\{p_k\}$ to a minimizer of f.

Theorem 18.14 *Let $\mathcal{M}$ be a Riemannian manifolds with sectional curvature $K \geq \kappa$ and $\{p_k\}$ be generated by the gradient method with Strategies 18.7 or 18.8. Then $\{p_k\}$ converges to a solution of the Problem* (18.1).

Proof Lemma 18.13, (18.8) and Definition 18.5 imply that $\{p_k\}$ is quasi-Fejér convergent to the set Ω^*. Let $\bar{p}$ a cluster point of $\{p_k\}$, by Theorem 18.12 we have $\bar{p} \in \Omega^*$. Therefore, since $\{p_k\}$ is quasi-Fejér convergent to Ω^*, we conclude from Theorem 18.6 that $\{p_k\}$ converges to $\bar{p}$ and the proof is completed. □

18.4.2 Iteration-Complexity Analysis

In this section, it will be also assumed that $\mathcal{M}$ is a Riemannian manifolds with sectional curvature $K \geq \kappa$ and $\{p_k\}$ is generated by the gradient method with Strategies 18.7 or 18.8. Our aim is to present iteration-complexity bounds related to the gradient method for minimizing a convex *function with Lipschitz continuous gradient with constant $L > 0$*. For this purpose, by using (18.5) and Remark 18.10, define

$$\xi := \begin{cases} \epsilon\,, & \text{for Strategy 18.7;} \\ [1-\delta]/L\,, & \text{for Strategy 18.8.} \end{cases} \qquad (18.11)$$

We recall that the constants ρ and $\mathcal{K}^q_{\rho,\kappa}$ are defined in (18.8) and (18.9), respectively.

Theorem 18.15 *For every $N \in \mathbb{N}$ and $q \in \Omega^*$, there holds*

$$f(p_N) - f^* \leq \frac{d^2(p_0, q) + \mathcal{K}^q_{\rho,\kappa}\rho}{2\xi N}. \tag{18.12}$$

Proof Since $f(p_k) - f^* \geq 0$, by using Lemma 18.13 and (18.11), we conclude that

$$2\xi\left(f(p_k) - f^*\right) \leq d^2(p_k, q) - d^2(p_{k+1}, q) + \mathcal{K}^q_{\rho,\kappa} t_k^2 \|\operatorname{grad} f(p_k)\|^2.$$

Thus, by summing both sides for $k = 0, 1, \ldots, N - 1$ and using (18.8), we have

$$2\xi \sum_{k=0}^{N-1} \left(f(p_k) - f^*\right) \leq d^2(p_0, q) + \mathcal{K}^q_{\rho,\kappa}\rho.$$

Therefore, due to $\{f(p_k)\}$ be a decreasing sequence, this inequality implies (18.12). □

Theorem 18.16 *For every $N \in \mathbb{N}$ and $q \in \Omega^*$, there holds*

$$\min\{\|\operatorname{grad} f(p_k)\| : k = 0, 1, \ldots, N\} \leq \left[\frac{2\left(d^2(p_0, q) + \mathcal{K}^q_{\rho,\kappa}\rho\right)}{\nu\xi^2}\right]^{\frac{1}{2}} \frac{1}{N},$$

where $\nu = 1/2$ for Strategy 18.7 and $\nu = \delta$ for Strategy 18.8.

Proof Let $N \in \mathbb{N}$ and denote by $\lceil N/2 \rceil$ the least integer that is greater than or equal to $N/2$. It follows from Lemma 18.11 that $\nu t_k \|\operatorname{grad} f(p_k)\|^2 \leq f(p_k) - f(p_{k+1})$, for all $k = 0, 1, \ldots$. Thus, by summing both sides of this inequality for $k = \lceil N/2 \rceil, \ldots, N$ and using (18.11), we obtain

$$\nu\xi \sum_{k=\lceil N/2 \rceil}^{N} \|\operatorname{grad} f(p_k)\|^2 \leq f(p_{\lceil N/2 \rceil}) - f(p_{N+1}) \leq f(p_{\lceil N/2 \rceil}) - f^*.$$

Thus, from Theorem 18.15 and considering that $N/2 \leq \lceil N/2 \rceil$ it follows that

$$\sum_{k=\lceil N/2 \rceil}^{N} \|\operatorname{grad} f(p_k)\|^2 \leq \frac{d^2(p_0, q) + \mathcal{K}^q_{\rho,\kappa}\rho}{2\nu\xi^2\lceil N/2 \rceil} \leq \frac{d^2(p_0, q) + \mathcal{K}^q_{\rho,\kappa}\rho}{\nu\xi^2 N}.$$

Hence, $\min\{\|\operatorname{grad} f(p_k)\|^2 : k = \lceil N/2 \rceil, \ldots, N\} \leq 2(d^2(p_0, q) + \mathcal{K}^q_{\rho,\kappa}\rho)/(\nu\xi^2 N^2)$, which implies the desired inequality. □

Bibliographic Notes and Remarks

In order to deal with constrained optimization problems in Euclidean space, Luenberger [56] proposed and established important convergence properties of the projected gradient method by using the Riemannian structure of the constraint set induced by the Euclidean structure. To the best of our knowledge, this was the first result involving concepts of Riemannian geometry to study optimization methods. After this seminal work, the gradient method has been studied in general Riemannian settings. Early works dealing with this method include [40, 67, 73, 76]. However, the obtained results in these previous works demand some sort of boundedness of the sequence, establishing only that all its cluster points are stationary. By assuming convexity of the objective function and that the manifold has *nonnegative curvature*, it has been proven in [28] that, for a suitable choice of the step size and without any boundedness assumption, the whole sequence converges to a solution. It is worth noting that this was the first Riemannian optimization result using the concept of curvature. Other variants and generalizations of this method can be found in [20, 49, 63, 88]. In the last years important properties of the gradient method in Riemannian settings have been obtained. For instance, in [92] the authors provided iteration-complexity bounds of the method for convex optimization problems on Hadamard manifolds. In [21], the authors established iteration-complexity bounds without assuming convexity of the objective function and curvature of the manifold. In [19] the gradient method is considered to compute the Karcher mean in the cone of symmetric positive definite matrices endowed with a suitable Riemannian metric. In [3] the authors study properties of the gradient method for the problem of finding the global Riemannian center of mass of a set of data points on a Riemannian manifold. The paper [15] extends the convergence analysis of the gradient method to Hadamard setting for functions which satisfy the Kurdyka-Lojasiewicz inequality. In [14] an iteration-complexity analysis of the method for convex optimization problems on Riemannian manifolds with nonnegative sectional curvature is presented. In [36] the full convergence and iteration-complexity analysis of the gradient method for convex optimization problems on Riemannian manifolds with lower bounded sectional curvature are presented without any assumptions on the boundedness of level sets.

18.5 Subgradient Method for Optimization

In this section we study properties of the subgradient method for solving Problem (18.1) with the two most commonly used strategies for choosing the step-size. We recall that $f : \mathcal{M} \to \mathbb{R}$ is a convex function. *Throughout this section we assume that* $\mathcal{M}$ *is a Riemannian manifold with sectional curvature* $K \geq \kappa$ *and* $\kappa \leq 0$. *We recall that* Ω^**, the solution set of Problem* (18.1)*, will be assumed to be nonempty only when explicitly stated.*

```
Algorithm: Subgradient method in Riemanian manifold $\mathcal{M}$
Step 0.  Let $p_0 \in \mathcal{M}$ and $s_0 \in \partial f(p_0)$. Set $k = 0$.
Step 1.  If $s_k = 0$, then stop; otherwise, choose a step-size $t_k > 0$ , $s_k \in \partial f(p_k)$ and compute
             $p_{k+1} := \exp_{p_k}\left(-t_k \frac{s_k}{\|s_k\|}\right);$
Step 2.  Set $k \leftarrow k + 1$ and proceed to Step 1.
```

In the following we present two different strategies for choosing the step-size $t_k > 0$.

Strategy 18.17 (Exogenous Step-Size)

$$t_k > 0, \qquad \sum_{k=0}^{\infty} t_k = +\infty, \qquad \sigma := \sum_{k=0}^{\infty} t_k^2 < +\infty. \tag{18.13}$$

The subgradient method with the step-size in Strategy 18.17 has been analyzed in several papers; see, for example, [26, 34, 79].

Strategy 18.18 (Polyak's Step-Size) Assume that $p_0 \in \mathcal{M}$, $\Omega^* \neq \varnothing$ and set

$$t_k = \alpha \frac{f(p_k) - f^*}{\|s_k\|}, \qquad 0 < \alpha < 2\frac{\tanh\left(\sqrt{|\kappa|}d_0\right)}{\sqrt{|\kappa|}d_0}, \qquad d_0 := d(p_0, \Omega^*), \tag{18.14}$$

where $d(p_0, \Omega^*) := \inf\{d(p_0, q) : q \in \Omega^*\} > 0$ and $\kappa \neq 0$, and $0 < \alpha < 2$ for $\kappa = 0$. □

The step-size rule of Strategy 18.18 was introduced in [66] and has been used in several papers, including [11, 14, 81].

Remark 18.19 Since $(0, +\infty) \mapsto \tanh(t)/t$ is decreasing, for any $\hat{d} > d_0$, we choose $0 < \alpha < 2\tanh(\sqrt{|\kappa|}\hat{d})/(\sqrt{\kappa|}\hat{d})) < 2\tanh(\sqrt{|\kappa|}d_0)/(\sqrt{\kappa|}d_0))$ in Strategy 18.18. As $\lim_{t\to 0}\tanh(t)/t = 1$, we choose $0 < \alpha < 2$ in (18.14) for a Riemannian manifold with non-negative curvature, i.e., $\kappa = 0$. Note that all results in this section can be obtained by assuming only that f is convex in a subset of $\mathcal{M}$, see the details in [37].

18.5.1 Asymptotic Convergence Analysis

Firstly, we assume that the sequence $\{p_k\}$ is generated by the subgradient method with Strategy 18.17. To proceed with the analysis, define

$$\Omega := \left\{q \in \mathcal{M} : \; f(q) \leq \inf_k f(p_k)\right\}.$$

The next result contains an inequality that is of fundamental importance to analyze the subgradient method. For each $q \in \Omega$ and $\kappa \in \mathbb{R}$, let us define

$$C_{q,\kappa} := \frac{\sinh\left(\sqrt{|\kappa|\sigma}\right)}{\sqrt{|\kappa|\sigma}} \left[1 + \cosh^{-1}\left(\cosh(|\kappa| d(p_0, q)) e^{\frac{1}{2}\sqrt{|\kappa|\sigma}\sinh(\sqrt{|\kappa|\sigma})}\right)\right],$$

where σ is given in (18.13) and $\kappa < 0$. Since $\lim_{\kappa \to 0} C_{q,\kappa} = 1$, we define $C_{q,0} := 1$. It is important to note that $C_{q,\kappa}$ is well defined only under the assumption $\Omega \neq \varnothing$. The inequality in the next lemma is a version of (18.10) for the subgradient method. Since its proof is quite technical it will be also omitted here, for details see [37, Lemma 3.2].

Lemma 18.20 *Let $\{p_k\}$ be generated by the subgradient method with Strategy 18.17. If $\Omega \neq \varnothing$ then, for each $q \in \Omega$ and $k = 0, 1, \ldots$, there holds*

$$d^2(p_{k+1}, q) \leq d^2(p_k, q) + C_{q,\kappa} t_k^2 + 2\,\frac{t_k}{\|s_k\|}[f(q) - f(p_k)], \qquad s_k \in \partial f(p_k).$$

Now, we are ready to establish the asymptotic analysis of the sequence $\{p_k\}$. We first consider the subgradient method with Strategy 18.17.

Theorem 18.21 *Let $\{p_k\}$ be generated by the subgradient method with Strategy 18.17. Then*

$$\liminf_k f(p_k) = f^*. \tag{18.15}$$

In addition, if $\Omega^* \neq \varnothing$ then the sequence $\{p_k\}$ converges to a point $p_* \in \Omega^*$.

Proof Assume by contradiction that $\liminf_k f(p_k) > f^* = \inf_{p \in \mathcal{M}} f(p)$. In this case, we have $\Omega \neq \varnothing$. Thus, from Lemma 18.20, we conclude that $\{p_k\}$ is bounded and, consequently, by using Proposition 18.2, the sequence $\{s_k\}$ is also bounded. Let $C_1 > \|s_k\|$, for $k = 0, 1, \ldots$. On the other hand, letting $q \in \Omega$, there exist $C_2 > 0$ and $k_0 \in \mathbb{N}$ such that $f(q) < f(p_k) - C_2$, for all $k \geq k_0$. Hence, using Lemma 18.20 and considering that $\|s_k\| < C_1$, for $k = 0, 1, \ldots$, we have

$$d^2(p_{k+1}, q) \leq d^2(p_k, q) + C_{q,\kappa} t_k^2 - 2\frac{C_2}{C_1} t_k, \qquad k = k_0, k_0 + 1, \ldots.$$

Let $\ell \in \mathbb{N}$. Thus, from the last inequality, after some calculations, we obtain

$$\frac{2\,C_2}{C_1} \sum_{j=k_0}^{\ell+k_0} t_j \leq d^2(p_{k_0}, q) - d^2(p_{k_0+\ell}, q) + C_{q,\kappa} \sum_{j=k_0}^{\ell+k_0} t_j^2 \leq d^2(p_{k_0}, q) + C_{q,\kappa} \sum_{j=k_0}^{\ell+k_0} t_j^2.$$

Hence, using the inequality in (18.13), we have a contraction. Therefore, (18.15) holds.

For proving the last statement, let us assume that $\Omega^* \neq \varnothing$. In this case, we have $\Omega \neq \varnothing$ and, from Lemma 18.20, $\{p_k\}$ is quasi-Féjer convergent to Ω and, consequently, a bounded sequence. The equality (18.15) implies that $\{f(p_k)\}$ possesses a decreasing monotonous subsequence $\{f(p_{k_j})\}$ such that $\lim_{j\to\infty} f(p_{k_j}) = f^*$. We can assume that $\{f(p_k)\}$ is decreasing, monotonous and converges to f^*. Being bounded, the sequence $\{p_k\}$ possesses a convergent subsequence $\{p_{k_\ell}\}$. Let us say that $\lim_{\ell\to\infty} p_{k_\ell} = p_*$, which by the continuity of f implies $f(p_*) = \lim_{\ell\to\infty} f(p_{k_\ell}) = f^*$, and then $p_* \in \Omega$. Hence, $\{p_k\}$ has an cluster point $p_* \in \Omega$, and due to $\{p_k\}$ be quasi-Féjer convergent to Ω, it follows from Theorem 18.6 that the sequence $\{p_k\}$ converges to p_*. □

Now, we will assume that $\{p_k\}$ is generated by the subgradient method with Strategy 18.18. Let α and d_0 be as in (18.14) and define

$$C_{\kappa,d_0} := \frac{2}{\alpha} - \frac{\sqrt{|\kappa|}d_0}{\tanh\left(\sqrt{|\kappa|}d_0\right)} > 0. \tag{18.16}$$

Remark 18.22 Since $\lim_{t\to 0}\tanh(t)/t = 1$, we conclude that for Riemannian manifolds with nonnegative curvature, namely, for $\kappa = 0$, (18.16) become $C_{\kappa,d_0} \equiv 2/\alpha - 1 > 0$.

The next lemma plays an important role in our analysis, its proof can be found in [37, Lemma 3.3]. For stating the lemma, we set

$$\bar{q} \in \Omega^* \quad \text{such that} \quad d_0 = d(p_0, \bar{q}). \tag{18.17}$$

Lemma 18.23 *Let $\{p_k\}$ be generated by the subgradient method with Strategy 18.18. Let $\bar{q} \in \Omega^*$ satisfy (18.17). Then the following inequality holds*

$$d^2(p_{k+1}, \bar{q}) \le d^2(p_k, \bar{q}) - C_{\kappa,d_0}\alpha^2 \frac{\left[f(p_k) - f^*\right]^2}{\|s_k\|^2}, \qquad k = 0, 1, \ldots. \tag{18.18}$$

Remark 18.24 For $\bar{\kappa} = 0$, we do not need a $\bar{q}$ satisfying (18.17) to prove Lemma 18.23. In this case, (18.18) becomes $d^2(p_{k+1}, q) \le d^2(p_k, q) - (2/\alpha - 1)t_k^2$, for $k = 0, 1, \ldots$ and $q \in \Omega^*$.

Theorem 18.25 *Let $\{p_k\}$ be generated by the subgradient method with Strategy 18.18. Then, $\lim_{k\to\infty} f(p_k) = f^*$. Consequently, each cluster point of the sequence $\{p_k\}$ is a solution of the Problem (18.1).*

Proof Let $\bar{q}$ satisfy (18.17). In particular, Lemma 18.23 implies that $\{d^2(p_k, \bar{q})\}$ is monotonically nonincreasing and, being nonnegative, it converges. Moreover, $\{p_k\}$ is bounded. Thus, from Proposition 18.2 we conclude that $\{s_k\}$ is also bonded. Hence, letting k go to $+\infty$ in (18.18), we conclude that $\lim_{k\to\infty} f(p_k) = f^*$. Now, let $\bar{p}$ be an accumulation point of $\{p_k\}$ and $\{p_{k_i}\}$ a subsequence of $\{p_k\}$ such that $\lim_{k_i\to+\infty} p_{k_i} = \bar{p}$. Therefore, $f(\bar{p}) = \lim_{k_i\to\infty} f(p_{k_i}) = f^*$ and then $\bar{p} \in \Omega^*$ and the proof is concluded. □

Corollary 18.26 *For $\kappa = 0$ the sequence $\{p_k\}$ converges to a point $q \in \Omega^*$.*

Proof Lemma 18.23 and Remark 18.24 imply that $\{p_k\}$ is bounded. Let $\bar{p}$ be an accumulation point of $\{p_k\}$ and $\{p_{k_i}\}$ a subsequence of $\{p_k\}$ such that $\lim_{k_i\to+\infty} p_{k_i} = \bar{p}$. Thus, Theorem 18.25 implies that $\bar{p} \in \Omega^*$. Using again Lemma 18.23, we obtain also that $\{d(p_k, \bar{p})\}$ is monotonically nonincreasing and, being nonegative, it converges. Since $\lim_{k\to\infty} d(p_{k_j}, \bar{p}) = 0$, we have $\lim_{k\to\infty} d(p_k, \bar{p}) = 0$. Therefore, $\{p_k\}$ converges. □

18.5.2 Iteration-Complexity Analysis

In this section, we present iteration-complexity bounds related to the subgradient method. Throughout this section we assume that $\Omega^* \neq \varnothing$. Set $p_* \in \Omega^*$ such that $\lim_{k\to\infty} p_k = p^*$ and $\tau > 0$ such that

$$\|s_k\| \le \tau, \qquad k = 0, 1, \ldots. \tag{18.19}$$

For instance, if $f : \mathcal{M} \to \mathbb{R}$ is Lipschitz continuous, then τ in (18.19) can be taken as the Lipschitz constant of f.

Theorem 18.27 *Let $\{p_k\}$ be generated by the subgradient method with Strategy 18.17. Then, for all $p_* \in \Omega^*$ and every $N \in \mathbb{N}$, the following inequality holds*

$$\min\left\{f(p_k) - f^* \;:\; k = 0, 1, \ldots, N\right\} \le \tau \frac{d^2(p_0, p_*) + C_{p_*,\kappa} \sum_{k=0}^N t_k^2}{2\sum_{k=0}^N t_k}.$$

Proof Let $p_* \in \Omega^*$. Since $\Omega^* \subset \Omega$, applying Lemma 18.20 with $q = p_*$, we obtain

$$d^2(p_{k+1}, p_*) \leqslant d^2(p_k, p_*) + C_{p_*,\kappa} t_k^2 + 2\frac{t_k}{\|s_k\|}[f^* - f(p_k)], \qquad s_k \in \partial f(p_k),$$

for all $k = 0, 1, \ldots$. Hence, summing up the above inequality for $k = 0, 1, \ldots, N$, after some algebraic manipulations, we have

$$2\sum_{k=0}^N \frac{t_k}{\|s_k\|}[f(p_k) - f^*] \le d^2(p_0, p_*) - d^2(p_{N+1}, p_*) + C_{p_*,\kappa} \sum_{k=0}^N t_k^2.$$

By (18.19) we have $\|s_k\| \le \tau$, for all $k = 0, 1, \ldots$. Therefore, we conclude

$$\frac{2}{\tau}\min\left\{f(p_k) - f^* \,:\, k = 0, 1, \ldots, N\right\}\sum_{k=0}^N t_k \le d^2(p_0, p_*) + C_{p_*,\kappa} \sum_{k=0}^N t_k^2,$$

which is equivalent to the desired inequality. □

The next result presents an iteration-complexity bound for the subgradient method with the Polyak's step-size rule.

Theorem 18.28 *Let $\{p_k\}$ be generated by the subgradient method with Strategy 18.18. Let $\bar{q} \in \Omega^*$ satisfy (18.17). Then, for every $N \in \mathbb{N}$, there holds*

$$\min\left\{f(p_k) - f^* \;:\; k = 0, 1, \ldots, N\right\} \le \frac{\tau d(p_0, \bar{q})}{\sqrt{C_{\kappa,d_0}}} \frac{1}{\sqrt{N+1}}.$$

Proof It follows from Lemma 18.23 and (18.19) that

$$[f(p_k) - f^*]^2 \le \frac{\tau^2}{C_{\kappa,d_0}\alpha^2}[d^2(p_k, \bar{q}) - d^2(p_{k+1}, \bar{q})], \qquad k = 0, 1, \ldots.$$

Performing the sum of the above inequality for $k = 0, 1, \ldots, N$, we obtain

$$\sum_{k=0}^{N}[f(p_k) - f^*]^2 \le \frac{\tau^2 d^2(p_0, \bar{q})}{C_{\kappa,d_0}}.$$

The statement of the theorem is an immediate consequence of the last inequality.

□

Remark 18.29 It is worth noting that if $\kappa = 0$ we have $C_{q,\kappa} = 1$ and then the inequality of Theorem 9 reduces to the inequality in [14, Theorem 3.4].

Bibliographic Notes and Remarks

The subgradient method was originally developed by Shor and others in the 1960s and 1970s and since of then, it and its variants have been applied to a far wider variety of problems in optimization theory; see[41, 66]. In order to deal with nonsmooth convex optimization problems on Riemanian manifolds, [34] extended and analyzed the subgradient method under the assumption that the sectional curvature of the manifolds is non-negative. As in the Euclidean context, the subgradient method is quite simple and possess nice convergence properties. After this work, the subgradient method in the Riemannian setting has been studied in different contexts; see, for instance, [9, 11, 37, 42, 79, 81, 91–93]. In [11] the subgradient method was introduced to solve convex feasibility problems on complete Riemannian manifolds with non-negative sectional curvatures. Significant improvements for this method were introduced in [81], by extending its analysis to manifolds with lower bounded sectional curvature. More recently, in [37, 79] an asymptotic and iteration-complexity analysis of the subgradient method for convex optimization was carried out in the context of manifolds with lower bounded sectional curvatures. In [37] the authors establish an iteration-complexity bound of the subgradient method with exogenous step-size and Polyak's step-size, for convex optimization problems on complete Riemannian manifolds with lower bounded sectional curvatures. These results, in some sense, increase the range of applicability of the method compared to the respective results obtained in [14, 92, 93]. We end this section by remarking that [91] proposes and analyzes an incremental subgradient method for Riemannian optimization.

18.6 Proximal Point Method for Optimization

In this section we study properties of the proximal point method for solving Problem (18.1), where $f : \mathcal{M} \to \mathbb{R}$ is a convex function. *Throughout this section we assume that $\mathcal{M}$ is a Hadamard manifold. We recall that Ω^*, the solution set of Problem* (18.1)*, will be assumed to be nonempty only when explicitly stated.*

Algorithm: Proximal point method in the Riemanian manifold $\mathcal{M}$

```
Step 0.  Let $p_0 \in \mathcal{M}$ and $\{\lambda_k\} \subset (0, +\infty)$. Set $k = 0$.
Step 1.  Compute
             $p_{k+1} = \operatorname{argmin}_{p \in \mathcal{M}} \left\{ f(p) + \frac{\lambda_k}{2} d^2(p_k, p) \right\};$      (18.20)
Step 2.  Set $k \leftarrow k + 1$ and proceed to Step 1.
```

The parameter λ_k will be chosen later, in order to guarantee a specific convergence property. The well-definedness of the iteration (18.20) is presented in detail in [35, Theorem 5.1]. On the other hand, since in Hadamard manifolds the square of the Riemannian distance is strongly convex, see [29, Corollary3.1], there is also an alternative proof of the well-definedness of the iteration (18.20). In the next lemma we present an important inequality, which is the main tool to establish both asymptotic convergence and iteration-complexity to the method. At this point we emphasize that to obtain this inequality it is fundamental that $\mathcal{M}$ is a Hadamard manifold or a more general manifold where the square of the distance is strongly convex. As the proof of this inequality is quite technical we skip it, see the details in [35, Lemma 6.2].

Lemma 18.30 *Let $\{p_k\}$ be the sequence generated by the proximal point method. Then, for every $q \in \mathcal{M}$, there holds*

$$d^2(p_{k+1}, q) \leq d^2(p_k, q) - d^2(p_{k+1}, p_k) + \frac{2}{\lambda_k}\left(f(q) - f(p_{k+1})\right), \qquad k = 0, 1, \ldots .$$

18.6.1 Asymptotic Convergence Analysis

In this section we present an asymptotic convergence analysis of the proximal point method. As aforementioned, the convergence result is an application of Lemma 18.30.

Theorem 18.31 *Let $\{p_k\}$ be the sequence generated by the proximal method. If the sequence $\{\lambda_k\}$ is chosen satisfying $\sum_{k=0}^{\infty}(1/\lambda_k) = +\infty$, then $\lim_{k\to+\infty} f(p_k) = f^*$. In addition, if Ω^* is nonempty then $\lim_{k\to+\infty} p_k = p_*$ and $p_* \in \Omega^*$.*

Proof It follows from (18.20) that $f(p_{k+1}) + (\lambda_k/2)d^2(p_k, p_{k+1}) \leq f(p) + \frac{\lambda_k}{2} d^2(p_k, p)$, for all $p \in \mathcal{M}$. Thus, $f(p_{k+1}) + (\lambda_k/2)d^2(p_k, p_{k+1}) \leq f(p_k)$, for all $k = 0, 1, \ldots$, which implies $f(p_{k+1}) \leq f(p_k)$, for all $k = 0, 1, \ldots$. Thus the sequence $\{f(p_k)\}$ is non increasing. To prove that $\lim_{k\to+\infty} f(p_k) = f^*$, we assume for contradiction that $\lim_{k\to+\infty} f(p_k) > f^*$. This, assumption implies that there exist $q \in \mathcal{M}$ and $\delta > 0$ such that $f(q) < f(p_k) - \delta$, for all $k = 0, 1, \ldots$. The combination of the last inequality with Lemma 18.30 yield $d^2(p_{k+1}, q) < d^2(p_k, q) - 2\delta/\lambda_k$, for all $k = 0, 1, \ldots$. Thus,

$$\sum_{k=0}^{j} \frac{1}{\lambda_k} \leqslant \frac{1}{2\delta}(d^2(p_0, q) - d^2(p_0, p_{j+1})) < \frac{1}{2\delta} d^2(p_0, q), \qquad \forall j \in \mathbb{N},$$

which contradicts the equality $\sum_{k=0}^{\infty}(1/\lambda_k) = +\infty$ and the desired equality holds. Now assume that Ω^* is nonempty. Thus taking $\bar{q} \in \Omega^*$ we have $f(\bar{q}) \leqslant f(p_k)$ for all $k = 0, 1, \ldots$. Hence, by Lemma 18.30 we obtain $d^2(p_{k+1}, \bar{q}) < d^2(p_k, \bar{q})$, for all $k = 0, 1, \ldots$. Therefore, the sequence $\{p_k\}$ is Féjer convergent to set Ω^*, and then $\{p_k\}$ is bounded. Let $\{p_{k_j}\}$ be a subsequence of $\{p_k\}$ such that $\lim_{k\to+\infty} p_{k_j} = p_*$. Since f is continuous and $\lim_{k\to+\infty} f(p_k) = f^*$ it follows that $f(p^*) = \lim_{k\to+\infty} f(p_{k_j}) = f^*$. Hence, $p^* \in \Omega^*$. Therefore the cluster point p_* of $\{p_k\}$ belongs to Ω^*, again by Theorem 18.6 we conclude that $\lim_{k\to+\infty} p_k = p_*$. □

18.6.2 Iteration-Complexity Analysis

In the next theorem we present an iteration-complexity bound for the proximal point method. As we will see, it is a straight application of the inequality in Lemma 18.30. For stating the theorem we need to assume that the solution set Ω^* is nonempty. Let $p_* \in \Omega^*$ and $f^* = f(p^*)$.

Theorem 18.32 *Let $\{p_k\}$ be the sequence generated by the proximal method with $\lambda_k \in (0, \lambda)$, for all $k = 0, 1, \ldots$ and $\lambda > 0$. Then, for every $N \in \mathbb{N}$, there holds*

$$f(p_N) - f^* \leq \frac{\lambda d^2(p_*, p_0)}{2(N+1)}.$$

As a consequence, given a tolerance $\epsilon > 0$, the number of iterations required by the proximal point method to obtain $p_N \in \mathcal{M}$ such that $f(p_N) - f^* \leq \epsilon$, is bounded by $\mathcal{O}(\lambda d^2(p_*, p_0)/\epsilon)$.

Proof It follows from Lemma 18.30 by taking $q = p^*$ and using $f^* = f(p^*)$, that

$$0 \leq f(p_{k+1}) - f^* \leq \frac{\lambda_k}{2}\left[d^2(p^*, p_k) - d^2(p_*, p_{k+1})\right], \qquad k = 0, 1, \ldots.$$

Hence, summing both sides of the last inequality for $k = 0, 1, \ldots, N$ and using that $\lambda_k \in (0, \lambda)$, we obtain

$$\sum_{k=0}^{N}(f(p_{k+1}) - f^*) \leq \frac{\lambda}{2}\left[d^2(p_0, p^*) - d^2(p_*, p_N)\right] \leq \frac{\lambda}{2}d^2(p_0, p^*). \qquad (18.21)$$

As in the proof of Theorem 18.31, we can prove that $f^* \leq f(p_{k+1}) \leq f(p_k)$, for all $k = 0, 1, \ldots$. Therefore, (18.21) implies that $(N + 1)(f(p_N) - f^*) \leq \lambda d^2(p_0, p^*)/2$, which proves the first statement of the theorem. The last statement of the theorem is an immediate consequence of the first one. □

Bibliographic Notes and Remarks

The proximal point method is one of the most interesting optimization methods, which was first proposed in the linear context by [58] and extensively studied by [68]. In the Riemannian setting, the proximal point method was first studied in [35] for convex optimization problems on Hadamard manifolds. This method has been quite explored since then in different contexts; see, for example, [7, 10, 12, 13, 47, 52, 54, 62, 74]. After introducing the method to optimization problem in Riemannian manifolds in [35], a significant improvement in this context was presented in [54]. In [54] the method was generalize to find a singularity of a monotone vector fields on Hadamard manifolds and new developments emerged after this work; see [7, 53, 82]. Besides generalizing this method, new concepts were also introduced allowing to extend the study of new methods, including [7, 55, 80, 83]. Another important development along these lines was presented in [5], which introduced the method into geodesic metric spaces of non-positive curvature, the so-called $CAT(0)$ spaces; other studies in this direction include [24, 25, 27, 51, 61, 65, 77]. Finally, we mention that the first iteration-complexity result to the proximal point method in manifolds context appeared in [5].

18.7 Conclusions

In this chapter we presented some results related to the asymptotic convergence and iteration-complexity of subgradient, gradient and proximal point methods. We know that a comprehensive study about first order methods on Riemannian manifolds should also include at least some results concerning the conjugate gradient and conditional gradient methods. At this point, we have included only a few comments on them and a more detailed study should be presented in the future. The conjugate gradient method on Riemannian setting was introduced by [73]. As in the Euclidean context, this method has become quite popular in the Riemannian scenario, papers addressing theoretical and applications of this method include [44, 72, 89, 94]. The Frank-Wolfe algorithm or conditional gradient is one of the oldest first-order methods for constrained convex optimization, dating back to the 1950s. It is surprising that this method has only recently been considered in the Riemannian context, see [84]. Numerical experiments have been presented in [84] indicating that this is a promising method for solving some important special classes of Riemannian convex optimization problems. We conclude this chapter with some perspectives and open problems. Under the hypothesis of boundedness from below of the sectional curvature of the Riemannian manifolds we were able to prove full convergence of the gradient and subgradient method. A natural issue to be investigated would be to extend the results of full convergence of these methods to more general manifolds, in particular, to extend the results for Riemannian manifolds with Ricci curvature bounded from below. It is also worth noting that the results on the proximal point method for minimizing convex functions so far were only for manifolds and/or geodesics spaces with negative curvature. Then, one would be interesting to investigate versions of this method for minimizing convex functions in general manifolds extending the results for Riemannian manifolds with nonnegative sectional curvature. We know that the distance on a Riemannian manifold with nonpositive curvature is, in general, not convex in the whole manifold. In this case, the difficulty of proving the well-definedess of the method emerges. Then, one possibility would be to develop a local theory for the method.

Acknowledgements This work was supported by CNPq 302473/2017 − 3, 408151/2016 − 1 and 424860/2018 − 0, FAPEG/CNPq/PPP—03/2015 and FAPEG/PRONEM—201710267000532. This work is part of a measure which is co-financed by tax revenue based on the budget approved by the members of the Saxon state parliament. Financial support is gratefully acknowledged.

References

1. Absil, P.A., Baker, C.G., Gallivan, K.A.: Trust-region methods on Riemannian manifolds. Found. Comput. Math. **7**(3), 303–330 (2007)
2. Absil, P.A., Mahony, R., Sepulchre, R.: Optimization Algorithms on Matrix Manifolds. Princeton University Press, Princeton (2008). With a foreword by Paul Van Dooren

3. Afsari, B., Tron, R., Vidal, R.: On the convergence of gradient descent for finding the Riemannian center of mass. SIAM J. Control Optim. **51**(3), 2230–2260 (2013)
4. Arsigny, V., Fillard, P., Pennec, X., Ayache, N.: Geometric means in a novel vector space structure on symmetric positive-definite matrices. SIAM J. Matrix Anal. Appl. **29**(1), 328–347 (2006)
5. Bačák, M.: The proximal point algorithm in metric spaces. Israel J. Math. **194**(2), 689–701 (2013)
6. Bačák, M., Bergmann, R., Steidl, G., Weinmann, A.: A second order nonsmooth variational model for restoring manifold-valued images. SIAM J. Sci. Comput. **38**(1), A567–A597 (2016)
7. Batista, E.E.A., Bento, G.d.C., Ferreira, O.P.: Enlargement of monotone vector fields and an inexact proximal point method for variational inequalities in Hadamard manifolds. J. Optim. Theory Appl. **170**(3), 916–931 (2016)
8. Baust, M., Weinmann, A., Wieczorek, M., Lasser, T., Storath, M., Navab, N.: Combined tensor fitting and TV regularization in diffusion tensor imaging based on a Riemannian manifold approach. IEEE Trans. Med. Imaging **35**(8), 1972–1989 (2016)
9. Bento, G.C., Cruz Neto, J.X.: A subgradient method for multiobjective optimization on Riemannian manifolds. J. Optim. Theory Appl. **159**(1), 125–137 (2013)
10. Bento, G.C., Cruz Neto, J.X.: Finite termination of the proximal point method for convex functions on Hadamard manifolds. Optimization **63**(9), 1281–1288 (2014)
11. Bento, G.C., Melo, J.G.: Subgradient method for convex feasibility on Riemannian manifolds. J. Optim. Theory Appl. **152**(3), 773–785 (2012)
12. Bento, G.C., Ferreira, O.P., Oliveira, P.R.: Proximal point method for a special class of nonconvex functions on Hadamard manifolds. Optimization **64**(2), 289–319 (2015)
13. Bento, G., da Cruz Neto, J., Oliveira, P.R.: A new approach to the proximal point method: convergence on general Riemannian manifolds. J. Optim. Theory Appl. **168**(3), 743–755 (2016)
14. Bento, G.C., Ferreira, O.P., Melo, J.G.: Iteration-complexity of gradient, subgradient and proximal point methods on Riemannian manifolds. J. Optim. Theory Appl. **173**(2), 548–562 (2017)
15. Bento, G.C., Bitar, S.D.B., Cruz Neto, J.X., Oliveira, P.R., Souza, J.C.: Computing Riemannian center of mass on Hadamard manifolds. J. Optim. Theory Appl. (2019)
16. Bergmann, R., Weinmann, A.: A second-order TV-type approach for inpainting and denoising higher dimensional combined cyclic and vector space data. J. Math. Imaging Vision **55**(3), 401–427 (2016)
17. Bergmann, R., Persch, J., Steidl, G.: A parallel Douglas-Rachford algorithm for minimizing ROF-like functionals on images with values in symmetric Hadamard manifolds. SIAM J. Imaging Sci. **9**(3), 901–937 (2016)
18. Bhattacharya, A., Bhattacharya, R.: Statistics on Riemannian manifolds: asymptotic distribution and curvature. Proc. Amer. Math. Soc. **136**(8), 2959–2967 (2008)
19. Bini, D.A., Iannazzo, B.: Computing the Karcher mean of symmetric positive definite matrices. Linear Algebra Appl. **438**(4), 1700–1710 (2013)
20. Bonnabel, S.: Stochastic gradient descent on Riemannian manifolds. IEEE Trans. Automat. Control **58**(9), 2217–2229 (2013)
21. Boumal, N., Absil, P.A., Cartis, C.: Global rates of convergence for nonconvex optimization on manifolds. IMA J. Numer. Anal. **39**(1), 1–33 (2018)
22. Bredies, K., Holler, M., Storath, M., Weinmann, A.: Total generalized variation for manifold-valued data. SIAM J. Imaging Sci. **11**(3), 1785–1848 (2018)
23. Burachik, R., Drummond, L.M.G., Iusem, A.N., Svaiter, B.F.: Full convergence of the steepest descent method with inexact line searches. Optimization **32**(2), 137–146 (1995)
24. Chaipunya, P., Kumam, P.: On the proximal point method in Hadamard spaces. Optimization **66**(10), 1647–1665 (2017)
25. Cholamjiak, P., Abdou, A.A.N., Cho, Y.J.: Proximal point algorithms involving fixed points of nonexpansive mappings in CAT(0) spaces. Fixed Point Theory Appl. **2015**(13), 227 (2015)

26. Correa, R., Lemaréchal, C.: Convergence of some algorithms for convex minimization. Math. Program. **62**(2, Ser. B), 261–275 (1993)
27. Cuntavepanit, A., Phuengrattana, W.: On solving the minimization problem and the fixed-point problem for a finite family of non-expansive mappings in CAT(0) spaces. Optim. Methods Softw. **33**(2), 311–321 (2018)
28. da Cruz Neto, J.X., de Lima, L.L., Oliveira, P.R.: Geodesic algorithms in Riemannian geometry. Balkan J. Geom. Appl. **3**(2), 89–100 (1998)
29. da Cruz Neto, J.X., Ferreira, O.P., Lucambio Pérez, L.R.: Contributions to the study of monotone vector fields. Acta Math. Hungar. **94**(4), 307–320 (2002)
30. Da Cruz Neto, J.X., Ferreira, O.P., Pérez, L.R.L., Németh, S.Z.: Convex- and monotone-transformable mathematical programming problems and a proximal-like point method. J. Global Optim. **35**(1), 53–69 (2006)
31. do Carmo, M.P.: Riemannian Geometry. Mathematics: Theory & Applications. Birkhäuser Boston, Boston (1992). Translated from the second Portuguese edition by Francis Flaherty
32. Edelman, A., Arias, T.A., Smith, S.T.: The geometry of algorithms with orthogonality constraints. SIAM J. Matrix Anal. Appl. **20**(2), 303–353 (1999)
33. Esposito, M., Hennersperger, C., Gobl, R., Demaret, L., Storath, M., Navab, N., Baust, M., Weinmann, A.: Total variation regularization of pose signals with an application to 3D freehand ultrasound. IEEE Trans. Med. Imaging **38**(10), 2245–2258 (2019)
34. Ferreira, O.P., Oliveira, P.R.: Subgradient algorithm on Riemannian manifolds. J. Optim. Theory Appl. **97**(1), 93–104 (1998)
35. Ferreira, O.P., Oliveira, P.R.: Proximal point algorithm on Riemannian manifolds. Optimization **51**(2), 257–270 (2002)
36. Ferreira, O.P., Louzeiro, M.S., Prudente, L.F.: Gradient method for optimization on riemannian manifolds with lower bounded curvature. SIAM J. Optim. **29**(4), 2517–2541 (2019). e-prints. arXiv:1806.02694
37. Ferreira, O.P., Louzeiro, M.S., Prudente, L.F.: Iteration-complexity of the subgradient method on Riemannian manifolds with lower bounded curvature. Optimization **68**(4), 713–729 (2019)
38. Fletcher, P.T.: Geodesic regression and the theory of least squares on Riemannian manifolds. Int. J. Comput. Vis. **105**(2), 171–185 (2013)
39. Freifeld, O., Black, M.J.: Lie bodies: a manifold representation of 3D human shape. In: Proceedings of European Conference on Computer Vision 2012. Springer, Berlin (2012)
40. Gabay, D.: Minimizing a differentiable function over a differential manifold. J. Optim. Theory Appl. **37**(2), 177–219 (1982)
41. Goffin, J.L.: Subgradient optimization in nonsmooth optimization (including the Soviet revolution). Doc. Math. (Extra vol.: Optimization stories), 277–290 (2012)
42. Grohs, P., Hosseini, S.: ε-Subgradient algorithms for locally lipschitz functions on Riemannian manifolds. Adv. Comput. Math. **42**(2), 333–360 (2016)
43. Hawe, S., Kleinsteuber, M., Diepold, K.: Analysis operator learning and its application to image reconstruction. IEEE Trans. Image Process. **22**(6), 2138–2150 (2013)
44. Honkela, A., Raiko, T., Kuusela, M., Tornio, M., Karhunen, J.: Approximate Riemannian conjugate gradient learning for fixed-form variational Bayes. J. Mach. Learn. Res. **11**, 3235–3268 (2010)
45. Huang, W., Gallivan, K.A., Absil, P.A.: A Broyden class of quasi-Newton methods for Riemannian optimization. SIAM J. Optim. **25**(3), 1660–1685 (2015)
46. Jeuris, B., Vandebril, R., Vandereycken, B.: A survey and comparison of contemporary algorithms for computing the matrix geometric mean. Electron. Trans. Numer. Anal. **39**, 379–402 (2012)
47. Kajimura, T., Kimura, Y.: Resolvents of convex functions in complete geodesic metric spaces with negative curvature. J. Fixed Point Theory Appl. **21**(1), 15 (2019). Art. 32
48. Karmarkar, N.: Riemannian geometry underlying interior-point methods for linear programming. In: Mathematical Developments Arising from Linear Programming (Brunswick, ME, 1988), Contemporary Mathematics, vol. 114, pp. 51–75. American Mathematical Society, Providence (1990)

49. Kum, S., Yun, S.: Incremental gradient method for Karcher mean on symmetric cones. J. Optim. Theory Appl. **172**(1), 141–155 (2017)
50. Lang, S.: Fundamentals of Differential Geometry, Graduate Texts in Mathematics, vol. 191. Springer, New York (1999)
51. Lerkchaiyaphum, K., Phuengrattana, W.: Iterative approaches to solving convex minimization problems and fixed point problems in complete CAT(0) spaces. Numer. Algorithms **77**(3), 727–740 (2018)
52. Leuştean, L., Nicolae, A., Sipoş, A.: An abstract proximal point algorithm. J. Global Optim. **72**(3), 553–577 (2018)
53. Li, C., Yao, J.C.: Variational inequalities for set-valued vector fields on Riemannian manifolds: convexity of the solution set and the proximal point algorithm. SIAM J. Control Optim. **50**(4), 2486–2514 (2012)
54. Li, C., López, G., Martín-Márquez, V.: Monotone vector fields and the proximal point algorithm on Hadamard manifolds. J. Lond. Math. Soc. (2) **79**(3), 663–683 (2009)
55. Li, C., Mordukhovich, B.S., Wang, J., Yao, J.C.: Weak sharp minima on Riemannian manifolds. SIAM J. Optim. **21**(4), 1523–1560 (2011)
56. Luenberger, D.G.: The gradient projection method along geodesics. Management Sci. **18**, 620–631 (1972)
57. Manton, J.H.: A framework for generalising the Newton method and other iterative methods from Euclidean space to manifolds. Numer. Math. **129**(1), 91–125 (2015)
58. Martinet, B.: Régularisation d'inéquations variationnelles par approximations successives. Rev. Française Informat. Recherche Opérationnelle **4**(Ser. R-3), 154–158 (1970)
59. Miller, S.A., Malick, J.: Newton methods for nonsmooth convex minimization: connections among U-Lagrangian, Riemannian Newton and SQP methods. Math. Program. **104**(2–3, Ser. B), 609–633 (2005)
60. Nesterov, Y.E., Todd, M.J.: On the Riemannian geometry defined by self-concordant barriers and interior-point methods. Found. Comput. Math. **2**(4), 333–361 (2002)
61. Pakkaranang, N., Kumam, P., Cho, Y.J.: Proximal point algorithms for solving convex minimization problem and common fixed points problem of asymptotically quasi-nonexpansive mappings in CAT(0) spaces with convergence analysis. Numer. Algorithms **78**(3), 827–845 (2018)
62. Papa Quiroz, E.A., Oliveira, P.R.: Proximal point methods for quasiconvex and convex functions with Bregman distances on Hadamard manifolds. J. Convex Anal. **16**(1), 49–69 (2009)
63. Papa Quiroz, E.A., Quispe, E.M., Oliveira, P.R.: Steepest descent method with a generalized Armijo search for quasiconvex functions on Riemannian manifolds. J. Math. Anal. Appl. **341**(1), 467–477 (2008)
64. Park, F.C., Bobrow, J.E., Ploen, S.R.: A lie group formulation of robot dynamics. Int. J. Rob. Res. **14**(6), 609–618 (1995)
65. Phuengrattana, W., Onjai-uea, N., Cholamjiak, P.: Modified proximal point algorithms for solving constrained minimization and fixed point problems in complete CAT(0) spaces. Mediterr. J. Math. **15**(3), 20 (2018). Art. 97
66. Poljak, B.T.: Subgradient methods: a survey of Soviet research. In: Nonsmooth Optimization (Proceedings of the IIASA Workshop, Laxenburg, 1977), IIASA Proc. Ser., vol. 3, pp. 5–29. Pergamon, Oxford (1978)
67. Rapcsák, T.: Smooth nonlinear optimization in $\mathbb{R}^n$. In: Nonconvex Optimization and Its Applications, vol. 19. Kluwer Academic Publishers, Dordrecht (1997)
68. Rockafellar, R.T.: Monotone operators and the proximal point algorithm. SIAM J. Control. Optim. **14**(5), 877–898 (1976)
69. Rothaus, O.S.: Domains of positivity. Abh. Math. Sem. Univ. Hamburg **24**, 189–235 (1960)
70. Said, S., Bombrun, L., Berthoumieu, Y., Manton, J.H.: Riemannian Gaussian distributions on the space of symmetric positive definite matrices. IEEE Trans. Inform. Theory **63**(4), 2153–2170 (2017)

71. Sakai, T.: Riemannian geometry. In: Translations of Mathematical Monographs, vol. 149. American Mathematical Society, Providence (1996). Translated from the 1992 Japanese original by the author
72. Sato, H.: A Dai-Yuan-type Riemannian conjugate gradient method with the weak Wolfe conditions. Comput. Optim. Appl. **64**(1), 101–118 (2016)
73. Smith, S.T.: Optimization techniques on Riemannian manifolds. In: Hamiltonian and Gradient Flows, Algorithms and Control, Fields Institute Communications, vol. 3, pp. 113–136. American Mathematical Society, Providence (1994)
74. Souza, J.C.O., Oliveira, P.R.: A proximal point algorithm for DC fuctions on Hadamard manifolds. J. Global Optim. **63**(4), 797–810 (2015)
75. Sra, S., Hosseini, R.: Conic geometric optimization on the manifold of positive definite matrices. SIAM J. Optim. **25**(1), 713–739 (2015)
76. Udrişte, C.: Convex functions and optimization methods on Riemannian manifolds. In: Mathematics and Its Applications, vol. 297. Kluwer Academic Publishers, Dordrecht (1994)
77. Ugwunnadi, G.C., Khan, A.R., Abbas, M.: A hybrid proximal point algorithm for finding minimizers and fixed points in CAT(0) spaces. J. Fixed Point Theory Appl. **20**(2), 19 (2018). Art. 82
78. Wang, J.H.: Convergence of Newton's method for sections on Riemannian manifolds. J. Optim. Theory Appl. **148**(1), 125–145 (2011)
79. Wang, X.M.: Subgradient algorithms on riemannian manifolds of lower bounded curvatures. Optimization **67**(1), 179–194 (2018)
80. Wang, J.H., López, G., Martín-Márquez, V., Li, C.: Monotone and accretive vector fields on Riemannian manifolds. J. Optim. Theory Appl. **146**(3), 691–708 (2010)
81. Wang, X., Li, C., Wang, J., Yao, J.C.: Linear convergence of subgradient algorithm for convex feasibility on Riemannian manifolds. SIAM J. Optim. **25**(4), 2334–2358 (2015)
82. Wang, J., Li, C., Lopez, G., Yao, J.C.: Proximal point algorithms on Hadamard manifolds: linear convergence and finite termination. SIAM J. Optim. **26**(4), 2696–2729 (2016)
83. Wang, X., López, G., Li, C., Yao, J.C.: Equilibrium problems on Riemannian manifolds with applications. J. Math. Anal. Appl. **473**(2), 866–891 (2019)
84. Weber, M., Sra, S.: Riemannian frank-wolfe with application to the geometric mean of positive definite matrices. ArXiv e-prints, pp. 1–21 (2018)
85. Weinmann, A., Demaret, L., Storath, M.: Total variation regularization for manifold-valued data. SIAM J. Imaging Sci. **7**(4), 2226–2257 (2014).
86. Weinmann, A., Demaret, L., Storath, M.: Mumford-Shah and Potts regularization for manifold-valued data. J. Math. Imaging Vision **55**(3), 428–445 (2016).
87. Wen, Z., Yin, W.: A feasible method for optimization with orthogonality constraints. Math. Program. **142**(1–2, Ser. A), 397–434 (2013)
88. Wilson, B., Leimeister, M.: Gradient descent in hyperbolic space, pp. 1–10 (2018). arXiv e-prints
89. Yao, T.T., Bai, Z.J., Zhao, Z.: A Riemannian variant of the Fletcher-Reeves conjugate gradient method for stochastic inverse eigenvalue problems with partial eigendata. Numer. Linear Algebra Appl. **26**(2), e2221, 19 (2019)
90. Zhang, T.: A majorization-minimization algorithm for computing the Karcher mean of positive definite matrices. SIAM J. Matrix Anal. Appl. **38**(2), 387–400 (2017)
91. Zhang, P., Bao, G.: An incremental subgradient method on Riemannian manifolds. J. Optim. Theory Appl. **176**(3), 711–727 (2018)
92. Zhang, H., Sra, S.: First-order methods for geodesically convex optimization. JMLR Workshop Conf. Proc. **49**(1), 1–21 (2016)
93. Zhang, H., Reddi, S.J., Sra, S.: Riemannian SVRG: fast stochastic optimization on Riemannian manifolds. ArXiv e-prints, pp. 1–17 (2016)
94. Zhu, X.: A Riemannian conjugate gradient method for optimization on the Stiefel manifold. Comput. Optim. Appl. **67**(1), 73–110 (2017)

Chapter 19
Recent Advances in Stochastic Riemannian Optimization

Reshad Hosseini and Suvrit Sra

Contents

Abstract Stochastic and finite-sum optimization problems are central to machine learning. Numerous specializations of these problems involve nonlinear constraints where the parameters of interest lie on a manifold. Consequently, stochastic manifold optimization algorithms have recently witnessed rapid growth, also in part due to their computational performance. This chapter outlines numerous stochastic optimization algorithms on manifolds, ranging from the basic stochastic gradient method to more advanced variance reduced stochastic methods. In particular, we present a unified summary of convergence results. Finally, we also provide several basic examples of these methods to machine learning problems, including learning parameters of Gaussians mixtures, principal component analysis, and Wasserstein barycenters.

R. Hosseini (✉)
School of ECE, College of Engineering, University of Tehran, Tehran, Iran

School of Computer Science, Institute of Research in Fundamental Sciences (IPM), Tehran, Iran
e-mail: reshad.hosseini@ut.ac.ir

S. Sra
Massachusetts Institute of Technology, Cambridge, MA, USA
e-mail: suvrit@mit.edu

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_19

19.1 Introduction

In this chapter we outline first-order optimization algorithms used for minimizing the expected loss (risk) and its special case, finite-sum optimization (empirical risk). In particular, we focus on the setting where the parameters to be optimized lie on a Riemannian manifold. This setting appears in a variety of problems in machine learning and statistics, including principal components analysis [33], low-rank matrix completion [9, 39], fitting statistical models like Gaussian mixture models [17, 18, 38], Karcher mean computation [22, 33], Wasserstein barycenters [40], dictionary learning [12], low rank multivariate regression [27], subspace learning [28], and structure prediction [34]; see also the textbook [1].

Typical Riemannian manifolds used in applications can be expressed by a set of constraints on Euclidean manifolds. Therefore, one can view a Riemannian optimization problem as a nonlinearly constrained one, for which one could use classical approaches. For instance, if the manifold constitutes a convex set in Euclidean space, one can use gradient projection like methods,[1] or other nonlinear optimization methods [6]. These methods could suffer from high computational costs, or as a more fundamental weakness, they may fail to satisfy the constraints exactly at each iteration of the associated algorithm. Another problem is that the Euclidean gradient does not take into account the geometry of the problem, and even if the projection can be done and the constraints can be satisfied at each iteration, the numerical conditioning may be much worse than a method that respects geometry [1, 42].

Riemannian optimization has shown great success in solving many practical problems because it respects the geometry of the constraint set. The definition of the inner product in Riemannian geometry makes the direction of the gradient to be more meaningful than Euclidean gradients because it considers the geometry imposed by constraints on the parameters of optimization. By defining suitable retractions (geodesic like curves on manifolds), the constraint is always satisfied. Sometimes the inner product is defined to also take into account the curvature information of the cost function. The natural gradient is an important example of the Riemannian gradient shown to be successful for solving many statistical fitting problems [2]. The natural gradient was designed for fitting statistical models and it is a Riemannian gradient on a manifold where the metric is defined by the Fisher information matrix.

[1] Some care must be applied here, because we are dealing with open sets, and thus projection is not well-defined.

Additional Background and Summary

Another key feature of Riemannian optimization is the generalization of the widely important concept of convexity to geodesic convexity. We will see later in this chapter that geodesic convexity help us derive convergence results for accelerated gradient descent methods akin to their famous Euclidean counterpart: Nesterov's accelerated gradient method. Similar to the Euclidean case, there are works that develop results for recognizing geodesic convexity of functions for some special manifolds [37]. Reformulating problems keeping an eye on geodesic convexity also yields powerful optimization algorithms for some practical problems [18].

After summarizing key concepts of Riemannian manifolds, we first sketch the Riemannian analogue of the widely used (Euclidean) stochastic gradient descent method. Though some forms of stochastic gradient descent (SGD) such as natural gradient were developed decades ago, the version of SGD studied here and its analysis has a relatively short history; Bonnabel [8] was the first to give a unifying framework for analyzing Riemannian SGD and provided an asymptotic analysis on its almost sure convergence. We recall his results after explaining SGD on manifolds. We then note how convergence results of [15] for Euclidean non-convex SGD generalize to the Riemannian case under similar conditions [18]. Among recent progress on SGD, a notable direction is that of faster optimization by performing variance reduction of stochastic gradients. We will later outline recent results of accelerating SGD on manifolds and give convergence analysis for geodesically non-convex and convex cases. Finally, we close by summarizing some applications drawn from machine learning that benefit from the stochastic Riemannian algorithms studied herein.

Apart from the algorithms given in this chapter, there exist several other methods that generalize well from the Euclidean to the Riemannian setting. For example in [4] the SAGA algorithm [13] is generalized to Riemannian manifolds along with convergence theory assuming geodesic convexity. In [23] a Riemannian stochastic quasi-Newton method is studied; in [21] an inexact Riemannian trust-region method is developed and applied to finite-sum problems. Adaptive stochastic gradient methods such as ADAM and RMSProp have also been generalized [5, 24, 25]. It was observed however that ADAM works inferior to plain SGD for fitting Gaussian mixture models [16], where momentum and Nesterov SGD offered the best variants that improve on the performance of plain SGD.

The convergence results presented in this chapter are for general Riemannian manifolds and hold for a fairly general class of cost functions. For specific manifolds and functions, one can obtain better convergence results for the algorithms. For example for the case of quadratic optimization with orthogonality constraint, the authors in [26] proved convergence results. The authors in [41] proved convergence for a block eigensolver.

19.2 Key Definitions

We omit standard definitions such as Riemannian manifolds, geodesics, etc.; and defer to a standard textbook such as [20]. Readers familiar with concepts from Riemannian geometry can skip this section and directly move onto Sect. 19.3; however, a quick glance will be useful for getting familiar with our notation.

A retraction is a smooth mapping Ret from the tangent bundle $T\mathcal{M}$ to the manifold $\mathcal{M}$. The restriction of a retraction to $T_x\mathcal{M}$, $\mathrm{Ret}_x : T_x\mathcal{M} \to \mathcal{M}$, is a smooth mapping that satisfies the following:

1. $\mathrm{Ret}_x(0) = x$, where 0 denotes the zero element of $T_x\mathcal{M}$.
2. $D\,\mathrm{Ret}_x(0) = \mathrm{id}_{T_x\mathcal{M}}$, where $D\,\mathrm{Ret}_x$ denotes the derivative of Ret_x and $\mathrm{id}_{T_x\mathcal{M}}$ denotes the identity mapping on $T_x\mathcal{M}$.

One possible candidate for retraction on Riemannian manifolds is the exponential map. The exponential map $\mathrm{Exp}_x : T_x\mathcal{M} \to \mathcal{M}$ is defined as $\mathrm{Exp}_x\, v = \gamma(1)$, where γ is the geodesic satisfying the conditions $\gamma(0) = x$ and $\dot{\gamma}(0) = v$.

A vector transport $\mathcal{T} : \mathcal{M}\times\mathcal{M}\times T\mathcal{M} \to T\mathcal{M}$, $(x, y, \xi) \mapsto \mathcal{T}_{x,y}(\xi)$ is a mapping that satisfies the following properties:

1. There exists an associated retraction Ret and a tangent vector ν satisfying $\mathcal{T}_{x,y}(\xi) \in T_{\mathrm{Ret}_x(\xi)}$, for all $\xi \in T_x\mathcal{M}$.
2. $\mathcal{T}_{x,x}\xi = \xi$, for all $\xi \in T_x\mathcal{M}$.
3. The mapping $\mathcal{T}_{x,y}(\cdot)$ is linear.

We use $\mathcal{T}_{x,y}^{\mathrm{Ret}_x}$ to denote the vector transport constructed by the differential of the retraction, i.e., $\mathcal{T}_{x,y}^{\mathrm{Ret}_x}(\xi) = D\,\mathrm{Ret}_x(\eta)[\xi]$, wherein $\mathrm{Ret}_x(\eta) = y$ (in the case of multiple η, we make it clear by writing the value of η), while $\mathcal{P}_{x,y}^{\mathrm{Ret}_x}$ denotes the parallel transport along the retraction curve (again, if there are multiple curves where $\mathrm{Ret}_x(\eta) = y$, we make it clear from context which curve is meant).

The gradient on a Riemannian manifold is defined as the vector $\nabla f(x)$ in tangent space such that

$$Df(x)\xi = \langle \nabla f(x), \xi \rangle, \quad \text{for } \xi \in T_x\mathcal{M},$$

where $\langle \cdot, \cdot \rangle$ is the inner product in the tangent space $T_x\mathcal{M}$. $Df(x)\xi$ is the directional derivative of f along ξ. Let $\gamma : [-1, 1] \to \mathcal{M}$ be a differentiable curve with $\gamma(0) = x$ and $\dot{\gamma}(0) = \xi$ (for example $\gamma(t) = \mathrm{Exp}(t\xi)$), then the directional derivative can be defined by

$$Df(x)\xi = \frac{d}{d\tau} f(\gamma(\tau))\bigg|_{\tau=0}.$$

Differentials at each point on the manifold forms the cotangent space. The cotangent space on the smooth manifold $\mathcal{M}$ at point x is defined as the dual space of the tangent space. Elements of the cotangent space are linear functionals on the tangent space.

The Hessian of a function is a symmetric bilinear form $D^2 f(x) : T_x\mathcal{M}\times T_x\mathcal{M} \to \mathbb{R}, (\xi, \eta) \to \langle \nabla_\eta \nabla f(x), \xi\rangle$, where ∇_η is the covariant derivative with respect to η [1]. The Hessian as a operator $\nabla^2 f(x) : T_x\mathcal{M} \to T_x\mathcal{M}$ is a linear operator that maps v in $T_x\mathcal{M}$ onto the Riesz representation $D^2 f(x)(v, .)$. Alternatively, the operator Hessian can be defined by

$$\frac{d}{d\tau}\langle \nabla f(\gamma(\tau)), \nabla f(\gamma(\tau))\rangle\bigg|_{\tau=0} = 2\langle \nabla f(x), (\nabla^2 f)\xi\rangle,$$

where $\gamma : [-1, 1] \to \mathcal{M}$ is a differentiable curve with $\gamma(0) = x$ and $\dot\gamma(0) = \xi$.In the following, we give some conditions and definitions needed for the complexity analysis of the algorithms in this book chapter.

Definition 19.1 ($\boldsymbol{\rho}$-Totally Retractive Neighborhood) A neighborhood Ω of a point x is called *ρ-totally retractive* if for all $y \in \Omega$, $\Omega \subset \mathbb{B}(0_y, \rho)$ and $\mathrm{Ret}_y(\cdot)$ is a diffeomorphism on $\mathbb{B}(0_y, \rho)$.

All optimization algorithms given in this book chapter start from an initial point and the point is updated based on a retraction along a direction with a certain step size. The following condition guarantees that all points along retraction in all interactions stay in a set.

Definition 19.2 (Iterations Stay Continuously in $\mathcal{X}$) The iterate $x_{k+1} = \mathrm{Ret}_{x_k}(\alpha_k\xi_k)$ is said to stay continuously in $\mathcal{X}$ if $\mathrm{Ret}_{x_k}(t\xi_k) \in \mathcal{X}$ for all $t \in [0, \alpha_k]$.

Most of the optimization algorithms explained in this chapter need a vector transport. The convergence analysis for many of them is available for the specific case of parallel transport. Some works that go beyond parallel transport still need some extra conditions on the vector transport as explained below. These conditions hold *a forteriori* for parallel transport.

Definition 19.3 (Isometric Vector Transport) The vector transport $\mathcal{T}$ is said to be *isometric* on $\mathcal{M}$ if for any $x, y \in \mathcal{M}$ and $\eta, \xi \in T_x\mathcal{M}$, $\langle \mathcal{T}_{x,y}(\eta), \mathcal{T}_{x,y}(\xi)\rangle = \langle \eta, \xi\rangle$.

Definition 19.4 ($\boldsymbol{\theta}$-Bounded Vector Transport) The vector transport $\mathcal{T}$ with its associated retraction Ret is said to be *θ-bounded* on $\mathcal{M}$ if for any $x, y = \mathrm{Ret}_x(\xi) \in \mathcal{M}$ and $\xi \in T_x\mathcal{M}$,

$$\|\mathcal{T}_{x,y}\eta - \mathcal{P}^{\mathrm{Ret}_x}_{x,y}\eta\| \le \theta\|\xi\|\|\eta\|, \tag{19.1}$$

where $\mathcal{P}$ is the parallel transport along this associated retraction curve.

Definition 19.5 ($\boldsymbol{\theta}$-Bounded Inverse Vector Transport) The inverse vector transport with its associated retraction Ret is said to be *θ-bounded* on $\mathcal{M}$ if for any $x, y = \mathrm{Ret}_x(\xi) \in \mathcal{M}$ and $\xi \in T_x\mathcal{M}$,

$$\|\left(\mathcal{T}_{x,y}\right)^{-1}\chi - \left(\mathcal{P}^{\mathrm{Ret}_x}_{x,y}\right)^{-1}\chi\| \le \theta\|\chi\|\|\xi\|,$$

where $\mathcal{P}$ is the parallel transport along this associated retraction curve.

The following proposition helps in checking if a vector transport satisfies some of the conditions expressed above.

Proposition 19.6 (Lemma 3.5 in Huang et al. [19]) *Assume that there exists a constant* $c_0 > 0$ *such that* $\mathcal{T}$ *satisfies* $\|\mathcal{T}_{x,y} - \mathcal{T}_{x,y}^{\mathrm{Ret}_x}\| \leq c_0\|\xi\|$, $\|\left(\mathcal{T}_{x,y}\right)^{-1} - \left(\mathcal{T}_{x,y}^{\mathrm{Ret}_x}\right)^{-1}\| \leq c_0\|\xi\|$, *for any* $x, y \in \mathcal{M}$ *and the retraction* $y = \mathrm{Ret}_x(\xi)$. *Then, the vector transport and its inverse are* θ*-bounded on* $\mathcal{M}$, *for a constant* $\theta > 0$.

We note that if the vector transport is C^0, then the condition of this proposition holds.

For the convergence analysis of the algorithms in this chapter, the cost function needs to satisfy some of the properties given below.

Definition 19.7 (G-Bounded Gradient) A function $f : \mathcal{X} \to \mathbb{R}$ is said to have a *G-bounded gradient* in $\mathcal{X}$ if $\|\nabla f(x)\| \leq G$, for all $x \in \mathcal{X}$.

Definition 19.8 (H-Bounded Hessian) A function $f : \mathcal{X} \to \mathbb{R}$ is said to have an *H-bounded Hessian* in $\mathcal{X}$ if $\|\nabla^2 f(x)\| \leq H$, for all $x \in \mathcal{X}$.

Definition 19.9 (Retraction L-Smooth) A function $f : \mathcal{X} \to \mathbb{R}$ is said to be *retraction L-smooth* if for any $x, y = \mathrm{Ret}_x(\xi)$ in $\mathcal{X}$, we have

$$f(y) \leq f(x) + \langle \nabla f(x), \xi \rangle + \frac{L}{2}\|\xi\|^2.$$

If the retraction is the exponential map, then the function is called **geodesically L-smooth**.

Definition 19.10 (Retraction L-Upper-Hessian Bounded) A function $f : \mathcal{X} \to \mathbb{R}$ is said to be upper-Hessian bounded in a subset $\mathcal{U} \subset \mathcal{X}$ if $\mathrm{Ret}_x(t\xi)$ stays in $\mathcal{X}$ for all $x, y = \mathrm{Ret}_x(\xi)$ in $\mathcal{U}$ and $t \in [0, 1]$, and there exists a constant $L > 0$ such that $\frac{d^2 f(\mathrm{Ret}_x(t\xi))}{dt^2} \leq L$.

Definition 19.11 (Retraction μ-Lower-Hessian Bounded) A function $f : \mathcal{X} \to \mathbb{R}$ is said to be lower-Hessian bounded in a subset $\mathcal{U} \subset \mathcal{X}$ if $\mathrm{Ret}_x(t\xi)$ stays in $\mathcal{X}$ for all $x, y = \mathrm{Ret}_x(\xi)$ in $\mathcal{U}$ and $t \in [0, 1]$, and there exists a constant $\mu > 0$ such that $\frac{d^2 f(\mathrm{Ret}_x(t\xi))}{dt^2} \geq \mu$.

Definition 19.12 (Retraction L_l-Lipschitz) A function $f : \mathcal{X} \to \mathbb{R}$ is said to be retraction L_l-Lipschitz in $\mathcal{X}$, if there exists $L_l > 0$ such that for all $x, y \in \mathcal{X}$,

$$\|\mathcal{P}_{x,y}^{\mathrm{Ret}_x} \nabla f(x) - \nabla f(y)\| \leq L_l\|\xi\|, \tag{19.2}$$

where $\mathcal{P}$ is the parallel transport along this associated retraction curve $y = \mathrm{Ret}_x(\xi)$.

If the retraction is the exponential map, then this condition is called **geodesically L_l-Lipschitz**. A function that is geodesically L_l-Lipschitz is also geodesically L-smooth with $L = L_l$ [44].

In the following, we give two propositions and a theorem for checking if a function satisfies some of the conditions explained before. The following proposition is based on a Lemma in [22].

Proposition 19.13 *Suppose that the function $f : \mathcal{X} \to \mathbb{R}$ is retraction L-upper-Hessian bounded in $\mathcal{U} \subset \mathcal{X}$. Then, the function is also retraction L-smooth in $\mathcal{U}$.*

Proposition 19.14 (Lemma 3.8 in Kasai et al. [22]) *Let* Ret *be a retraction on $\mathcal{M}$ and the vector transport associated with the retraction and its inverse be θ-bounded. Assume a function is twice continuously differentiable with H-bounded Hessian. Then the function is retraction L_l-Lipschitz with $L_l = H(1 + \Xi\theta)$ with Ξ being an upper bound for $\|\xi\|$ in (19.2).*

For showing retraction L-smoothness, we can use the following theorem.

Theorem 19.15 (Lemma 2.7 in Boumal et al. [11]) *Let $\mathcal{M}$ be a compact Riemannian submanifold of a Euclidean space. Let* Ret *be a retraction on $\mathcal{M}$. If a function has a Euclidean Lipschitz continuous gradient in the convex hull of $\mathcal{M}$, then the function is retraction L-smooth for some constant L for any retraction.*

The aforementioned conditions of function are quite general. In the following we give some conditions on functions that help to develop stronger convergence results.

Definition 19.16 (g-Convex) A set $\mathcal{X}$ is geodesically convex (g-convex) if for any $x, y \in \mathcal{X}$, there is a geodesic γ with $\gamma(0) = x$, $\gamma(1) = y$ and $\gamma(t) \in \mathcal{X}$ for $t \in [0, 1]$. A function $f : \mathcal{X} \to R$ is called geodesically convex in this set if

$$f(\gamma(t)) \leq (1 - t) f(x) + t f(y).$$

Definition 19.17 (μ-Strongly g-Convex) A function $f : \mathcal{X} \to R$ is called geodesically μ-strongly convex if for any $x, y = \mathrm{Exp}_x(\xi) \in \mathcal{X}$ and g_x subgradient of f at x (gradient if f is smooth), it holds

$$f(y) \geq f(x) + \langle g_x, \xi \rangle + \frac{\mu}{2} \|\xi\|^2.$$

Definition 19.18 (τ-Gradient Dominated) A function $f : \mathcal{X} \to R$ is called τ-gradient dominated if x^* is a global minimizer of f and for every $x \in \mathcal{X}$ we have

$$f(x) - f(x^*) \leq \tau \|\nabla f(x)\|^2. \tag{19.3}$$

The following proposition shows that strongly convex functions are also gradient dominated. Therefore, the convergence analysis developed for gradient dominated functions also holds for strongly convex functions [44].

Algorithm 1 Riemannian SGD

```
Given: Smooth manifold $\mathcal{M}$ with retraction Ret; initial value $x_0$; a differentiable cost function $f$; number of iterations $T$.
for $t = 0, 1, \dots, T-1$ do
    Obtain the direction $\xi_t = \nabla f_i(x_t)$, where $\nabla f_i(x_t)$ is the noisy version of the cost gradient
    Use a step-size rule to choose the step-size $\alpha_t$
    Calculate $x_{t+1} = \mathrm{Ret}_{x_t}(-\alpha_t \xi_t)$
end for
return $x_T$
```

Proposition 19.19 *τ-gradient domination is implied by $\frac{1}{2\tau}$-strong convexity as in Euclidean case.*

19.3 Stochastic Gradient Descent on Manifolds

In the most general form, consider the following constrained optimization problem:

$$\min_{x \in \mathcal{M}} f(x). \tag{19.4}$$

We assume $\mathcal{M}$ is a Riemannian manifold and that at each step of SGD we obtain a noisy version of the Riemannian gradient. Riemannian SGD uses the following simple update rule:

$$x_{t+1} = \mathrm{Ret}_{x_t}\big(-\eta_t \nabla f_{i_t}(x_t)\big), \tag{19.5}$$

where ∇f_{i_t} is a noisy version of the Riemannian gradient at time step t and the noise terms at different time steps are assumed to be independent. Note that there is stochasticity in each update. Therefore, the value x_t can be seen as a sample from a distribution depending on the gradient noise until time step t. A sketch of Riemannian SGD is given in Algorithm 1. For providing convergence results for all algorithms, it is assumed the stochastic gradients in all iterations are unbiased, i.e.,

$$\mathbb{E}[\nabla f_{i_t}(x_t) - \nabla f(x_t)] = 0.$$

This unbiasedness condition is assumed in all theorems and we do not state it explicitly in the statements of the theorems.

The cost function used in many practical machine learning problems which is solved by SGD can be defined by

$$f(x) = \mathbb{E}[f(z; x)] = \int f(z; x) dP(z), \tag{19.6}$$

where x denotes the parameters, dP is a probability measure and $f(z; x)$ is the risk function. For this cost function $f_{i_t} = f(z_{i_t}, x_t)$ is the risk evaluated at the current sample z_{i_t} from the probability law dP. Apparently, the stochastic gradients for this cost function satisfy the condition that stochastic gradients are unbiased. A special case of the above-mentioned cost function is the following finite-sum problem:

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(z_i, x). \tag{19.7}$$

If we assume $z_1, \ldots, z_n$ to be our data, then the empirical distribution over the data $P(Z = z_i) = \frac{1}{n}$ gives rise to the above noted cost function. Therefore, the theoretical analysis for SGD works both for online-algorithms and also finite-sum optimization problems. To further elucidate this consider the following example.

Example: Maximum Likelihood Parameter Estimation

Consider we want to estimate the parameters of a model distribution given by the density $q(z; x)$, where x denotes the parameters. In the online learning framework, we observe a sample z_t from the underlying distribution $p(z)$ at each time step. Observing this new sample, the parameter set is updated by a rule. The update rule should be designed such that in the limit of observing enough samples, the parameters converge to the optimal parameters. The optimal parameters are commonly defined as the parameters that minimize the Kullback-Leibler divergence between the estimated and the true distributions. The following cost function minimizes this divergence:

$$f(x) = \mathbb{E}[-\log q(z; x)] = -\int \log q(z; x)p(z)dz,$$

where q is the density of model distribution and p is the true density. Apparently, this cost function is in the form of cost function defined in (19.6). One of the common update rules for online learning is SGD. For Riemannian SGD, we have $\nabla f(z_t, x_t) = \nabla f_{i_t}(x_t)$ and we use the update rule as in (19.5).

In the finite sample case, consider $z_1, \ldots, z_n$ to be i.i.d. samples from the underlying density $q(z; x)$. A common approach for estimating the parameters is the maximum-likelihood estimate where we are minimizing the following cost function:

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} -\log q(z_i; x).$$

The cost function is a finite-sum cost that can be minimized using SGD. Therefore, it is important to know the conditions under which SGD guarantees convergence.

The following theorem gives the convergence to stationary points of the cost function.

Theorem 19.20 (Theorem 2 in Bonnabel [8]) *Consider the optimization problem in* (19.4)*, where the cost function is the expected risk* (19.6)*. Assume*

- *The manifold $\mathcal{M}$ is a connected Riemannian manifold with injectivity radius uniformly bounded from below by $I > 0$.*
- *The steps stay within a compact set.*
- *The gradients of the f_is are G-bounded.*

Let the step-sizes in Algorithm 1 satisfy the following standard condition

$$\sum \alpha_t^2 < \infty \textit{ and } \sum \alpha_t = \infty, \tag{19.8}$$

Then $f(x_t)$ converges a.s. and $\nabla f(x_t) \to 0$ a.s.

Staying within a compact set of the previous theorem is a strong requirement. Under milder conditions, [18] were able to prove the rate of convergence.

Theorem 19.21 (Theorem 5 in Hosseini and Sra [18]) *Assume that the following conditions hold*

- *The functions f_i are retraction L-smooth.*
- *The expected square norm of the gradients of the f_is are G^2-bounded.*

Then for the following constant step-size in Algorithm 1

$$\alpha_t = \frac{c}{\sqrt{T}},$$

we have

$$\min_{0 \le t \le T-1} \mathbb{E}[\|\nabla f(x_t)\|^2] \le \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[\|\nabla f(x_t)\|^2] \le \frac{1}{\sqrt{T}} \left(\frac{f(x_0) - f(x^*)}{c} + \frac{Lc}{2} G^2 \right). \tag{19.9}$$

where $f(x_0)$ is the function value at the initial point and $f(x^*)$ is the minimum function value.

The following theorem shows that it is possible to get a convergence rate without needing bounded gradients with a randomized rule. For this theorem, the stochastic gradients needs to have σ-bounded variance, i.e.,

$$\mathbb{E}[\|\nabla f_{i_t}(x_t) - \nabla f(x_t)\|^2] \le \sigma^2, \qquad 0 \le \sigma < \infty.$$

The conditions and the resulting rate are similar to that of Euclidean case [15], and no further assumptions are necessary.

Theorem 19.22 (Theorem 4 in Hosseini and Sra [18]) *Assume that the following conditions hold.*

- *The functions f_i are retraction L-smooth.*
- *The functions f_i have σ-bounded variance.*

Assume a slightly modified version of SGD that outputs a point x_a by randomly picking one of the iterates, say x_t, with probability $p_t := (2\alpha_t - L\alpha_t^2)/Z_T$, where $Z_T = \sum_{t=1}^{T}(2\alpha_t - L\alpha_t^2)$. Furthermore, choose $\alpha_t = \min\{L^{-1}, c\sigma^{-1}T^{-1/2}\}$ in Algorithm 1 for a suitable constant c. Then, we obtain the following bound on $\mathbb{E}[\|\nabla f(x_a)\|^2]$, which measures the expected gap to stationarity:

$$\mathbb{E}[\|\nabla f(x_a)\|^2] \leq \frac{2L\Delta_1}{T} + \left(c + c^{-1}\Delta_1\right)\frac{L\sigma}{\sqrt{T}} = O\left(\frac{1}{T}\right) + O\left(\frac{1}{\sqrt{T}}\right). \tag{19.10}$$

For Hadamard manifolds (complete, simply-connected Riemannian manifolds with nonpositive sectional curvature), one can prove a.s. convergence under milder conditions. Hadamard manifolds have strong properties, for example the exponential map at any point is globally invertible. Concerning convergence for Hadamard manifolds there is the following result requiring milder assumptions.

Theorem 19.23 (Theorem 3 in Bonnabel [8]) *Consider the optimization problem in (19.4), where the cost function is the expected risk (19.6). Assume*

- *The exponential map is used for the retraction.*
- *The manifold $\mathcal{M}$ is a Hadamard manifold.*
- *There is a lower bound on the sectional curvature denoted by $\kappa < 0$.*
- *There is a point $y \in \mathcal{M}$ such that the negative gradient points towards y when $d(x, y)$ becomes larger than $s > 0$, i.e.,*

$$\inf_{d(x,y)>s} \langle \mathrm{Exp}_x^{-1}(y),\ \nabla f(x)\rangle < 0$$

- *There is a continuous function $g : \mathcal{M} \to \mathbb{R}$ that satisfies*

$$g(x) \geq \max\big\{1, \mathbb{E}\big[\|\nabla f(z; x)\|^2(1 + \sqrt{\kappa}(d(x, y) + \|\nabla f(z; x)\|))\big],$$
$$\mathbb{E}\big[\big(2\|\nabla f(z; x)\|d(x, y) + \|\nabla f(z; x)\|^2\big)^2\big]\big\}$$

Then for the step size rule $\alpha_t = -\frac{\beta_t}{g(x_t)}$ in Algorithm 1, wherein β_t satisfying (19.8), $f(x_t)$ converges a.s. and $\nabla f(x_t) \to 0$ a.s.

Algorithm 2 Riemannian SVRG

```
1:  Given: Smooth manifold $\mathcal{M}$ with retraction Ret and vector transport $\mathcal{T}$; initial value $x_0$; a
    finite-sum cost function $f$; update frequency $m$; number of epochs $S$ and $K$.
2:  for $k = 0, \ldots,$ K-1 do
3:      $\tilde{x}^0 = x_k$
4:      for $s = 0, \ldots,$ S-1 do
5:          Calculate the full Riemannian gradient $\nabla f(\tilde{x}^s)$
6:          Store $x_0^{s+1} = \tilde{x}^s$
7:          for $t = 0, \ldots, m-1$ do
8:              Choose $i_t \in \{1, \ldots, n\}$ uniformly at random
9:              Calculate $\xi_t^{s+1} = \nabla f_{i_t}(x_t^{s+1}) - \mathcal{T}_{\tilde{x}^s, x_t^{s+1}}(\nabla f_{i_t}(\tilde{x}^s) - \nabla f(\tilde{x}^s))$
10:             Use a step-size rule to choose the step-size $\alpha_t^{s+1}$
11:             Calculate $x_{t+1}^{s+1} = \mathrm{Ret}_{x_t^{s+1}}(-\alpha_t^{s+1}\xi_t^{s+1})$
12:         end for
13:         Option I-a: Set $\tilde{x}^{s+1} = x_m^{s+1}$
14:         Option II-a: Set $\tilde{x}^{s+1} = x_t^{s+1}$ for randomly chosen $t \in \{0, \ldots, m-1\}$
15:     end for
16:     Option I-b: Set $x_{k+1} = \tilde{x}^S$
17:     Option II-b: Set $x_{k+1} = \tilde{x}_t^s$ for randomly chosen $s \in \{0, \ldots, S-1\}$ and $t \in \{0, \ldots, m-1\}$
18: end for
19: return $x_K$
```

19.4 Accelerating Stochastic Gradient Descent

Mainly for finite-sum problems but also for expected risk (19.6) problems, accelerated algorithms have been developed with faster convergence rates than plain SGD. In this section, we review several popular accelerated algorithms that are based on *variance reduction* ideas. Stochastic variance reduced gradient (SVRG) is a popular variance reduction technique that has a superior convergence than plain SGD. A Riemannian version of SVRG (R-SVRG) was proposed in [33] and generalized to use retractions and vector transports in [36]. Variance reduction can be seen in the line 9 of Algorithm 2, where the average gradient is used for adjust the current gradient. Consider a stochastic gradient that has high variance; then subtracting the difference between this gradient and the average gradient at a reference point from this gradient in the current point reduces the effect of high variance. Because we are on a Riemannian manifold, gradients live in different tangent spaces, and a vector transport is needed to make the subtraction meaningful as can be seen in the line 9 of Algorithm 2.

The authors in [33] were able to prove that R-SVRG has the same convergence as in the Euclidean case [32]. Though, the statement on the convergence rate needs additional assumptions and a bound depending on the sectional curvature.

Theorem 19.24 (Theorem 2 in Zhang et al. [33]) *Consider the optimization problem in* (19.4)*, where the cost function is the finite sum* (19.7)*. Consider, we run Riemannian SVRG to solve this problem with* $K = 1$*, Option I-a, Option II-b. Assume*

- *The exponential map is used for the retraction and the parallel transport is used for the vector transport.*
- *The iterations stay in a compact subset $\mathcal{X}$, and the diameter of $\mathcal{X}$ is bounded by D, that is $\max_{x,y \in \mathcal{X}} d(x, y) \leq D$.*
- *The exponential map on $\mathcal{X}$ is invertible.*
- *The sectional curvature is upper-bounded.*
- *There is a lower bound on the sectional curvature denoted by κ.*
- *The functions f_i are geodesically L-smooth.*
- *The function f attains its minimum at $x^* \in \mathcal{X}$.*

Define ζ to be a constant that captures the impact of the manifold curvature.

$$\zeta = \begin{cases} \frac{\sqrt{|\kappa|}D}{\tanh\left(\sqrt{|\kappa|}D\right)}, & \kappa < 0. \\ 1, & \kappa \geq 0. \end{cases} \tag{19.11}$$

Then there exist universal constants $\mu_0 \in (0, 1)$, $\nu > 0$ such that if we set $\alpha_t = \frac{\mu_0}{Ln^{\alpha_1}\zeta^{\alpha_2}}, \alpha_1 \in (0, 1], \alpha_2 \in (0, 2]$ and $m = \lfloor \frac{n^{3\alpha_1}}{3\mu_0 \zeta^{1-2\alpha_2}} \rfloor$ in Algorithm 2, we have

$$\mathbb{E}[\|\nabla f(x_1)\|^2] \leq \frac{Ln^{\alpha_1}\zeta^{\alpha_2}[f(x_0) - f(x^*)]}{T\nu},$$

where $T = mS$ is the number of iterations.

The abovementioned theorem was stated based on the exponential map and the parallel transport that can be expensive making SVRG impractical for some applications. In [36] the following convergence result is proved when using retractions and vector transports.

Theorem 19.25 (Theorem 4.6 in Sato et al. [36]) *Consider the optimization problem in (19.4), where the cost function is the finite sum (19.7). Consider, we run the Riemannian SVRG algorithm to solve this problem with $K = 1$, Option I-a and Option I-b. Assume*

- *The retraction is of the class C^2.*
- *The iterations stay in a compact subset $\mathcal{X}$.*
- *For each $s \geq 0$, there exists $\eta_t^{s+1} \in T_{\tilde{x}^s}\mathcal{M}$ such that $\mathrm{Ret}_{\tilde{x}^s}(\eta_t^{s+1}) = x_t^{s+1}$.*
- *There exists $I > 0$ such that, for any $x \in \mathcal{X}$, $\mathrm{Ret}_x(.)$ is defined in a ball $\mathbb{B}(0_x, I) \in T_x\mathcal{M}$, which is centered at the origin 0_x in $T_x\mathcal{M}$ with radius I.*
- *The vector transport is continuous and isometric on $\mathcal{M}$.*
- *The functions f_i are twice-differentiable.*

Assume the step-size α_t^s in Algorithm 2 is chosen by the rule (19.8). Then $f(x_t^s)$ converges a.s. and $\nabla f(x_t^s) \to 0$ a.s.

Note that existence of η_t^{s+1} is guaranteed if Ret_x has ρ-totally retractive neighborhood for all $x \in \mathcal{X}$. For the special case of the exponential map and the parallel transport, many of the conditions of the aforementioned theorem are automatically satisfied or simplified: The parallel transport is an isometry, the exponential map is of class C^2, and the third and fourth conditions can be satisfied by having a connected manifold with the injectivity radius uniformly bounded from below by $I > 0$.

Stochastic recursive gradient (SRG) [29] is another variance reduction algorithm similar to SVRG proposed. It was recently shown that the algorithm achieves the optimal bound for the class of variance reduction methods that only assume the Lipschitz continuous gradients [30]. Recently, the Riemannian counterpart of this algorithm (R-SRG) shown in Algorithm 3 has also been developed [22]. The following theorem gives a convergence result with the minimalistic conditions needed for the proof.

Theorem 19.26 (Theorem 4.5 in Kasai et al. [22]) *Consider the optimization problem in* (19.4)*, where the cost function is the finite sum* (19.7)*. Consider, we run the Riemannian SRG algorithm to solve this problem with* $S = 1$*. Assume*

- *The iterations stay continuously in a subset* $\mathcal{X}$*.*
- *The vector transport is* θ*-bounded.*
- *The vector transport is isometric on* $\mathcal{X}$*.*
- *The functions* f_i *are retraction* L*-smooth.*
- *The functions* f_i *are retraction* L_l*-Lipschitz.*
- *The gradients of the* f_i*s are* G*-bounded.*
- *The function* f *attains its minimum at* $x^* \in \mathcal{X}$*.*

Assume a constant step-size $\alpha \leq \frac{2}{L+\sqrt{L^2+8m(L_l^2+G^2\theta^2)}}$ *in Algorithm* 3*. Then, we have*

$$\mathbb{E}[\|\nabla f(\tilde{x})\|^2] \leq \frac{2}{\alpha(m+1)}[f(x_0) - f(x^*)].$$

A very similar idea to R-SRG was used in another algorithm called Riemannian SPIDER (R-SPIDER) [45]. The Euclidean counterpart of the R-SPIDER algorithm was shown to have near optimal complexity bound. It can be applied to both the finite-sum and the stochastic optimization problems [14]. The details of the R-SPIDER method are given in Algorithm 4. The algorithm uses retraction and vector transport while the original algorithm and proofs of [45] were for the exponential mapping and the parallel transport. For the analysis of general non-convex functions in this section, we set $T = 1$ meaning that we have a single outer-loop.

Theorem 19.27 (Theorem 1 in Zhou et al. [45]) *Consider the optimization problem in* (19.4)*, where the cost function is the finite sum* (19.7)*. Consider, we run the Riemannian SPIDER algorithm with option I to solve this problem. Assume*

- *The exponential map is used for the retraction and the parallel transport is used for the vector transport.*
- *The functions* f_i *are geodesically* L*-Lipschitz.*

Algorithm 3 Riemannian SRG

```
Given: Smooth manifold $\mathcal{M}$ with retraction Ret and vector transport $\mathcal{T}$; initial value $\tilde{x}^0$; a finite-sum cost function $f$; update frequency $m$; number of epochs $S$.
for $s = 0, \dots,$ S-1 do
    Store $x_0 = \tilde{x}^s$
    Calculate the full Riemannian gradient $\nabla f(x_0)$
    Store $\xi_0 = \nabla f(x_0)$
    Store $x_1 = \mathrm{Ret}_{x_0}(-\alpha_0 \xi_0)$
    for $t = 1, \dots, m-1$ do
        Choose $i_t \in \{1, \dots, n\}$ uniformly at random
        Calculate $\xi_t = \nabla f_{i_t}(x_t) - \mathcal{T}_{x_{t-1}, x_t}\big(\nabla f_{i_t}(x_{t-1}) - \xi_{t-1}\big)$
        Use a step-size rule to choose the step-size $\alpha_t$
        Calculate $x_{t+1} = \mathrm{Ret}_{x_t}(-\alpha_t \xi_t)$
    end for
    Set $\tilde{x}^{s+1} = x_t$ for randomly chosen $t \in \{0, \dots, m\}$
end for
return $\tilde{x}^S$
```

– *The stochastic gradients have σ-bounded variance.*

Let $T = 1$, $s = \min\big(n, \frac{16\sigma^2}{\epsilon^2}\big)$, $p = n_0 s^{\frac{1}{2}}$,$\alpha_k = \min\big(\frac{\epsilon}{2Ln_0}, \frac{\|\xi_k\|}{4Ln_0}\big)$,$|\mathcal{S}_1| = s$, $|\mathcal{S}_2| = \frac{4s^{\frac{1}{2}}}{n_0}$ and $n_0 \in [1, 4s^{\frac{1}{2}}]$ in Algorithm 4. Then, we achieve $\mathbb{E}[\|\nabla f(\tilde{x}^1)\|] \leq \epsilon$ in at most $K = \frac{14Ln_0\Delta}{\epsilon^2}$ iterations in expectation, where $\Delta = f(x_0) - f(x^*)$ with $x^* = \arg\min_{x \in \mathcal{M}} f(x)$.

For the online case, the following theorem considers the iteration complexity of the algorithm.

Theorem 19.28 (Theorem 2 in Zhou et al. [45]) *Consider the optimization problem in (19.4), where the cost function is the expected risk (19.6). Assume the same conditions as in Theorem 19.27. Consider we run the Riemannian SPIDER algorithm with option I to solve this problem. Let $T = 1$, $p = \frac{n_0\sigma}{\epsilon}$,$\alpha_k = \min\big(\frac{\epsilon}{2Ln_0}, \frac{\|\xi_k\|}{4Ln_0}\big)$,$|\mathcal{S}_1| = \frac{64\sigma^2}{\epsilon^2}$, $|\mathcal{S}_2| = \frac{4\sigma}{\epsilon n_0}$ for $n_0 \in [1, 4\frac{\sigma}{\epsilon}]$ in Algorithm 4. Then, we achieve $\mathbb{E}[\|\nabla f(\tilde{x}^1)\|] \leq \epsilon$ in at most $K = \frac{14Ln_0\Delta}{\epsilon^2}$ iterations in expectation, where $\Delta = f(x_0) - f(x^*)$ with $x^* = \arg\min_{x \in \mathcal{M}} f(x)$.*

The authors of [44] give the following convergence theorem for the same algorithm. The following theorems are for finite-sum and online settings.

Theorem 19.29 (Theorem 2 in Zhang et al. [44]) *Consider the same problem and assume the same conditions as in Theorem 19.27. Consider, we run the Riemannian SPIDER algorithm with option II to solve this problem. Let $T = 1$, $p = \lceil n^{1/2} \rceil$, $\alpha_k = \min\{\frac{1}{2L}, \frac{\epsilon}{\|\xi_k\|L}\}$, $|\mathcal{S}_1| = n$, and $|\mathcal{S}_2| = \lceil n^{1/2} \rceil$ for each iteration in Algorithm 4. Then, we achieve $\mathbb{E}[\|\nabla f(\tilde{x}^1)\|^2] \leq 10\epsilon^2$ in at most $K = \frac{4L\Delta}{\epsilon^2}$ iterations, where $\Delta = f(x_0) - f(x^*)$ with $x^* = \arg\min_{x \in \mathcal{M}} f(x)$.*

Algorithm 4 Riemannian SPIDER

```
**Given:** Smooth manifold $\mathcal{M}$ with retraction Ret and vector transport $\mathcal{T}$; initial value $\tilde{x}^0$; noisy version of the cost function $f_i$; iteration interval $p^t$, mini-batch sizes $|\mathcal{S}_1^t|$ and $|\mathcal{S}_{2,k}^t|$; number of epochs $T$ and $K^t$.
**for** $t = 0, \dots, T-1$ **do**
    $x_0 = \tilde{x}^t$
    **for** $k = 0, \dots, K^t - 1$ **do**
        **if** $\mathrm{mod}(k, p^t) = 0$ **then**
            Draw minibatch size $|\mathcal{S}_1^t|$ and compute $\xi_k = \nabla f_{\mathcal{S}_1^t}(x_k)$
        **else**
            Draw minibatch size $|\mathcal{S}_2^t|$ and compute $\nabla f_{\mathcal{S}_2^t}(x_k)$
            Compute $\xi_k = \nabla f_{\mathcal{S}_2^t}(x_k) - \mathcal{T}_{x_{k-1},x_k}\big(\nabla f_{\mathcal{S}_2^t}(x_{k-1}) - \xi_{k-1}\big)$
        **end if**
        **if** $\xi_k \leq 2\epsilon_k$ **then**
            Option II: $\tilde{x}^{t+1} = x_k$, break
        **end if**
        Use a step-size rule to choose the step-size $\alpha_k^t$
        Calculate $x_{k+1} = \mathrm{Ret}_{x_k}(-\alpha_k^t \xi_k)$
    **end for**
    Option I: Output $\tilde{x}^{t+1} = x_k$ for randomly chosen $k \in \{0, \dots, K-1\}$
**end for**
**return** $\tilde{x}^T$
```

Theorem 19.30 (Theorem 1 in Zhang et al. [44]) *Consider the same problem and assume the same conditions as in Theorem 19.28. Consider, we run the Riemannian SPIDER algorithm with option II to solve this problem. Let* $T = 1$*,* $p = \frac{1}{\epsilon}$*,* $\alpha_k = \min\{\frac{1}{2L}, \frac{\epsilon}{\|\xi_k\|L}\}$*,* $|\mathcal{S}_1| = \frac{2\sigma^2}{\epsilon^2}$*, and* $|\mathcal{S}_2| = \frac{2}{\epsilon}$ *for each iteration in Algorithm 4. Then, we achieve* $\mathbb{E}[\|\nabla f(\tilde{x}^1)\|^2] \leq 10\epsilon^2$ *in at most* $K = \frac{4L\Delta}{\epsilon^2}$ *iterations, where* $\Delta = f(x_0) - f(x^*)$ *with* $x^* = \arg\min_{x \in \mathcal{M}} f(x)$*.*

Among the convergence results presented in this section, R-SPIDER is the only algorithm that has strong convergence without the need for the strong condition that the iterates stay in a compact set. This condition is hard to ensure even for simple problems. Another important point to mention is that the step-sizes suggested by the theorems are very small, and in practice much larger step-sizes with some decaying rules are usually used.

19.5 Analysis for G-Convex and Gradient Dominated Functions

For g-convex or gradient dominated functions, we obtain faster convergence rates for the algorithms explained in the previous sections. For plain SGD, [43] proved faster convergence for g-convex functions as stated in the following theorem.

Theorem 19.31 (Theorem 14 in Zhang et al. [43]) *Consider the R-SGD Algorithm for solving the optimization problem in* (19.4)*, where the cost function is the expected risk* (19.6)*. Assume*

- *The function f is g-convex.*
- *The exponential map is used for the retraction.*
- *The iterations stay in a compact subset $\mathcal{X}$, and the diameter of $\mathcal{X}$ is bounded by D, that is* $\max_{x,y\in\mathcal{X}} d(x,y) \le D$*.*
- *There is a lower bound on the sectional curvature denoted by κ.*
- *The functions f_i are geodesically L-smooth.*
- *The function f attains its minimum at $x^* \in \mathcal{X}$.*
- *The functions f_i have σ-bounded variance.*
- *The manifold is Hadamard (Riemannian manifolds with global non-positive curvature).*

Define ζ to be a constant that captures the impact of manifold curvature defined by

$$\zeta = \frac{\sqrt{|\kappa|}D}{\tanh\big(\sqrt{|\kappa|}D\big)}. \tag{19.12}$$

Then the R-SGD algorithm with $\alpha_t = \frac{1}{L+\frac{\sigma}{D}\sqrt{(t+1)\zeta}}$ in Algorithm 1 satisfies

$$\mathbb{E}[f(\bar{x}_T) - f(x^*)] \le \frac{\zeta L D^2 + 2D\sigma\sqrt{\zeta T}}{2(\zeta + T - 1)},$$

where $\bar{x}_1 = x_1$, $\bar{x}_{t+1} = \mathrm{Exp}_{\bar{x}_t}(\frac{1}{t+1}\mathrm{Exp}_{\bar{x}_t}^{-1}(x_{t+1}))$, for $1 \le t \le T-1$ and $\bar{x}_T = \mathrm{Exp}_{\bar{x}_{T-1}}(\frac{\zeta}{\zeta+T-1}\mathrm{Exp}_{\bar{x}_{T-1}}^{-1}(x_T))$.

The aforementioned theorem shows that we need a decaying step size for obtaining faster convergence for the R-SGD algorithm, while Theorem 19.22 needed constant step size for getting a convergence rate for general non-convex functions. Decaying step-size is usually used in practice and the above theorem can be a motivation, because near local minima the function can be assumed to be g-convex. For the case of strongly g-convex functions, the authors of [43] proved a stronger convergence result stated below.

Theorem 19.32 (Theorem 12 in Zhang et al. [43]) *Consider the R-SGD Algorithm for solving the optimization problem in* (19.4)*, where the cost function is the expected risk* (19.6)*. Assume*

- *The function f is μ-strongly g-convex.*
- *The exponential map is used for the retraction.*
- *The iterations stay in a compact subset $\mathcal{X}$, and the diameter of $\mathcal{X}$ is bounded by D, that is* $\max_{x,y\in\mathcal{X}} d(x,y) \le D$*.*
- *There is a lower bound on the sectional curvature denoted by κ.*
- *The function f attains its minimum at $x^* \in \mathcal{X}$.*

- *The expected square norm of the gradients of the f_is are G^2-bounded.*
- *The manifold is Hadamard (Riemannian manifolds with global non-positive curvature).*

Then the R-SGD algorithm with $\alpha_t = \frac{2}{\mu(t+2)}$ in Algorithm 1 satisfies

$$\mathbb{E}[f(\bar{x}_T) - f(x^*)] \leq \frac{2\zeta G}{(T+2)},$$

where $\bar{x}_0 = x_0$, $\bar{x}_{t+1} = \operatorname{Exp}_{\bar{x}_t}(\frac{2}{t+2}\operatorname{Exp}_{\bar{x}_t}^{-1}(x_{t+1}))$ and ζ is a constant given in (19.12).

For strongly g-convex functions, [33] proved a linear convergence rate for the R-SVRG algorithm given in the following theorem.

Theorem 19.33 (Theorem 1 in Zhang et al. [33]) *Consider the optimization problem in* (19.4)*, where the cost function is the finite sum* (19.7)*. Consider, we run the Riemannian SVRG algorithm to solve this problem with $K = 1$, Option I-a and Option I-b. Assume the same conditions as in Theorem 19.24. Furthermore, assume that the function f is μ-strongly g-convex. If we use an update frequency and a constant step size in Algorithm 2 such that the following holds*

$$a = \frac{3\zeta\alpha L^2}{\mu - 2\zeta\alpha L^2} + \frac{(1 + 4\zeta\alpha^2 - 2\alpha\mu)^m(\mu - 5\zeta\alpha L^2)}{\mu - 2\zeta\alpha L^2} < 1,$$

then the iterations satisfy

$$\mathbb{E}[f(\tilde{x}^S) - f(x^*)] \leq \frac{L}{2}\mathbb{E}[d^2(\tilde{x}^S, x^*)] \leq \frac{L}{2}a^S d^2(x^0, x^*).$$

For a class of functions more general than strongly g-convex functions, that is gradient-dominated functions, it is also possible to prove that R-SVRG has a strong linear convergence rate.

Theorem 19.34 (Theorem 3 in Zhang et al. [33]) *Consider the optimization problem in* (19.4)*, where the cost function is the finite sum* (19.7)*. Consider, we run the Riemannian SVRG algorithm to solve this problem with Option II-a, Option I-b. Assume the same conditions as in Theorem 19.24. Furthermore, assume that the function f is τ-gradient dominated. If we use the parameters $\alpha = \frac{\mu_0}{Ln^{2/3}\zeta^{1/3}}$, $m = \lfloor\frac{n}{3\mu_0}\rfloor$, $S = \lceil(6 + \frac{18\mu_0}{n-3})\frac{L\tau\zeta^{1/2}\mu_0}{\nu n^{1/3}}\rceil$ for some universal constants $\mu_0 \in (0, 1)$ and $\nu > 0$ in Algorithm 2, then we have*

$$\mathbb{E}[\|\nabla f(x^K)\|^2] \leq 2^{-K}\|\nabla f(x^0)\|^2,$$

$$\mathbb{E}[f(x^K) - f(x^*)] \leq 2^{-K}[\|f(x^0) - f(x^*)\|].$$

The aforementioned strong convergence results for R-SVRG are valid when using the exponential map and the parallel transport. For the general retraction and the vector transport there is not any global rate of convergence result yet. However, the authors in [36, Theorem 5.14] proved a local linear convergence result for the R-SVRG algorithm.

For the R-SRG algorithm, [22] gives a convergence result for the g-convex case as stated in the following.

Theorem 19.35 (Theorem 4.1 in Kasai et al. [22]) *Consider the optimization problem in* (19.4)*, where the cost function is the finite sum* (19.7)*. Assume the same conditions as in Theorem 19.26 hold, and furthermore assume that*

$$\|\mathcal{P}_{x,y}^{\mathrm{Ret}_x}\nabla f_i(x) - \nabla f_i(y)\|^2 \leq L\langle \mathcal{P}_{x,y}^{\mathrm{Ret}_x}\nabla f_i(x) - \nabla f_i(y),\ \mathrm{Exp}_y^{-1} x\rangle,$$

where L is the constant for the retraction smooth function f. For the Euclidean case, this condition is equal to have a convex and L-smooth function. Consider, we run the Riemannian SRG algorithm to solve this problem using the parameters α and m in Algorithm 3 such that $\alpha < 2/L$ and $(\beta - L^2)\alpha^2 + 3L\alpha - 2 \leq 0$, where

$$\beta := 2\big((2L_l + 2\theta G + L)\theta G + \nu L\big)m. \tag{19.13}$$

Then for $s > 0$,

$$\mathbb{E}[\|\nabla f(\tilde{x}^s)\|^2] \leq \frac{2}{\alpha(m+1)}\mathbb{E}[\|f(\tilde{x}^{s-1}) - f(x^*)\|] + \frac{\alpha L}{2-\alpha L}\mathbb{E}[\|\nabla f(\tilde{x}^{s-1})\|^2].$$

For μ-strongly g-convex functions, the authors of [22] proved linear convergence as stated below. The nice feature of the R-SRG algorithm is that it is the only method that achieves linear convergence without needing the exponential map and the parallel transport.

Theorem 19.36 (Theorem 4.3 in Kasai et al. [22]) *Consider the optimization problem in* (19.4)*, where the cost function is the finite sum* (19.7)*. Assume the same conditions as in Theorem 19.35 and furthermore assume that the function f is μ-strongly convex. Consider, we run the Riemannian SRG algorithm to solve this problem using the parameters α and m in Algorithm 3 such that such that $\alpha_m := \frac{1}{\mu\alpha(m+1)} + \frac{\alpha L}{2-\alpha L} < 1$. Then,*

$$\mathbb{E}[\|\nabla f(\tilde{x}^s)\|^2] \leq \sigma_m^s \mathbb{E}[\|\nabla f(\tilde{x}^0)\|^2].$$

Similarly for τ gradient dominated functions, the authors of [22] obtained linear convergence.

Theorem 19.37 (Theorem 4.6 in Kasai et al. [22]) *Consider the optimization problem in* (19.4)*, where the cost function is the finite sum* (19.7)*. Assume the same conditions as in Theorem 19.26 hold and furthermore assume that the function is*

τ-gradient dominated. Consider, we run Riemannian SRG algorithm to solve this problem with the same α in Algorithm 3 as that of Theorem 19.26 and assume $\bar{\sigma}_m := \frac{2\tau}{\alpha(m+1)} < 1$. Then for $s > 0$,

$$\mathbb{E}[\|\nabla f(\tilde{x}^s)\|^2] \leq \bar{\sigma}_m^s \mathbb{E}[\|\nabla f(\tilde{x}^0)\|^2].$$

For τ-gradient dominated functions, [45] was able to prove stronger convergence results for the R-SPIDER algorithm. The following two theorems are convergence results for the finite-sum and online cases. Unlike the analysis for the general non-convex case, here the authors use a fixed step-size and adaptive batch sizes.

Theorem 19.38 (Theorem 3 in Zhou et al. [45]) *Consider the finite sum problem (19.7) solved using the R-SPIDER algorithm with option I. Assume the same conditions as in Theorem 19.27, and furthermore assume that the function f is τ-gradient dominated. At iteration t of Algorithm 4, set $\epsilon_0 = \frac{\sqrt{\Delta}}{2\sqrt{\tau}}$, $\epsilon_t = \frac{\epsilon_0}{2^t}$, $s_t = \min(n, \frac{32\sigma^2}{\epsilon_{t-1}^2})$, $p^t = n_0^t s_t^{\frac{1}{2}}$,$\alpha_k = \frac{\|\xi_k\|}{2Ln_0}$,$|\mathcal{S}_1^t| = s_t$, $|\mathcal{S}_{2,k}^t| = \min\big(\frac{8p^t\|\xi_{k-1}\|^2}{(n_0^t)^2\epsilon_{t-1}^2}, n\big)$ and $K^t = \frac{64Ln_o^t\Delta^t}{\epsilon_{t-1}^2}$ where $n_0^t \in [1, \frac{8\sqrt{s}\|\xi_{k-1}\|^2}{\epsilon_{t-1}^2}]$ and $\Delta = f(x_0) - f(x^*)$ with $x^* = \arg\min_{x\in\mathcal{M}} f(x)$. Then the sequence $\tilde{x}^t$ satisfies*

$$\mathbb{E}[\|\nabla f(\tilde{x}^t)\|^2] \leq \frac{\Delta}{4^t\tau}.$$

Theorem 19.39 (Theorem 4 in Zhou et al. [45]) *Consider the optimization problem in (19.4) solved using the R-SPIDER algorithm with option I. Assume the same conditions as in Theorem 19.28, and furthermore assume that the function f is τ-gradient dominated. At iteration t of Algorithm 4, set $\epsilon_0 = \frac{\sqrt{\Delta}}{2\sqrt{\tau}}$, $\epsilon_t = \frac{\epsilon_0}{2^t}$, $p^t = \frac{\sigma n_0^t}{\epsilon_{t-1}}$, $\alpha_k^t = \frac{\|\xi_k\|}{2L_l n_0^t}$, $|\mathcal{S}_1^t| = \frac{32\sigma^2}{\epsilon_{t-1}^2}$, $|\mathcal{S}_{2,k}^t| = \frac{8\sigma\|\xi_{k-1}\|^2}{n_0^t\epsilon_{t-1}^3}$ and $K^t = \frac{64Ln_o^t\Delta^t}{\epsilon_{t-1}^2}$ where $n_0 \in [1, \frac{8\sigma\|\xi_{k-1}\|^2}{\epsilon_{t-1}^3}]$ and $\Delta = f(x_0) - f(x^*)$ with $x^* = \arg\min_{x\in\mathcal{M}} f(x)$. Then the sequence $\tilde{x}^t$ satisties,*

$$\mathbb{E}[\|\nabla f(\tilde{x}^t)\|^2] \leq \frac{\Delta}{4^t\tau}.$$

The authors of [44] give the following analysis of the R-SPIDER algorithm for τ-gradient dominated functions.

Theorem 19.40 (Theorem 3 in Zhang et al. [44]) *Consider the same problem and assume the same conditions as in Theorem 19.28. Consider, we run the Riemannian SPIDER algorithm with option II to solve this problem. Let $p = \lceil n^{1/2}\rceil$,$\epsilon_t = \sqrt{\frac{M_0}{10\tau 2^t}}$,$\alpha_t = \frac{\epsilon_t}{L}$,$|\mathcal{S}_1| = n$, and $|\mathcal{S}_2| = \lceil n^{1/2}\rceil$ in each iteration of Algorithm 4, where $M_0 > f(x_0) - f(x^*)$ with $x^* = \arg\min_{x\in\mathcal{M}} f(x)$. Then the algorithm returns $\tilde{x}^T$ that satisfies*

$$\mathbb{E}[f(\tilde{x}^T) - f(x^*)] \leq \frac{M_0}{2^T}.$$

The authors of [44] also give another proof for the R-SPIDER algorithm with different parameters that give better iteration complexity for τ-gradient dominated functions with respect to n.

Theorem 19.41 (Theorem 4 in Zhang et al. [44]) *Consider the same problem and assume the same conditions as in Theorem 19.28. Consider, we run the Riemannian SPIDER algorithm with option II to solve this problem. In Algorithm 4, let* $T = 1$, $p = \lceil 4L\tau \log(4) \rceil$, $\alpha = \frac{1}{2L}$, $|\mathcal{S}_1| = n$, *and* $|\mathcal{S}_{2,k}| = \left\lceil \min\left\{ n, \frac{4\tau p L^2 \| \mathrm{Exp}^{-1}_{x_{k-1}}(x_k) \|^2 2^{\lceil k/p \rceil}}{M_0} \right\} \right\rceil$, *where* $M_0 > f(x_0) - f(x^*)$ *with* $x^* = \arg\min_{x \in \mathcal{M}} f(x)$. *Then, the algorithm returns* $\tilde{x}^K$ *after* $K = pS$ *iterations that satisfies*

$$\mathbb{E}[f(\tilde{x}^K) - f(x^*)] \leq \frac{M_0}{2^S}.$$

The theorems of the algorithms in the previous sections showing convergence speed of different algorithms are summarized in Tables 19.1 and 19.2. The incremental first order oracle (IFO) complexity for different algorithms are calculated by counting the number of evaluations needed to reach the ϵ accuracy of gradient ($\mathbb{E}[\|\nabla f(x)\|^2] \leq \epsilon$) or function ($\mathbb{E}[f(x) - f(x^*)] \leq \epsilon$) in the theorems given in the previous sections.

Table 19.1 Comparison of the IFO complexity for different Riemannian stochastic optimization algorithms under finite-sum and online settings

	Method	general non-convex	g-convex	Theorem
Finite-sum	R-SGD* [18]	$\mathcal{O}\left(\frac{L}{\epsilon} + \frac{L^2\sigma^2}{\epsilon^2}\right)$	–	19.22
	R-SRG [22]	$\mathcal{O}\left(n + \frac{L^2}{\epsilon^2}\right)$	$\mathcal{O}\left(\left(n + \frac{1}{\epsilon}\right)\log\left(\frac{1}{\epsilon}\right)\right)$	19.26, 19.35
	R-SRG* [22]	$\mathcal{O}\left(n + \frac{L^2\rho_l^2+\theta^2}{\epsilon^2}\right)$	$\mathcal{O}\left(\frac{\left(n+\frac{1}{\epsilon}\right)\log\left(\frac{1}{\epsilon}\right)}{\log(c(1-\beta/L^2))}\right)$	19.26, 19.35
	R-SVRG [33]	$\mathcal{O}\left(n + \frac{\zeta^{1/2}n^{2/3}}{\epsilon}\right)$	–	19.24
	R-SPIDER [45]	$\mathcal{O}\left(\min\left(n + \frac{L\sqrt{n}}{\epsilon}, \frac{L\sigma}{\epsilon^{3/2}}\right)\right)$	–	19.27, 19.38
	R-SPIDER [44]	$\mathcal{O}\left(n + \frac{L\sqrt{n}}{\epsilon}\right)$	–	19.29

(continued)

Table 19.1 (continued)

Online	R-SGD* [18]	$\mathcal{O}\left(\frac{L}{\epsilon}+\frac{L^2\sigma^2}{\epsilon^2}\right)$	–	19.22
	R-SPIDER [45]	$\mathcal{O}\left(\frac{L\sigma}{\epsilon^{3/2}}\right)$	–	19.28
	R-SPIDER [44]	$\mathcal{O}\left(\frac{L\sigma^2}{\epsilon^{3/2}}\right)$	–	19.30

The ϵ-accuracies of gradients are reported for general non-convex and g-convex functions. Star in front of the method names means using the general retraction and the parallel transport, and no star means using the exponential map and the parallel transport in the method. The parameter ζ (19.11) is determined by manifold curvature and diameter, σ is the standard deviation of stochastic gradients, θ is the constant in θ-bounded vector transport, $\rho_l = L_l/L$ for retraction L-smooth and retraction L_l-Lipschitz function, the parameter β is defined in (19.13) and $c > 1$ is a constant. Apparently for the parallel transport $\theta = 0$ and $\rho_l = 1$

Table 19.2 Comparison of the IFO complexity for different Riemannian stochastic optimization algorithms under finite-sum and online settings

	Method	τ-gradient dominated	μ-strongly g-convex	Theorem
Finite-sum	R-SGD [44]	–	$\frac{\zeta G}{\epsilon}$	19.32
	R-SRG [22]	$\mathcal{O}\left(\left(n+L_\tau^2\right)\log\left(\frac{1}{\epsilon}\right)\right)$	$\mathcal{O}\left(\left(n+L_\mu\right)\log\left(\frac{1}{\epsilon}\right)\right)$	19.36, 19.37
	R-SRG* [22]	$\mathcal{O}\left(\left(n+\tau^2(L^2\rho_l^2+\theta^2)\right)\log\left(\frac{1}{\epsilon}\right)\right)$	$\mathcal{O}\left(\frac{(n+L_\mu)\log(\frac{1}{\epsilon})}{\log(c(1-\beta/L^2))}\right)$	19.36, 19.37
	R-SVRG [33]	$\mathcal{O}\left((n+L_\tau\zeta^{1/2}n^{2/3})\log\left(\frac{1}{\epsilon}\right)\right)$	$\mathcal{O}((n+\zeta L_\mu^2)\log\left(\frac{1}{\epsilon}\right))$	19.33, 19.34
	R-SPIDER [45]	$\mathcal{O}\left(\min\left(\left(n+L_\tau\sqrt{n}\right)\log\left(\frac{1}{\epsilon}\right),\,\frac{L_\tau\sigma}{\epsilon^{1/2}}\right)\right)$	←	19.38
	R-SPIDER [44]	$\mathcal{O}\left(\left(n+\min\left(L_\tau\sqrt{n},\,L_\tau^2\right)\right)\log\left(\frac{1}{\epsilon}\right)\right)$	←	19.40, 19.41
Online	R-SGD [44]	–	$\frac{\zeta G}{\epsilon}$	19.32
	R-SPIDER [45]	$\mathcal{O}\left(\frac{L_\tau\sigma}{\epsilon^{1/2}}\right)$	←	19.39

The ϵ-accuracies of functions are reported for μ-strongly g-convex and τ-gradient dominated functions. The results of Theorems 19.34, 19.36, 19.37, 19.38 are originally given for the ϵ-accuracy of gradient, and they also hold for the ϵ-accuracy of function because of (19.3). The parameters $L_\tau = 2\tau L$ and $L_\mu = \frac{L}{\mu}$ are condition numbers, G is the bound for the norm of the stochastic gradients, and other parameters are the same as those given in Table 19.1. From Proposition 19.19, it is clear that the complexity results for τ-gradient dominated functions also hold for μ-strongly g-convex functions, and to obtain complexity results it is enough to change L_τ to L_μ in the equations

19.6 Example Applications

We list below a few finite-sum optimization problems drawn from a variety of applications. Riemannian stochastic methods turn out to be particularly effective for solving these problems. We only include the formulation, and refer the reader to the cited works for details about implementation and empirical performance. The manifolds occurring in the examples below are standard, and the reader can find explicit implementations of retractions, vector transport, etc., within the MANOPT software [10], for instance.

Stochastic PCA

Suppose we have observations $z_1, \dots, z_n \in \mathbb{R}^d$. The stochastic PCA problem is to compute the top eigenvector of the matrix $\sum_{i=1}^n z_i z_i^T$. This problem can be written as a finite-sum optimization problem on the sphere $\mathbb{S}^{d-1}$ as follows

$$\min_{x^T x=1} \quad -x^T \left(\sum_{i=1}^{n} z_i z_i^T\right) x = -\sum_{i=1}^{n} (z_i^T x)^2. \tag{19.14}$$

Viewing (21.100) as a Riemannian optimization problem was proposed in [33], who solved it using R-SVRG, in particular, by proving that the cost function satisfies a Riemannian gradient-dominated condition (probabilistically). One can extend this problem to solve for the top-k eigenvectors by considering is as an optimization problem on the Stiefel manifold.

A challenge for the methods discussed in the present paper, except R-SGD and R-SPIDER explained in Sect. 19.4 is the requirements for the iterates to remain within a predefined compact set. While the whole manifold is compact, for obtaining a precise theoretical characterization of the computational complexity of the algorithms involved, the requirement to remain within a compact set is important.

GMM

Let $z_1, \dots, z_n$ be observations in $\mathbb{R}^d$ that we wish to model using a Gaussian mixture model (GMM). Consider the mixture density

$$p(z; \{\mu_j, \Sigma_j\}_{j=1}^k) := \sum_{j=1}^{k} \pi_j \mathcal{N}(z; \mu_j, \Sigma_j),$$

where $\mathcal{N}(z; \mu, \Sigma)$ denotes the Gaussian density evaluated at z and parameterized by μ and Σ. This leads to the following maximum likelihood problem:

(continued)

$$\max_{\{\pi_j,\mu_j,\Sigma_j\}_{j=1}^k} \sum\nolimits_{i=1}^n \log p(z_i; \{\mu_j, \Sigma_j\}_{j=1}^k). \tag{19.15}$$

In [18], the authors reformulate (21.52) to cast it as a problem well-suited for solving using R-SGD. They consider the reformulated problem

$$\max_{\{\omega_j, S_j \succ 0\}_{j=1}^k} \sum_{i=1}^n \log\Big(\sum\nolimits_{j=1}^k \frac{\exp(\omega_j)}{\sum_{k=1}^k \exp(\omega_k)} q(y_i; S_j)\Big), \tag{19.16}$$

where $y_i = [z_i; 1]$, and $q(y; S_j)$ is the centered normal distribution parameterized by $S_j = \begin{bmatrix} \Sigma_j + \mu_j\mu_j^T & \mu_j \\ \mu_j^T & 1 \end{bmatrix}$. With these definitions, problem (21.61) can be viewed as an optimization problem on the product manifold $\big(\prod_{j=1}^k \mathbb{P}^{d+1}\big) \times \mathbb{R}^{k-1}$.

Importantly, in [18] it was shown that SGD generates iterates that remain bounded, which is crucial, and permits one to invoke the convergence analysis without resorting to projection onto a compact set.

Karcher Mean

Let $A_1, \ldots, A_n$ be hermitian (strictly) positive definite (hpd) matrices. This set is a manifold, commonly endowed with the Riemmanian metric $\langle \eta, \xi \rangle = \operatorname{tr}(\eta X^{-1} \xi X^{-1})$. This metric leads to the distance $d(X, Y) := \|\log(X^{-1/2} Y X^{-1/2})\|_F$ between hpd matrices X and Y. The Riemannian centroid (also called the "Karcher mean") is defined as the solution to the following finite-sum optimization problem:

$$\min_{X \succ 0} \sum\nolimits_{i=1}^n w_i d^2(X, A_i), \tag{19.17}$$

where the weights $w_i \geq 0$ and $\sum_{i=1}^n w_i = 1$. This problem is often used as a defacto benchmark problem for testing Riemannian optimization problems (see e.g., [33]). The objective function in (19.14) is both geodesically L-smooth as well as strongly convex, both properties can be exploited to obtain faster convergence [22, 33].

It is important to note that this problem is over the manifold of hpd matrices, which is a noncompact manifold. Hence, to truly invoke the convergence theorems (except for R-SGD and R-SPIDER explained in Sect. 19.4), we need to ensure lower bounds on the curvature as well as ensure that iterates remain within a compact set. Lower bounds on the curvature can be obtained in terms

(continued)

of $\min_{1\leq i\leq n} \lambda_{\min}(A_i)$; ensuring that the iterates remain within a compact set can be ensured via projection. Fortunately, for (19.17), a simple compact set containing the solution is known, since we know that (see e.g., [7]) its solution X^* satisfies $\textsc{Hm}(A_1, \ldots, A_n) \preceq X^* \preceq \textsc{Am}(A_1, \ldots, A_n)$, where Hm and Am denote the Harmonic and Arithmetic Means, respectively. A caveat, however, is that R-SVRG and related methods do not permit a projection operation and assume their iterates to remain in a compact set by fiat; R-SGD, however, allows metric projection and can be applied. Nevertheless, in practice, one can invoke any of the methods discussed in this chapter.

We note in passing here that the reader may also be interested in considering the somewhat simpler "Karcher mean" problems that arise when learning hyperbolic embeddings [35], as well as Fréchet-means on other manifolds [3, 31].

Wasserstein Barycenters

Consider two centered multivariate Gaussian distributions with covariance matrices Σ_1 and Σ_2. The Wasserstein W_2 optimal transport distance between them is given by

$$d_W^2(\Sigma_1, \Sigma_2) := \operatorname{tr}(\Sigma_1 + \Sigma_2) - 2\operatorname{tr}[(\Sigma_1^{1/2}\Sigma_2\Sigma_1^{1/2})^{1/2}]. \tag{19.18}$$

The Wasserstein barycenter of n different centered Gaussians is then given by the solution to the optimization problem

$$\min_{X\succ 0} \quad \sum\nolimits_{i=1}^{n} w_i d_W^2(X, \Sigma_i). \tag{19.19}$$

While (21.83) is a (Euclidean) convex optimization problem, it lends itself to more efficient solution by viewing it as a Riemannian convex optimization problem [40]. A discussion about compact sets similar to the Karcher mean example above applies here too.

Riemannian Dictionary Learning

Dictionary learning problems seek to encode input observations using a sparse combination of an "overcomplete basis". The authors of [12] study a Riemannian version of dictionary learning, where input hpd matrices must be encoded as sparse combinations of a set of hpd "dictionary atoms." This problem may be cast as the finite-sum minimization problem

(continued)

$$\min_{B,\alpha_1,\ldots,\alpha_n} \quad \sum\nolimits_{i=1}^{n} d^2\left(X_i, \sum\nolimits_{j=1}^{m} \alpha_{ij} B_j\right) + R(B, \alpha_1, \ldots, \alpha_n). \tag{19.20}$$

In other words, we seek to approximate each input matrix $X_i \approx \sum_{j=1}^{m} \alpha_{ij} B_j$, using $B_j \succ 0$ and nonegative coefficients α_{ij}. The function $R(\cdot)$ is a suitable regularizer on the tensor B and the coefficient matrix α, and $d(\cdot, \cdot)$ denotes the Riemannian distance.

For this particular problem, we can invoke any of the discussed stochastic methods in practice; though previously, results only for SGD have been presented [12]. By assuming a suitable regularizer $R(\cdot, \cdot)$ we can ensure that the problem has a solution, and that the iterates generated by the various methods remain bounded.

Acknowledgements SS acknowledges partial support from an NSF-CAREER grant, the DARPA-Lagrange program, and Amazon Research.

References

1. Absil, P.A., Mahony, R., Sepulchre, R.: Optimization Algorithms on Matrix Manifolds. Princeton University Press, Princeton (2009)
2. Amari, S.I.: Natural gradient works efficiently in learning. Neural Comput. **10**(2), 251–276 (1998)
3. Arnaudon, M., Barbaresco, F., Yang, L.: Medians and means in Riemannian geometry: existence, uniqueness and computation. In: Matrix Information Geometry, pp. 169–197. Springer, Berlin (2013)
4. Babanezhad, R., Laradji, I.H., Shafaei, A., Schmidt, M.: Masaga: a linearly-convergent stochastic first-order method for optimization on manifolds. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 344–359. Springer, Berlin (2018)
5. Bécigneul, G., Ganea, O.E.: Riemannian adaptive optimization methods (2018). Preprint. arXiv:1810.00760
6. Bertsekas, D.P.: Nonlinear Programming, 2nd edn. Athena Scientific, Nashua (1999)
7. Bhatia, R.: Positive Definite Matrices. Princeton University Press, Princeton (2007)
8. Bonnabel, S.: Stochastic gradient descent on Riemannian manifolds. IEEE Trans. Autom. Control **58**(9), 2217–2229 (2013)
9. Boumal, N., Absil, P.A.: RTRMC: a Riemannian trust-region method for low-rank matrix completion. In: Advances in Neural Information Processing Systems, pp. 406–414 (2011)
10. Boumal, N., Mishra, B., Absil, P.A., Sepulchre, R.: Manopt, a matlab toolbox for optimization on manifolds. J. Mach. Learn. Res. **15**(1), 1455–1459 (2014)
11. Boumal, N., Absil, P.A., Cartis, C.: Global rates of convergence for nonconvex optimization on manifolds. IMA J. Numer. Anal. **39**(1), 1–33 (2019)
12. Cherian, A., Sra, S.: Riemannian dictionary learning and sparse coding for positive definite matrices. IEEE Trans. Neur. Net. Lear. Syst. **28**(12), 2859–2871 (2017)
13. Defazio, A., Bach, F., Lacoste-Julien, S.: SAGA: a fast incremental gradient method with support for non-strongly convex composite objectives. In: Advances in Neural Information Processing Systems, pp. 1646–1654 (2014)

14. Fang, C., Li, C.J., Lin, Z., Zhang, T.: Spider: near-optimal non-convex optimization via stochastic path-integrated differential estimator. In: Advances in Neural Information Processing Systems, pp. 687–697 (2018)
15. Ghadimi, S., Lan, G.: Stochastic first- and zeroth-order methods for nonconvex stochastic programming. SIAM J. Optim. **23**(4), 2341–2368 (2013)
16. Guadarrama: Fitting large-scale gaussian mixtures with accelerated gradient descent. Master's Thesis, University of Edinburgh (2018)
17. Hosseini, R., Sra, S.: Matrix manifold optimization for Gaussian mixtures. In: Advances in Neural Information Processing Systems, pp. 910–918 (2015)
18. Hosseini, R., Sra, S.: An alternative to EM for Gaussian mixture models: batch and stochastic Riemannian optimization. Math. Program. (2019)
19. Huang, W., Gallivan, K.A., Absil, P.A.: A broyden class of quasi-Newton methods for riemannian optimization. SIAM J. Optim. **25**(3), 1660–1685 (2015)
20. Jost, J.: Riemannian Geometry and Geometric Analysis. Springer, Berlin (2011)
21. Kasai, H., Mishra, B.: Inexact trust-region algorithms on Riemannian manifolds. In: Advances in Neural Information Processing Systems, pp. 4249–4260 (2018)
22. Kasai, H., Sato, H., Mishra, B.: Riemannian stochastic recursive gradient algorithm. In: International Conference on Machine Learning, pp. 2516–2524 (2018)
23. Kasai, H., Sato, H., Mishra, B.: Riemannian stochastic quasi-Newton algorithm with variance reduction and its convergence analysis. In: Twenty-First International Conference on Artificial Intelligence and Statistics, vol. 84, pp. 269–278 (2018)
24. Kasai, H., Jawanpuria, P., Mishra, B.: Riemannian adaptive stochastic gradient algorithms on matrix manifolds. In: International Conference on Machine Learning, pp. 3262–3271 (2019)
25. Kumar Roy, S., Mhammedi, Z., Harandi, M.: Geometry aware constrained optimization techniques for deep learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4460–4469 (2018)
26. Liu, H., So, A.M.C., Wu, W.: Quadratic optimization with orthogonality constraint: explicit łojasiewicz exponent and linear convergence of retraction-based line-search and stochastic variance-reduced gradient methods. Math. Program. 1–48 (2018)
27. Meyer, G., Bonnabel, S., Sepulchre, R.: Linear regression under fixed-rank constraints: a Riemannian approach. In: International Conference on Machine Learning (2011)
28. Mishra, B., Kasai, H., Jawanpuria, P., Saroop, A.: A Riemannian gossip approach to subspace learning on Grassmann manifold. Mach. Learn. **108**(10), 1783–1803 (2019)
29. Nguyen, L.M., Liu, J., Scheinberg, K., Takáč, M.: SARAH: a novel method for machine learning problems using stochastic recursive gradient. In: International Conference on Machine Learning, pp. 2613–2621 (2017)
30. Nguyen, L.M., van Dijk, M., Phan, D.T., Nguyen, P.H., Weng, T.W., Kalagnanam, J.R.: Optimal finite-sum smooth non-convex optimization with SARAH. arXiv preprint arXiv: 1901.07648 (2019)
31. Nielsen, F., Bhatia, R.: Matrix Information Geometry. Springer, Berlin (2013)
32. Reddi, S.J., Hefny, A., Sra, S., Poczos, B., Smola, A.: Stochastic variance reduction for nonconvex optimization. In: International Conference on Machine Learning, pp. 314–323 (2016)
33. Zhang, H., Reddi, S., Sra, S.: Riemannian SVRG: Fast stochastic optimization on Riemannian manifolds. In: Advances in Neural Information Processing Systems, pp. 4592–4600 (2016)
34. Rudi, A., Ciliberto, C., Marconi, G., Rosasco, L.: Manifold structured prediction. In: Advances in Neural Information Processing Systems, pp. 5610–5621 (2018)
35. Sala, F., De Sa, C., Gu, A., Re, C.: Representation tradeoffs for hyperbolic embeddings. In: International Conference on Machine Learning, vol. 80, pp. 4460–4469 (2018)
36. Sato, H., Kasai, H., Mishra, B.: Riemannian stochastic variance reduced gradient algorithm with retraction and vector transport. SIAM J. Optim. **29**(2), 1444–1472 (2019)
37. Sra, S., Hosseini, R.: Conic geometric optimization on the manifold of positive definite matrices. SIAM J. Optim. **25**(1), 713–739 (2015)

38. Sra, S., Hosseini, R., Theis, L., Bethge, M.: Data modeling with the elliptical gamma distribution. In: Artificial Intelligence and Statistics, pp. 903–911 (2015)
39. Vandereycken, B.: Low-rank matrix completion by Riemannian optimization. SIAM J. Optim. **23**(2), 1214–1236 (2013)
40. Weber, M., Sra, S.: Riemannian Frank-Wolfe methods with application to the Karcher and Wasserstein means. arXiv: 1710.10770 (2018)
41. Xu, Z., Gao, X.: On truly block eigensolvers via Riemannian optimization. In: International Conference on Artificial Intelligence and Statistics, pp. 168–177 (2018)
42. Yuan, X., Huang, W., Absil, P.A., Gallivan, K.A.: A Riemannian quasi-Newton method for computing the karcher mean of symmetric positive definite matrices. Technical Reporet FSU17-02, Florida State University (2017)
43. Zhang, H., Sra, S.: First-order methods for geodesically convex optimization. In: Conference on Learning Theory, pp. 1617–1638 (2016)
44. Zhang, J., Zhang, H., Sra, S.: R-SPIDER: A fast Riemannian stochastic optimization algorithm with curvature independent rate. arXiv: 1811.04194 (2018)
45. Zhou, P., Yuan, X.T., Feng, J.: Faster first-order methods for stochastic non-convex optimization on Riemannian manifolds. In: The 22nd International Conference on Artificial Intelligence and Statistics, pp. 138–147 (2019)

Chapter 20
Averaging Symmetric Positive-Definite Matrices

Xinru Yuan, Wen Huang, Pierre-Antoine Absil, and Kyle A. Gallivan

Contents

This work was supported by the Fundamental Research Funds for the Central Universities (No. 20720190060).

X. Yuan · K. A. Gallivan
Department of Mathematics, Florida State University, Tallahassee, FL, USA
e-mail: yuan.xinru43@gmail.com; kgallivan@fsu.edu

W. Huang (✉)
School of Mathematical Sciences, Xiamen University, Xiamen City, People's Republic of China
e-mail: wen.huang@xmu.edu.cn

P.-A. Absil
Department of Mathematical Engineering, ICTEAM Institute, Université catholique de Louvain, Louvain-la-Neuve, Belgium
e-mail: pa.absil@uclouvain.be

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_20

Abstract Symmetric positive definite (SPD) matrices have become fundamental computational objects in many areas, such as medical imaging, radar signal processing, and mechanics. For the purpose of denoising, resampling, clustering or classifying data, it is often of interest to average a collection of symmetric positive definite matrices. This paper reviews and proposes different averaging techniques for symmetric positive definite matrices that are based on Riemannian optimization concepts.

20.1 Introduction

A symmetric matrix is *positive definite* (SPD) if all its eigenvalues are positive. The set of all $n \times n$ SPD matrices is denoted by

$$\mathcal{S}_{++}^n = \{A \in \mathbb{R}^{n\times n} \mid A = A^T, A \succ 0\},$$

where $A \succ 0$ denotes that all the eigenvalues of A are positive; and an ellipse or an ellipsoid $\{x \in \mathbb{R}^n \mid x^T A x = 1\}$ is used to represent a 2×2 SPD matrix or larger SPD matrix, see Fig. 20.1. SPD matrices have become fundamental computational objects in many areas. For example, they appear as diffusion tensors in medical imaging [25, 32, 60], as data covariance matrices in radar signal processing [15, 42], and as elasticity tensors in elasticity [50]. In these and similar applications, it is often of interest to average or find a central representative for a collection of SPD matrices, e.g., to aggregate several noisy measurements of the same object. Averaging also appears as a subtask in interpolation methods [2] and segmentation [16, 59]. In clustering methods, finding a cluster center as a representative of each cluster is crucial. Hence, it is desirable to find a center that is intrinsically representative and can be computed efficiently.

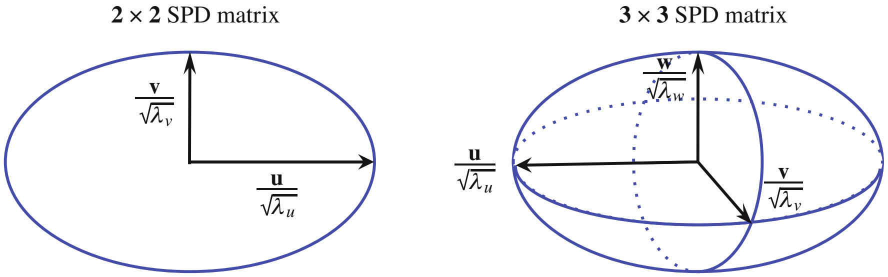

2 × 2 SPD matrix
$\frac{\mathbf{v}}{\sqrt{\lambda_v}}$
$\frac{\mathbf{u}}{\sqrt{\lambda_u}}$
3 × 3 SPD matrix
$\frac{\mathbf{w}}{\sqrt{\lambda_w}}$
$\frac{\mathbf{u}}{\sqrt{\lambda_u}}$
$\frac{\mathbf{v}}{\sqrt{\lambda_v}}$

Fig. 20.1 Visualization of an SPD matrix. The axes represent the directions of eigenvectors and the lengths of the axes are the reciprocals of the square roots of the corresponding eigenvalues

20.2 ALM Properties

A natural way to average a collection of SPD matrices, $\{A_1, \ldots, A_K\}$, is to take their arithmetic mean, i.e., $G(A_1, \ldots, A_K) = (A_1 + \cdots + A_K)/K$. However, this is not appropriate in applications where invariance under inversion is required, i.e., $G(A_1, \ldots, A_K)^{-1} = G(A_1^{-1}, \ldots, A_K^{-1})$. In addition, the arithmetic mean may cause a “swelling effect" that should be avoided in diffusion tensor imaging. Swelling is defined as an increase in the matrix determinant after averaging, see Fig. 20.2 or [32] for more examples. An alternative is to generalize the definition of the geometric mean from scalars to matrices, which yields $G(A_1, \ldots, A_K) = (A_1 \ldots A_K)^{1/K}$. However, this generalized geometric mean is not invariant under permutation since matrices are not commutative in general. Ando et al. [8] introduced a list of fundamental properties, referred to as the ALM list, that a matrix “geometric” mean should possess:

P1 Consistency with scalars. If $A_1, \ldots, A_K$ commute then $G(A_1, \ldots, A_K) = (A_1 \cdots A_K)^{1/K}$.

P2 Joint homogeneity. $G(\alpha_1 A_1, \ldots, \alpha_K A_K) = (\alpha_1 \cdots \alpha_K)^{1/K} G(A_1, \ldots, A_K)$.

P3 Permutation invariance. For any permutation $\pi(A_1, \ldots, A_K)$ of $(A_1, \ldots, A_K)$, $G(A_1, \ldots, A_K) = G(\pi(A_1, \ldots, A_K))$.

P4 Monotonicity. If $A_i \geq B_i$ for all i, then $G(A_1, \ldots, A_K) \geq G(B_1, \ldots, B_K)$ in the positive semidefinite ordering, i.e., $A \geq B$ iff $A - B \succeq 0$, i.e., $A \geq B$ means that $A - B$ is positive semidefinite (all its eigenvalues are nonnegative).

P5 Continuity from above. If $\{A_1^{(n)}\}, \ldots, \{A_K^{(n)}\}$ are monotonic decreasing sequences (in the positive semidefinite ordering) converging to $A_1, \ldots, A_K$, respectively, then $G(A_1^{(n)}, \ldots, A_K^{(n)})$ converges to $G(A_1, \ldots, A_K)$.

P6 Congruence invariance. $G(S^T A_1 S, \ldots, S^T A_K S) = S^T G(A_1, \ldots, A_K) S$ for any invertible S.

P7 Joint concavity. $G(\lambda A_1 + (1-\lambda) B_1, \ldots, \lambda A_K + (1-\lambda) B_K) \geq \lambda G(A_1, \ldots, A_K) + (1-\lambda) G(B_1, \ldots, B_K)$.

P8 Invariance under inversion. $G(A_1, \ldots, A_K)^{-1} = G(A_1^{-1}, \ldots, A_K^{-1})$.

P9 Determinant identity. $\det G(A_1, \ldots, A_K) = (\det A_1 \cdots \det A_K)^{1/K}$.

These properties are known to be important in numerous applications, e.g. [19, 44, 50]. In the case of $K = 2$, the geometric mean is uniquely defined by the above properties and given by the following expression [17]

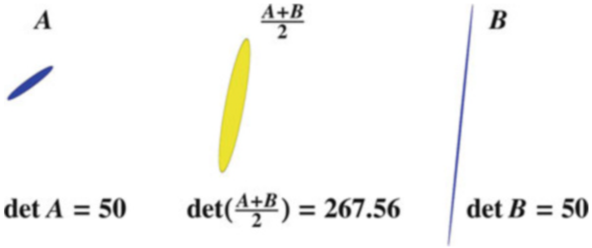


Fig. 20.2 An example of the swelling effect of the arithmetic mean

$$G(A, B) = A^{\frac{1}{2}}(A^{-\frac{1}{2}}BA^{-\frac{1}{2}})^{\frac{1}{2}}A^{\frac{1}{2}}, \tag{20.1}$$

where $Z^{\frac{1}{2}}$ for $Z \succ 0$ is the unique SPD matrix such that $Z^{\frac{1}{2}}Z^{\frac{1}{2}} = Z$. However, the ALM properties do not uniquely define a mean for $K \geq 3$. There can be many different definitions of means that satisfy all the properties. The Karcher mean, discussed in Sect. 20.3.1, is one of them.

20.3 Geodesic Distance Based Averaging Techniques

Since $\mathcal{S}^n_{++}$ is an open submanifold of the vector space of $n \times n$ symmetric matrices, its tangent space at a point X, denoted by $\mathrm{T}_X\,\mathcal{S}^n_{++}$, can be identified with the set of $n \times n$ symmetric matrices. The manifold $\mathcal{S}^n_{++}$ becomes a Riemannian manifold when endowed with the affine-invariant metric,[1] see [59], given by

$$g_X(\xi_X, \eta_X) = \mathrm{trace}(\xi_X X^{-1}\eta_X X^{-1}). \tag{20.2}$$

The length of a continuously differentiable curve $\gamma : [0, 1] \to \mathcal{M}$ on a Riemannian manifold is

$$\int_0^1 \sqrt{g_{\gamma(t)}(\dot{\gamma}(t), \dot{\gamma}(t))}dt.$$

It is known that, for all X and Y on the Riemannian manifold $\mathcal{S}^n_{++}$ with respect to the metric (20.2), there is a unique shortest curve such that $\gamma(0) = X$ and $\gamma(1) = Y$. This curve, given by

$$X^{\frac{1}{2}}(X^{-\frac{1}{2}}YX^{-\frac{1}{2}})^t X^{\frac{1}{2}},$$

is termed a *geodesic*. Its length, given by

$$\delta(X, Y) = \|\log(X^{-1/2}YX^{-1/2})\|_{\mathrm{F}},$$

is termed the *geodesic distance* between X and Y; see, e.g., [18, Proposition 3] or [59, §3.3].

[1] The family of Riemanian metrics that satisfy the affine invariance property is described in [34]; see also Sect. 20.5. The Riemannian metric (20.2) is also called the natural metric [31], the trace metric [43], or the Rao–Fisher metric [63].

20.3.1 Karcher Mean (L^2 Riemannian Mean)

The Karcher mean of $\{A_1, \ldots, A_K\}$, also called the Fréchet mean, the Riemannian barycenter, or the Riemannian center of mass, is defined as the minimizer of the sum of squared distances

$$\mu = \underset{X \in \mathcal{S}^n_{++}}{\arg\min}\, F(X), \quad \text{with } F : \mathcal{S}^{\mathrm{n}}_{++} \to \mathbb{R},\ X \mapsto \frac{1}{2K} \sum_{i=1}^{K} \delta^2(X, A_i), \tag{20.3}$$

where δ is the geodesic distance associated with metric (20.2). It is proved in [17, 18] that F is strictly convex and therefore has a unique minimizer. Hence, a point $\mu \in \mathcal{S}^n_{++}$ is a Karcher mean if it is a stationary point of F, i.e., $\operatorname{grad} F(\mu) = 0$, where $\operatorname{grad} F$ denotes the Riemannian gradient of F with respect to the metric (20.2). The Karcher mean in (20.3) satisfies all properties in the ALM list [19, 44], and therefore is often used in practice. However, a closed-form solution for problem (20.3) is not known in general, and for this reason, the Karcher mean is usually computed by iterative methods.

Various methods have been used to compute the Karcher mean of SPD matrices. Most of them resort to the framework of Riemannian optimization (see, e.g., [1]). One exception in [77] resorts to a majorization minimization algorithm. This algorithm is easy to use in the sense that it is a parameter-free algorithm. However, it is usually not as efficient as other Riemannian-optimization-based methods [38]. Several stepsize selection rules have been investigated for the Riemannian steepest descent (RSD) method. A constant stepsize strategy is proposed in [62] and a convergence analysis is given. An adaptive stepsize selection rule based on the explicit expression of the Riemannian Hessian of the cost function F is studied in [61, Algorithm 2], and is shown to be the optimal stepsize for strongly convex cost functions in Euclidean space, see [52, Theorem 2.1.14]. That is, the stepsize is chosen as $\alpha_k = 2/(M_k + L_k)$, where M_k and L_k are the lower and upper bounds on the eigenvalues of the Riemannian Hessian of F, respectively. A Riemannian version of the Barzilai-Borwein stepsize (RBB) has been considered in [38]. A version of Newton's method for the Karcher mean computation is also provided in [61]. A Richardson-like iteration is derived and evaluated empirically in [21], and is available in the Matrix Means Toolbox.[2] Yuan has shown in [73] that the Richardson-like iteration is a steepest descent method with stepsize $\alpha_k = 1/L_k$. In [48], a computationally cheap per iteration sequence is analyzed. The method is an incremental gradient algorithm for the cost function (20.3) based on a shuffled inductive sequence. It is shown that a few iterations gives a matrix that is the best initialization for the state-of-the-art optimization algorithms when compared to commonly-used initial guesses, such as arithmetic-harmonic mean.

[2] http://bezout.dm.unipi.it/software/mmtoolbox/.

A survey of several optimization algorithms for averaging SPD matrices is presented in [39], including Riemannian versions of steepest descent, conjugate gradient, BFGS, and trust-region Newton methods. The authors conclude that the first order methods, steepest descent and conjugate gradient, are the preferred choices for problem (20.3) in terms of computation time. The benefit of fast convergence of Newton's method and BFGS is nullified by their high computational costs per iteration, especially as the size of the matrices increases. It is also empirically observed in [39] that the Riemannian metric yields much faster convergence for the tested algorithms compared with the induced Euclidean metric, which is given by $g_X(\eta_X, \xi_X) = \mathrm{trace}(\xi_X \eta_X)$.

It is known that a large condition number of the Hessian of the objective function slows down the first order optimization methods. Therefore, a recent paper [75] justifies the observations in [39] by analyzing the condition number of the Hessian in (20.3). Specifically, it is proven therein that in double precision arithmetic, the condition number of the Hessian of the objective function in (20.3) under the affine-invariance metric (20.2) is bounded above by a small positive number whereas the condition number of the Hessian under the Euclidean metric is bounded below by a potential large positive number, which linearly depends on the square of the condition number of the minimizer matrix μ. In addition, a limited-memory Riemannian BFGS method is proposed in [74] and empirically shown to be competitive with or superior to other state-of-the-art methods.

20.3.2 Riemannian Median (L^1 Riemannian Mean)

In the Euclidean space, it is known that the median is preferred to the mean in the presence of outliers due to the robustness of the former and the sensitivity of the latter. This is illustrated in Fig. 20.3, where the mean is dragged towards the outliers lying at the top right corner, while the median appears to be a better estimator of centrality. It is shown in [45] that half of the points must be corrupted in order to corrupt the median.

Given a set of points $\{a_1, \dots, a_K\} \in \mathbb{R}^n$, with the usual Euclidean distance $\|\cdot\|$, the geometric median is defined as the point $m \in \mathbb{R}^n$ minimizing the sum of distance

$$f(x) = \sum_{i=1}^{K} \|x - a_i\|.$$

The geometric median is not available in closed form in general, even for Euclidean points. The geometric median can be computed by an iterative algorithm introduced by Weiszfeld [71], which is essentially a Euclidean steepest descent. Later Ostresh [57] improved Weiszfeld's algorithm and proposed an update iteration with convergence result.

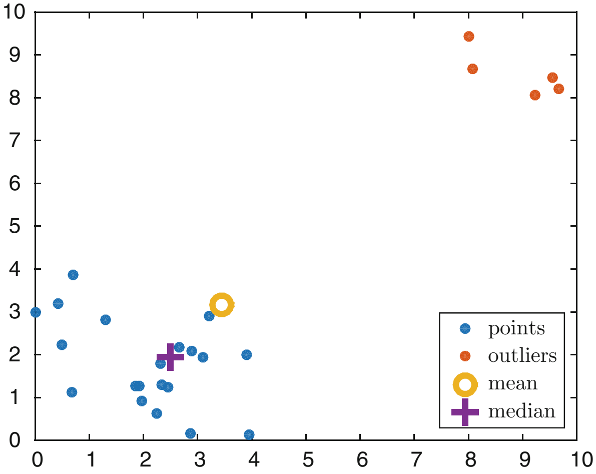


Fig. 20.3 The geometric mean and median in $\mathbb{R}^2$ space

This notion of the geometric median can be extended to the $\mathcal{S}_{++}^{\mathrm{n}}$ manifold. Given a set of SPD matrices $\{A_1, \ldots, A_K\}$, their Riemannian median is defined as the minimizer to the sum of distances

$$\mu_1 = \arg\min_{X \in \mathcal{S}_{++}^{\mathrm{n}}} \sum_{i=1}^{K} \delta(A_i, X), \tag{20.4}$$

where $\delta(\cdot,\cdot)$ is the geodesic distance. It was proven in [33] that the Riemannian median defined by (20.4) exists and is unique in the case of a non-positively curved manifold such as $\mathcal{S}_{++}^{\mathrm{n}}$ when all the data points A_i do not lie on the same geodesic. Note that the cost function in (20.4) is not differentiable at the data matrices, i.e., $X = A_i$ for $i = 1, \ldots, K$.

The computation of medians on $\mathcal{S}_{++}^{\mathrm{n}}$ has not received as much attention as the mean [23, 33, 73]. Fletcher et al. [33] generalized the Weiszfeld-Ostresh's algorithm to the Riemannian median computation on an arbitrary manifold, and proved that the algorithm converges to the unique solution when it exists. Charfi et al. [23] considered the computation of multiple averaging techniques, including the Riemannian median. A Euclidean steepest descent method and a fixed point algorithm are proposed. However, for the Euclidean steepest descent method, it is not guaranteed that each iterate stays on $\mathcal{S}_{++}^{\mathrm{n}}$. No stepsize selection rule is given for the steepest descent method. In [73], Yuan explores Riemannian optimization techniques, in particular smooth and nonsmooth Riemannian quasi-Newton based methods, to compute the Riemannian median, and empirically shows that the limited-memory Riemannian BFGS method is more robust and more efficient than the Riemannian Weiszfeld-Ostresh algorithm.

20.3.3 Riemannian Minimax Center (L^∞ Riemannian Mean)

Finding the unique smallest enclosing ball of a finite set of points in a Euclidean space is a fundamental problem in computational geometry and has been explored in e.g., [13, 14, 54, 66, 72]. This can be formulated as finding the minimizer of the cost function $f(x) = \max_{1\leq i\leq K} \|x - a_i\|$. Many data sets from machine learning, medical imaging, or computer vision consist of points on a nonlinear manifold [58, 68]. Therefore, finding the smallest enclosing ball of a collection of points on a manifold is of interest and has been studied in [11]. The center of the smallest enclosing ball is defined to be the L^∞ Riemannian center of mass or the minimax center.

Specifically, given a set of SPD matrices $\{A_1, \ldots, A_K\}$, the minimax center is defined as the point minimizing the maximum geodesic distance δ to the point set

$$\mu_\infty = \underset{X\in\mathcal{S}^{\mathrm{n}}_{++}}{\arg\min} \max_{1\leq i\leq K} \delta(A_i, X). \tag{20.5}$$

In general, there is no known closed form of the solution. In Euclidean space, a fast and simple iterative procedure for solving (20.5) has been proposed in [13]. The procedure is extended to arbitrary Riemannian manifolds in [11] with a study of the convergence rate. The existence and uniqueness of the minimax center defined in (20.5) have been studied in [3, 4, 11]. The SPD minimax center has been used in [9] to denoise tensor images.

The optimization problem in (20.5) is defined on the Riemannian manifold $\mathcal{S}^n_{++}$. Therefore, Riemannian optimization techniques are natural options for solving this problem. Unlike the cases of the Karcher mean and the median, the solution of (20.5) usually lies at a non-differentiable point. Therefore, one must utilize nonsmooth optimization techniques on Riemannian manifolds. In [73], Yuan uses the modified Riemannian BFGS method [37] and the subgradient-based Riemannian BFGS method [36] to solve the SPD minimax center problem more efficiently than the state-of-the-art method of Arnaudon and Nielsen [11].

20.4 Divergence-Based Averaging Techniques

The averaging techniques based on the geodesic distance provide an attractive approach to averaging a collection of SPD matrices since (1) the approach yields nice geometric interpretations of the optimization problems and (2) its L^2-based Riemannian mean (Karcher mean) satisfies all the desired geometric properties in the ALM list [8].

A divergence is similar to a distance and provides a measure of dissimilarity between two elements. However, in general, it need not satisfy symmetry or the triangle inequality. In recent years, matrix divergences have been of increasing interest due to their simplicity, efficiency and robustness to outliers, e.g., see [7, 10, 23, 27, 28, 56, 69, 70]. The idea of using divergences to define the mean of a collection of SPD matrices has been studied in the literature [24, 26, 50, 51, 64, 65].

20.4.1 Divergences

20.4.1.1 The α-Divergence Family

Let $\varphi : \Omega \to \mathbb{R}$ be a strictly convex and differentiable real-valued function defined on a convex set $\Omega \subset \mathbb{R}^m$. The α divergence family [76] is defined to be

$$\delta^2_{\varphi,\alpha}(x, y) = \frac{4}{1-\alpha^2}[\frac{1-\alpha}{2}\varphi(x) + \frac{1+\alpha}{2}\varphi(y) - \varphi(\frac{1-\alpha}{2}x + \frac{1+\alpha}{2}y)], \tag{20.6}$$

where $\alpha \in (-1, 1)$. The α-divergence possesses a dual symmetry with respect to the change $\alpha \to -\alpha$, i.e., $\delta_{\varphi,\alpha}(x, y) = \delta_{\varphi,-\alpha}(y, x)$.

For the values $\alpha = 1$ and $\alpha = -1$, the α-divergence is defined by taking the limit as $\alpha \to 1$ and $\alpha \to -1$, i.e.,

$$\delta^2_{\varphi,1}(x, y) = \varphi(x) - \varphi(y) - \langle \nabla\varphi(y), x - y\rangle \text{ and } \delta^2_{\varphi,-1}(x, y) = \delta^2_{\varphi,\mathrm{B}}(y, x). \tag{20.7}$$

Note that $\delta^2_{\phi,-1}(x, y)$ in (20.7) is actually the Bregman divergence defined in [22], denoted by $\delta^2_{\varphi,\mathrm{B}}(x, y)$.

Both the α-divergence (20.6) and the Bregman divergence (20.7) can be naturally extended to $\mathcal{S}^{\mathrm{n}}_{++}$, e.g., see [24, 50, 55]. Given a strictly convex (in the classical Euclidean sense) and differentiable real-valued function $\phi : \mathcal{S}^{\mathrm{n}}_{++} \to \mathbb{R}$ and $X, Y \in \mathcal{S}^{\mathrm{n}}_{++}$, the α-divergence with $-1 < \alpha < 1$ is defined as

$$\delta^2_{\phi,\alpha}(X, Y) = \frac{4}{1-\alpha^2}[\frac{1-\alpha}{2}\phi(X) + \frac{1+\alpha}{2}\phi(Y) - \phi(\frac{1-\alpha}{2}X + \frac{1+\alpha}{2}Y)]. \tag{20.8}$$

The Bregman divergence, $\delta^2_{\phi,\mathrm{B}}$ on $\mathcal{S}^n_{++}$, is therefore defined as

$$\delta^2_{\phi,\mathrm{B}}((X, Y) = \phi(X) - \phi(Y) - \langle \nabla\phi(Y), X - Y\rangle, \tag{20.9}$$

where $\langle X, Y\rangle = \mathrm{tr}(XY)$. Different choices of ϕ give different divergences. Commonly used convex functions on $\mathcal{S}^{\mathrm{n}}_{++}$ are [55]:

- quadratic entropy:

$$\phi(X) = \mathrm{tr}(X^T X), \tag{20.10}$$

- log-determinant (also called Burg) entropy:

$$\phi(X) = -\log\det X, \tag{20.11}$$

- von Neumann entropy:

$$\phi(X) = \operatorname{tr}(X \log X - X). \tag{20.12}$$

20.4.1.2 Symmetrized Divergence

A divergence is not symmetric in general. There are two common ways to symmetrize a divergence [28]:

- Type 1:

$$\delta^2_{\mathrm{S}\phi}(X, Y) = \frac{1}{2}(\delta^2_{\phi}(X, Y) + \delta^2_{\phi}(Y, X)), \tag{20.13}$$

- Type 2:

$$\delta^2_{\mathrm{S}\phi}(X, Y) = \frac{1}{2}(\delta^2_{\phi}(X, \frac{X+Y}{2}) + \delta^2_{\phi}(Y, \frac{X+Y}{2})). \tag{20.14}$$

20.4.1.3 The LogDet α-Divergence

When the associated function $\phi(X)$ in (20.8) is the log-determinant (LogDet) function (20.11), we get the LogDet α-divergence [24]:

$$\delta^2_{\mathrm{LD},\alpha}(X, Y) = \frac{4}{1-\alpha^2} \log \frac{\det(\frac{1-\alpha}{2}X + \frac{1+\alpha}{2}Y)}{[\det(X)]^{\frac{1-\alpha}{2}}[\det(Y)]^{\frac{1+\alpha}{2}}}, \text{ for } -1 < \alpha < 1. \tag{20.15}$$

The most frequently mentioned advantage of the LogDet α-divergence (20.15) compared to the geodesic distance δ is its computational efficiency. The computation of (20.15) requires three Cholesky factorizations (for $\frac{1-\alpha}{2}X + \frac{1+\alpha}{2}Y$, X, and Y), while computing the geodesic distance involves eigenvalue decomposition. In addition, the LogDet α-divergence enjoys several desired invariance properties [24]:

1. Invariance under congruence transformations

$$\delta^2_{\mathrm{LD},\alpha}(SAS^T, SBS^T) = \delta^2_{\mathrm{LD},\alpha}(A, B) \text{ for any invertible } S. \tag{20.16}$$

2. Dual-invariance under inversion

$$\delta^2_{\mathrm{LD},\alpha}(A^{-1}, B^{-1}) = \delta^2_{\mathrm{LD},-\alpha}(A, B). \tag{20.17}$$

3. Dual symmetry

$$\delta^2_{\mathrm{LD},\alpha}(A,B) = \delta^2_{\mathrm{LD},-\alpha}(B,A). \tag{20.18}$$

The LogDet α-divergence (20.15) is asymmetric except for $\alpha = 0$. But it can be symmetrized using (20.13) and (20.14), and the corresponding two symmetric forms of the LogDet α-divergence are

$$\delta^2_{\mathrm{S1LD},\alpha}(X,Y) = \frac{2}{1-\alpha^2}\log\frac{\det\left[(\frac{1-\alpha}{2}X + \frac{1+\alpha}{2}Y)(\frac{1-\alpha}{2}Y + \frac{1+\alpha}{2}X)\right]}{\det(XY)}, \tag{20.19}$$

and

$$\delta^2_{\mathrm{S2LD},\alpha}(X,Y) = \frac{2}{1-\alpha^2}\log\frac{\det\left[(\frac{3-\alpha}{4}X + \frac{1+\alpha}{4}Y)(\frac{3-\alpha}{4}Y + \frac{1+\alpha}{4}X)\right]}{[\det(XY)]^{\frac{1-\alpha}{2}}[\det(\frac{X+Y}{2})]^{1+\alpha}}. \tag{20.20}$$

The divergence $\delta^2_{\mathrm{LD},0}$ is also called the Stein divergence and is studied in [64, 65]. It is shown in [65] that $\delta^2_{\mathrm{LD},0}$ is the square of a distance function (i.e., $\delta_{\mathrm{LD},0}$ is a distance function in the sense that $\delta_{\mathrm{LD},0}$ is symmetric, nonnegative, definite, and satisfies the triangle inequality), and it shares several common geometric properties with the geodesic distance δ, such as P6 (congruence invariance) and P8 (inversion invariance) in the ALM properties, see [65, Table 4.1].

20.4.1.4 The LogDet Bregman Divergence

The LogDet Bregman divergence is defined using $\phi(X) = -\log\det X$, and is given by

$$\delta^2_{\mathrm{LD,B}}(X,Y) = \mathrm{tr}(Y^{-1}X - I) - \log\det(Y^{-1}X). \tag{20.21}$$

The LogDet Bregman divergence is also called the Kullback-Leibler divergence in [51]. It is easy to verify that the LogDet Bregman divergence is invariant under congruence transformations. In addition, the LogDet Bregman divergence is asymmetric. When it is symmetrized using (20.13) and (20.14), we have

$$\delta^2_{\mathrm{S1LD,B}}(X,Y) = \frac{1}{2}\mathrm{tr}(Y^{-1}X + X^{-1}Y - 2I), \tag{20.22}$$

and

$$\delta^2_{\mathrm{S2LD,B}}(X,Y) = \log\det(\frac{X+Y}{2}) - \frac{1}{2}\log\det(XY). \tag{20.23}$$

Notice that (20.23) coincides with the LogDet α-divergence with $\alpha = 0$. The Type 1 symmetrized LogDet Bregman divergence (20.22) is also called the Jeffrey divergence (or J-divergence) in [35, 70]. It is easily verified that both (20.22) and (20.23) are invariant under congruence and inversion.

20.4.1.5 The von Neumann α-Divergence

The von Neumann function $\phi(X) = \mathrm{tr}(X \log X - X)$ arises in quantum mechanics [53]. Its domain is the set of positive semidefinite matrices by using the convention that $0 \log 0 = 0$. The von Neumann α-divergence is defined as

$$\begin{aligned}\delta^2_{\mathrm{VN},\alpha}(X, Y) =& \frac{4}{1-\alpha^2}\,\mathrm{tr}\left\{\frac{1-\alpha}{2} X \log X + \frac{1+\alpha}{2} Y \log Y\right. \\ & \left. -(\frac{1-\alpha}{2} X + \frac{1+\alpha}{2} Y) \log(\frac{1-\alpha}{2} X + \frac{1+\alpha}{2} Y)\right\}. \qquad (20.24)\end{aligned}$$

From (20.24), we can verify that the von Neumann α-divergence satisfies the following invariance properties:

1. Invariance under rotations

$$\delta^2_{\mathrm{VN},\alpha}(OXO^T, OYO^T) = \delta^2_{\mathrm{VN},\alpha}(X, Y) \text{ for any } O \in \mathrm{SO}(n). \qquad (20.25)$$

2. Dual symmetry

$$\delta^2_{\mathrm{VN},\alpha}(X, Y) = \delta^2_{\mathrm{VN},-\alpha}(Y, X). \qquad (20.26)$$

It is clear from the dual symmetry that the von Neumann divergence is asymmetric except for $\alpha = 0$, which is given by

$$\delta^2_{\mathrm{VN},0}(X, Y) = 4\,\mathrm{tr}\{\frac{1}{2} X \log X + \frac{1}{2} Y \log Y - (\frac{X+Y}{2}) \log(\frac{X+Y}{2})\}. \qquad (20.27)$$

We note that the computation of the von Neumann α-divergence (20.24) requires three eigenvalue decompositions, which makes it more expensive than the computation of the geodesic distance δ, the LogDet α-divergence $\delta^2_{\mathrm{LD},\alpha}$, and the LogDet Bregman divergence $\delta^2_{\mathrm{LD,B}}$. Therefore, we neglect the sided means based on this divergence in Sect. 20.4.2.

20.4.1.6 The von Neumann Bregman Divergence

The von Neumann Bregman divergence [55], denoted by $\delta^2_{\mathrm{VN,B}}$, is defined using $\phi(X) = \mathrm{tr}(X \log X - X)$ for the Bregman divergence (20.9) and is given by

$$\delta^2_{\mathrm{VN,B}}(X, Y) = \mathrm{tr}(X(\log X - \log Y) - X + Y). \tag{20.28}$$

Note that (20.28) is referred to as the von Neumann divergence in [29, 40, 55] and the quantum relative entropy in [53]. The von Neumann Bregman divergence (20.28) is invariant under rotations, and its computation requires two eigenvalue decompositions. It is shown in [29] that (20.28) is finite if and only if the range of Y contains the range of X, i.e., $\mathrm{range}(X) \subseteq \mathrm{range}(Y)$. For this reason, the von Neumann Bregman divergence is often used in low-rank matrix nearness problems, e.g., see [29, 40, 41].

The von Neumann Bregman divergence is not symmetric, and its symmetrized versions are given by

$$\delta^2_{\mathrm{S1VN,B}}(X, Y) = \frac{1}{2}\,\mathrm{tr}(X(\log X - \log Y) + Y(\log Y - \log X)), \tag{20.29}$$

and

$$\delta^2_{\mathrm{S2VN,B}}(X, Y) = \mathrm{tr}(\frac{1}{2}X\log X + \frac{1}{2}Y\log Y - (\frac{X+Y}{2})\log(\frac{X+Y}{2})). \tag{20.30}$$

Note that (20.29) is finite if and only if $\mathrm{range}(X) = \mathrm{range}(Y)$. That is, the Type 1 symmetrized von Neumann Bregman divergence $\delta^2_{\mathrm{S1VN,B}}(X, Y)$ enjoys a range-space preserving property, which is important for the analysis of rank deficient matrices [40]. In addition, we note that the symmetrized von Neumann Bregman divergence (20.30) coincides with the von Neumann α-divergence with $\alpha = 0$, i.e., Eq. (20.27).

20.4.2 Left, Right, and Symmetrized Means Using Divergences

Given a divergence function on $\mathcal{S}^{\mathrm{n}}_{++}$, one can define the mean of a collection of SPD matrices $\{A_1, \ldots, A_K\}$ in a way similar to that used for the Karcher mean. Due to the asymmetry of divergence functions, the notion of right mean and left mean are used and coincide if the divergence is symmetric.

Definition 20.1 The right mean of a collection of SPD matrices $\{A_1, \ldots, A_K\}$ associated with divergence function $\delta^2_\phi(x, y)$ is defined as the minimizer of the sum of divergences

$$\mu^{\mathrm{r}} = \underset{X\in\mathcal{S}^{\mathrm{n}}_{++}}{\arg\min}\, f(X), \quad \text{with } f : \mathcal{S}^{\mathrm{n}}_{++} \to \mathbb{R},\; X \mapsto \sum_{i=1}^{K} \delta^2_\phi(A_i, X). \tag{20.31}$$

Definition 20.2 The left mean of a collection of SPD matrices $\{A_1, \ldots, A_K\}$ associated with divergence function $\delta_\phi^2(x, y)$ is defined as the minimizer of the sum of divergences

$$\mu^{\mathrm{l}} = \operatorname*{arg\,min}_{X \in \mathcal{S}_{++}^{\mathrm{n}}} f(X), \quad \text{with } f : \mathcal{S}_{++}^{\mathrm{n}} \to \mathbb{R},\ X \mapsto \sum_{i=1}^{K} \delta_\phi^2(X, A_i). \tag{20.32}$$

Definition 20.3 The symmetrized mean of a collection of SPD matrices $\{A_1, \ldots, A_K\}$ associated with divergence function $\delta_\phi^2(x, y)$ is defined as the minimizer of the sum of divergences

$$\mu^{\mathrm{s}} = \operatorname*{arg\,min}_{X \in \mathcal{S}_{++}^{\mathrm{n}}} f(X), \quad \text{with } f : \mathcal{S}_{++}^{\mathrm{n}} \to \mathbb{R},\ X \mapsto \sum_{i=1}^{K} \delta_{S\phi}^2(X, A_i). \tag{20.33}$$

where $\delta_{S\phi}^2$ is defined as (20.13) or (20.14). □

20.4.2.1 The LogDet α-Divergence

When δ_ϕ^2 is the LogDet α-divergence $\delta_{\mathrm{LD},\alpha}^2$, the optimization problems in Definitions 20.1, 20.2 and 20.3 have been studied in [24], where it is proved that the optimization problems have unique minimizers. Sra [65] analyzes the optimization problem for $\alpha = 0$, and proves that $\delta_{\mathrm{LD},0}^2$ is jointly geodesically convex under the affine-invariant metric $g_X(\xi, \eta) = \mathrm{tr}(\xi X^{-1} \eta X^{-1})$ where $\xi, \eta \in \mathrm{T}_X\, \mathcal{S}_{++}^{\mathrm{n}}$. In [73], Yuan extends the result and shows that $\delta_{\mathrm{LD},\alpha}^2$ is jointly geodesically convex for any $-1 < \alpha < 1$. Hence, any local minimum point is also a global minimum point.

A closed-form solution is unknown, except for $K = 2$. Unlike the Karcher mean computation that is extensively tackled by Riemannian optimization methods, the LogDet α-divergence based mean is often computed by fixed point algorithms, see [24, 55]. A Euclidean Newton's method is considered in [24] which, however, fails to converge in some numerical experiments. The special case of $\alpha = 0$ is studied in [24] and a fixed point algorithm to compute the divergence-based mean is given and its convergence investigated. This fixed point algorithm is applied to computing the divergence-based mean in [26, 27, 64, 65]. Yuan [73] studies solving the sided mean problem using Riemannian optimization algorithms and explains the fixed point algorithm in [24] in a Riemannian optimization framework. The Riemannian approaches, in particular the limited-memory Riemannian BFGS method, are shown to outperform other state-of-the-art methods for a wide range of problems.

20.4.2.2 The LogDet Bregman Divergence

Means based on the LogDet Bregman divergence have the following closed forms [51, Lemma 17.4.3]:

Lemma 20.4 ([51, Lemma 17.4.3]) *Let $\{A_1, \ldots, A_K\}$ be a collection of SPD matrices, let $\mathcal{A}(A_1, \ldots, A_K) = \frac{1}{K} \sum\limits_{i=1}^{K} A_i$ be their arithmetic mean, let $\mathcal{H}(A_1, \ldots, A_K) = K(\sum\limits_{i=1}^{K} A_i^{-1})^{-1}$ be their harmonic mean, and let $G(A, B)$ denote the geometric mean of A and B (20.1).*

1. The right mean based on $\delta^2_{\mathrm{LD,B}}$ (20.21) is given by the arithmetic mean, i.e.,

$$\mathcal{A}(A_1, \ldots, A_K) = \operatorname*{arg\,min}_{X \in \mathcal{S}^{\mathrm{n}}_{++}} \sum_{i=1}^{K} \delta^2_{\mathrm{LD,B}}(A_i, X). \tag{20.34}$$

2. The left mean based on $\delta^2_{\mathrm{LD,B}}$ (20.21) is given by the harmonic mean, i.e.,

$$\mathcal{H}(A_1, \ldots, A_K) = \operatorname*{arg\,min}_{X \in \mathcal{S}^{\mathrm{n}}_{++}} \sum_{i=1}^{K} \delta^2_{\mathrm{LD,B}}(X, A_i). \tag{20.35}$$

3. The symmetric mean based on $\delta^2_{\mathrm{S1LD,B}}$ (20.22) is given by the geometric mean of the arithmetic mean and the harmonic mean, i.e.,

$$G(\mathcal{A}(A_1, \ldots, A_K), \mathcal{H}(A_1, \ldots, A_K)) = \operatorname*{arg\,min}_{X \in \mathcal{S}^{\mathrm{n}}_{++}} \sum_{i=1}^{K} \delta^2_{\mathrm{S1LD,B}}(A_i, X). \tag{20.36}$$

20.4.2.3 The von Neumann Bregman Divergence

Given a collection of SPD matrices $\{A_1, \ldots, A_K\} \in \mathcal{S}^{\mathrm{n}}_{++}$, the right mean μ^r and left mean μ^l associated with the von Neumann Bregman divergence are given by, respectively,

$$\mu^{\mathrm{r}} = \operatorname*{arg\,min}_{X \in \mathcal{S}^{\mathrm{n}}_{++}} \delta^2_{\mathrm{VN,B}}(A_i, X) = \operatorname*{arg\,min}_{X \in \mathcal{S}^{\mathrm{n}}_{++}} \sum_{i=1}^{K} \mathrm{tr}(A_i \log A_i - A_i \log X - A_i + X) \tag{20.37}$$

and

$$\mu^{\mathrm{l}} = \operatorname*{arg\,min}_{X \in \mathcal{S}^{\mathrm{n}}_{++}} \delta^2_{\mathrm{VN,B}}(X, A_i) = \operatorname*{arg\,min}_{X \in \mathcal{S}^{\mathrm{n}}_{++}} \sum_{i=1}^{K} \mathrm{tr}(X \log X - X \log A_i - X + A_i). \tag{20.38}$$

In [73], it is pointed out that the left mean based on the von Neumann Bregman divergence has a closed form, which coincides with the Log-Euclidean Fréchet mean in [12]. A closed form of the right mean based on von Neumann Bregman divergence is not known. In addition, no efficient algorithm for computing the right mean currently exists since the closed form of the gradient of $\mathrm{tr}(A_i \log X)$ is not known.

20.4.3 Divergence-Based Median and Minimax Center

Similar to the geodesice-distance-based median and minimax center, one can define median and minimax center based on various types of divergences,

$$\text{right median: } \operatorname*{arg\,min}_{X \in \mathcal{S}_{++}^n} \sum_{i=1}^{K} \delta_{\phi,\alpha}(A_i, X), \tag{20.39}$$

$$\text{right minimax center: } \operatorname*{arg\,min}_{X \in \mathcal{S}_{++}^n} \max \delta_{\phi,\alpha}(A_i, X), \tag{20.40}$$

where $\delta_{\phi,\alpha}$ can be any of the divergences in Sect. 20.4.1. The left mean and left minimax center can be defined in a similar way.

In [23], Charfi et al. considered the computation of medians based not only on the geodesic distance, but also on Log-Euclidean distance and the Stein divergence. The Stein divergence median is also studied in [65], and a convergence proof of the fixed point iteration in [23] is given. A median based on the total Kullback-Leibler divergence is proposed in [69], which has a closed form expression. Yuan [73] reviews various types of the divergence-based medians and minimax centers and uses Riemannian optimization techniques to compute those based on the LogDet α-divergences. It is shown empirically that Riemannian optimization methods are usually more efficient than other state-of-the-art methods.

20.5 Alternative Metrics on SPD Matrices

Besides the geodesic distance and divergences, there exist other metrics to measure the similarity between two SPD matrices.

Log-Euclidean Metric

The Log-Euclidean metric proposed in [12] utilizes the observation that the matrix logarithm $\log\colon \mathcal{S}_{++}^n \to \mathbb{R}^{n\times n}$ is a one-to-one mapping. Therefore, the distance between two SPD matrices X, Y can be defined by

$$\delta_{\text{LogEuc}}(X, Y) = \| \log(X) - \log(Y)\|_F.$$

The Karcher mean defined by this distance has a closed form and coincides with the left mean based on the von Neumann Bregman divergence in Sect. 20.4.2.3.

Wasserstein Metric
The Wasserstein metric defines a general distance between arbitrary probability distributions on a general metric space. Note that the centered multivariate normal distribution $\mathcal{N}(0, X)$, $X \in \mathcal{S}_{++}^n$ is uniquely characterized by $X \in \mathcal{S}_{++}^n$. Therefore, when the Wasserstein metric is used to measure the distance between the multivariate normal distributions with zero mean, it defines a distance metric on $\mathcal{S}_{++}^n$, given by [46]

$$\delta_{\text{Wass}}(X, Y) = \left[\text{tr}(X) + \text{tr}(Y) - 2\,\text{tr}[(X^{\frac{1}{2}} Y X^{\frac{1}{2}})^{\frac{1}{2}}]\right]^{\frac{1}{2}}.$$

The Karcher mean (also called the barycenter) in the Wasserstein space is introduced in [5] and has been used to define the mean on the manifold of $\mathcal{S}_{++}^n$. A fixed point algorithm for computing the Karcher mean of a finite set of probabilities was proposed in [6], and used to find the Karcher mean of SPD matrices. The Wassertein distance can also be interpreted as the geodesic distance in the quotient geometry studied in [20, §4] and [47].

Affine Invariant Metric Family
The affine invariance metric family in $\mathcal{S}_{++}^n$ has been studied in [34] and the corresponding geodesic distance is given by

$$\delta_{\text{AIF}}(X, Y) = \left[\frac{\alpha}{4}\,\text{tr}((\log(X^{-1/2} Y X^{-1/2}))^2) + \frac{\beta}{4}(\text{tr}(\log(X^{-1/2} Y X^{-1/2})))^2\right]^{\frac{1}{2}},$$

where $\alpha > 0$ and $\beta > -\alpha/n$. The metric in (20.2) corresponds to $\alpha = 4$ and $\beta = 0$. In general, the relationship between the Karcher mean based on δ_{AIF}, the choice of parameters of α and β, and the ALM properties, is not fully understood.

Other Metrics
Other possibilities include the Bogoliubov-Kubo-Mori [49], the polar affine metric [78] and the broader class of the power Euclidean metrics [30], and the families of balanced metrics introduced in [67].

20.6 Conclusion

In this paper, we have briefly summarized the optimization problems of geodesic-distance-based and divergence-based mean, median and minimax center, and the existing optimization techniques. We have pointed out that the optimization problems in this paper can be nicely solved by Riemannian optimization techniques since the domain $\mathcal{S}_{++}^n$ is a well-studied smooth manifold.

References

1. Absil, P.-A., Mahony, R., Sepulchre, R.: Optimization Algorithms on Matrix Manifolds. Princeton University Press, Princeton (2008)
2. Absil, P.A., Gousenbourger, P.Y., Striewski, P., Wirth, B.: Differentiable piecewise-Bézier surfaces on Riemannian manifolds. SIAM J. Imag. Sci. **9**(4), 1788–1828 (2016)
3. Afsari, B.: Riemannian L^p center of mass: existence, uniqueness, and convexity. Proc. Am. Math. Soc. **139**(2), 655–673 (2011)
4. Afsari, B., Tron, R., Vidal, R.: On the convergence of gradient descent for finding the Riemannian center of mass. SIAM J. Control. Optim. **51**(3), 2230–2260 (2013)
5. Agueh, M., Carlier, G.: Barycenters in the Wasserstein space. SIAM J. Math. Anal. **43**(2), 904–924 (2011)
6. Alvarez-Esteban, P.C., Del Barrio, E., Cuesta-Albertos, J.A., Matran, C.: A fixed-point approach to barycenters in Wasserstein space. J. Math. Anal. Appl. **441**(2), 744–762 (2016)
7. Alyani, K., Congedo, M., Moakher, M.: Diagonality measures of Hermitian positive-definite matrices with application to the approximate joint diagonalization problem. Linear Algebra Appl. **528**(1), 290–320 (2017)
8. Ando, T., Li, C.K., Mathias, R.: Geometric means. Linear Algebra Appl. **385**, 305–334 (2004)
9. Angulo, J.: Structure tensor image filtering using Riemannian L_1 and L_∞ center-of-mass. Image Anal. Stereology **33**(2), 95–105 (2014)
10. Arandjelovic, O., Shakhnarovich, G., Fisher, J., Cipolla, R., Darrell, T.: Face recognition with image sets using manifold density divergence. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005, vol. 1, pp. 581–588. IEEE, Piscataway (2005)
11. Arnaudon, M., Nielsen, F.: On approximating the Riemannian 1-center. Comput. Geom. **46**(1), 93–104 (2013)
12. Arsigny, V., Fillard, P., Pennec, X., Ayache, N.: Log-Euclidean metrics for fast and simple calculus on diffusion tensors. Magn. Reson. Med. **56**(2), 411–421 (2006)
13. Badoiu, M., Clarkson, K.L.: Smaller core-sets for balls. In: Proceedings of Fourteenth ACM-SIAM Symposium on Discrete Algorithms (2003)
14. Badoiu, M., Clarkson, K.L.: Optimal core-sets for balls. Comput. Geom. **40**(1), 14–22 (2008)
15. Barbaresco, F.: Innovative tools for radar signal processing based on Cartan's geometry of SPD matrices and information geometry. In: Proceedings of IEEE Radar Conference, pp. 1–6 (2008)
16. Barmpoutis, A., Vemuri, B.C., Shepherd, T.M., Forder, J.R.: Tensor splines for interpolation and approximation of DT-MRI with applications to segmentation of isolated rat hippocampi. IEEE Trans. Med. Imaging **26**(11), 1537–1546 (2007)
17. Bhatia, R.: Positive Definite Matrices. Princeton University Press, Princeton (2007)
18. Bhatia, R., Holbrook, J.: Riemannian geometry and matrix geometric means. Linear Algebra Appl. **413**(2–3), 594–618 (2006)
19. Bhatia, R., Karandikar, R.L.: Monotonicity of the matrix geometric mean. Math. Ann. **353**(4), 1453–1467 (2012)
20. Bhatia, R., Jain, T., Lim, Y.: On the Bures Wasserstein distance between positive definite matrices. Expo. Math. **37**(2), 165–191 (2019)
21. Bini, D.A., Iannazzo, B.: Computing the Karcher mean of symmetric positive definite matrices. Linear Algebra Appl. **438**(4), 1700–1710 (2013)
22. Bregman, L.V.: The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. USSR Comput. Math. Math. Phys. **7**(3), 200–217 (1967)
23. Charfi, M., Chebbi, Z., Moakher, M., Vemuri, B.C.: Bhattacharyya median of symmetric positive-definite matrices and application to the denoising of diffusion-tensor fields. In: 2013 IEEE 10th International Symposium on Biomedical Imaging (ISBI), pp. 1227–1230. IEEE, Piscataway (2013)

24. Chebbi, Z., Moakher, M.: Means of Hermitian positive-definite matrices based on the log-determinant α-divergence function. Linear Algebra Appl. **436**(7), 1872–1889 (2012)
25. Cheng, G., Salehian, H., Vemuri, B.: Efficient recursive algorithms for computing the mean diffusion tensor and applications to DTI segmentation. In: Computer Vision–ECCV 2012, pp. 390–401 (2012)
26. Cherian, A., Sra, S., Banerjee, A., Papanikolopoulos, N.: Efficient similarity search for covariance matrices via the Jensen-Bregman LogDet divergence. In: 2011 IEEE International Conference on Proceedings of Computer Vision (ICCV), pp. 2399–2406. IEEE, Piscatawsay (2011)
27. Cherian, A., Sra, S., Banerjee, A., Papanikolopoulos, N.: Jensen-Bregman logdet divergence with application to efficient similarity search for covariance matrices. IEEE Trans. Pattern Anal. Mach. Intell. **35**(9), 2161–2174 (2013)
28. Cichocki, A., Cruces, S., Amari, S.-i.: Log-determinant divergences revisited: alpha-beta and gamma log-det divergences. Entropy **17**(5), 2988–3034 (2015)
29. Dhillon, I.S., Tropp, J.A.: Matrix nearness problems with Bregman divergences. SIAM J. Matrix Anal. Appl. **29**(4), 1120–1146 (2007)
30. Dryden, I.L., Pennec, X., Peyrat, J.M.: Power euclidean metrics for covariance matrices with application to diffusion tensor imaging (2010). arXiv:1009.3045v1
31. Faraut, J., Koranyi, A.: Analysis on Symmetric Cones. Oxford University Press, New York (1994)
32. Fletcher, P.T., Joshi, S.: Riemannian geometry for the statistical analysis of diffusion tensor data. Signal Process. **87**(2), 250–262 (2007)
33. Fletcher, P.T., Venkatasubramanian, S., Joshi, S.: The geometric median on Riemannian manifolds with application to robust atlas estimation. NeuroImage **45**(1), S143–S152 (2009)
34. Forstner, W., Moonen, B.: A metric for covariance matrices. In: Grafarend, E.W., Krum, F.W., Schwarze, V.S. (eds), Geodesy-The Challenge of the 3rd Millennium. Springer, Berlin (2003)
35. Harandi, M., Basirat, M., Lovell, B.C.: Coordinate coding on the Riemannian manifold of symmetric positive-definite matrices for image classification. In: Riemannian Computing in Computer Vision, pp. 345–361. Springer, Berlin (2016)
36. Hosseini, S., Huang, W., Yousefpour, R.: Line search algorithms for locally Lipschitz functions on Riemannian manifolds. SIAM J. Optim. **28**(1), 596–619 (2018)
37. Huang, W.: Optimization algorithms on Riemannian manifolds with applications. Ph.D. Thesis. Department of Mathematics, Florida State University (2014)
38. Iannazzo, B., Porcelli, M.: The Riemannian Barzilai–Borwein method with nonmonotone line search and the matrix geometric mean computation. IMA J. Numer. Anal. **38**(1), 495–517 (2018)
39. Jeuris, B., Vandebril, R., Vandereycken, B.: A survey and comparison of contemporary algorithms for computing the matrix geometric mean. Electron. Trans. Numer. Anal. **39**, 379–402 (2012)
40. Kulis, B., Sustik, M., Dhillon, I.: Learning low-rank kernel matrices. In: Proceedings of the 23rd International Conference on Machine Learning, pp. 505–512. ACM, New York (2006)
41. Kulis, B., Sustik, M.A., Dhillon, I.S.: Low-rank kernel learning with Bregman matrix divergences. J. Mach. Learn. Res. **10**, 341–376 (2009)
42. Lapuyade-Lahorgue, J., Barbaresco, F.: Radar detection using Siegel distance between autoregressive processes, application to HF and X-band radar. In: Proceedings of IEEE Radar Conference, pp. 1–6 (2008)
43. Lawson, J.D., Lim, Y.: The geometric mean, matrices, metrics, and more. Am. Math. Mon. **108**(108), 797–812 (2001)
44. Lawson, J., Lim, Y.: Monotonic properties of the least squares mean. Math. Ann. **351**(2), 267–279 (2011)
45. Lopuhaa, H.P., Rousseeuw, P.J.: Breakdown points of affine equivariant estimators of multivariate location and covariance matrices. Ann. Stat., 229–248 (1991)
46. Luigi, M., Montrucchio, L., Pistone, G.: Wasserstein riemannian geometry of Gaussian densities. Inf. Geo. **1**(2), 137–179 (2018)

47. Massart, E., Absil, P.-A.: Quotient geometry with simple geodesics for the manifold of fixed-rank positive-semidefinite matrices. Technical Report UCL-INMA-2018.06-v2, U.C. Louvain (2018)
48. Massart, E.M., Hendrickx, J.M., Absil, P.-A.: Matrix geometric means based on shuffled inductive sequences. Linear Algebra Appl. **542**, 334–359 (2018)
49. Michor, P.W., Petz, D., Andai, A.: On the curvature of a certain Riemannian space of matrices. Infin. Dimens. Anal. Quantum Probab. Relat. Top. **3**(02), 199–212 (2000)
50. Moakher, M.: On the averaging of symmetric positive-definite tensors. J. Elast. **82**(3), 273–296 (2006)
51. Moakher, M., Batchelor, P.G.: Symmetric positive-definite matrices: from geometry to applications and visualization. In: Visualization and Processing of Tensor Fields, pp. 285–298. Springer, Berlin (2006)
52. Nesterov, Y.: Introductory lectures on convex programming volume I: Basic course. Lect. Notes **87**, 236 (1998)
53. Nielsen, M.A., Chuang, I.: Quantum computation and quantum information. Cambridge University Press, second edition (2010)
54. Nielsen, F., Nock, R.: Approximating smallest enclosing balls with applications to machine learning. Int. J. Comput. Geom. Appl. **19**(05), 389–414 (2009)
55. Nielsen, F., Liu, M., Ye, X., Vemuri, B.C.: Jensen divergence based SPD matrix means and applications. In: Proceedings of 21st International Conference on Pattern Recognition (ICPR), pp. 2841–2844. IEEE, Piscataway (2012)
56. Nielsen, F., Nock, R.: Total Jensen divergences: definition, properties and clustering. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2016–2020. IEEE, Piscataway (2015)
57. Ostresh Jr, L.M.: On the convergence of a class of iterative methods for solving the Weber location problem. Oper. Res. **26**(4), 597–609 (1978)
58. Pennec, X.: Statistical Computing on Manifolds: From Riemannian Geometry to Computational Anatomy, pp. 347–386. Springer, Berlin (2009)
59. Pennec, X., Fillard, P., Ayache, N.: A Riemannian framework for tensor computing. Int. J. Comput. Vis. **66**(1), 41–66 (2006)
60. Rathi, Y., Tannenbaum, A., Michailovich, O.: Segmenting images on the tensor manifold. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007)
61. Rentmeesters, Q.: Algorithms for data fitting on some common homogeneous spaces. Ph.D. Thesis. Universite catholiqué de Louvain (2013)
62. Rentmeesters, Q., Absil, P.-A.: Algorithm comparison for Karcher mean computation of rotation matrices and diffusion tensors. In: 19th European Signal Processing Conference, pp. 2229–2233 (2011)
63. Said, S., Bombrun, L., Berthoumieu, Y., Manton, J.H.: Riemannian Gaussian distributions on the space of symmetric positive definite matrices. IEEE Trans. Inf. Theory **63**(4), 2153–2170 (2017)
64. Sra, S.: A new metric on the manifold of kernel matrices with application to matrix geometric means. In: Advances in Neural Information Processing Systems, pp. 144–152 (2012)
65. Sra, S.: Positive definite matrices and the S-divergence. Proc. Am. Math. Soc. **144**(7), 2787–2797 (2015)
66. Sylvester, J.J.: A question in the geometry of situation. Q. J. Math. **1**, 17 (1857)
67. Tsang, I.W., Kocsor, A., Kwok, J.T.: Exploration of Balanced Metrics on Symmetric Positive Definite Matrices. Geometric Science of Information, pp. 484–493. Springer, Cham
68. Turaga, P., Veeraraghavan, A., Srivastava, A., Chellappa, R.: Statistical computations on Grassmann and Stiefel manifolds for image and video-based recognition. IEEE Trans. Pattern Anal. Mach. Intell. **33**(11), 2273–86 (2011)
69. Vemuri, B.C., Liu, M., Amari, S.-I., Nielsen, F.: Total Bregman divergence and its applications to DTI analysis. IEEE Trans. Med. Imaging **30**(2), 475–483 (2011)

70. Wang, Z., Vemuri, B.C.: An affine invariant tensor dissimilarity measure and its applications to tensor-valued image segmentation. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004, vol. 1, pp. I–I. IEEE, Piscataway (2004)
71. Weiszfeld, E.: Sur le point pour lequel la somme des distances de n points donnés est minimum. Tohoku Math. J. First Series **43**, 355–386 (1937)
72. Welzl, E.: Smallest enclosing disks (balls and ellipsoids). In: New Results and New Trends in Computer Science (1991)
73. Yuan, X.: Riemannian optimization methods for averaging symmetric positive definite matrices. Ph.D. Thesis. Department of Mathematics, Florida State University (2018)
74. Yuan, X., Huang, W., Absil, P.-A., Gallivan, K.A.: A Riemannian limited-memory BFGS algorithm for computing the matrix geometric mean. Procedia Comput. Sci. **80**, 2147–2157 (2016)
75. Yuan, X., Huang, W., Absil, P.-A., Gallivan, K. A.: Computing the matrix geometric mean: Riemannian vs Euclidean conditioning, implementation techniques, and a Riemannian BFGS method. Technical Report UCL-INMA-2019.05, U.C. Louvain (2019). https://www.math.fsu.edu/~whuang2/papers/CMGM.htm
76. Zhang, J.: Divergence function, duality, and convex analysis. Neural Comput. **16**(1), 159–195 (2004)
77. Zhang, T.: A majorization-minimization algorithm for computing the Karcher mean of positive definite matrices. SIAM J. Matrix Anal. Appl. **38**(2), 387–400 (2017)
78. Zhang, Z., Su, J., Klassen, E., Le, H., Srivastava, A.: Rate-invariant analysis of covariance trajectories. J. Math. Imaging Vision **60**(8), 1306–1323 (2018)

Chapter 21
Rolling Maps and Nonlinear Data

Knut Hüper, Krzysztof A. Krakowski, and Fátima Silva Leite

Contents

K. Hüper (✉)
Institute of Mathematics, Julius-Maximilians-Universität Würzburg, Würzburg, Germany
e-mail: hueper@mathematik.uni-wuerzburg.de

K. A. Krakowski
Wydział Matematyczno-Przyrodniczy, Uniwersytet Kardynała Stefana Wyszyńskiego, Warsaw, Poland
e-mail: k.krakowski@uksw.edu.pl

F. Silva Leite
Department of Mathematics and Institute of Systems and Robotics, University of Coimbra, Coimbra, Portugal
e-mail: fleite@mat.uc.pt

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_21

Abstract In this chapter we study solutions to certain interpolation problems on Riemannian manifolds. Our methodology is based on rolling motions of those manifolds considered as rigid bodies, subject to holonomic as well as non-holonomic constraints. Although our approach is quite general, we focus our attention to three specific examples, namely spheres, Graßmannians and special orthogonal groups due to their importance in applications.

21.1 Introduction

Solving interpolation problems on Riemannian manifolds in an efficient way became an important research objective in recent years. Designing algorithms which can be easily implemented on a computer is of high interest in physics or astronomy, but mainly in engineering areas, such as robotics, computer vision, signal and image processing, machine learning, just to mention a few. Many well known approaches for solving interpolation problems on Euclidean spaces have been already extended to Riemannian manifolds. For a first example we mention the generalized de Casteljau algorithm, see e.g. [22, 58], which is theoretically appealing. However, although it leads to Bézier curves on curved spaces, the implementation of those algorithms often lacks efficiency as the curves are defined by highly nonlinear implicit equations. Another example includes variational methods of Hamiltonian or Lagrangian type to produce interpolation curves which are optimal in a certain sense. However, those curves are often solutions of high order nonlinear implicit differential equations. The approach we take here is based on an extrinsic point of view. The Riemannian manifold where the data to be interpolated is defined is embedded isometrically into a Euclidean space. As a consequence all (affine) tangent spaces are also embedded into the same space. In the sequel, rolling the manifold along one of its affine tangent spaces will play a prominent role. Rolling can be seen as a rigid motion under additional constraints. These side conditions are of holonomic as well as of non-holonomic type. The holonomic ones are specified by imposing that during the rolling motion the manifold together with that affine tangent space are kept tangent to each other along a prescribed curve on the manifold. The non-holonomic constraints are specified by imposing no-slip and no-twist. In other words the latter means that the manifold is neither allowed to spin nor to slide. These constraints are made mathematically precise in the next section. It turns out that the approach in this chapter is often very efficient, as for many cases the interpolating curves are given in closed form. The reader might argue that one does not get an isometric embedding for a given Riemannian manifold in a simple way. However, firstly, Nash's theorem, see e.g. [30], ensures the existence of such an isometric embedding in the finite dimensional setting, together with bounds on its dimension. Secondly, it seems fair to say that in many important cases, such as for instance for compact symmetric spaces, an isometric embedding is naturally associated with the given Riemannian manifold, see e.g. [46].

Many of the results presented here have already appeared in the literature. Nevertheless, we tried to keep the material in this chapter as selfcontained as possible.

21.2 Rolling Manifolds Along Affine Tangent Spaces

21.2.1 Mathematical Setting

Let us fix the mathematical background and terminology needed for our subsequent considerations and calculations. Denote by $\mathcal{M}$ a finite dimensional Riemannian manifold and by $T_p^{\mathrm{aff}}\mathcal{M}$ its affine tangent space at $p \in \mathcal{M}$, both embedded into some fixed Euclidean vector space V. Moreover, consider a finite dimensional Lie group $\mathcal{G}$. We will make the following assumptions.

Assumption *The group $\mathcal{G}$ acts transitively and isometrically on $\mathcal{M}$. The action $\sigma \colon \mathcal{G} \times \mathcal{M} \to \mathcal{M}$ can be linearly extended to an isometric action on V.* □

As a subgroup of the special Euclidean group $SE(V)$, i.e., rotations and translations of the vector space V, we define another group $SE(V) \supset \widehat{\mathcal{G}} := \mathcal{G} \ltimes V$ (semidirect product) acting isometrically on V via

$$\widehat{\sigma} \colon \widehat{\mathcal{G}} \times V \to V, \quad \big((g, s), v\big) \mapsto gv + s. \tag{21.1}$$

Clearly, group multiplication in $\widehat{\mathcal{G}}$ is given by $(g_1, s_1) \cdot (g_2, s_2) = (g_1 g_2, g_1 s_2 + s_1)$ with inverse $(g, s)^{-1} = \big(g^{-1}, -g^{-1}s\big)$. To fix notation, we define as usual:

$GL_n := \big\{X \in \mathbb{R}^{n\times n} \,\vert\, \det(X) \neq 0\big\}$,	general linear group,
$SO_n := \big\{X \in GL_n \vert X^\top X = I, \det(X) = 1\big\}$,	special orthogonal group,
$\mathfrak{gl}_n := \mathbb{R}^{n\times n}$,	Lie algebra of GL_n,
$\mathfrak{so}_n := \big\{X \in \mathfrak{gl}_n \vert X = -X^\top\big\}$,	Lie algebra of SO_n,
$Sym_n := \big\{X \in \mathfrak{gl}_n \vert X = X^\top\big\}$,	vector space of symmetric matrices,
$[X, Y] := XY - YX, \text{ for all } X, Y \in \mathfrak{gl}_n$,	Lie bracket or commutator,
$\mathrm{ad}_X \colon \mathfrak{gl}_n \to \mathfrak{gl}_n, \; Y \mapsto [X, Y]$,	ad-operator.

We proceed with three well known examples important for our considerations.

Example The Euclidean sphere:

$\mathcal{M} := S^{n-1} \subset \mathbb{R}^n =: V$ with isometry group $\mathcal{G} := SO_n$ and transitive action $\sigma\colon \mathcal{G} \times S^{n-1} \to S^{n-1}$ defined by $(R, p) \mapsto Rp$ and $\widehat{\mathcal{G}} := \mathcal{G} \ltimes \mathbb{R}^n = SE_n$. The embedding space V is equipped with the usual Euclidean inner product $\langle v, w\rangle_{\mathbb{R}^n} := v^\top w$. We have the tangent space at $p \in S^{n-1}$

$$T_p S^{n-1} = (I_n - pp^\top)\mathbb{R}^n = \{w \in \mathbb{R}^n | w \perp p\}, \tag{21.2}$$

with $T_p^{\text{aff}} S^{n-1} = p + T_p S^{n-1}$, and the normal space

$$T_p^\perp S^{n-1} = pp^\top \mathbb{R}^n = \mathbb{R}p. \tag{21.3}$$

Corresponding orthogonal projection operators are

$$\begin{aligned} \Pi_{T_p S^{n-1}}&\colon \mathbb{R}^n \to \mathbb{R}^n,\ x \mapsto (I - px^\top)p = x - px^\top p, \\ \Pi_{T_p^\perp S^{n-1}}&\colon \mathbb{R}^n \to \mathbb{R}^n,\ x \mapsto px^\top p. \end{aligned} \tag{21.4}$$

Example The Graßmannian:

$\mathcal{M} := Gr_{n,k}$ is defined to be the set of all proper k-dimensional subspaces of $\mathbb{R}^n$, considered here as the set of rank k orthogonal projection operators on $\mathbb{R}^n$. Take

$$V := Sym_n, \quad \dim V = n(n+1)/2, \tag{21.5}$$

equipped with the Frobenius inner product induced from $\mathbb{R}^{n\times n}$, namely $\langle S, T\rangle_{Sym_n} := \operatorname{tr}(S^\top T) = \operatorname{tr}(ST)$. Then we have

$$Gr_{n,k} := \{P \in Sym_n | P^2 = P, \operatorname{tr} P = k\}, \quad \dim Gr_{n,k} = k(n-k). \tag{21.6}$$

For later purposes we introduce three proper subspaces of $\mathfrak{gl}_n$, Sym_n, and $\mathfrak{so}_n$, respectively. See [8] for a slightly different notation. For fixed $P \in Gr_{n,k}$ we define

$$\begin{aligned} gl_P &:= \big\{M \in \mathfrak{gl}_n | M = PM + MP\big\}, \\ sym_P &:= Sym_n \cap gl_P, \\ so_P &:= \mathfrak{so}_n \cap gl_P. \end{aligned} \tag{21.7}$$

There exist several useful relations and properties for the above subspaces proved in [8], some of them are listed here.

Lemma 21.1 *Let $P \in Gr_{n,k}$ and $M \in gl_P$ then*

1. $\operatorname{ad}_P^2 M = M$,
2. *the restriction* $\operatorname{ad}_P |_{gl_P}$ *is a global isometry,*
3. $\operatorname{ad}_P sym_P = so_P$.

The special orthogonal group acts transitively and isometrically via conjugation

$$\sigma\colon SO_n \times Gr_{n,k} \to Gr_{n,k}, \quad (R, P) \mapsto RPR^\top. \tag{21.8}$$

Here $\widehat{\mathcal{G}} = SO_n \ltimes Sym_n \subset SE_{n(n+1)/2}$.

An obvious corollary of the definitions in (21.7) is:

Corollary 21.2 *Let $P \in Gr_{n,k}$ and $\Theta \in SO_n$ then*

$$gl_{\Theta P \Theta^\top} = \Theta gl_P \Theta^\top, \quad sym_{\Theta P \Theta^\top} = \Theta sym_P \Theta^\top, \quad so_{\Theta P \Theta^\top} = \Theta so_P \Theta^\top. \tag{21.9}$$

We will use several equivalent descriptions of the tangent space, see [8, 35], namely

$$\begin{aligned} T_P Gr_{n,k} &= \mathrm{ad}_P^2 Sym_n = \{S \in Sym_n | SP + PS = S\} \\ &= sym_P = \mathrm{ad}_P \mathfrak{so}_n = \mathrm{ad}_P\, so_P, \end{aligned} \tag{21.10}$$

with $T_P^{\mathrm{aff}} Gr_{n,k} = P + T_P Gr_{n,k}$, and for the normal space

$$T_P^\perp Gr_{n,k} = (\mathrm{id} - \mathrm{ad}_P^2) Sym_n. \tag{21.11}$$

The related orthogonal projection operators are as

$$\begin{aligned} \Pi_{T_P Gr_{n,k}}\colon Sym_n \to Sym_n, &\quad S \mapsto \mathrm{ad}_P^2\, S = [P, [P, S]], \\ \Pi_{T_P^\perp Gr_{n,k}}\colon Sym_n \to Sym_n, &\quad S \mapsto (\mathrm{id} - \mathrm{ad}_P^2) S = S - [P, [P, S]]. \end{aligned} \tag{21.12}$$

The following characterizations will turn out to be useful.

Lemma 21.3 *We have*

1. $\left[T_P Gr_{n,k}, so_p\right] \subset T_P^\perp Gr_{n,k}$,
2. $\mathrm{ad}_P \left[T_P Gr_{n,k}, T_P Gr_{n,k}\right] = 0$.

Proof We exploit the invariance properties, see Corollary 21.2, i.e., it is sufficient to show the statements at the standard projector $P_0 := \left[\begin{smallmatrix} I & 0 \\ 0 & 0 \end{smallmatrix}\right]$.

For the first statement consider $A \in T_{P_0} Gr_{n,k}$ and $B \in so_{P_0}$. They are then of the form $A = \left[\begin{smallmatrix} 0 & C \\ C^\top & 0 \end{smallmatrix}\right]$ and $B = \left[\begin{smallmatrix} 0 & D \\ -D^\top & 0 \end{smallmatrix}\right]$. Their commutator $[A, B] = \left[\begin{smallmatrix} -CD^\top - DC^\top & 0 \\ 0 & C^\top D + D^\top C \end{smallmatrix}\right]$ then lies in the kernel of $\mathrm{ad}_{P_0}^2$.

For the second statement consider the same A and another $E = \left[\begin{smallmatrix} 0 & F \\ F^\top & 0 \end{smallmatrix}\right] \in T_{P_0} Gr_{n,k}$. It follows that their commutator being equal to $\left[\begin{smallmatrix} CF^\top - FC^\top & 0 \\ 0 & C^\top F - F^\top C \end{smallmatrix}\right]$ lies in the kernel of ad_{P_0}. □

Remark 21.4 Note that we cannot replace the inclusion in Lemma 21.3 by equality, simply because elements in $T_P^{\perp} Gr_{n,k}$ do not necessarily have zero trace, whereas in $\left[T_P Gr_{n,k}, so_p\right]$ all elements are traceless.

Example The special orthogonal group:
$\mathcal{M} := SO_n \subset \mathbb{R}^{n\times n} =: V$ with Euclidean metric $\langle X, Y\rangle_{\mathbb{R}^{n\times n}} := \operatorname{tr}(X^\top Y)$. Again we have several equivalent descriptions of the tangent space at $\Theta \in SO_n$, namely

$$T_\Theta SO(n) = \Theta \cdot \mathfrak{so}_n = \mathfrak{so}_n \cdot \Theta = \frac{\operatorname{id} - \Theta(\cdot)^\top \Theta}{2} \cdot \mathbb{R}^{n\times n}, \tag{21.13}$$

with $T_\Theta^{\text{aff}} SO(n) = \Theta + T_\Theta SO(n)$, and normal space

$$T_\Theta^{\perp} SO(n) = Sym_n \cdot \Theta = \Theta \cdot Sym_n = \frac{\operatorname{id} + \Theta(\cdot)^\top \Theta}{2} \cdot \mathbb{R}^{n\times n}. \tag{21.14}$$

Consequently, the corresponding orthogonal projection operators are

$$\begin{aligned} \Pi_{T_\Theta SO_n} &: \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}, \quad X \mapsto \frac{X - \Theta X^\top \Theta}{2}, \\ \Pi_{T_\Theta^{\perp} SO_n} &: \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}, \quad X \mapsto \frac{X + \Theta X^\top \Theta}{2}. \end{aligned} \tag{21.15}$$

The product of two copies of the special orthogonal group acts transitively and isometrically on itself via left-right multiplication

$$\sigma : (SO_n \times SO_n) \times SO_n \to SO_n, \quad ((R_1, R_2), \Theta) \mapsto R_1 \Theta R_2^\top. \tag{21.16}$$

The extended group reads $\widehat{\mathcal{G}} = (SO_n \times SO_n) \ltimes \mathbb{R}^{n\times n} \subset SE_{n^2}$. □

21.2.2 Rolling Manifolds

Here we first fix some notation and further terminology for the analysis we use later on. In all the above cases we have the isometric action of the corresponding $\widehat{\mathcal{G}}$ on V, i.e.,

$$\widehat{\sigma} : \widehat{\mathcal{G}} \times V \to V, \quad (\widehat{g}, v) \mapsto \widehat{g} \circ v =: \widehat{g} v \tag{21.17}$$

and the closely related map

$$\widehat{\sigma}_{\widehat{g}} : V \to V, \quad v \mapsto \widehat{g} v. \tag{21.18}$$

We compute the tangent map of (21.18), or differential for short, as

$$\begin{aligned} \mathrm{d}\widehat{\sigma}_{\widehat{g}}(v)\colon T_v V \cong V \to T_{\widehat{g}v} V \cong V, \\ w \mapsto \mathrm{d}\widehat{\sigma}_{\widehat{g}}(v)w := \mathrm{d}_v(\widehat{\sigma}_{\widehat{g}})w := \tfrac{\mathrm{d}}{\mathrm{d}\epsilon}\widehat{\sigma}_{\widehat{g}}(v+\epsilon w)\Big|_{\epsilon=0}. \end{aligned} \tag{21.19}$$

In the sequel we will be interested in restricting smooth vector fields v defined on the embedding V to $\mathcal{M}$. In particular, we explicitly need an additive decomposition into tangent and normal part of the derivative of v. By the extrinsic view we adopt in this paper, i.e., embedding the Riemannian manifold $\mathcal{M}$ isometrically into some Euclidean space V, covariant derivatives of vector fields on $\mathcal{M}$ get a particular simple meaning. The covariant derivative of the vector field v evaluated at $p \in \mathcal{M}$ is defined to be the orthogonal projection of the ordinary derivative of v evaluated at p into the tangent space $T_p\mathcal{M}$. The formulas (21.4), (21.12), (21.15) are now particularly useful. Consequently, using standard notation we get for the covariant derivative of v evaluated at p

$$\frac{\mathrm{D}}{\mathrm{d}t}v(t)|_{t=0} := \Pi_{T_{\alpha(0)}\mathcal{M}}\frac{\mathrm{d}}{\mathrm{d}t}(v\circ\alpha)(t)|_{t=0} \tag{21.20}$$

with an arbitrary smooth curve $\alpha\colon (-\epsilon,\epsilon) \to \mathcal{M}$ and $\alpha(0) = p$. Analogously,

$$\frac{\mathrm{D}^{\perp}}{\mathrm{d}t}v(t)|_{t=0} := \Pi_{T^{\perp}_{\alpha(0)}\mathcal{M}}\frac{\mathrm{d}}{\mathrm{d}t}(v\circ\alpha)(t)|_{t=0}. \tag{21.21}$$

Definition 21.5 Let $J := [0,\tau]$ denote a nonzero interval. A smooth mapping

$$\chi\colon J \to \widehat{\mathcal{G}}, \quad t \mapsto \chi(t) := \big(R(t), v(t)\big) \tag{21.22}$$

satisfying for all $t \in J$ the three properties below, is called a rolling of $\mathcal{M}$ on $T_p^{\mathrm{aff}}\mathcal{M}$ (or simply a rolling map) without slipping and without twisting.

1. **Rolling conditions:** There exists a smooth curve $\alpha\colon J \to \mathcal{M}$ with $p = \alpha(0)$ such that

   a. $\beta(t) := \chi(t)\alpha(t) \in T_p^{\mathrm{aff}}\mathcal{M}$,
   b. $\mathrm{d}_{\alpha(t)}\chi(t)T_{\alpha(t)}\mathcal{M} = T_{\beta(t)}(T_p^{\mathrm{aff}}\mathcal{M}) \cong T_p\mathcal{M}$.

   The curve α is called the rolling curve, whereas β is the development curve of α.
2. **No-slip condition:**

$$\dot{\beta}(t) = \mathrm{d}_{\alpha(t)}\chi(t)\dot{\alpha}(t). \tag{21.23}$$

3. **No-twist conditions:**

   a. **Tangential part:** For any *tangent* vector field $Z(t)$ along $\alpha(t)$

$$\mathrm{d}_{\alpha(t)}\chi(t)\tfrac{\mathrm{D}}{\mathrm{d}t}Z(t) = \tfrac{\mathrm{D}}{\mathrm{d}t}\mathrm{d}_{\alpha(t)}\chi(t)Z(t). \tag{21.24}$$

b. **Normal part:** For any *normal* vector field $Z(t)$ along $\alpha(t)$

$$\mathrm{d}_{\alpha(t)}\chi(t)\tfrac{\mathrm{D}^{\perp}}{\mathrm{d}t}Z(t) = \tfrac{\mathrm{D}^{\perp}}{\mathrm{d}t}\mathrm{d}_{\alpha(t)}\chi(t)Z(t). \tag{21.25}$$

At this stage some remarks are in order.

Remark 21.6 Definition 21.5 is equivalent to Sharpe's, see [60]. For a proof of this equivalence we refer to [28]. Moreover, in [60] it is shown that for a given smooth curve α in $\mathcal{M}$, with $\alpha(0) = p$, there exists a unique rolling of $\mathcal{M}$ on $T_p^{\mathrm{aff}}\mathcal{M}$.

Remark 21.7 In Definition 21.5, conditions 2.–3., smoothness of α can be relaxed to piecewise smoothness, simply by replacing *for all* $t \in J$ essentially by *for almost all* $t \in J$.

Remark 21.8 From the no-twist conditions it follows that a tangent (resp. normal) vector field Z along the rolling curve α is parallel iff $\mathrm{d}_{\alpha(t)}\chi(t)Z(t)$ is a tangent (resp. normal) parallel vector field along the development curve β.

Remark 21.9 As we roll $\mathcal{M}$ along an affine tangent space, the covariant derivatives on the right hand sides in (21.24) and (21.25) are equal to ordinary derivatives as the corresponding vector fields live in a linear space.

We now revisit the above three examples presenting for each one explicitly the rolling map associated to a given curve α.

Rolling S^{n-1}

Any smooth curve $\alpha\colon J \to S^{n-1}$ with $\alpha(0) = p$ is of the form $\alpha(t) = R(t)p$ with smooth $R\colon J \to SO_n$ satisfying $R(0) = I$.

Theorem 21.10 *Consider the unique solution $\big(R(t), v(t)\big)$ of the initial value problem*

$$\begin{aligned} \dot{R}(t) &= R(t)\big(u(t)p^{\top} - pu^{\top}(t)\big), & R(t) &\in SO_n,\ R(0) = I, \\ \dot{v}(t) &= u(t), & u(t) &\in T_pS^{n-1},\ v(0) = 0. \end{aligned} \tag{21.26}$$

Then $\chi(t) = \big(R^{\top}(t), v(t)\big) \in SE_n$ is the unique rolling map of S^{n-1} on $T_p^{\mathrm{aff}}S^{n-1}$ along the smooth curve $\alpha(t) = R(t)p$. Equations (21.26) are called the kinematic equations of S^{n-1}.

Proof Obviously, $\alpha(t) \in S^{n-1}$. Moreover,

$$\beta(t) = \chi(t)\alpha(t) = \big(R^{\top}(t), v(t)\big)\circ\alpha(t) = R^{\top}(t)\alpha(t)+v(t) = p+v(t). \tag{21.27}$$

By the initial condition $v(0) = 0$ together with $u(t) \in T_pS^{n-1}$ it is ensured that $v(t) \in T_pS^{n-1}$. Hence $\beta(t) \in T_p^{\mathrm{aff}}S^{n-1}$. The second rolling condition is fulfilled as well. Indeed,

$$\mathrm{d}_{\alpha(t)}\chi(t)\big(T_{\alpha(t)}S^{n-1}\big) = R^{\top}(t)\big(T_{R(t)p}S^{n-1}\big) = T_pS^{n-1}. \tag{21.28}$$

We now check the no-slip condition. From (21.26) and (21.27) we get

$$\dot{\beta}(t) = \dot{v}(t) = u(t). \tag{21.29}$$

On the other hand, using (21.28) and (21.26), we get

$$\begin{aligned} \mathrm{d}_{\alpha(t)}\chi(t)\dot{\alpha}(t) &= R^\top(t)\dot{\alpha}(t) = R^\top(t)\dot{R}(t)p \\ &= \big(u(t)p^\top - pu^\top(t)\big)p = u(t) \end{aligned} \tag{21.30}$$

as required.

It remains to check the no-twist conditions. For the tangent part let $z(t)$ be any *tangent* vector field along $\alpha(t)$. We first evaluate the left hand side (LHS) of (21.24). In fact, by ommitting t-dependence for convenience

$$\begin{aligned} \mathrm{d}_{\alpha(t)}\chi(t)\big(\tfrac{\mathrm{D}}{\mathrm{d}t}z(t)\big) &= R^\top \tfrac{\mathrm{D}}{\mathrm{d}t}z = R^\top\big(I - \alpha\alpha^\top\big)\dot{z} \\ &= R^\top\dot{z} - pp^\top R^\top\dot{z} = R^\top\dot{z} - p\langle p, R^\top\dot{z}\rangle. \end{aligned} \tag{21.31}$$

Correspondingly for the RHS of (21.24)

$$\begin{aligned} \tfrac{\mathrm{D}}{\mathrm{d}t}\big(\mathrm{d}_{\alpha(t)}\chi(t)z(t)\big) &= \tfrac{\mathrm{d}}{\mathrm{d}t}(R^\top z) = \dot{R}^\top z + R^\top\dot{z} \\ &= \big(pu^\top - up^\top\big)R^\top z + R^\top\dot{z} = p\langle u, R^\top z\rangle + R^\top\dot{z}. \end{aligned} \tag{21.32}$$

By requiring (21.31) being equal to (21.32), it remains to show that $-p\langle p, R^\top\dot{z}\rangle = p\langle u, R^\top z\rangle$. This is correct as $\langle\alpha, z\rangle = 0$ implies $\langle\dot{\alpha}, z\rangle = -\langle\alpha, \dot{z}\rangle$, and together with the definition $\alpha = Rp$ and imposing (21.26) the result follows.

For the normal part of the no-twist condition, we proceed as follows. Let $z(t)$ be now any *normal* vector field along $\alpha(t)$. Hence, we know that such a z is always of the form $z(t) = \gamma(t)\alpha(t)$ with some suitable smooth scalar valued function γ. For the derivative we get $\dot{z} = \dot{\gamma}\alpha + \gamma\dot{\alpha} = \dot{\gamma}\alpha + \gamma\dot{R}p = \dot{\gamma}\alpha + \gamma Ru$. Evaluating the LHS of (21.25) gives

$$\begin{aligned} \mathrm{d}_{\alpha(t)}\chi(t)\tfrac{\mathrm{D}^\perp}{\mathrm{d}t}z(t) &= R^\top\tfrac{\mathrm{D}^\perp}{\mathrm{d}t}z = R^\top\alpha\alpha^\top\dot{z} = p\alpha^\top\dot{z} \\ &= p(\dot{\gamma} + \gamma\underbrace{\alpha^\top Ru}_{=0}) = p\dot{\gamma}. \end{aligned} \tag{21.33}$$

Correspondingly, for the RHS of (21.24)

$$\begin{aligned} \tfrac{\mathrm{D}^\perp}{\mathrm{d}t}\big(\mathrm{d}_{\alpha(t)}\chi(t)z(t)\big) &= \tfrac{\mathrm{d}}{\mathrm{d}t}(R^\top z) = \dot{R}^\top z + R^\top\dot{z} = \big(pu^\top - up^\top\big)R^\top z + R^\top\dot{z} \\ &= \big(pu^\top - up^\top\big)p\gamma + R^\top(\dot{\gamma}\alpha + \gamma Ru) = -u\gamma + \dot{\gamma}p + \gamma u \\ &= p\dot{\gamma} \end{aligned} \tag{21.34}$$

as required. □

The kinematic equations (21.26) form a differential system on the Lie group SE_n. They are in accordance with [42], p. 467.

Corollary 21.11 *If $u(t) = u$ is constant, the kinematic equations* (21.26) *can be solved explicitly to obtain $R(t) = \mathrm{e}^{t\Omega}$, where $\Omega := up^\top - pu^\top$, and $v(t) = tu$. In this case, the rolling curve $\alpha(t) = \mathrm{e}^{t\Omega}\, p$ is a geodesic.*

Proof The first statement is an immediate consequence of Theorem 21.10. For the second statement it is enough to show that for $\alpha(t) = \mathrm{e}^{t\Omega}\, p$, we get $\frac{\mathrm{D}}{\mathrm{d}t}\dot{\alpha} = \ddot{\alpha} - \langle \alpha, \ddot{\alpha}\rangle\alpha = 0$. But this follows from the following computations: $\dot{\alpha}(t) = \mathrm{e}^{t\Omega}\, u$, $\ddot{\alpha}(t) = -\,\mathrm{e}^{t\Omega}\, p\langle u, u\rangle$, $\langle \alpha(t), \ddot{\alpha}(t)\rangle = -\langle u, u\rangle$. So, $\ddot{\alpha} - \langle \alpha, \ddot{\alpha}\rangle\alpha = \mathrm{e}^{t\Omega}(-\langle u, u\rangle p + \langle u, u\rangle p) = 0$. □

Rolling $Gr_{n,k}$

Any smooth curve $\alpha\colon J \to Gr_{n,k}$ with $\alpha(0) = P$ is of the form $\alpha(t) = R(t)PR^\top(t)$ with smooth $R\colon J \to SO_n$ satisfying $R(0) = I$.

Theorem 21.12 *Consider the unique solution $\big(R(t), S(t)\big)$ of the initial value problem*

$$\begin{aligned} &\dot{R}(t) = R(t)[U(t), P], \quad && R(t) \in SO_n,\ \ R(0) = I,\\ &\dot{S}(t) = U(t), && U(t) \in T_p Gr_{n,k},\ \ S(t) \in Sym_n,\ \ S(0) = 0. \end{aligned} \tag{21.35}$$

Then $\chi(t) = \big(R^\top(t), S(t)\big) \in SE_{n(n+1)/2}$ is the rolling map of $Gr_{n,k}$ on $T_P^{\mathrm{aff}}Gr_{n,k}$ along the smooth curve $\alpha(t) = R(t)PR^\top(t)$. Equations (21.35) are called the kinematic equations of $Gr_{n,k}$.

Proof Obviously, $\alpha(t) \in Gr_{n,k}$. Moreover,

$$\beta(t) = \chi(t)\alpha(t) = R^\top(t)\alpha(t)R(t) + S(t) = P + S(t). \tag{21.36}$$

By the initial condition $S(0) = 0$ together with $U(t) \in T_P Gr_{n,k}$ we have $S(t) \in T_p Gr_{n,k}$. Then $\beta(t) \in T_P^{\mathrm{aff}} Gr_{n,k}$. Similar to the S^{n-1} case the second rolling condition is fulfilled as well. Indeed,

$$\mathrm{d}_\alpha \chi\big(T_\alpha Gr_{n,k}\big) = R^\top(T_{RPR^\top} Gr_{n,k})R = T_P Gr_{n,k}. \tag{21.37}$$

We now check the no-slip condition. From (21.35) and (21.36) we get

$$\dot{\beta}(t) = \dot{S}(t) = U(t). \tag{21.38}$$

On the other hand, using (21.37) we get

$$\begin{aligned} \mathrm{d}_{\alpha(t)}\chi(t)\dot{\alpha}(t) = R^\top\dot{\alpha}R = R^\top(\dot{R}PR^\top + RP\dot{R}^\top)R = R^\top\dot{R}P + P\dot{R}^\top R\\ = [U, P]P + P[U, P]^\top = \operatorname{ad}_P^2 U = U \end{aligned} \tag{21.39}$$

as required.

It remains to check the no-twist conditions. For the tangent part, let $Z(t)$ be any tangent vector field along $\alpha(t)$. I.e., $Z \in T_\alpha Gr_{n,k}$ implying $R^\top Z R \in T_P Gr_{n,k}$. In particular, there exists an $\Omega(t) \in \mathfrak{so}_n$ with $R^\top Z R = [\Omega, P]$. Hence, using the kinematic equations (21.35) we get

$$R^\top \dot{Z} R = \big[[U, P], [\Omega, P]\big] + [\dot{\Omega}, P]. \tag{21.40}$$

Evaluating the LHS of (21.24) gives

$$\begin{aligned} \mathrm{d}_{\alpha(t)}\chi(t)\big(\tfrac{\mathrm{D}}{\mathrm{d}t} Z(t)\big) &= R^\top\big(\tfrac{\mathrm{D}}{\mathrm{d}t} Z\big)R = R^\top(\mathrm{ad}_\alpha^2\, \dot{Z})R = \mathrm{ad}_P^2(R^\top \dot{Z} R) \\ &= \mathrm{ad}_P^2\left(\big[[U, P], [\Omega, P]\big] + [\dot{\Omega}, P]\right) \\ &= \mathrm{ad}_P^2\,\big[[U, P], [\Omega, P]\big] - \mathrm{ad}_P^3\,\dot{\Omega} = [\dot{\Omega}, P]. \end{aligned} \tag{21.41}$$

The vanishing of the first summand in the last line of (21.41) follows from the fact that $[U, P] \in so_P$ and $[\Omega, P] \in T_P Gr_{n,k}$ and by Lemma 21.3. Note that the minimal polynomial of the ad_P-operator is equal to $\mathrm{ad}_P^3 - \mathrm{ad}_P$, see [35] for a proof. Correspondingly, for the RHS of (21.24), and using (21.35),

$$\begin{aligned} \tfrac{\mathrm{D}}{\mathrm{d}t}\mathrm{d}_{\alpha(t)}\chi(t)Z(t) &= \tfrac{\mathrm{d}}{\mathrm{d}t}(R^\top Z R) = \dot{R}^\top Z R + R^\top Z \dot{R} + R^\top \dot{Z} R \\ &= [U, P]^\top R^\top Z R + R^\top Z R[U, P] + R^\top \dot{Z} R \\ &= \big[[\Omega, P], [U, P]\big] + \big[[U, P], [\Omega, P]\big] + [\dot{\Omega}, P] = [\dot{\Omega}, P]. \end{aligned} \tag{21.42}$$

It remains to show the normal part of the no-twist condition. Let Z be any *normal* vector field along α, i.e., $Z \in T_\alpha^\perp Gr_{n,k}$ and $R^\top Z R \in T_P^\perp Gr_{n,k}$. Moreover, there exists a $B \in Sym_n$ such that $Z = (\mathrm{id} - \mathrm{ad}_\alpha^2)B$, or equivalently, such that $R^\top Z R = (\mathrm{id} - \mathrm{ad}_P^2)(R^\top B R)$. Now $R^\top \dot{\alpha} R = \mathrm{ad}_P^2\, U = U$ by (21.35). Consequently,

$$\dot{Z} = \big(\mathrm{id} - \mathrm{ad}_\alpha^2\big)\dot{B} - [\dot{\alpha}, [\alpha, B]] - [\alpha, [\dot{\alpha}, B]], \tag{21.43}$$

implying by invariance

$$R^\top \dot{Z} R = \big(\mathrm{id} - \mathrm{ad}_P^2\big)R^\top \dot{B} R - [U, [P, R^\top B R]] - [P, [U, R^\top B R]]. \tag{21.44}$$

Afterall, the LHS of (21.25) for $Gr_{n,k}$ gives

$$\begin{aligned} \tfrac{\mathrm{D}}{\mathrm{d}t}\mathrm{d}_{\alpha(t)}\chi(t)Z(t) &= R^\top\big(\mathrm{id} - \mathrm{ad}_\alpha^2\big)(\dot{Z})R = (\mathrm{id} - \mathrm{ad}_P^2)(R^\top \dot{Z} R) \\ &= R^\top \dot{Z} R + \mathrm{ad}_P^2\left(\big[U, [P, R^\top B R]\big] + \big[P, [U, R^\top B R]\big]\right) \\ &= R^\top \dot{Z} R + \mathrm{ad}_P^2\,\big([U, [P, R^\top B R]]\big) + \big[P, [U, R^\top B R]\big] \\ &= R^\top \dot{Z} R + \big[P, [U, R^\top B R]\big]. \end{aligned} \tag{21.45}$$

Here we have used $\big[U, [P, R^\top B R]\big] \in T_P^\perp Gr_{n,k}$, by Lemma 21.3.

For the RHS of (21.25) for $Gr_{n,k}$ we calculate, using (21.35) and the Jacobi identity,

$$\begin{aligned}
\tfrac{\mathrm{d}}{\mathrm{d}t}(R^\top ZR) &= R^\top \dot{Z}R + \dot{R}^\top ZR + R^\top Z\dot{R} = R^\top \dot{Z}R + \big[R^\top ZR, [U, P]\big] \\
&= R^\top \dot{Z}R + \big[\big(\mathrm{id} - \mathrm{ad}_P^2\big)(R^\top BR), [U, P]\big] \\
&= R^\top \dot{Z}R - \big[U, \underbrace{[P, (\mathrm{id} - \mathrm{ad}_P^2)(R^\top BR)]}_{=0 \text{ as } \mathrm{ad}_P^3 = \mathrm{ad}_P}\big] \\
&\quad - \big[P, [(\mathrm{id} - \mathrm{ad}_P^2)(R^\top BR), U]\big] \\
&= R^\top \dot{Z}R - \big[P, [R^\top BR, U]\big] + \underbrace{\big[P, [\mathrm{ad}_P^2(R^\top BR), U]\big]}_{=0 \text{ by Lemma 21.3}} \\
&= R^\top \dot{Z}R + \big[P, [U, R^\top BR]\big]
\end{aligned} \tag{21.46}$$

as required. □

Corollary 21.13 *If $U(t) = U$ is constant, the kinematic equations (21.35) can be solved explicitly to obtain $R(t) = \mathrm{e}^{t\Omega}$, where $\Omega := [U, P]$, and $S(t) = tU$. In this case, the rolling curve $\alpha(t) = \mathrm{e}^{t\Omega}\, P\, \mathrm{e}^{-t\Omega}\, P$ is a geodesic.*

Proof The first statement is an immediate consequence of Theorem 21.12. For the second statement it is enough to show that $\frac{\mathrm{D}}{\mathrm{d}t}\dot{\alpha} := \Pi_{T_\alpha Gr_{n,k}}(\ddot{\alpha}) = 0$, which, according to (21.12), is equivalent to having $\mathrm{ad}_\alpha^2\, \ddot{\alpha} = 0$. A proof of this equality can be found, for instance, in [8]. An even shorter proof is included next.

For $\alpha(t) = \mathrm{e}^{t\Omega}\, P\, \mathrm{e}^{-t\Omega}$, we have $\dot{\alpha}(t) = \mathrm{e}^{t\Omega}\, \mathrm{ad}_\Omega\, P\, \mathrm{e}^{-t\Omega}$, and $\ddot{\alpha}(t) = \mathrm{e}^{t\Omega}\, \mathrm{ad}_\Omega^2\, P\, \mathrm{e}^{-t\Omega}$. So, $\mathrm{ad}_\alpha^2\, \ddot{\alpha} = 0$ is equivalent to $\mathrm{ad}_P^2\, \mathrm{ad}_\Omega^2\, P = 0$. But, by the Jacobi identity, $\mathrm{ad}_P\, \mathrm{ad}_\Omega^2\, P = [\mathrm{ad}_P^2\, \Omega, \Omega]$, and since $\Omega = [U, P] \in so_P$, according to Lemma 21.1, $\mathrm{ad}_P^2\, \Omega = \Omega$, so $\mathrm{ad}_P\, \mathrm{ad}_\Omega^2\, P = 0$, proving that α is indeed a geodesic. □

Rolling SO_n

Any smooth curve $\alpha\colon J \to SO_n$ with $\alpha(0) = P$ is of the form $\alpha(t) = V(t)PW^\top(t)$ with smooth $V, W\colon J \to SO_n$ satisfying $V(0) = W(0) = I$.

Theorem 21.14 *Consider the unique solution $\big(V(t), W(t), X(t)\big)$ of the initial value problem*

$$\begin{aligned}
&\dot{V}(t) = \tfrac{1}{2}V(t)U(t), && V(t) \in SO_n,\ V(0) = I, \\
&\dot{W}(t) = -\tfrac{1}{2}W(t)P^\top U(t)P, && W(t) \in SO_n,\ W(0) = I, \\
&\dot{X}(t) = U(t)P, && U(t) \in \mathfrak{so}_n,\ X(t) \in \mathbb{R}^{n\times n},\ X(0) = 0.
\end{aligned} \tag{21.47}$$

Then $\chi(t) = \big(V^\top(t), W(t), X(t)\big) \in SE_{n^2}$ *is the rolling map of* SO_n *on* $T_P^{\mathrm{aff}} SO_n$ *along the smooth curve* $\alpha(t) = V(t)PW^\top(t)$*. Equations (21.47) are called the kinematic equations of* SO_n*.*

Proof Obviously, $\alpha(t) \in SO_n$. Moreover,

$$\beta(t) = \chi(t)\alpha(t) = V^\top(t)\alpha(t)W(t) + X(t) = P + X(t). \tag{21.48}$$

By the initial condition $X(0) = 0$ together with $U(t)P \in T_P SO_n$, we have $X(t) \in T_P SO_n$. Moreover, $\beta(t) \in T_P^{\mathrm{aff}} SO_n$. In addition, the second rolling condition is fulfilled as well as

$$\mathrm{d}_\alpha\chi\big(T_\alpha SO_n\big) = V^\top(T_\alpha SO_n)W = V^\top(T_{VPW^\top} SO_n)W = T_P SO_n. \tag{21.49}$$

We now check the no-slip condition. From (21.47) and (21.48) we get

$$\dot\beta(t) = \dot X(t) = U(t)P. \tag{21.50}$$

On the other hand, using (21.49) and (21.47) we get

$$\begin{aligned} \mathrm{d}_{\alpha(t)}\chi(t)\dot\alpha(t) &= V^\top\dot\alpha W = V^\top\big(\dot V P W^\top + V P \dot W^\top\big)W = V^\top \dot V P + P\dot W^\top W \\ &= \tfrac{1}{2}UP - \tfrac{1}{2}U^\top P = UP \end{aligned} \tag{21.51}$$

as required.

It remains to check the no-twist conditions. For the tangent part let $Z(t)$ be any *tangent* vector field along $\alpha(t)$. I.e., $Z \in T_\alpha SO_n$ implying $V^\top(t)Z(t)W(t) \in T_P Gr_{n,k}$. In particular there exists a curve $\Omega(t) \in \mathfrak{so}_n$ with

$$V^\top(t)Z(t)W(t) = \Omega(t)P. \tag{21.52}$$

Taking derivates on both sides and using the kinematic equations (21.47) give

$$\dot V^\top Z W + V^\top Z \dot W + V^\top \dot Z W = \dot\Omega P \iff V^\top \dot Z W = \dot\Omega P + \tfrac{1}{2}(U\Omega + \Omega U)P. \tag{21.53}$$

Evaluating the LHS of (21.24) gives

$$\begin{aligned} \mathrm{d}_{\alpha(t)}\chi(t)\tfrac{\mathrm{D}}{\mathrm{d}t}Z(t) &= V^\top\tfrac{1}{2}\big(\dot Z - \alpha\dot Z^\top\alpha\big)W = \tfrac{1}{2}\big(V^\top\dot Z W - P\big(V^\top \dot Z W\big)^\top P\big) \\ &= \tfrac{1}{2}\Big(\dot\Omega P + \tfrac{1}{2}(U\Omega + \Omega U)P - P\big(\dot\Omega P + \tfrac{1}{2}(U\Omega+\Omega U)P\big)^\top P\Big) = \dot\Omega P. \end{aligned} \tag{21.54}$$

Correspondingly, for the RHS of (21.24) using (21.53)

$$\tfrac{\mathrm{D}}{\mathrm{d}t}\mathrm{d}_{\alpha(t)}\chi(t)Z(t) = \tfrac{\mathrm{d}}{\mathrm{d}t}(V^\top Z W) = \dot\Omega P \tag{21.55}$$

as required. It remains to show the normal part of the no-twist condition. Let Z be any *normal* vector field along α, i.e., $Z \in T_\alpha^\perp SO_n$ and $V^\top ZW = S(t)P \in T_P^\perp SO_n$ with curve $S(t) \in \mathrm{Sym}_n$. Analogously to the tangent part, we take derivatives

$$\dot{V}^\top ZW + V^\top Z\dot{W} + V^\top \dot{Z}W = \dot{S}P \iff V^\top \dot{Z}W = \dot{S}P + \tfrac{1}{2}(US + SU)P. \tag{21.56}$$

Evaluating the LHS of (21.25) gives

$$\begin{aligned} \mathrm{d}_{\alpha(t)}\chi(t)\tfrac{\mathrm{D}^\perp}{\mathrm{d}t}Z(t) &= V^\top \tfrac{1}{2}\big(\dot{Z} + \alpha\dot{Z}^\top\alpha\big)W = \tfrac{1}{2}\big(V^\top\dot{Z}W + P\big(V^\top\dot{Z}W\big)^\top P\big) \\ &= \tfrac{1}{2}\Big(\dot{S}P + \tfrac{1}{2}(US+SU)P + P\big(\dot{S}P + \tfrac{1}{2}(US+SU)P\big)^\top P\Big) = \dot{S}P. \end{aligned} \tag{21.57}$$

Correspondingly, for the RHS of (21.25) using (21.56)

$$\tfrac{\mathrm{D}^\perp}{\mathrm{d}t}\mathrm{d}_{\alpha(t)}\chi(t)Z(t) = \tfrac{\mathrm{d}}{\mathrm{d}t}(V^\top ZW) = \dot{S}P \tag{21.58}$$

as required. □

Corollary 21.15 *If $U(t) = U$ is constant, the kinematic equations (21.47) can be solved explicitly to obtain $V(t) = \mathrm{e}^{\frac{1}{2}tU}$, $W(t) = P^\top \mathrm{e}^{-\frac{1}{2}tU} P$, and $X(t) = tUP$. In this case, the rolling curve $\alpha(t) = \mathrm{e}^{tU} P$ is a geodesic.*

Proof This follows from Theorem 21.14 and from the fact that $\dot{\alpha} = U\alpha$ implies $\ddot{\alpha} = U^2\alpha$, which clearly belongs to $T_\alpha^\perp SO_n$. So, $\frac{\mathrm{D}}{\mathrm{d}t}\dot{\alpha} = 0$ and α is a geodesic. □

Remark 21.16 Clearly, for the three kinematic equations (21.26), (21.35), and (21.47), standard results for solving initial value problems apply, in particular, existence and uniqueness of solutions.

Remark 21.17 In all the three examples presented, we start with an arbitrary curve α as the rolling curve, but later the nonholonomic constraints of no-twist force α to have a special form. This might seem a contradiction, but indeed it has a very interesting explanation that involves the theory of symmetric spaces. Going into these details here is out of scope for this chapter, but we give a brief explanation, referring the interested reader to [34, 43] for further developments.

We mention a specific case. If $\mathcal{G}$ is a Lie group acting transitively and effectively on a Riemannian manifold $\mathcal{M}$ with $\mathcal{K}$ the isotropy subgroup of $\mathcal{G}$ at a point $P \in \mathcal{M}$, then $\mathcal{M} = \mathcal{G}/\mathcal{K}$ and the Lie algebra of $\mathcal{G}$, here denoted by $\mathfrak{g}$, admits a direct sum decomposition $\mathfrak{g} = \mathfrak{k} \oplus \mathfrak{p}$, called Cartan decomposition. Here $\mathfrak{k}$ is the Lie algebra of $\mathcal{K}$, $\mathfrak{p}$ is a vector subspace of $\mathfrak{g}$, and furthermore $[\mathfrak{k}, \mathfrak{p}] \subset \mathfrak{p}$, and $[\mathfrak{p}, \mathfrak{p}] \subset \mathfrak{k}$. It turns out that $\mathfrak{p} \cong T_P\mathcal{M}$.

A curve $t \mapsto g(t)$ in $\mathcal{G}$ is said to be horizontal if $g(t)^{-1}\dot{g}(t) \in \mathfrak{p}$. An important fact, whose proof can be found in [43, page 131], is that any curve in $\mathcal{M}$ is the projection of a horizontal curve in $\mathcal{G}$. This statement is what resolves the seeming contradiction mentioned above.

Since S^{n-1}, $Gr_{n,k}$ and SO_n are all (compact) symmetric spaces, the last statement applies. So, even though the rolling curve for each example was considered the projection of any curve in the group $\mathcal{G}$ that acts on the manifold, the "rotational part" of the kinematic equations shows that it could be considered the projection of a horizontal curve from the very beginning. In order to fully understand that, one would have to go into the Cartan decomposition for each example and conclude that the curve on $\mathcal{G}$ that projects on α is indeed horizontal.

21.2.3 Parallel Transport

Consider a tangent (resp. normal) vector field $Z(t)$ along the rolling curve $\alpha(t)$. Denote by $\chi(t)$ the corresponding rolling map. Then Z is tangent (resp. normal) *parallel* along α iff $\frac{\mathrm{D}}{\mathrm{d}t} Z(t) = 0$ (resp. $\frac{\mathrm{D}^{\perp}}{\mathrm{d}t} Z(t) = 0$). Equivalently, from the no-twist conditions we therefore get

$$\tfrac{\mathrm{D}}{\mathrm{d}t}\big(\mathrm{d}_{\alpha(t)}\chi(t)Z(t)\big) = 0 \quad \Big(\text{resp. } \tfrac{\mathrm{D}^{\perp}}{\mathrm{d}t}\big(\mathrm{d}_{\alpha(t)}\chi(t)Z(t)\big) = 0\Big). \tag{21.59}$$

Explicit formulas for the parallel transport of an arbitrary tangent $Z(0) \in T_{\alpha(0)}M$ (resp. normal $Z(0) \in T^{\perp}_{\alpha(0)}M$) is then straightforward by the following fact (rolling conditions).

$$\begin{aligned} \mathrm{d}_{\alpha(t)}\chi(t) \colon T_{\alpha(t)}M &\to T_{\chi(t)\alpha(t)}\big(\chi(t)M\big) = T_{\beta(t)}\big(T^{\mathrm{aff}}_{\alpha(0)}M\big) = T_{\alpha(0)}M, \\ Z(t) &\mapsto \mathrm{d}_{\alpha(t)}\chi(t)Z(t) = Z(0), \end{aligned} \tag{21.60}$$

or equivalently

$$Z(0) \mapsto \big(\mathrm{d}_{\alpha(t)}\chi(t)\big)^{-1} Z(0) = Z(t). \tag{21.61}$$

That is, $\big(\mathrm{d}_{\alpha(t)}\chi(t)\big)^{-1}$ is the isometric isomorphism denoting the parallel vector transport along α. In particular, for the three examples:

Sphere S^{n-1} Consider the rolling curve $\alpha(t) = R(t)p$. An arbitrary tangent vector field has the form $Z(t) = R(t)\Omega(t)p \in T_{\alpha}S^{n-1}$ with $\Omega = u(t)p^{\top} - pu^{\top}(t) \in \mathfrak{so}_n$ and $u(t) \in T_pS^{n-1}$. Consequently, $Z(t)$ is parallel along α iff $u(t)$ is constant. Similarly, an arbitrary normal vector field $Z(t) = R(t)\gamma(t)p \in T^{\perp}_{\alpha}S^{n-1}$ with $\gamma \in \mathbb{R}$ is parallel along α iff γ is constant. Explicitly, given any $Z_0 \in T_p\mathbb{R}^n \cong \mathbb{R}^n$ its parallel transport $Z(t)$ along $\alpha(t)$ is then $Z(t) = R(t)Z_0$.

Graßmannian $Gr_{n,k}$ With rolling curve $\alpha(t) = R(t)PR^{\top}(t)$, an arbitrary tangent vector field $Z(t)$ along $\alpha(t)$ has the form $Z(t) = R(t)[\Omega(t), P]R^{\top}(t) \in T_{\alpha}Gr_{n,k}$ with $\Omega \in so_P$. So $Z(t)$ is parallel along α iff Ω is constant. Analogously, an arbitrary normal vector field along $\alpha(t)$ has the form $Z(t) = R(t)(S(t) - \mathrm{ad}^2_P\, S(t))R^{\top}(t) \in T^{\perp}_{\alpha}Gr_{n,k}$ with $S \in \mathrm{Sym}_n$. It is parallel iff $S(t) - \mathrm{ad}^2_P\, S(t)$ is constant. More general, consider $Z_0 \in T_P\, \mathrm{Sym}_n \cong \mathrm{Sym}_n$, then its parallel transport along α is as $Z(t) = R(t)Z_0R^{\top}(t)$.

Special Orthogonal Group SO_n In this case the rolling curve $\alpha(t) = V(t)PW^\top(t)$. An arbitrary tangent vector field $Z(t)$ along $\alpha(t)$ is of the form $Z(t) = V(t)\Omega(t)PW^\top(t)$ with $\Omega \in \mathfrak{so}_n$ Consequently, $Z(t)$ is parallel along α iff Ω is constant. Similarly, an arbitrary normal vector field looks like $Z(t) = V(t)S(t)PW^\top(t)$ with $S \in \mathrm{Sym}_n$. Consequently, $Z(t)$ is parallel along α iff S is constant. Finally, let $Z_0 \in T_P\mathbb{R}^{n\times n} \cong \mathbb{R}^{n\times n}$, then its parallel transport along $\alpha(t)$ is as $Z(t) = V(t)Z_0W^\top(t)$.

Remark 21.18 The fact that the computation of parallel transport of vector fields along curves is that simple for the above examples, as we have shown, was already noticed in [57, page 630]. There, the author is referring to the kinematic interpretation of the Levi-Civita connection for the special case of 2-surfaces in $\mathbb{R}^3$. These results can also be found for arbitrary dimension in [60], prop. 3.7, app. B.

21.3 Rolling to Solve Interpolation Problems on Manifolds

As before, the manifolds $\mathcal{M}$ and $T^{\text{aff}}_{p_0}\mathcal{M}$ are considered to be embedded into some Euclidean vector space V. In this section we assume that if $\chi(t) = \big(R^\top(t), s(t)\big)$ is a rolling map of $\mathcal{M}$ on $T^{\text{aff}}_{p_0}\mathcal{M}$, along the curve $\alpha(t)$, then $\alpha(t) = R(t)p_0$. Note that the rolling maps for the three examples considered in previous sections, $\mathcal{M} = S^n$, $\mathcal{M} = SO_n$, and $\mathcal{M} = Gr_{n,k}$, satisfy this property. With this assumption, we propose an algorithm to generate an interpolating curve on $\mathcal{M}$ that is given explicitly in terms of the coordinates of the embedding space.

21.3.1 Formulation of the Problem

Problem 21.19 Given a set of $k+1$ distinct points $p_i \in \mathcal{M}$, two vectors ξ_0 and ξ_k tangent to $\mathcal{M}$ at p_0 and p_k, respectively, and fixed times t_i, where $0 = t_0 < \cdots < t_k = \tau$, find a C^2-smooth curve

$$\gamma : [0, \tau] \to \mathcal{M} \tag{21.62}$$

satisfying interpolation conditions

$$\gamma(t_i) = p_i, \qquad 1 \le i \le k-1, \tag{21.63}$$

and boundary conditions

$$\gamma(0) = p_0,\ \gamma(\tau) = p_k, \quad \dot\gamma(0) = \xi_0,\ \dot\gamma(\tau) = \xi_k. \tag{21.64}$$

□

21.3.2 Motivation

If $\mathcal{M}$ is the Euclidean space $\mathbb{R}^n$, the unique solution of this problem is a cubic spline, i.e., a curve whose restriction to each subinterval $[t_i, t_{i+1}]$ is a cubic polynomial. It can be obtained, for instance, through the de Casteljau algorithm [25], or using a variational/Hamiltonian approach since the cubic spline is the curve that minimizes the cost functional

$$\int_0^\tau \left\langle \ddot{\gamma}(t), \ddot{\gamma}(t) \right\rangle dt. \tag{21.65}$$

The de Casteljau algorithm is a geometric construction simple to implement, based on successive linear interpolation, cf. [25, 27]. The optimality property of the Euclidean cubic splines makes them particularly useful in applications. But its generalization to Riemannian manifolds, where in the cost functional above $\ddot{\gamma}$ is replaced by the intrinsic acceleration $\frac{D}{dt}\dot{\gamma}$ and $\langle \cdot, \cdot \rangle$ is the Riemannian metric on $\mathcal{M}$, is even more useful since nonlinear data appears naturally in many engineering applications. Cubic splines on manifolds have been studied by several authors, starting with the pioneer work in [56], followed by further developments, cf. [12, 19, 20]. Contrary to the Euclidean situation, the Euler-Lagrange equations associated to this variational problems are highly nonlinear and can only be solved explicitly in some trivial cases. This is the main drawback of the variational approach.

To overcome this problem, the de Casteljau algorithm has also been generalized to Riemannian manifolds, cf. [22, 58]. The basic idea is to replace linear interpolation by geodesic interpolation, which for specific manifolds requires that one knows explicit formulas for the geodesic that joins two points. The implementation of this algorithm also requires to solve nonlinear implicit equations which makes it computationally expensive. Moreover, it does not give the interpolation curve in an explicit form.

Interpolation schemes for manifolds with a complicated geometry, based on the idea of projecting the interpolating data to a simpler manifold using local diffeomorphisms, are available in the literature, cf. [26, 54]. But the most successful scheme combines rolling with other local diffeomorphisms parameterized by time. They are based on rolling and unwrapping techniques. The term unwrapping, coined in [41], is used to describe how the diffeomorphism maps data from the manifold to data on its affine tangent space at a point. A similar concept appearing in the setting of optimization on manifolds, namely a retraction, appeared in [3].

These schemes were used for the first time in [41] for the sphere S^2, generalized in [36] for spheres of any dimension, and in [37] for SO_n and $Gr_{n,k}$. They can be used to solve interpolation problems on general manifolds as long as the kinematic equations of rolling can be solved explicitly. The result is an interpolating curve given in closed form.

21.3.3 Solving the Interpolation Problem

To solve Problem 21.19, we propose the following algorithm which is based on rolling and unwrapping techniques. The resulting curve will be given explicitly in terms of the coordinates of the embedding space. The algorithm can be described in the following 5 steps.

Algorithm: Interpolation Algorithm

1. Compute an arbitrary smooth curve $\alpha : [0, \tau] \to \mathcal{M}$, connecting p_0 with p_k, such that $\alpha(0) = p_0$ and $\alpha(\tau) = p_k$.
2. Unwrap the boundary data from $\mathcal{M}$ to $T^{\text{aff}}_{p_0}\mathcal{M}$, by rolling $\mathcal{M}$ along $\alpha : [0, \tau] \to \mathcal{M}$ with rolling map $\chi(t) = \big(R^\top(t), s(t)\big)$. This produces a smooth curve $\beta : [0, \tau] \to T^{\text{aff}}_{p_0}\mathcal{M}$ and the rolling conditions ensure that all the boundary conditions (21.64) are mapped to $T^{\text{aff}}_{p_0}\mathcal{M}$ as follows:

$$\begin{aligned} p_0 &\mapsto \beta(0) = \chi(0)p_0 = p_0 =: q_0, \quad & p_k &\mapsto \beta(\tau) = \chi(\tau)p_k =: q_k \\ \xi_0 &\mapsto d_{\alpha(t)}\chi(t)\big|_{t=0}\xi_0 = \xi_0 =: \eta_0, \quad & \xi_k &\mapsto d_{\alpha(t)}\chi(t)\big|_{t=\tau}\xi_k =: \eta_k. \end{aligned} \tag{21.66}$$

3. Unwrap the remaining data $p_1, \dots, p_{k-1}$ onto $T^{\text{aff}}_{p_0}\mathcal{M}$, using a suitable local diffeomorphism

$$\phi : M \supset \Omega \to T^{\text{aff}}_{p_0}\mathcal{M}, \qquad p_0 \in \Omega \text{ open}, \tag{21.67}$$

satisfying

$$\phi(p_0) = p_0 \quad \text{and} \quad d_{p_0}\phi = \text{id}, \tag{21.68}$$

as follows:

$$p_i \mapsto \phi\big(R^\top(t_i)p_i\big) + s(t_i) =: q_i. \tag{21.69}$$

4. Solve the interpolation problem (similar to Problem 21.19) on $T^{\text{aff}}_{p_0}\mathcal{M}$ using the mapped data $\{q_0, \dots, q_k, \eta_0, \eta_k\}$. This will generate a C^2-smooth curve

$$y : [0, \tau] \to T^{\text{aff}}_{p_0}\mathcal{M} \tag{21.70}$$

that satisfies the following boundary and interpolation conditions:

$$y(0) = p_0 = q_0,\ y(\tau) = q_k, \quad \dot{y}(0) = \xi_0 = \eta_0,\ \dot{y}(\tau) = \eta_k, \tag{21.71}$$

and

$$y(t_i) = q_i, \quad i = 1, \cdots, k-1. \tag{21.72}$$

5. Wrap $y([0, \tau])$ back onto the manifold $\mathcal{M}$, using the inverse of the rolling map and the inverse of the diffeomorphism, to obtain the solution γ of Problem 21.19 on $\mathcal{M}$ by means of the following explicit formula:

$$\gamma(t) := \chi(t)^{-1}\Big(\phi^{-1}\big(y(t) - s(t)\big) + s(t)\Big) \tag{21.73}$$

Note that, since $y(t)$ is a curve in $T^{\text{aff}}_{p_0}\mathcal{M}$ and the assumption at the beginning of this section sets $s(t) \in T_{p_0}\mathcal{M}$, then $t \mapsto y(t) - s(t)$ is a curve in $T^{\text{aff}}_{p_0}\mathcal{M}$ and so $\gamma(t)$ is well defined.

Theorem 21.20 *If $\alpha(t) = R(t)p_0$, the curve $t \mapsto \gamma(t)$ defined by (21.73) solves Problem 21.19.*

Proof The inverse of the rolling map is given by $\chi(t)^{-1} = \big(R(t), -R(t)s(t)\big)$. So, we may write

$$\gamma(t) = R(t)\Big(\phi^{-1}\big(y(t) - s(t)\big) + s(t)\Big) - R(t)s(t) = R(t)\left(\phi^{-1}(y(t) - s(t))\right). \tag{21.74}$$

Therefore, since $y(t_i) = q_i$ and (21.69) holds, we have $y(t_i) - s(t_i) = \phi\big(R^\top(t_i)p_i\big)$ and, consequently,

$$\gamma(t_i) = R(t_i)\Big(\phi^{-1}\big(y(t_i) - s(t_i)\big)\Big) = R(t_i)R^\top(t_i)p_i = p_i,$$

that is, the curve γ interpolates the points p_i at time t_i, for all $i = 0, \ldots, k$.

We now prove that the initial and final velocity of the curve γ also match the boundary conditions. Note that the constraint $\alpha(t) = R(t)p_0$ implies that $s(t) \in T_{p_0}\mathcal{M}$ and $\beta(t) = p_0 + s(t)$. For simplicity, define $z(t) := y(t) - s(t)$, so that, $z(0) = p_0$ and $\dot{z}(0) = 0$. Taking derivatives on both sides of (21.74), we obtain

$$\dot{\gamma}(t) = \dot{R}(t)\Big(\phi^{-1}\big(z(t)\big)\Big) + R(t)d_{z(t)}\phi^{-1}(\dot{z}(t)). \tag{21.75}$$

Evaluating at $t = 0$ and taking into account that $\dot{R}(0)p_0 = \eta_0$ and ϕ satisfies conditions (21.68), it follows that

$$\dot{\gamma}(0) = \dot{R}(0)\Big(\phi^{-1}(p_0)\Big) + R(0)d_{p_0}\phi^{-1}(0) = \eta_0. \tag{21.76}$$

Similarly, evaluating at $t = \tau$, we easily obtain

$$\dot{\gamma}(\tau) = \dot{R}(\tau)\Big(\phi^{-1}\big(z(\tau)\big)\Big) + R(\tau)d_{p_0}\phi^{-1}(0) = \eta_k. \tag{21.77}$$

Finally, the resulting curve γ is C^2-smooth by construction, since χ and ϕ are smooth and β is C^2-smooth. This concludes the proof. □

Remark 21.21

1. It is important to point out that step 4. of the algorithm can be easily implemented, since a cubic spline on a flat space can be obtained explicitly and uniquely from the smooth, boundary and interpolation conditions.
2. However, the interpolating curve γ depends on the choice of the rolling curve used in steps 1. and 2. and the choice of the local diffeomorphism used in step 3..

21.3.4 Examples

Using the results of previous sections about rolling S^{n-1}, SO_n, and $Gr_{n,k}$ on the corresponding affine space at a point, we present explicit formulas for an interpolating curve γ.

Sphere S^{n-1} $\chi(t) = \big(R^\top(t), v(t)\big)$ is the rolling map along the curve $\alpha(t) = R(t)p$. So, the interpolating curve generated by the previous algorithm is given by

$$\gamma(t) = R(t)\Big(\phi^{-1}\big(y(t) - v(t)\big)\Big), \tag{21.78}$$

where R, v are the solutions of the kinematic equations (21.26), that satisfy $R(0) = I, v(0) = 0$.

Graßmannian $Gr_{n,k}$ $\chi(t) = \big(R^\top(t), S(t)\big)$ is the rolling map along the curve $\alpha(t) = R(t)PR^\top(t)$. So, the interpolating curve generated by the previous algorithm is given by

$$\gamma(t) = R(t)\Big(\phi^{-1}\big(y(t) - S(t)\big)\Big) - > R^\top(t), \tag{21.79}$$

where R, S are the solutions of the kinematic equations (21.35), that satisfy $R(0) = I, S(0) = 0$.

Special Orthogonal Group SO_n $\chi(t) = \big(V^\top(t), W^\top(t)\big), X(t))$ is the rolling map along the curve $\alpha(t) = V(t)PW^\top(t)$. So, the interpolating curve generated by the previous algorithm is given by

$$\gamma(t) = V(t)\Big(\phi^{-1}\big(y(t) - X(t)\big)\Big)W^\top(t), \tag{21.80}$$

where V, W, X are the solutions of the kinematic equations (21.47), that satisfy $V(0) = W(0) = I, X(0) = 0$.

The interpolating curve γ now depends on the choice of the rolling curve used in steps 1. and 2. and the choice of the local diffeomorphism used in step 3. Since the kinematic equations of rolling can be explicitly solved when the rolling curve is a geodesic, this is the natural choice when implementing the algorithm. In which concerns the diffeomorphism, we can choose, for instance, the stereographic projection or the orthogonal projection for the sphere and the Riemannian normal coordinates for all the examples.

21.3.5 Implementation of the Algorithm on S^2

Here we present an example, for the two-sphere $S^2 = \{x \in \mathbb{R}^3 | x_1^2 + x_2^2 + x_3^2 = 1\}$ rolling on its tangent plane $T_{p_0}^{\text{aff}}S^2$ at the south pole $p_0 = [0, 0, -1]^\top \in S^2$. We want to solve Problem 21.19 for $\mathcal{M} = S^2$ using Algorithm:. Two choices have to be made: the rolling curve α and the diffeomorphism ϕ. For the first, the obvious choice is the geodesic that joins p_0 (at $t = 0$) to p_k (at $t = \tau$). In this case, the rolling map is given by

$$\chi(t) = \big(R(t)^\top, s(t)\big) = \big(\mathrm{e}^{-t\Omega}, t\Omega p_0\big), \tag{21.81}$$

where Ω is the constant matrix

$$\Omega = \begin{bmatrix} 0 & 0 & -u_1 \\ 0 & 0 & -u_2 \\ u_1 & u_2 & 0 \end{bmatrix} \in \mathfrak{so}_3. \tag{21.82}$$

The development $\beta([0, \tau])$ is a straight line segment in $T^{\text{aff}}_{p_0} S^2$, parameterized by t, starting at $t = 0$ in p_0 as one would expect.

Let us now fix the diffeomorphism. We do it in two situations, the stereographic projection denoted by ϕ^{stereo}, and the normal projection denoted by ϕ^{ortho}.

$$\phi^{\text{stereo}} : S^2 \setminus \{[0, 0, 1]^\top\} \to T^{\text{aff}}_{p_0} S^2, \qquad \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \mapsto \begin{bmatrix} \frac{2x_1}{1-x_3} \\ \frac{2x_2}{1-x_3} \\ -1 \end{bmatrix}, \tag{21.83}$$

with inverse

$$(\phi^{\text{stereo}})^{-1} : T^{\text{aff}}_{p_0} S^2 \to S^2 \setminus \{[0, 0, 1]^\top\}, \qquad \begin{bmatrix} \xi_1 \\ \xi_2 \\ -1 \end{bmatrix} \mapsto \begin{bmatrix} 4\xi_1 \\ 4\xi_2 \\ \xi_1^2 + \xi_2^2 - 4 \end{bmatrix} \frac{1}{\xi_1^2+\xi_2^2+4}. \tag{21.84}$$

Orthogonal projection onto the sphere is defined by

$$\phi^{\text{ortho}} : S^2 \setminus \{x \in S^2 \mid x_3 \geq 0\} \to T^{\text{aff}}_{p_0} S^2, \qquad \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \mapsto \begin{bmatrix} -\frac{x_1}{x_3} \\ -\frac{x_2}{x_3} \\ -1 \end{bmatrix}, \tag{21.85}$$

with inverse

$$(\phi^{\text{ortho}})^{-1} : T^{\text{aff}}_{p_0} S^2 \to S^2 \setminus \{x \in S^2 \mid x_3 \geq 0\}, \qquad \begin{bmatrix} \xi_1 \\ \xi_2 \\ -1 \end{bmatrix} \mapsto \begin{bmatrix} \xi_1 \\ \xi_2 \\ -1 \end{bmatrix} \frac{1}{\sqrt{\xi_1^2+\xi_2^2+1}}. \tag{21.86}$$

Obviously, for the south pole $p_0 = [0, 0, -1]^\top$,

$$\phi^{\text{stereo}}(p_0) = \phi^{\text{ortho}}(p_0) = p_0. \tag{21.87}$$

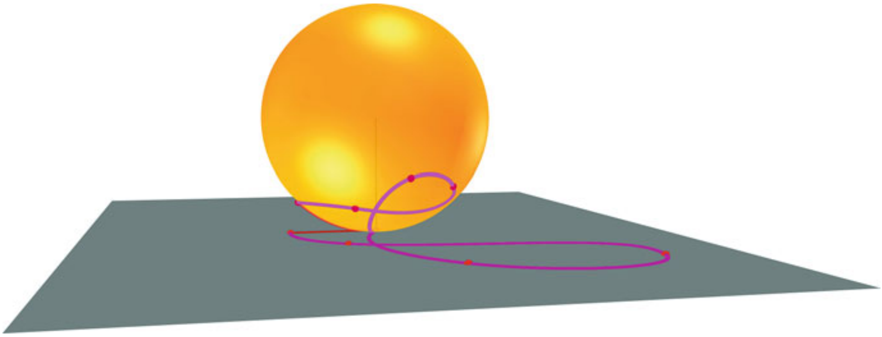


Fig. 21.1 Wrapping back by orthogonal projection, see (21.85) and (21.86). The sphere is still at rest at the south pole

Moreover, the differential of these diffeomorphisms satisfy the following, for any tangent vector $h \in T_{p_0}S^2$:

$$d_{p_0}\phi^{\text{stereo}}(h) = d_{p_0}\phi^{\text{ortho}}(h) = h, \tag{21.88}$$

so, both diffeomorphisms satisfy the conditions (21.68).

Interpolating the mapped data on $T^{\text{aff}}_{p_0}S^2$ can be done by computing a cubic spline, for instance, by means of the classical de Casteljau algorithm, see [25] or [27]).

According to Problem 21.19 we are given n points on S^2 together with n instants of time. For demonstration purposes and keeping the plots clear we decided to choose $n = 5$. Following the above algorithm we compute the great circle segment α connecting the initial point (south pole) with the final point. The development β is then a straight line segment in the affine tangent plane attached to the south pole. Great circles and straight lines are blue coloured in all five figures (see the Appendix). In Fig. 21.1 the sphere is attached to the tangent plane at p_0 at time t_0. One can see the cubic spline (red) lying in the tangent plane and the solution curve of the interpolation problem living on the sphere (red as well). Figure 21.2 shows the sphere after rolling along the blue straight line segment. The black ray emanating from the mid point of the sphere clarifies that we have used orthogonal projection. In contrast, Figs. 21.3 and 21.4 show the result by using stereographic projection instead of orthogonal projection. The black ray emanating from the top of the sphere connects corresponding points on the sphere and the tangent plane. Cubic spline and solution curve are both plotted in white. The last picture Fig. 21.5 allows for a qualitative comparison of the two methods.

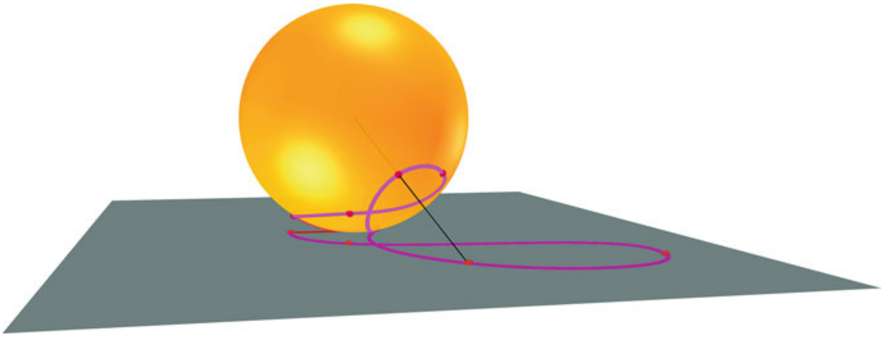


Fig. 21.2 Wrapping back by orthogonal projection, see (21.85) and (21.86). The sphere is rolling along the straight line segment

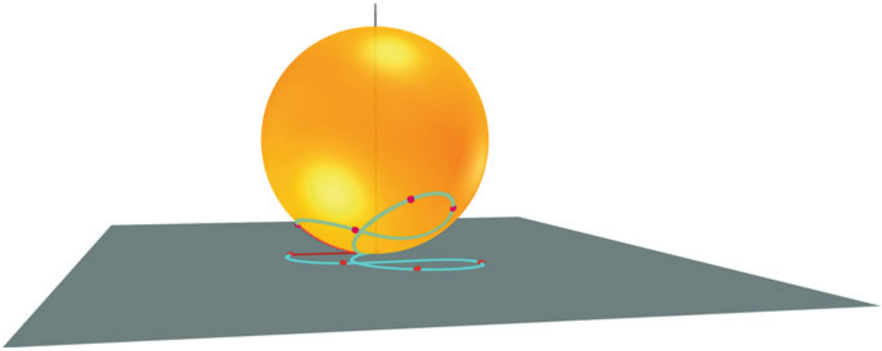


Fig. 21.3 Wrapping back by stereographic projection, see (21.83) and (21.84). The sphere is still at rest at the south pole

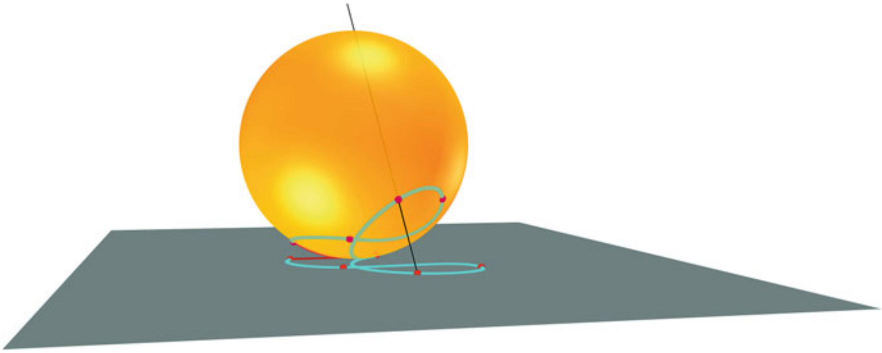


Fig. 21.4 Wrapping back by stereographic projection, see (21.83) and (21.84). The sphere is rolling along the straight line segment

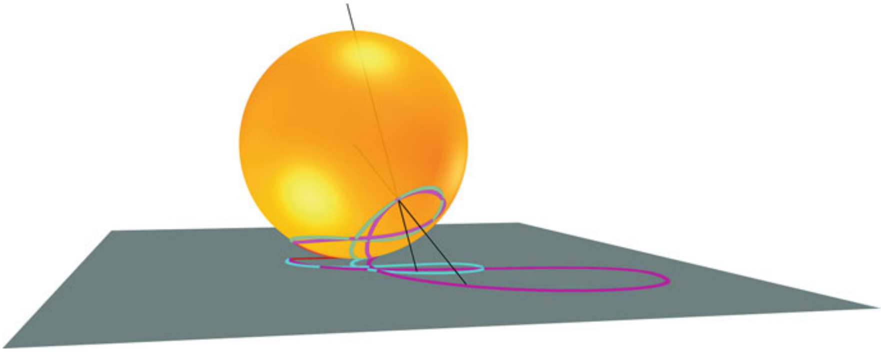


Fig. 21.5 Comparison of different interpolation curves which both solve the problem

21.4 Some Extensions

21.4.1 Rolling a Hypersurface

Let $\mathcal{H}^{n-1}$ denote a smooth orientable manifold of dimension $n-1$ embedded in $\mathbb{R}^n$ and $\eta\colon \mathcal{H}^{n-1} \to S^{n-1}$ the Gauß map that assigns a *unit normal vector* to every point in $\mathcal{H}^{n-1}$. Let $\alpha\colon J \to \mathcal{H}^{n-1}$ be a smooth curve with $\alpha(0) = p$ and $\eta\colon J \to S^{n-1}$ be the Gauß map along α. Then, the following theorem holds.

Theorem 21.22 *Consider the unique solution $\big(R(t), v(t)\big)$ of the initial value problem*

$$\begin{aligned} \dot R(t) &= \big(\dot\eta(t)\,\eta(t)^\top - \eta(t)\,\dot\eta(t)^\top\big)R(t), \quad && R(t) \in SO_n,\ \ R(0) = I, \\ \dot v(t) &= -\dot R^\top(t)\,\alpha(t), && v(t) \in \mathbb{R}^n,\ \ v(0) = 0. \end{aligned} \tag{21.89}$$

Then $\chi(t) = \big(R^\top(t), v(t)\big) \in SE_n$ is the unique rolling map of $\mathcal{H}^{n-1}$ on $T_p^{\mathrm{aff}}\mathcal{H}^{n-1}$ along the smooth curve $\alpha(t)$. Equation (21.89) are called the kinematic equations of the hypersurface.

For an alternative proof, and some useful geometric properties of the special case of the ellipsoid, see [50].

Proof The development curve $\beta\colon J \to \mathbb{R}^n$ is given by $\beta(t) = \chi(t)\,\alpha(t) = R^\top(t)\,\alpha(t) + v(t)$. By the second equality in (21.89), (we have omitted dependance on t for clarity)

$$\dot\beta = \dot R^\top\alpha + R^\top\dot\alpha + \dot v = \dot R^\top\alpha + R^\top\dot\alpha - \dot R^\top\alpha = R^\top\dot\alpha, \tag{21.90}$$

verifying the no-slip condition (21.23) of Definition 21.5.

To verify the rolling conditions consider the Gauß map η, which is a unit normal vector field η along α and note that $R^\top \eta$ remains constant during rolling. To see this calculate the derivative

$$\tfrac{\mathrm{d}}{\mathrm{d}t}\big(R^\top \eta\big) = \dot{R}^\top \eta + R^\top \dot{\eta} = R^\top\big(\eta\langle \dot{\eta}, \eta\rangle - \dot{\eta}\langle \eta, \eta\rangle\big) + R^\top \dot{\eta} = 0.$$

Therefore the rolling motion has no effect on the image of the tangent space $R^\top\big(T_\alpha \mathcal{H}^{n-1}\big)$, identified as a subspace of $\mathbb{R}^n$. By the initial condition, $R^\top\big(T_\alpha \mathcal{H}^{n-1}\big)$ and $T_p\mathcal{H}^{n-1}$ coincide at $t = 0$ thus they remain parallel because the latter tangent space is stationary. Since by (21.90)

$$\langle R^\top \eta, \dot{\beta}\rangle = \langle R^\top \eta, R^\top \dot{\alpha}\rangle = \langle \eta, \dot{\alpha}\rangle = 0$$

it follows that $\dot{\beta} \in R^\top\big(T_\alpha \mathcal{H}^{n-1}\big)$ and by the previous conclusion $\dot{\beta}$ also belongs to $T_p\mathcal{H}^{n-1}$. But $\beta(0) = p$ thus $\beta(t) \in T_p^{\text{aff}}\mathcal{H}^{n-1}$ verifying the first rolling condition. This also proves the second rolling condition $R^\top\big(T_\alpha \mathcal{H}^{n-1}\big) = T_\beta\big(T_p^{\text{aff}}\mathcal{H}^{n-1}\big)$.

It remains to verify the tangential part of the no-twist condition of Definition 21.5. Let ζ be a tangent vector field along α. Then

$$\mathrm{d}_\alpha \chi \tfrac{\mathrm{D}}{\mathrm{d}t}\zeta = R^\top\big(\dot{\zeta} - \dot{\zeta}^\perp\big) = R^\top \dot{\zeta} - R^\top \eta \langle \dot{\zeta}, \eta\rangle = R^\top \dot{\zeta} + R^\top \eta\langle \zeta, \dot{\eta}\rangle.$$

On the other hand, by Remark 21.9

$$\begin{aligned}\tfrac{\mathrm{D}}{\mathrm{d}t}\mathrm{d}_\alpha \chi \zeta = \tfrac{\mathrm{d}}{\mathrm{d}t}\big(R^\top \zeta\big) &= \dot{R}^\top \zeta + R^\top \dot{\zeta} = R^\top\big(\eta\,\dot{\eta}^\top - \dot{\eta}\,\eta^\top\big)\zeta + R^\top \dot{\zeta} \\ &= R^\top\big(\eta\langle \dot{\eta}, \zeta\rangle - \dot{\eta}\langle \eta, \zeta\rangle\big) + R^\top \dot{\zeta} = R^\top \eta \langle \dot{\eta}, \zeta\rangle + R^\top \dot{\zeta}\end{aligned}$$

thus confirming (21.24) and completing the proof. □

We now show that when $\mathcal{H}^{n-1}$ is the unit sphere S^{n-1}, the kinematic equations (21.89) reduce to the kinematic equations (21.26). In this case $\eta(t) = \alpha(t) = C(t)\,p$, for some curve $C\colon J \to SO_n$ with $C(0) = I$. Then, omitting dependance on t for clarity

$$\dot{\eta}\eta^\top - \eta\dot{\eta}^\top = \dot{C}pp^\top C^\top - Cpp^\top \dot{C}^\top = C\left(C^\top \dot{C}pp^\top - pp^\top \dot{C}^\top C\right)C^\top.$$

Take $R = C$ then the equations in (21.89) become

$$\dot{C} = C\big(C^\top \dot{C}pp^\top - pp^\top \dot{C}^\top C\big), \qquad \dot{v} = -\dot{C}^\top Cp = C^\top \dot{C}p,$$

or equivalently $\dot{C} = C\big(up^\top - pu^\top\big)$ and $\dot{v} = u$, where we denote $u = C^\top \dot{C}p$. Thus we arrive at the system of kinematic equations (21.26) in Sect. 21.2.2.

21.4.2 The Case of an Ellipsoid

Given a positive definite diagonal matrix $D = \operatorname{diag}(d_1, d_2, \dots, d_n) \succ 0$, define an ellipsoid $\mathcal{E}^{n-1}$, with positive semi-axes $d_1, d_2, \dots, d_n$, by

$$\mathcal{E}^{n-1} := \left\{ x \in \mathbb{R}^n \;\middle|\; \langle x, D^{-2}x \rangle = 1 \right\}. \tag{21.91}$$

The normal space at $p \in \mathcal{E}^{n-1}$ is spanned by $D^{-2}p$ and the Gauß map η along a curve $\alpha \colon J \to \mathcal{E}^{n-1}$ is given by

$$\eta(t) = \frac{D^{-2}\alpha(t)}{\|D^{-2}\alpha(t)\|}. \tag{21.92}$$

Then the first equation of (21.89) becomes somewhat unpleasant

$$\dot{R}(t) = \Big(\tfrac{D^{-2}\dot{\alpha}(t)}{\|D^{-2}\alpha(t)\|^2} \big(D^{-2}\alpha(t)\big)^\top - \tfrac{D^{-2}\alpha(t)}{\|D^{-2}\alpha(t)\|^2} \big(D^{-2}\dot{\alpha}(t)\big)^\top \Big) R(t). \tag{21.93}$$

One can remedy these complexities in the following way, see also [48]. Suppose that $\alpha(t) = D^2 C(t) D^{-1} p / \|DC(t)D^{-1}p\|$, for some $p \in \mathcal{E}^{n-1}$, $C \colon J \to SO_n$ with $C(0) = I$. Then

$$\eta(t) = \frac{C(t)D^{-1}p}{\|C(t)D^{-1}p\|} = C(t)D^{-1}p \tag{21.94}$$

and

$$\dot{\eta}(t)\eta^\top(t) - \eta(t)\dot{\eta}^\top(t) = \dot{C}(t)D^{-1}pp^\top D^{-1}C^\top(t) - C(t)D^{-1}pp^\top D^{-1}\dot{C}^\top(t). \tag{21.95}$$

Denoting $q = D^{-1}p$ one gets a similar expression to that of the previous example of the sphere, namely

$$\dot{\eta}(t)\eta^\top(t) - \eta(t)\dot{\eta}^\top(t) = \dot{C}(t)qq^\top C^\top(t) - C(t)qq^\top \dot{C}^\top(t). \tag{21.96}$$

Thus we have just shown that the problem of rolling $\mathcal{E}^{n-1}$ can be reduced, at least in principle, to the problem of rolling the unit sphere. However, finding explicit solutions for the rolling map even along simple curves, such as geodesics, is a difficult task, making it harder to implement the interpolating algorithm without sophisticated methods, as e.g. geometric integration algorithms or Lie integrators, cf. [33]. To overcome this problem, it is convenient to consider the ellipsoid embedded in a space with a different metric. This is explained in the next subsection.

21.4.2.1 Changing the Metric on the Ellipsoid

We give here yet another example of rolling that combines the cases of the sphere and ellipsoid described before. In essence, this is a short study of rolling the unit sphere in $\mathbb{R}^n$ equipped with a left-invariant metric. Let us start with the following observation.

Let $\mathcal{G}_D := D\,SO_n\,D^{-1} = \{DXD^{-1} \mid X \in SO_n\}$. Then $\mathcal{G}_D$ is a group that acts transitively on $\mathcal{E}^{n-1}$, although not isometrically, according to $x \mapsto DRD^{-1}x$, where $R \in SO_n$. However, by a change of metric on the ambient space $\mathbb{R}^n$ to

$$(v, w) \mapsto \langle v, w\rangle_{D^{-2}} := \langle v, D^{-2}w\rangle = v^\top D^{-2}w, \tag{21.97}$$

for any vectors $v, w \in \mathbb{R}^n$, definition (21.91) becomes

$$\mathcal{E}^{n-1} := \{x \in \mathbb{R}^n \mid \langle x, x\rangle_{D^{-2}} = 1\},$$

the sphere in $\mathbb{R}^n$ endowed with the left-invariant metric (21.97). With this metric, the normal space at $p \in \mathcal{E}^{n-1}$ is spanned by p and

$$T_p\mathcal{E}^{n-1} = \left\{D\Omega D^{-1}p \;\middle|\; \Omega \in \mathfrak{so}_n\right\},$$

where $D\,\mathfrak{so}_n\,D^{-1}$ is the Lie algebra of $\mathcal{G}_D$. The Gauß map η along $\alpha\colon J \to \mathcal{E}^{n-1}$ is $\eta(t) = \alpha(t)$, precisely as in the case of the Euclidean sphere, as expected. The following result whose proof can be found in [49] describes the kinematic equations for rolling the ellipsoid over its affine tangent space at the south pole $p = -d_n e_n$.

Theorem 21.23 *Let $A(t) := u(t)p^\top D^{-1} - D^{-1}pu^\top(t)$, where $u = [u_1, \cdots, u_{n-1}, 0]^\top$. If $R\colon J \to SO_n$ and $s\colon J \to \mathbb{R}^n$ are the unique solutions of the following set of equations*

$$\begin{aligned} \dot{R}(t) &= A(t)R(t), \\ \dot{s}(t) &= -DA(t)D^{-1}p, \end{aligned} \tag{21.98}$$

with $R(0) = I$ and $s(0) = 0$, then, $\chi\colon J \to \mathcal{G}_D \ltimes \mathbb{R}^n$ given by

$$\chi(t) = \left(DR(t)D^{-1}, s(t)\right) \tag{21.99}$$

is the rolling map of the ellipsoid rolling on its affine tangent space at p, with rolling curve $\alpha(t) = DR^{-1}(t)D^{-1}p$ and its development $\beta(t) = s(t) + p$.

When $u(t) = u$ is constant, the kinematic equations can be solved explicitly and the rolling curve, which is a geodesic in this case, is given in closed form.

Thus, the implementation in Sect. 21.3.5 of the interpolation algorithm on the sphere is applicable to an ellipsoid, with some adjustments, as the following example illustrates.

21.4.2.2 Implementation of the Interpolation Algorithm on $\mathcal{E}^2$

We shall be interested in

$$\mathcal{E}^2 := \left\{ x \in \mathbb{R}^3 \mid \tfrac{x_1^2}{d_1^2} + \tfrac{x_2^2}{d_2^2} + \tfrac{x_3^2}{d_3^2} = 1 \right\} \tag{21.100}$$

rolling on its affine tangent plane $T_{p_0}^{\text{aff}}\mathcal{E}^2$ at point $p_0 = [0, 0, -d_3]^\top \in \mathcal{E}^2$, embedded in $\mathbb{R}^3$ endowed with the metric (21.97), see [49]. The rolling curve is a geodesic and the development curve therefore is a straight line in $T_{p_0}^{\text{aff}}\mathcal{E}^2$. Geodesic segments γ_i joining two non-antipodal points $p_i, p_{i+1} \in \mathcal{E}^2$ at t_i and t_{i+1} are great arcs, given by

$$\gamma_i(t) = \tfrac{1}{\sin\theta_i}\Big(p_i \sin\left(\tfrac{t_{i+1}-t}{t_{i+1}-t_i}\theta_i\right) + p_{i+1} \sin\left(\tfrac{t-t_i}{t_{i+1}-t_i}\theta_i\right)\Big), \qquad t_i \le t \le t_{i+1},$$

where $\theta_i = \arccos\langle p_i, p_{i+1}\rangle_{D^{-2}}$. The stereographic projection is given by

$$\phi^{\text{stereo}} \colon \mathcal{E}^2 \setminus \left\{[0, 0, d_3]^\top\right\} \to T_{p_0}^{\text{aff}}\mathcal{E}^2$$
$$\phi^{\text{stereo}} \colon \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \mapsto \begin{bmatrix} \frac{2d_3x_1}{d_3-x_3} \\ \frac{2d_3x_2}{d_3-x_3} \\ -d_3 \end{bmatrix}, \tag{21.101}$$

with inverse

$$(\phi^{\text{stereo}})^{-1} \colon \begin{bmatrix} \xi_1 \\ \xi_2 \\ -d_3 \end{bmatrix} \mapsto \frac{1}{d_2^2\xi_1^2 + d_1^2\xi_2^2 + 4d_1^2d_2^2} \begin{bmatrix} 4d_1^2d_2^2\xi_1 \\ 4d_1^2d_2^2\xi_2 \\ (d_2^2\xi_1^2 + d_1^2\xi_2^2 - 4d_1^2d_2^2)d_3 \end{bmatrix}. \tag{21.102}$$

Note that when d_1, d_2 and d_3 are all set to 1 then formulae (21.101), and (21.102) simplify to (21.83), and (21.84), respectively. An example of interpolation on an ellipsoid with the above algorithm using the change of metric is illustrated with Fig. 21.6.

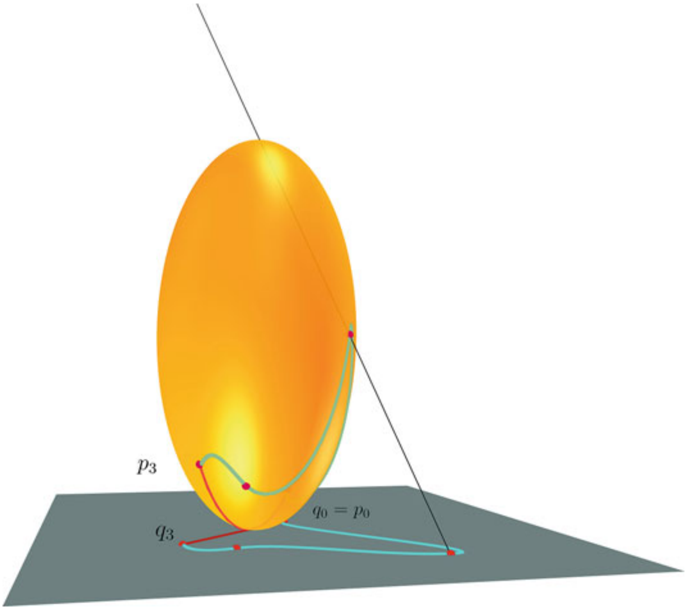


Fig. 21.6 Wrapping back by stereographic projection, see (21.101) and (21.102). The ellipsoid is rolling along the straight line segment

21.4.3 Related Work

In this section we start with a brief review of work related to rolling motions and alternative methods to solve interpolating problems on Riemannian manifolds. The cited literature below is far from being exhaustive. To shed some light on the rich potential of the material in this chapter towards applications, we mention at the end of this section a number of scientific areas where our developments can be used successfully.

21.4.3.1 Rolling Symmetric Spaces

In this chapter we focussed our attention on three classes of manifolds that are particular cases of compact symmetric spaces. Rolling motions of these more general Riemannian manifolds have already been studied in [51], using an extrinsic approach similar to the one presented here. The work [40] addresses the rolling motions of even broader classes of Riemannian manifolds from a theoretical perspective.

21.4.3.2 Rolling Stiefel Manifolds

Stiefel manifolds $St_{n,k}$, consisting of orthonormal k-frames in $\mathbb{R}^n$ play a key role in many engineering applications. Besides $S^{n-1} = St_{n,1}, St_{n,n-1} \cong SO_n$ and $O_n = St_{n,n}$, Stiefel manifolds are in general not symmetric spaces. Finding the kinematic equations for rolling Stiefel manifolds is a much harder challenge. In spite of that, there has been some successful attempts to overcome those difficulties, e.g. [38, 39, 64], but there are still open questions that have not been addressed in a satisfactory manner.

21.4.3.3 Rolling Pseudo-orthogonal and Symplectic Manifolds

The concept of rolling extends quite naturally to pseudo-Riemannian or even to symplectic manifolds. Examples are Lorentzian spheres, pseudo-orthogonal groups and symplectic groups. For instance see [21, 47], and [53], respectively.

21.4.3.4 Control Theoretic Aspects Including Controllability of Rolling

Rolling maps and the corresponding kinematic equations also play an important role in certain optimal control and controllability problems, often related to geometric mechanics; see [5, 9, 24, 42], and more recently, [44, 45, 67]. Ongoing work relating rolling motions to sub-Riemannian optimal control problems will highlight other interesting connections.

21.4.3.5 Intrinsic Rolling

In more recent years, an intrinsic viewpoint has also been successfully applied to study rolling motions of Riemannian and pseudo-Riemannian manifolds. In this approach, the geometric description of the rolling does not depend on any particular embedding. Examples of works that follow this viewpoint are, for instance, [11], [13–15], and [28]. This extends to the study of controllability of the rolling, see for instance [16, 17, 31], and even the rolling of a manifold over another one of different dimension, such as in [18] and [55].

21.4.3.6 Variational and Hamiltonian Approach to Interpolation

Cubic splines and their higher order extensions have been generalized to Riemannian manifolds and are instrumental in many areas of science and technology. Variational cubic splines, in particular, are minimizers of the acceleration under suitable interpolation and boundary conditions. Although the theory of splines on manifolds is already well established, there are still geometric integration challenges to be addressed [12, 19, 20, 56]. The alternative Hamiltonian approach becomes very cumbersome as the degree of the polynomial splines increases. The optimal control problems associated to cubic spline curves already face the complexity that results from the appearence of higher order bundles, cf. [1, 2, 23]. In [6], the Hamiltonian viewpoint was applied to simple variational splines and the corresponding Hamiltonian dynamics studied in detail.

21.4.3.7 The de Casteljau Algorithm and Bézier-like Curves on Riemannian Manifolds

The classical de Casteljau algorithm has a significant importance in modeling and computer aided geometric design. Its generalisation to solve interpolation problems on manifolds was motivated by many engineering applications where the data does not lie on Euclidean spaces, cf. [22, 58, 59].

Recently, another method to solve the problem of fitting a smooth curve to data points on a Riemannian manifold appeared in [29]. Although this method uses the de Casteljau construction to produce Bézier-like curves, a careful choice of the control points guarantees that the solutions also satisfy optimality criteria.

21.4.3.8 Some Applications

There is a considerable amount of real life problems where one has to deal with nonlinear data living in a Riemannian manifold, Lie group or more generally symmetric space. Works might include computer vision [52], pattern recognition [7, 65], robotics [61], even spin dynamics [10], pure mathematics, namely some interplay between geometry and algebra, see [4], or sliding-rolling [32].

For instance, in computer vision applications the data may correspond to a sequence of images of a dynamical scene captured at different times and the objective is to reconstruct the scene from that limited number of observations, [63]. Face recognition problems or age estimation from several facial images is another example where interpolating data is important. In medical applications, interpolation methods may help to follow the evolution of a disease or to set appropriate diagnostics. One application where rolling motions have already been used for recognizing human actions from 3D skeletal data can be found in [65]. In robotics and in space science path planning is particularly important and can be achieved through interpolating methods in the Lie group of Euclidean motions, cf. [62, 66].

Acknowledgements The work of the third author has been supported by OE - national funds of FCT/MCTES (PIDDAC) under project UID/EEA/00048/2019.

References

1. Abrunheiro, L., Camarinha, M., Clemente-Gallardo, J.: Cubic polynomials on Lie groups: reduction of the Hamiltonian system. J. Phys. A Math. Theor. **44**, 355203 (2011)
2. Abrunheiro, L., Camarinha, M., Clemente-Gallardo, J.: Corrigendum: cubic polynomials on Lie groups: reduction of the Hamiltonian system. J. Phys. A Math. Theor. **46**, 189501 (2013)
3. Adler, R.L., Dedieu, J.P., Margulies, J.Y., Martens, M., Shub, M.: Newton's method on Riemannian manifolds and a geometric model for the human spine. IMA J. Numer. Anal. **22**, 359–390 (2002)
4. Agrachev, A.A.: Rolling balls and octonions. Tr. Mat. Inst. Steklova **258**(Anal. i Osob. Ch. 1), 17–27 (2007)
5. Agrachev, A.A., Sachkov, Y.L.: Control Theory from the Geometric Viewpoint. Berlin, Springer (2004)
6. Balseiro, P., Struchi, T., Cabrera, A., Koiller, J.: About simple variational splines from the Hamiltonian viewpoint. J. Geom. Mech. **9**(3), 257–290 (2017)
7. Batista, J., Krakowski, K.A., Silva Leite, F.: Exploring quasi-geodesics on Stiefel manifolds in order to smooth interpolate between domains. In: 2017 IEEE 56th Annual Conference on Decision and Control (CDC), pp. 6395–6402 (2017)
8. Batzies, E., Hüper, K., Machado, L., Silva Leite, F.: Geometric mean and geodesic regression on Grassmannians. Linear Algebra Appl. **466**, 83–101 (2015)
9. Bloch, A.M.: Nonholonomic Mechanics and Control, vol. 24, 2nd edn. Springer, New York (2015)
10. Bloch, A.M., Rojo, A.G.: Kinematics of the rolling sphere and quantum spin. Commun. Inf. Syst. **10**(4), 221–238 (2010)
11. Bryant, R.L., Hsu, L.: Rigidity of integral curves of rank 2 distributions. Invent. Math. **114**(1), 435–461 (1993)
12. Camarinha, M.: The Geometry of Cubic Polynomials on Riemannian Manifolds. Ph.D. thesis, University of Coimbra, Portugal (1996)
13. Chitour, Y., Kokkonen, P.: Rolling manifolds on space forms. Annales de l'Institut Henri Poincare (C) Non Linear Anal. **29**(6), 927–954 (2012)
14. Chitour, Y., Kokkonen, P.: Rolling of manifolds and controllability in dimension three. Mém. Soc. Math. Fr. Nouv. Sér. **147**, 1–162 (2016)
15. Chitour, Y., Godoy Molina, M., Kokkonen, P.: The rolling problem: overview and challenges. In: Stefani, G., Boscain, U., Gauthier, J.P., Sarychev, A., Sigalotti, M. (eds.) Geometric Control Theory and Sub-Riemannian Geometry, pp. 103–122. Springer, Berlin (2014)
16. Chitour, Y., Godoy Molina, M., Kokkonen, P.: The rolling problem: overview and challenges. In: Geometric Control Theory and Sub-Riemannian Geometry. Proceedings of the Meeting on Geometric Control Theory and Sub-Riemannian Geometry, Dedicated to Andrei A. Agrachev on the Occasion of his 60th birthday, Cortona, Italy, May 21–25, 2012, pp. 103–122. Springer, Berlin (2014)
17. Chitour, Y., Godoy Molina, M., Kokkonen, P.: On the controllability of the rolling problem onto the hyperbolic n-space. SIAM J. Control Optim. **53**(2), 948–968 (2015)
18. Chitour, Y., Godoy Molina, M., Kokkonen, P., Markina, I.: Rolling against a sphere: the non-transitive case. J. Geom. Anal. **26**(4), 2542–2562 (2016)
19. Crouch, P., Silva Leite, F.: Geometry and the dynamic interpolation problem. In: Proceeding of American Control Conference, pp. 1131–1136. IEEE, Boston (1991)
20. Crouch, P., Silva Leite, F.: The dynamic interpolation problem: on Riemannian manifolds, Lie groups and symmetric spaces. J. Dyn. Control. Syst. **1**(2), 177–202 (1995)
21. Crouch, P., Silva Leite, F.: Rolling maps for pseudo-orthogonal groups. In: 51st IEEE Conference on Decision and Control, pp. 7485–7491. IEEE, Hawaii (2012)
22. Crouch, P., Kun, G., Silva Leite, F.: The de Casteljau algorithm on Lie groups and spheres. J. Dyn. Control. Syst. **5**(3), 397–429 (1999)

23. Crouch, P., Silva Leite, F., Camarinha, M.: Hamiltonian structure of generalized cubic polynomials. In: Proceeding IFAC Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control, 16-18 March 2000. Princeton University, USA (2000)
24. Cushman, R., Duistermaat, H., Śniatycki, J.: Geometry of nonholonomically constrained systems. Hackensack, World Scientific (2010)
25. de Casteljau, P.: Outillages méthodes calcul. Technical report, A. Citroen, Paris (1959)
26. Egerstedt, M., Martin, C.: Control theoretic splines. Princeton University Press, Princeton (2010)
27. Farin, G.: Curves and Surfaces for CAGD: A Practical Guide, 5th edn. Morgan Kaufmann Publishers Inc., San Francisco (2002)
28. Godoy Molina, M., Grong, E., Markina, I., Silva Leite, F.: An intrinsic formulation of the problem on rolling manifolds. J. Dyn. Control Syst. **18**(2), 181–214 (2012)
29. Gousenbourger, P.Y., Massart, E., Absil, P.A.: Data fitting on manifolds with composite Bézier-like curves and blended cubic splines. J. Math. Imaging Vis. **61**(5), 645–671 (2019)
30. Gromov, M., Rokhlin, V.: Embeddings and immersions in Riemannian geometry. Russ. Math. Surv. **25**(5), 1–57 (1970)
31. Grong, E.: Controllability of rolling without twisting or slipping in higher dimensions. SIAM J. Control Optim. **50**(4), 2462–2485 (2012)
32. Hacisalihoğlu, H.H.: On the rolling of one curve or surface upon another. Proc. R. Ir. Acad. Sect. A **71**, 13–17 (1971)
33. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations. 2nd ed. Springer, Berlin (2006)
34. Helgason, S.: Differential geometry, Lie groups and symmetric spaces. Academic Press, New York (1978)
35. Helmke, U., Hüper, K., Trumpf, J.: Newton's method on Graßmann manifolds (2007). ArXiv:0709.2205v2
36. Hüper, K., Silva Leite, F.: Smooth interpolating curves with applications to path planning. In: 10-th IEEE Mediterranean Conference on Control and Automation. Instituto Superior Técnico, Lisboa (2002). Proceedings on CDROM
37. Hüper, K., Silva Leite, F.: On the geometry of rolling and interpolation curves on S^n, SO_n, and Grassmann manifolds. J. Dyn. Control Syst. **13**(4), 467–502 (2007)
38. Hüper, K., Ullrich, F.: Real Stiefel manifolds: An extrinsic point of view. In: 13th APCA International Conference on Automatic Control and Soft Computing (CONTROLO), pp. 13–18. Ponta Delgada, Azores (2018)
39. Hüper, K., Kleinsteuber, M., Silva Leite, F.: Rolling Stiefel manifolds. Int. J. Syst. Sci. **39**(9), 881–887 (2008)
40. Hüper, K., Krakowski, K.A., Silva Leite, F.: Rolling maps in a Riemannian framework. In: J. Cardoso, K. Hüper, P. Saraiva (eds.) Textos de Matemática. Série B 43, pp. 15–30. Universidade de Coimbra, Departamento de Matemática, Coimbra (2011)
41. Jupp, P., Kent, J.: Fitting smooth paths to spherical data. Appl. Stat. **36**(1), 34–46 (1987)
42. Jurdjevic, V.: Geometric Control Theory. Cambridge University Press, Cambridge (1997)
43. Jurdjevic, V.: Optimal Control and Geometry: Integrable Systems. Cambridge University Press, Cambridge (2016)
44. Jurdjevic, V., Zimmerman, J.: Rolling problems on spaces of constant curvature. In: Bullo, F., Fujimoto, K. (eds.) 3rd Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control, pp. 137–142. IFAC, Nagoya (2006)
45. Kleinsteuber, M., Hüper, K., Silva Leite, F.: Complete controllability of the rolling n-sphere—a constructive proof. In: Bullo, F., Fujimoto, K. (eds.) 3rd IFAC Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control, pp. 143–146. IFAC, Nagoya (2006)
46. Kobayashi, S.: Isometric imbeddings of compact symmetric spaces. Tôhoku Math. J. **20**(2), 21–25 (1968)
47. Korolko, A., Silva Leite, F.: Kinematics for rolling a Lorentzian sphere. In: 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC), Orlando, Fl., USA, pp. 6522–6527 (2011)

48. Krakowski, K.A., Silva Leite, F.: Why controllability of rolling may fail: a few illustrative examples. In: Pré-Publicações do Departamento de Matemática, 12–26, pp. 1–30. University of Coimbra, Coimbra (2012)
49. Krakowski, K.A., Silva Leite, F.: An algorithm based on rolling to generate smooth interpolating curves on ellipsoids. Kybernetika **50**(4), 544–562 (2014)
50. Krakowski, K., Silva Leite, F.: Geometry of the Rolling Ellipsoid. Kybernetika **52**(2), 209–223 (2016)
51. Krakowski, K.A., Machado, L., Silva Leite, F.: Rolling symmetric spaces. In: Geometric Science of Information. Lecture Notes in Computer Science, vol. 9389, pp. 550–557. Springer, Berlin (2015)
52. Machado, L., Pina, F., Silva Leite, F.: Rolling maps for the essential manifold. In: Dynamics, games and science. CIM Series Mathematics Science, vol. 1, pp. 399–415. Springer, Cham (2015)
53. Marques, A., Silva Leite, F.: Constructive proof for complete controllability of a rolling pseudo-hyperbolic space. 2018 13th APCA International Conference on Control and Soft Computing (CONTROLO), pp. 19–24 (2018)
54. Marthinsen, A.: Interpolation in Lie groups. SIAM J. Numer. Anal. **37**(1), 269–285 (1999)
55. Mortada, A., Kokkonen, P., Chitour, Y.: Rolling manifolds of different dimensions. Acta Appl. Math. **139**(1), 105–131 (2015)
56. Noakes, L., Heinzinger, G., Paden, B.: Cubic splines on curved spaces. IMA J. of Math. Control Inf. **6**, 465–473 (1989)
57. Nomizu, K.: Kinematics and differential geometry of submanifolds. Tôhoku Math. J. (2) **30**(4), 623–637 (1978)
58. Park, F., Ravani, B.: Bézier curves on Riemannian manifolds and Lie groups with kinematics applications. ASME J. Mech. Des. **117**, 36–40 (1995)
59. Popiel, T., Noakes, L.: Bézier curves and C^2 interpolation in Riemannian manifolds. J. Approx. Theory **148**(2), 111–127 (2007)
60. Sharpe, R.: Differential Geometry: Cartan's Generalization of Klein's Erlangen Program. Springer, Berlin (1997)
61. Shen, Y., Hüper, K., Leite, F.S.: Smooth Interpolation of Orientation by Rolling and Wrapping for Robot Motion Planning. In: IEEE International Conference on Robotics and Automation (ICRA2006), pp. 113–118. IEEE, Orlando (2006)
62. Smith, S., Broucke, M., Francis, B.: Curve shortening and the rendezvous problem for mobile autonomous robots. IEEE Trans. Autom. Control **52**, 1154–1159 (2007)
63. Srivastava, A., Turaga, P.: Riemannian computing in computer vision. Springer, Berlin (2016)
64. Ullrich, F.: Rolling maps for real Stiefel manifolds. Master's thesis, Institute of Mathematics, Julius-Maximilians-Universität Würzburg, Germany (2017)
65. Vemulapalli, R., Chellappa, R.: Rolling rotations for recognizing human actions from 3d skeletal data. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, pp. 4471–4479 IEEE, Las Vegas (2016)
66. Younes, L.: Shapes and diffeomorphisms. Springer, Berlin (2010)
67. Zimmerman, J.: Optimal control of the sphere S^n rolling on E^n. Math. Control Signals Syst. **17**, 14–37 (2005)

Part VI
Applications

Chapter 22
Manifold-Valued Data in Medical Imaging Applications

Maximilian Baust and Andreas Weinmann

Contents

Abstract The last decade has witnessed a considerable amount of research being devoted on how to process big and often unstructured data. However, one often neglects the fact that a considerable portion of today's data deluge is actually structured, particularly when we consider measured data. The reason for this fact

M. Baust (✉)
TU Munich, Chair of Computer Aided Medical Procedures, Garching, Germany
e-mail: maximilian.baust@tum.de

A. Weinmann
Department of Mathematics and Natural Sciences, Hochschule Darmstadt, Darmstadt, Germany
e-mail: andreas.weinmann@h-da.de

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_22

is that sensors either directly record structured data, e.g., object poses, or are used to record data of a certain type, i.e., magnetic resonance images of the human heart. In both cases, the measured data is subject to physical or physiological constraints causing the measured data to enjoy a manifold structure. In this chapter, we make extensive use of this observation and discuss several applications within the realm of medical imaging. We briefly discuss the general mathematical structure of these problems and elaborate on why recently proposed formulations of manifold-valued regularizers are applicable to them. We describe the employed numerical schemes to provide application-focused readers with a guide to manifold-valued regularization techniques. Further, we discuss three entirely disjoint applications: regularization of pose signals for 3D freehand ultrasound compounding, estimation and regularization of diffusion tensors measured by magnetic resonance imaging, and estimation and regularization of shape signals. Finally, we discuss possible extensions.

22.1 Introduction

The foundation of this chapter is the observation that sensor or imaging data itself as well as data derived from such data is highly structured. This observation is particularly true in the realm of medical imaging, because many physiological systems are subject to physical and biochemical constraints. In the course of this chapter we will make heavy use of this observation and demonstrate how variational techniques for manifold-valued data can be applied to medical imaging as well as the understanding of it.

22.1.1 Motivation

Let us start by considering the case of 3D freehand ultrasound, where a series of 2D ultrasound images is compounded into a 3D ultrasound volume by using the data of an external tracking system. This imaging modality has become very popular over the course of the last two decades due to the availability of low-cost 2D ultrasound systems as well as reasonably cheap optical or electromagnetic tracking systems of sufficient quality. The compounding process itself, i.e., the process of generating a 3D volume out of the 2D measurements, can be explained in a very picturesque way: We can consider all 2D ultrasound images as being attached to a clothesline. Next, we define a 3D grid of sampling points with respect to an arbitrarily chosen world coordinate system. Finally, we compute the intensity value at each 3D grid point via a specified interpolation method taking into account image data from surrounding 2D ultrasound images. Thereby, the position and orientation, i.e., the pose, of each ultrasound image is determined by the output of the employed tracking system. The whole process is obviously highly influenced by how accurately the position and orientation of the individual ultrasound frames can be determined or measured in 3D space. Inaccuracies in the measurements process thus affect the interpolation process of all grid points in the proximity of one 2D ultrasound frame. By noting that the series of poses measured by the tracking system can be interpreted as a time-series with values in the manifold SE(3), we will see that it is possible to improve the compounding outcome via regularization techniques for manifold-valued data.

As a second imaging technique, we consider diffusion tensor imaging which is a special form of magnetic resonance imaging that exploits the diffusion of water molecules within an organism's tissue. Based on multiple diffusion-weighted acquisitions, it is possible to fit diffusion tensors at each location within the measurement volume, which indicate the main directions of water diffusion. By taking the orientations of multiple neighboring tensors into account, it is then possible to track fibrous structures within the imaged tissue, e.g., in order to visualize connected areas in the human brain as an important planning step before cranial interventions. This imaging modality is inherently manifold-valued as each grid point of the imaging volume carries a positive definite 3×3 matrix which represents the covariance matrix of a normal distribution. In the corresponding section, we see how to extend pure regularization techniques for manifold-valued data to simultaneously fit and regularize diffusion tensors and demonstrate how this approach benefits the creation of connectomes, i.e., wiring diagrams of the human brain via fibre tracking.

While the manifold structure is quite obvious in the case of pose signals and DTI, there are situations where this is less obvious. Consider two and three dimensional shapes of organs imaged by various medical imaging modalities. A good example is the human heart; due to its bio-mechanical and functional properties its shape cannot vary arbitrarily. Thus, an image of the heart, be it acquired with magnetic resonance imaging (MRI), computed tomography (CT) or ultrasound (US), cannot vary arbitrarily either. As a consequence, two-dimensional (short-axis) images of the human heart acquired with MRI have a specific appearance and, roughly speaking, less degrees of freedom than the plain number of pixels. This observation translates well to two-dimensional shapes obtained from segmentations of the myocardial wall or other parts of the human circulatory system, such as the abdominal aorta. By approximating these manifold structures with classical parametric shape manifolds, we will demonstrate how techniques for regularization of manifold-valued signals can be applied to geometry processing and image segmentation.

22.1.2 General Model

Each of the applications described previously has its peculiarities in regards to the chosen manifold (including the Riemannian metric). However, all of these applications have a lot of commonalities as well and we will briefly describe them in this section. For the moment, we restrict ourselves to a one-dimensional setting (as appearing in time series of poses or shapes) and consider manifold-valued signals

$\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_n) \in \mathcal{M}^n$. The problems discussed in this chapter are either of the form

$$\min_{\mathbf{x} \in \mathcal{M}^n} \sum_{i=1}^{n} D(\mathbf{x}_i, \mathbf{y}_i) + \lambda \sum_{i=1}^{n-1} R(\mathbf{x}_i, \mathbf{x}_{i+1}), \tag{22.1}$$

or

$$\min_{\mathbf{x} \in \mathcal{M}^n} \sum_{i=1}^{n} D(\mathcal{A}(\mathbf{x}_i), y_i) + \lambda \sum_{i=1}^{n-1} R(\mathbf{x}_i, \mathbf{x}_{i+1}), \tag{22.2}$$

as well as their multivariate analogues. In both scenarios, D is an atomic data fidelity term, ensuring that the computed solution $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ is well explained by the input data $\mathbf{y}$, R is a regularizing term incorporating a priori regularity knowledge such as piecewise smoothness of the computed solution $\mathbf{x}$ and $\lambda > 0$ is a regularization parameter allowing the user to trade data fidelity for regularity. In the first scenario, the input data is necessarily manifold-valued, e.g., in case of pose signals. In the second scenario, there is an operator $\mathcal{A}$ which can be an indirect measurement operator or a "construction" operator as in the case of DTI, as well as m measurements y_i, $i = 1, \ldots, n$. (More precisely, in DTI, a reasonable model involves the operator $\mathcal{A}$ which maps a diffusion tensor to a set of corresponding DWIs.) Generally, in case of the second scenario, we typically have a series of real valued measurements $y = (y_1, \ldots, y_n) \in \mathrm{R}^{d \times n}$ from which the manifold-valued signal is estimated via the imaging operator $\mathcal{A}$.

The atomic data terms in the first scenario, as well as the regularizers in both the first and the second one, are defined via a previously chosen Riemannian metric d. We recall that a Riemannian metric is defined as a smoothly varying scalar product in every point of $\mathcal{M}$. A Riemannian manifold $\mathcal{M}$ becomes a metric space by defining the distance $\mathrm{d}(\mathbf{p}, \mathbf{q})$ between points $\mathbf{p}$, $\mathbf{q}$ as the smallest length of the arcs connecting $\mathbf{p}$, $\mathbf{q}$. In our setting, we employ the distance d to define

$$D(\mathbf{x}_i, \mathbf{y}_i) = h \circ \mathrm{d}(\mathbf{x}_i, \mathbf{y}_i) \tag{22.3}$$

and

$$R(\mathbf{x}_i, \mathbf{x}_{i+1}) = h \circ \mathrm{d}(\mathbf{x}_i, \mathbf{x}_{i+1}). \tag{22.4}$$

Here, h is one of the following functions: $h(s) = s$ leading to an ℓ_1-type penalization, $h(s) = s^2/2$ leading to an ℓ_2-type penalization, and

$$h(s) = \begin{cases} s^2, & s < 1/\sqrt{2}, \\ \sqrt{2}s - 1/2, & \text{otherwise}, \end{cases} \tag{22.5}$$

yielding the manifold-valued equivalent of the well-known Huber-norm [43], i.e., a differentiable compromise between the ℓ_1-norm and the ℓ_2-norm. As we will see in the remainder of this chapter, Huber-type weighting of the Riemannian metric has proven to yield very good results.

In case of the second scenario, i.e., the manifold-valued signal being estimated during the optimization process, the data terms will be discussed in the respective subsections. Further, we consider the multivariate situation in Sect. 22.3 in the context of DTI. Methods to compute (approximate) solutions of the respective model are presented in the corresponding sections.

22.1.3 Organization of the Chapter

We start out by considering pose signals in Sect. 22.2. We explain a Riemannian manifold structure on the space of poses, formulate the total variation model and related models for pose signals, and explain the algorithmic approach. We apply the developed method for the compounding of ultrasound images. Then, in Sect. 22.3, we deal with diffusion tensor imaging. We discuss the data manifold and introduce a multivariate TV model incorporating indirect measurements in the DTI setting. We present a generalized forward-backward scheme for solving the model. We show the benefit of the method for nerve fiber tracking. Finally, in Sect. 22.4, we consider geometry processing and medical image segmentation. We discuss the manifold structure of the considered shape spaces, derive a basic model for denoising and a corresponding algorithm. We apply the method for regularizing slicewise segmentations of a part of an abdominal aorta. Finally, we point out extensions to joint segmentation and regularization as well as to the multivariate setup.

22.2 Pose Signals and 3D Ultrasound Compounding

In this section we consider denoising time series of poses and apply it to ultrasound compounding. In order to explain the model and the algorithm, we need some basics on the underlying manifold SE(3). We gather these facts and specify the model in Sect. 22.2.1, the numerical approach is explained in Sect. 22.2.2 and experiments are conducted in Sect. 22.2.3. We conclude with a discussion in Sect. 22.2.4. This section is mostly based on [35].

22.2.1 Problem-Specific Manifold and Model

The output of a tracking system is a series of Euclidean transformations, often called poses, consisting of a rotation matrix R and a translation vector t. The set of all such tuples of the form (R, t) constitute the Euclidean motion group SE(3). As a group, SE(3) is a semi-direct product of the rotation group SO(3) and the translation group on $\mathbb{R}^3$ represented by $t \in \mathbb{R}^3$. This means that the rotational component of the second pose acts on the translational component of the first pose before adding them up. This structure is reflected by the matrix representation of SE(3) of the form

$$\mathrm{SE}(3) = \left\{ \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix} : \ R \in \mathrm{SO}(3), \ t \in \mathbb{R}^3 \right\}. \tag{22.6}$$

Then, SE(3) can be viewed as a subgroup of the group of invertible 4×4 matrices $\mathrm{GL}(4, \mathbb{R})$ with the matrix multiplication as group operation.

We consider the manifold $\mathcal{M} = \mathrm{SE}(3)$ of Euclidean transformations and endow it with a Riemannian metric as follows. For a pose $\mathbf{p} \in \mathcal{M}$ we denote the tangent space at $\mathbf{p}$ by $T_{\mathbf{p}}\mathcal{M}$. We have a closer look at the elements of the tangent space $T_{\mathbf{e}}\mathcal{M}$ at the identity $\mathbf{e}$: Any $v \in T_{\mathbf{e}}\mathcal{M}$ is of the form

$$v = \begin{pmatrix} \omega_v & t_v \\ 0 & 0 \end{pmatrix}, \tag{22.7}$$

where $t_v \in \mathbb{R}^3$ and the skew symmetric matrix ω_v is given by

$$\omega_v = \begin{pmatrix} 0 & -\omega_v^z & \omega_v^y \\ \omega_v^z & 0 & -\omega_v^x \\ -\omega_v^y & \omega_v^x & 0 \end{pmatrix}. \tag{22.8}$$

Here $\omega_v^x, \omega_y^v, \omega_z^v$ denote a kind of infinitesimal angular displacements with respect to the corresponding axis. A Riemannian metric in the identity is given by

$$\mathfrak{g}_{\mathbf{e}}(v, w) = \mathrm{trace}(\omega_v^T \omega_w) + t_v \cdot t_w, \tag{22.9}$$

where $v, w \in T_{\mathbf{e}}\mathcal{M}$. For a general pose $\mathbf{q} = \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix}$, and tangent vectors $\mathbf{q}v, \mathbf{q}w$ sitting in $\mathbf{q}$ which are represented as matrix products of $\mathbf{q}$ and v, w as in (22.7), we let

$$\mathfrak{g}_{\mathbf{q}}(\mathbf{q}v, \mathbf{q}w) = \mathfrak{g}_{\mathbf{e}}(v, w) \tag{22.10}$$

to obtain a left-invariant Riemannian metric. This means that the metric is invariant with respect to the choice of a global coordinate frame. However, $\mathfrak{g}$ is not right-invariant, which means that it is not invariant regarding the choice of a body-fixed reference frame. In the context of 3D freehand ultrasound compounding, this means that by using this metric the computed results are independent of the choice of a global reference frame, such as the one provided by the tracking system itself, but not independent regarding the calibration of the ultrasound transducer. We point out that the proposed approach can also be carried out for right-invariant metrics (by symmetry) if desired. Achieving both left- and right-invariance is not possible, but this is not a shortcoming of the proposed method, rather than an intrinsic structural issue of SE(3). (We note that such problems do not appear for compact or commutative groups in which case every left or right invariant metric is bi-invariant; however, SE(3) is neither compact nor commutative.) In this chapter, we concentrate on the left-invariant metric defined in (22.9) and refer the interested reader to [17, 58, 87, 88] for further reading.

Next, we consider the exponential mapping $\exp_{\mathbf{p}} : T_{\mathbf{p}}\mathcal{M} \to \mathcal{M}$, which maps a tangent vector w in the tangent space $T_{\mathbf{p}}\mathcal{M}$ of the pose $\mathbf{p}$ to a new pose $\exp_{\mathbf{p}}(w)$ by following the geodesic starting at $\mathbf{p}$ with direction w w.r.t. the above Riemannian metric. (We note that this exponential function does not agree with the Lie group exponential in SE(3).) We consider a vector $v \in \mathfrak{se}(3)$, where $\mathfrak{se}(3)$ denotes the tangent space at the identity and where v is given by (22.7) with $t_v \in \mathbb{R}^3$ being the translational part and ω_v being the $\mathfrak{so}_3$ part of v. Here, $\mathfrak{so}_3$ denotes the tangent space of SO(3) at the identity which equals the space of skew-symmetric matrices. Given a pose $\mathbf{p} = \begin{pmatrix} R_p & t_p \\ 0 & 1 \end{pmatrix} \in \mathrm{SE}(3)$, we have with $w = \mathbf{p}v$ that

$$\exp_{\mathbf{p}}(\mathbf{p}v) = \begin{pmatrix} R_p \exp(\omega_v) & t_p + t_v \\ 0 & 1 \end{pmatrix}, \tag{22.11}$$

where $\exp(\omega_v)$ on the right hand side denotes the matrix exponential function of $\omega_v \in \mathfrak{so}_3$. To compute the matrix exponential of the skew-symmetric matrix ω_v, we use the Rodrigues' formula as explained in [54]. We note that the geodesics induced by the Riemannian metric (22.9) on SE(3) are precisely the geodesics in the product manifold of SO(3) and $\mathbb{R}^3$, where SO(3) is equipped with its bi-invariante metric.

The inverse of the Riemannian exponential mapping in SE(3) at $\mathbf{p}$, denoted by $\log_{\mathbf{p}}$, maps another pose $\mathbf{q}$ to the tangent vector $\log_{\mathbf{p}}(\mathbf{q})$ in $T_{\mathbf{p}}\mathcal{M}$ sitting at the pose $\mathbf{p}$ pointing towards $\mathbf{q}$. Note that this mapping is locally defined in a neighborhood of $\mathbf{p}$. We get for poses $\mathbf{p}$, $\mathbf{q}$ that

$$\log_{\mathbf{p}}(\mathbf{q}) = \begin{pmatrix} R_p \log(R_q R_p^T) & t_q - t_p \\ 0 & 0 \end{pmatrix}, \tag{22.12}$$

where the log on the right hand side denotes the principal logarithm of matrices (which may be viewed as componentwise principal logarithm on the eigenvalues). Concerning the computation of the principal matrix logarithm of the rotation matrix $R_q R_p^T$, we again refer to [54]. We have that $\exp_{\mathbf{p}}(\log_{\mathbf{p}}(\mathbf{q})) = \mathbf{q}$. Further, if $\mathbf{q}$ is in the domain of $\log_{\mathbf{p}}$, we have for the distance $\mathrm{d}(\mathbf{p}, \mathbf{q})$ between $\mathbf{p}$ and $\mathbf{q}$ that

$$\mathrm{d}(\mathbf{p}, \mathbf{q}) = \sqrt{\mathfrak{g}_{\mathbf{p}}(\log_{\mathbf{p}}(\mathbf{q}), \log_{\mathbf{p}}(\mathbf{q}))} = \sqrt{\| \log(R_q R_p^T)\|_F^2 + \|t_p - t_q\|^2}, \tag{22.13}$$

where $\|\log(R_q R_p^T)\|_F$ denotes the Frobenius norm of $\log(R_q R_p^T)$, and where $\|t_p - t_q\|$ denotes the euclidean norm of $t_p - t_q$.

Model for Pose Denoising. We specify the general model explained in Sect. 22.1.2 to the situation of pose denoising. Since we have a time series $\mathbf{y} = (\mathbf{y}_i)_{i=1}^n$ of poses as an input, we consider the problem

$$\min_{\mathbf{x} \in \mathcal{M}^n} \sum_{i=1}^{n} h \circ \mathrm{d}(\mathbf{x}_i, \mathbf{y}_i) + \lambda\, h' \circ \mathrm{d}(\mathbf{x}_i, \mathbf{x}_{i+1}), \tag{22.14}$$

where d is the distance on the manifold of poses induced by the Riemannian metric and which is given by (22.13); further, h, h' are potentials given by either $h(s) = s$, $h(s) = s^2/2$ or by the Huber potential given by (22.5), and for h' analogously; $\lambda > 0$ is the regularization parameter. In order to distinguish these three cases in an intuitive manner, we denote the case $h(s) = s$ by ℓ_1, the case $h(s) = s^2/2$ by ℓ_2 and the case (22.5) by HUBER. As $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_{i+1})$ can be considered as a manifold-valued, first-order forward difference, we interpret R as a first order approximation of the classical Tikhonov regularizer in case of $h'(s) = s^2/2$, or the total variation in case of $h'(s) = s$, respectively. In case of (22.5), R can be regarded as a pose-valued differentiable approximation of the total variation regularizer, which can be used to avoid staircasing problems associated with total variation denoising, cf. Chambolle and Pock [62]. Similar to the data term, we use abbreviations for the different regularization types, i.e., TV (ℓ_1-case), TKHV (ℓ_2-case), and HUBER.

22.2.2 Numerical Approach

As a strategy to minimize the functional in (22.14) we employ a cyclic proximal point algorithm (CPPA) as proposed in [83]; see also Chapter 2 of this book for details and for a discussion of further algorithmic approaches. CPPAs for Hadamard spaces have been proposed in [7] and inspired their use in [83]. A reference for CPPAs in vector spaces is [19]. We recall that a CPPA consists of a cyclic application of proximal mappings to the individual atomic data terms $D(\mathbf{x}_i, \mathbf{y}_i) = h \circ \mathrm{d}(\mathbf{x}_i, \mathbf{y}_i)$ and the atomic regularization terms $R(\mathbf{x}_i, \mathbf{x}_{i+1}) = h' \circ \mathrm{d}(\mathbf{x}_i, \mathbf{x}_{i+1})$. A pseudo-code is given in Algorithm 1. The function `compLambda` determines the step size and the functions `proxData` and `proxR1` realize and the proximal mappings on SE(3).

We now discuss this approach in more details. We recall that the general definition of the proximal mapping of a function F with parameter λ is given by

$$\mathrm{prox}_{\lambda F}(u) = \mathrm{argmin}_v F(v) + \tfrac{1}{2\lambda}\mathrm{d}(v, u)^2, \tag{22.15}$$

i.e., the result of the proximal mapping is a point v^* which minimizes the right hand side and is a compromise between closeness to u and having a small functional value. The crucial point in our situation of TV minimization is that the proximal

Algorithm 1 CPPA scheme for solving (22.14)

```
Input: Signal $\mathbf{y}$, regularization parameter $\lambda$, number of steps $l$, and chosen potential functions $h, h'$.
Ouput: Denoised signal $\mathbf{x}$.
for $j = 1, \dots, l$
  $\lambda_j \leftarrow$ CompLambda($j$)    //compute rel. parameter
  for $i = 1, \dots, n$
    $\mathbf{x}_i \leftarrow$ ProxData($\lambda_j$,$\mathbf{x}_i$,$\mathbf{y}_i$)    //proximal mapping data term
  for $i = 1, \dots, n-1$
    ($\mathbf{x}_i$,$\mathbf{x}_{i+1}$) $\leftarrow$ ProxFirstOrder($\lambda_j$, $\lambda$, $\mathbf{x}_i$,$\mathbf{x}_{i+1}$) //proximal mapping TV atom
```

mappings of the data and the first order difference terms can be explicitely computed in terms of geodesics in the manifold; for derivations, we refer to [83]. We have the following explicit formulae for the appearing proximal mappings where the exp and the log function in SE(3) are implemented via (22.11) and (22.12), respectively. The proximal mapping for the data term atom $D_i(\mathbf{x}_i) = h \circ \mathrm{d}(\mathbf{x}_i, \mathbf{y}_i)$, for a given data point $\mathbf{y}_i$ is given by

$$\mathrm{prox}_{\lambda D_i}(\mathbf{x}_i) = \exp_{\mathbf{x}_i}\left(t\ \log_{\mathbf{x}_i} \mathbf{y}_i\right), \tag{22.16}$$

where the parameter t is chosen, depending on the kind of data term used. For the ℓ^2-type data term $t = \lambda/(1+\lambda)$, and for the ℓ^1 type data term $t = \lambda/\mathrm{d}(\mathbf{x}_i, \mathbf{y}_i)$, if $\lambda < \mathrm{d}(\mathbf{x}_i, \mathbf{y}_i)$, and $t = \mathrm{d}(\mathbf{x}_i, \mathbf{y}_i)$ else. For the Huber type data term, we have

$$t = \begin{cases} \frac{2\lambda}{1+2\lambda}, & \text{if } \mathrm{d}(\mathbf{x}_i, \mathbf{y}_i) < \frac{\omega(1+2\lambda)}{\sqrt{2}}, \\ \min\left(\mathrm{d}(\mathbf{x}_i, \mathbf{y}_i), \sqrt{2}\lambda\right)/\mathrm{d}(\mathbf{x}_i, \mathbf{y}_i), & \text{otherwise.} \end{cases} \tag{22.17}$$

The proximal mapping for the TV, the Huber and the analogue of the classical Tichonov regularizer atoms $R_i(\mathbf{x}_i, \mathbf{x}_{i+1}) = h \circ \mathrm{d}(\mathbf{x}_i, \mathbf{x}_{i+1})$, are given by

$$(\mathrm{prox}_{\lambda R_i}\mathbf{x})_i = \exp_{\mathbf{x}_i}\left(t\ \log_{\mathbf{x}_i} \mathbf{x}_{i+1}\right), \tag{22.18}$$

$$(\mathrm{prox}_{\lambda R_i}\mathbf{x})_{i+1} = \exp_{\mathbf{x}_{i+1}}\left(t\ \log_{\mathbf{x}_{i+1}} \mathbf{x}_i\right), \tag{22.19}$$

where $t = \lambda/(1+2\lambda)$ for the ℓ_2-type regularization. For the TV regularization $t = \lambda$, if $\lambda < \mathrm{d}(\mathbf{x}_i, \mathbf{x}_{i+1})/2$, $t = \mathrm{d}(\mathbf{x}_i, \mathbf{x}_{i+1})/2$ otherwise. In case of Huber regularization, we have

$$t = \begin{cases} \frac{2\lambda}{1+4\lambda}, & \text{if } \mathrm{d}(\mathbf{x}_i, \mathbf{x}_{i+1}) < \frac{1+4\lambda}{\sqrt{2}}, \\ \min\left(\mathrm{d}(\mathbf{x}_i, \mathbf{x}_{i+1})/2, \sqrt{2}\lambda\right)/\mathrm{d}(\mathbf{x}_i, \mathbf{x}_{i+1}), & \text{otherwise.} \end{cases} \tag{22.20}$$

During the iteration of Algorithm 1, the step size parameter λ_r of the proximal mappings is successively decreased. In this way, the penalty for deviation from the previous iterate is successively increased. It is chosen in a way such that the sequence λ_r is square-summable but not summable. This is moderate enough to not enforce convergence by step size decay. More precisely, we use the sequence $\lambda_r = 0.25r^{-0.95}$ in the algorithmic realization.

At this point, let us note that the proximal mappings are possibly multivalued in the general manifold-setting since the uniqueness of the minimizers cannot always be guaranteed. However, in the scenarios we consider, this is a rather mathematical pathology which we did not observe in practice. In fact, such situations only occur on negligible sets of data; we refer to the discussion in [83].

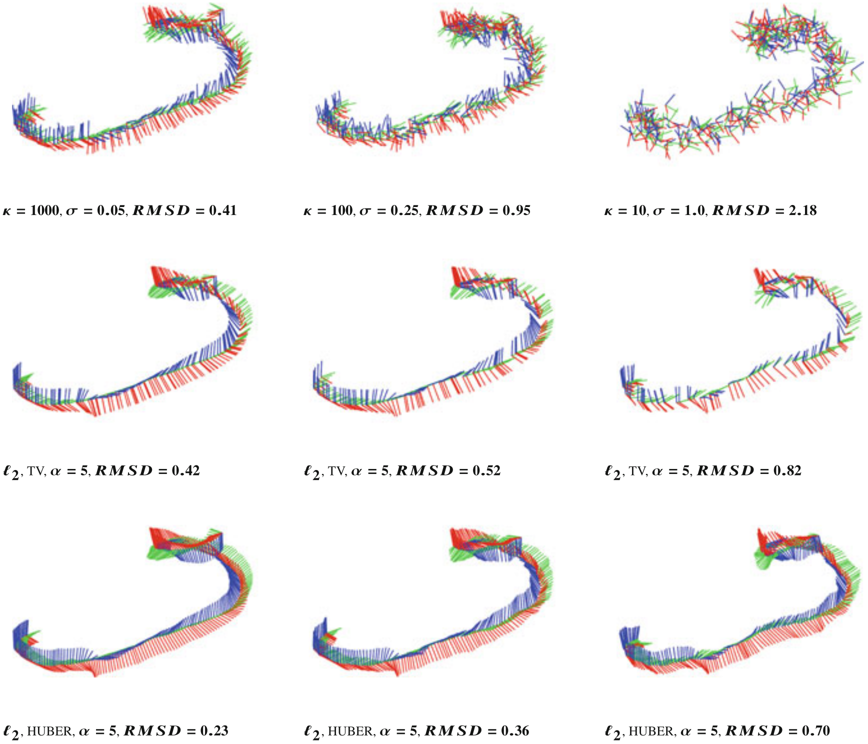
22.2.3 Experiments

We start by considering entirely synthetic experiments using an artificial ground truth pose signal. We added noise to the translational as well as the rotational components of each pose as shown in the first row of Fig. 22.1. More precisely, for the rotational components, we employed Gibbs sampling for a vector-valued von Mises-Fisher distribution with the probability density function being proportional to

$$\exp((\kappa, 0, 0, 0)^T \cdot p) \text{ for } \|p\|_2 = 1, \tag{22.21}$$

where $p \in \mathbf{S}^3$ denotes the directional component of a pose $\mathbf{p}$ [21]. By choosing $\kappa = 1000, 100, 10$ and $\sigma = 0.05, 0.25, 1.0$ for the Gaussian noise which we used on the translational part we obtain the three test data items depicted in the first row of Fig. 22.1. In order to determine the best parameter settings we performed a large grid search with the various weightings for the data term, i.e., ℓ_1, ℓ_2 and HUBER and different regularization types, i.e., TV, TKHV and HUBER. In addition to first order regularization, we also tried out second order regularization approaches [8, 18]. While we noticed that including second order regularization yield slightly better results than the best parameter combination for first order regularization, it increases the total runtime significantly. Thus, we derived a set of recommended parameters for first order Huber-type regularization which yields a good compromise between regularization performance and algorithm runtime, cf. third row of Fig. 22.1. We display the results of first order TV regularization in the second row of Fig. 22.1 and again refer the interested reader to [35] for further details.

For demonstrating the potential of manifold-valued regularization for pose signals, we consider 3D freehand ultrasound imaging. We designed a setup that facilitates to track an Ultrasonix L14-5/38 GPS linear probe with multiple tracking systems simultanously: besides an integrated EM sensor, we used an external EM sensor (in conjunction with a NDI Ascension EM tracking system) as well as an optical marker, visible to an NDI Polaris optical tracking system standing on a tripod, and a KUKA iiwa 7 R800 as a mechanical tracking system. For a detailed



$\kappa = 1000, \sigma = 0.05, \boldsymbol{RMSD} = 0.41$ $\kappa = 100, \sigma = 0.25, \boldsymbol{RMSD} = 0.95$ $\kappa = 10, \sigma = 1.0, \boldsymbol{RMSD} = 2.18$

ℓ_2, TV, $\alpha = 5, \boldsymbol{RMSD} = 0.42$ ℓ_2, TV, $\alpha = 5, \boldsymbol{RMSD} = 0.52$ ℓ_2, TV, $\alpha = 5, \boldsymbol{RMSD} = 0.82$

ℓ_2, HUBER, $\alpha = 5, \boldsymbol{RMSD} = 0.23$ ℓ_2, HUBER, $\alpha = 5, \boldsymbol{RMSD} = 0.36$ ℓ_2, HUBER, $\alpha = 5, \boldsymbol{RMSD} = 0.70$

Fig. 22.1 Visual comparison of regularization results: synthetic data with various noise levels is displayed in the first row. Results obtained with ℓ_2-type data term and total variation regularization are displayed in the second row. The best results, also indicated by the respective root mean squared deviation values, are obtained with Huber regularization (third row)

description of the setup, and in particular of its calibration procedure, we refer the interested reader to [35] again. To visualize the impact of the regularization, particularly on the small EM coil integrated in the US transducer, we built an artificial Lego phantom as described in [86], immersed it in water and performed the image acquisition with the aforementioned system. Due to the regularity of the Lego pieces a good visual feedback of the quality of the compounding can be obtained. One can observe in Fig. 22.2 that the reconstruction using the tracking data of the integrated EM coil is significantly degraded in comparison to the compounded volume obtained from the regularized data.

The same setup was used for a scan of a human forearm, where the robot was set into gravity-compensation-mode in order to simulate a freehand acquisition, cf. Fig. 22.3. The highly degraded tracking data of the internal tracking system leads to jagged reconstructions. After applying the proposed regularization method, one obtains a significant reduction of these artifacts. It is important to note though, that the presented regularization technique can only correct for the more local

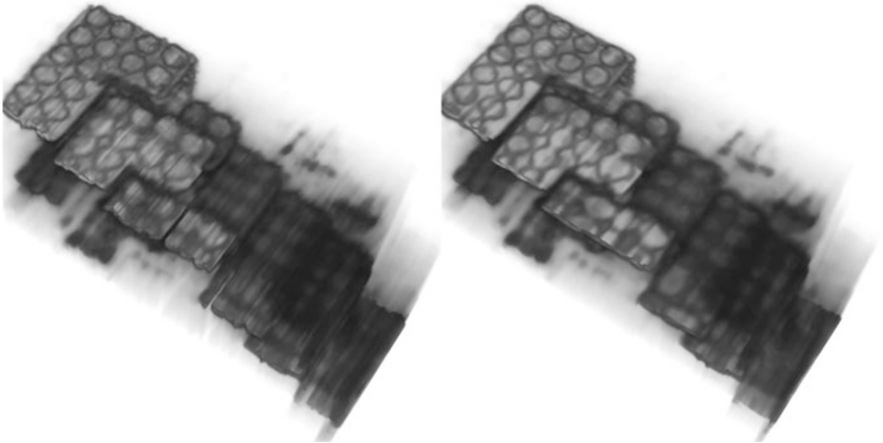


Fig. 22.2 Lego Phantom experiment: rendering of a 3D compounding of the built Lego phantom immersed in water. The compounding computed with the original data of the electromagnetic tracking system integrated in the ultrasound probe is shown on the left. The compounding obtained after applying the regularization is shown on the right

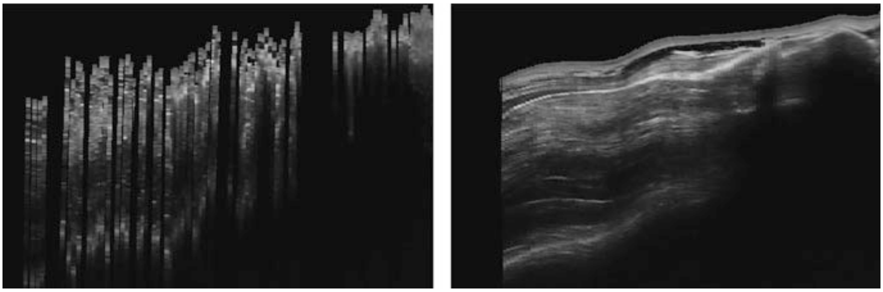


Fig. 22.3 Human Forearm experiment: due to the presence of a heavy metallic object, which has been deliberately introduced into the magnetic field of the tracking system, the compounding using the original tracking data of the internal tracking system shows severe artifacts. By applying manifold-valued regularization, these artifacts can be corrected to a large extend

perturbations causing the jags, but it has obviously no means of distinguishing coarse scale perturbations, yielding the wavy appearance, from intentional motions made by the operator.

22.2.4 Discussion

We have demonstrated that applying the proposed variational regularization methods to the output of noisy tracking systems can be highly beneficial for freehand imaging modalities, such as 3D freehand ultrasound imaging. This is particularly

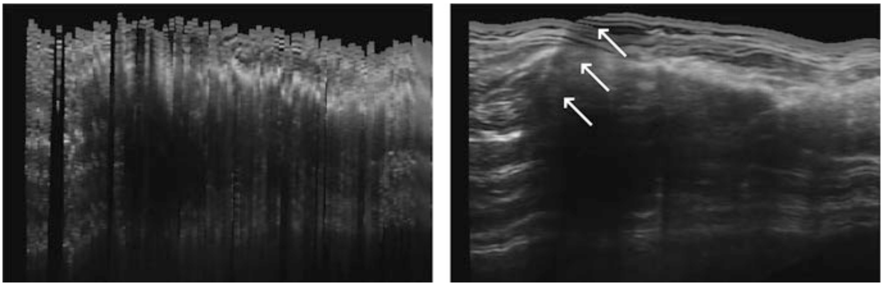


Fig. 22.4 Discontinuity preservation: In contrast to naive smoothing algorithms, the presented regularization maintains discontinuities in the pose data. Whereas the reconstruction using original data of the integrated EM system shows significant artifacts as well as a clearly visible discontinuity in the tissue layers caused by a sudden movement of the probe (left panel), the compounding using the regularized data yields much more evident tissue layers (right panel), while still preserving the discontinuity which manifests itself in a clearly visible darkening of the tissue layers (indicated by the white arrows)

true in case of EM tracking when using small coils, which is the case for the coil integrated in the ultrasound transducer itself.

In contrast to related approaches for pose filtering, the proposed approach has several desirable properties: In contrast to Gaussian filtering of the six parameters individually it respects the intrinsic geometry of the space of poses as it models each pose as an element of SE(3). In contrast to Kalman filtering [80], to Kalman filtering in combination with fusing additional information [1, 34, 48, 50, 79], or to Bayesian approaches [66], the presented method exhibits information flow in both directions—not only in forward direction which is due to the fact that the regularization terms couple all poses. Information flow in both directions also facilitates the preservation of sharp discontinuities due to sudden pose changes as demonstrated in Fig. 22.4. Furthermore, it is model-free as opposed to model-based approaches, cf. Lugez et al. [51] and Sadjadi et al. [65].

Concerning limitations, we notice that the method is suited to remove local inconsistencies or noise in the tracking data, but not to remove global distortions. As shown in Fig. 22.3, after applying the proposed regularization technique, local inconsistencies are removed yielding a smoother appearance of the image, but global distortions are still present.

We conclude this section with a recommendation for obtaining good parameter settings: We suggest to start with an ℓ_2-type data term, first order Huber-TV regularization ($\alpha = 5$), and 2000 iteration steps which should yield already very good results. If desired or necessary, we suggest to refine these parameters and possibly add some second order regularization in order to obtain even better results. This strategy is particularly suited for time-critical applications, because the first order method regularizes even long pose signals with more than 100 entries in below one second.

22.3 Diffusion Tensor Imaging

Diffusion tensor imaging (DTI) is an imaging modality based on nuclear magnetic resonance and it has become extremely popular during the last decades. Applications of DTI include the determination of fiber tract orientations, e.g., in order to plan surgeries, for the detection of ischemia, or the investigation of neurodegenerative diseases such as schizophrenia or autism.

In this section, we introduce models of the form (22.2) incorporating indirect measurement operators in the setting of DTI. We discuss the data manifold, the measurement operator as well as the variational model in Sect. 22.3.1. In Sect. 22.3.2 we present a generalized forward-backward scheme proposed in [16] for the numerical treatment. In Sect. 22.3.3 we consider experimental results; in particular, we see the benefit of the proposed method by applying it to fiber tracking.

22.3.1 Problem-specific Manifold and Model

We start out to discuss the data space as well as the measurement process, and discuss details on the manifold structure afterwards.

In DTI, each pixel (or voxel) contains a positive definite matrix, i.e., a *diffusion tensor*. These positive matrices can be interpreted as covariance matrices of multivariate zero mean normal distributions and they model the diffusivity of water molecules in space, which is determined by a series of diffusion weighted images (DWIs). We denote the space of 3 × 3 diffusion tensors by Pos_3 and equip it with the Riemannian metric

$$\mathfrak{g}_X(W, V) = \operatorname{trace}(X^{-\frac{1}{2}} W X^{-1} V X^{-\frac{1}{2}}), \tag{22.22}$$

where the symmetric matrices W, V represent tangent vectors at the point $X \in \mathrm{Pos}_3$ which constitutes a positive matrix. For details, we refer to [61] for instance.

Measurements. The DTIs are generated from a series of diffusion weighted images (DWIs.) Each of the DWIs, denoted by y^k, measures the directional diffusivity with respect to a given direction v_k. The relationship between a diffusion tensor X and a corresponding measurement y^k is modeled by the Stejskal-Tanner equation

$$y^k = A^0 \exp(-b\, v_k^T X v_k), \tag{22.23}$$

with a global constant $b > 0$ and a voxel-wise constant $A^0 > 0$, where A^0 represents the intensity of the unweighted measurement. For details we refer to [70] or the discussion in [16]. This means that fitting the tensor X_{ij} at location (i, j) can be achieved by minimizing the atomic data term

$$D(\mathcal{A}(X_{ij}), \{y_{ij}^k\}_k)) = \sum_k \left| bv_k^T X_{ij} v_k - \log(A_{ij}^0 / y_{ij}^k) \right|^2, \tag{22.24}$$

with b, $A_{ij}^0 > 0$. As a consequence, we can write the overall data term (which is a two-dimensional specification of (22.2) to DTI) as

$$\mathbf{D}(\mathcal{A}(X_{ij})_{ij}, (\{y_{ij}^k\}_k)_{ij}) = \sum_{i,j} \sum_k \left| \mathcal{A}^k(X_{ij}) - \log(A_{ij}^0 / y_{ij}^k) \right|^2, \tag{22.25}$$

where $\mathcal{A}^k(X_{ij}) = bv_k^T X_{ij} v_k$ denotes the imaging operator as introduced in (22.2). From a statistical point of view, this data term assumes that the noise on the logarithm of the DWIs is Gaussian noise. It often causes a so-called shrinking effect which manifests itself in the reconstructed tensor being too small due to assuming a Gaussian distribution. A refined noise model for DWIs is Rician noise. An atomic data term which is suited for this noise model is

$$\mathbf{D}(\mathcal{A}(X_{ij})_{ij}, (\{y_{ij}^k\}_k)_{ij}) = \sum_{i,j} D(\mathcal{A}(X_{ij}), \{y_{ij}^k\}_k), \tag{22.26}$$

where

$$D(\mathcal{A}(X_{ij}), \{y_{ij}^k\}_k) = -\sum_k \log \left(\frac{y_{ij}^k}{\sigma^2} \exp\left(-\frac{p_{ij}^k(X_{ij})^2 + (y_{ij}^k)^2}{2\sigma^2} \right) I_0 \left(\frac{p_{ij}^k(X_{ij}) y_{ij}^k}{\sigma^2} \right) \right). \tag{22.27}$$

Here, $p_{ij}^k(X_{ij}) = A_{ij}^0 \exp(-bv_k^T X_{ij} v_k)$ [36, 67], and I_0 denotes the modified "cosh"-like Bessel function of the first kind of order zero; cf. [36]. This term is not based on a measurement operator combined with an Euclidean norm; however we can well deal with it since its structure is similar to the data term in (22.2) as it combines the measurement operator with a differentiable function.

Manifold Structure. Equipped with the Riemannian metric (22.22) the space of positive matrices becomes a Cartan-Hadamard manifold which is a simply connected complete Riemannian manifold of non-positive sectional curvature. Cartan-Hadamard manifolds are a particularly nice class of manifolds in which for instance geodesics do not intersect and are always length-minimizing. In particular, the Riemannian exponential function and its inverse are diffeomorphisms. For details, we refer to the book [26] for instance.

For the space of positive matrices, the Riemannian exponential mapping $\exp_X$ is given by

$$\exp_X(W) = X^{\frac{1}{2}} \exp(X^{-\frac{1}{2}} W X^{-\frac{1}{2}}) X^{\frac{1}{2}}. \tag{22.28}$$

X is a positive matrix and the symmetric matrix W represents a tangent vector in X. The mapping exp on the right-hand side denotes the matrix exponential function. Furthermore, there is also a closed form expression for the inverse of the Riemannian exponential mapping: we have, for positive matrices X, Y that

$$\exp_X^{-1}(Y) := \log_X(Y) := X^{\frac{1}{2}} \log(X^{-\frac{1}{2}} Y X^{-\frac{1}{2}}) X^{\frac{1}{2}}. \tag{22.29}$$

The log symbol on the right-hand side denotes the matrix logarithm; we note that it is well-defined since the argument is a positive matrix. The matrix exponential and logarithm can be computed by diagonalizing the symmetric matrix under consideration and then applying the scalar exponential and logarithm functions to the eigenvalues. The distance between two positive matrices X and Y is given by

$$\mathrm{d}(X, Y) = \sqrt{\sum\nolimits_{l=1}^{3} \log(\kappa_l)^2}, \tag{22.30}$$

where the κ_l denote the eigenvalues of the matrix $X^{-\frac{1}{2}} Y X^{-\frac{1}{2}}$.

Model. For the DTI case, we restrict ourselves to considering total variation regularization; we switch from a univariate (one-dimensional) to a multivariate two-or three-dimensional scenario and incorporate indirect measurement terms. Switching from a one-dimensional scenario to a two- or three-dimensional one has certain implications for the regularizer. In the bivariate setting, we may use a neigborhood system, for instance $e_1 = (1, 0)$, $e_2 = (1, 1)$, $e_3 = (0, 1)$, $e_4 = (-1, 1)$, or $e_1 = (1, 0)$, $e_2 = (0, 1)$, to obtain a finite difference discretization

$$\mathrm{TV}(X) = \sum_l \sum_{i,j} \alpha_l \mathrm{d}(X_{(i,j)+e_l}, X_{i,j}). \tag{22.31}$$

Here, the weights for the four-directional discretization are given by $\alpha_1 = \alpha_2 = \sqrt{2} - 1$ and $\alpha_3 = \alpha_4 = 1 - \sqrt{2}/2$ and for the discretization using the coordinate directions by $\alpha_1 = \alpha_2 = 1$. The first neighborhood system yields a near isotropic finite difference discretization. In our experiments, we compared both neighborhood systems and found that the anisotropy effects are rather mild, and do not justify the additional computational cost due to additional directions in a larger neighborhood system. In the three-dimensional setting, we may use the coordinate axes and the planar diagonals as directions, or simply the coordinate axes. We use the latter in our experiments. Concerning details on finite-difference discretizations, we refer the interested reader to [24, 71, 84]. As models for regularizing diffusion tensor images or volumes we consider

$$\min_{X \in \mathrm{Pos}_3^{n \times m}} \sum_{i,j=1}^{n,m} D(\mathcal{A}(X_{ij}), \{y_{ij}^k\}_k) + \lambda \mathrm{TV}(X), \tag{22.32}$$

where TV is given by (22.31) and its three-dimensional analogue, respectively and $\lambda > 0$ denotes a regularization parameter. As data term in 3D we replace the pixel-wise sum by the voxel-wise sum of the atomic data terms given by (22.24) and (22.27), respectively.

22.3.2 Algorithmic Approach

In [16], we have proposed a generalized forward-backward algorithm for the model (22.32). We first describe the scheme and then explain how to compute the particular quantities. In a vector space setting, forward-backward algorithms have been considered in [63]; see also the references therein.

Generalized Forward-Backward Algorithm. The basic idea is as follows. We decompose the functional (22.32) into the data term

$$F(X) := \sum_{i,j=1}^{n,m} F_{ij}(X_{ij}) := \sum_{i,j=1}^{n,m} D(\mathcal{A}(X_{ij}), \{y_{ij}^k\}_k) \tag{22.33}$$

and the regularizer $\mathrm{TV}(X)$. As a forward step, we perform a gradient step for the data term F of (22.33). The computation of the required gradients is discussed below. As a backward step, we apply the proximal mappings for the atoms of the TV term in a cyclic way. More precisely, for each summand of (22.31) we apply its proximal mapping in a cyclic way. (We can see this as one sweep of the CPPA for the TV term as discussed in Sect. 22.2.2.) The proximal mappings of the summands of (22.31) are explicitly given as discussed in Sect. 22.2.2 since, in the manifold of positive matrices, we have explicit formulae for the distance, the exponential mapping and its inverse as explained before. Then, the generalized forward-backward algorithm consists of iterating the gradient step w.r.t. the data term and the cyclic application of the proximal mappings on the TV regularizer. We refer to Algorithm 2 where the concept of the algorithm is presented, and for further details, we refer to [16].

Computing Gradients. We explain how to compute the Riemannian gradients ∇F_{ij} of the summands F_{ij} of the data term F. More precisely, we need the gradients of

$$F_{ij}(X_{ij}) = D(\mathcal{A}(X_{ij}), \{y_{ij}^k\}_k) \tag{22.34}$$

with the data term D either given by (22.24) and (22.27), respectively. We first notice that Pos_n is an open subset of the linear space of symmetric matrices Sym_n. Hence, the differential in the manifold setting agrees with the differential in the vector space setting. So let V be an arbitrarily chosen element in the tangent space of X_{ij}. Then the first variation of F_{ij} at X_{ij} in the direction V is expressed via the Riemannian metric (22.22) by Algorithm 2

Algorithm 2 Generalized forward backward algorithm (with cyclic backward step) for the TV problem (22.32) for joint diffusion tensor fitting and denoising

```
**Input:** DWI images $\mathbf{y}^k = (y^k_{ij})$, regularization parameter $\lambda$, number of steps $S$.
**Ouput:** Signal $X$ (solution of (22.32)).
Initialize $\mathbf{X}$
**for** $m = 1, \dots, S$
  $\lambda_m \leftarrow$ CompLambda($m$) //compute stepsize
  //Riemannian gradient step
  **for** $i = 1, \dots, I,\ j = 1, \dots, J$
    $G_{ij} \leftarrow$ CompGradient($X_{ij}$,$y_{ij}$)  // Riem. grad. of (22.24),(22.27)
    $\lambda_m \leftarrow$ min( searchLine($G_{ij}$,$U_{ij}$,$F_{ij}$,$\lambda_m$))  // adapt step length
  **for** $i = 1, \dots, I,\ j = 1, \dots, J$
    $X_{i,j} \leftarrow \exp_{X_{i,j}}(-\lambda_m G_{i,j})$  //update according to (22.42)
  //proximal mappings for TV atoms (numbered by $n = 1, .., N$)
  **for** $n = 1, \dots, N$
    $t \leftarrow \min(\lambda_m \lambda \alpha_{l_n}, \frac{1}{2})$  // determine $t$, cf. (22.18)
    $\bar{X}_{i_n,j_n} \leftarrow \exp_{X_{i_n,j_n}}\left(t \log_{X_{i_n,j_n}} X_{(i_n,j_n)+v_{l_n}}\right)$
    $\bar{X}_{(i_n,j_n)+v_{l_n}} \leftarrow \exp_{X_{i_n,j_n+v_{l_n}}}\left(t \log_{X_{i_n,j_n+v_{l_n}}} X_{(i_n,j_n)}\right)$
    $X_{i_n,j_n} \leftarrow \bar{X}_{i_n,j_n}$, $X_{(i_n,j_n)+v_{l_n}} \leftarrow \bar{X}_{(i_n,j_n)+v_{l_n}}$ // update
```

$$\tfrac{d}{dt}|_{t=0} F_{ij}(X_{ij} + tV) = \tfrac{d}{dt}|_{t=0} D(\mathscr{A}(X_{ij} + tV), \{y^k_{ij}\}_k) = \mathfrak{g}_{X_{ij}}(\nabla F_{ij}(X_{ij}), V). \tag{22.35}$$

We first consider the situation when the data term D is given by (22.24). For the computation of ∇F_{ij}, we first use that any diffusion tensor is a symmetric matrix which implies that there is a canonical isomorphism $\mathfrak{I}$ mapping a tensor W to its six components in its upper triangular part, $\mathfrak{I}(W) = (W^{(1,1)}, \dots, W^{(1,3)}, W^{(2,2)}, \dots, W^{(3,3)})$. Using this isomorphism we reformulate F_{ij} in the form

$$\frac{1}{2}\|Ax_{ij} - \bar{y}_{ij}\|_2^2. \tag{22.36}$$

Here x_{ij} is the vector in $\mathbb{R}^6$ with components from the upper triangle of X_{ij}, $A \in \mathbb{R}^{N \times 6}$ is determined by all N gradient vectors y_k (independent of i, j) and $\bar{y}_{ij} \in \mathbb{R}^N$ is a vector with components $\bar{y}^k_{ij} = \log(y^k_{ij}/A_0)$. The first variation of this least squares problem can now be represented by the euclidean scalar product of $\mathbb{R}^6$ as

$$\tfrac{d}{dt}|_{t=0} F_{ij}(x_{ij} + tv) = \langle m, v \rangle\,, \tag{22.37}$$

where $m = A^T(Ax_{ij} - \bar{y}_{ij})$ for arbitrarily chosen $v \in \mathbb{R}^6$. Defining $M = \mathfrak{I}^{-1}(m)$ and $V = \mathfrak{I}^{-1}(v)$ we rewrite this first variation as $\langle m, v \rangle = \operatorname{trace}(M^T V)$. In other words, M^T is the Euclidean gradient of F_{ij}. Employing the Riesz representation theorem and the definition of the Riemannian metric (22.22) we find

$$\mathrm{trace}\left(X_{ij}^{-1}\nabla F_{ij} X_{ij}^{-1} V\right) = \mathrm{trace}(M^T V). \tag{22.38}$$

(Note that we here suppress the dependence on ∇F_{ij} and M^T on X_{ij} in the notation.) As this equality holds true for all tangent vectors V at the point X_{ij},

$$\nabla F_{ij}(X_{ij}) = X_{ij} M^T X_{ij}. \tag{22.39}$$

If the data term D is given by (22.27) we proceed similarly and obtain

$$\nabla F_{ij}(X_{ij}) = X_{ij} \hat{M}^T X_{ij}, \tag{22.40}$$

where

$$\hat{M}^T = \sum_k \frac{-bp_{ij}^k(X_{ij})}{\sigma^2} \left(p_{ij}^k(X_{ij}) - \frac{I_0'}{I_0}\left(\frac{p_{ij}^k(X_{ij})F_{ij}^k}{\sigma^2}\right) F_{ij}^k\right) v_k v_k^T \tag{22.41}$$

and I_0'/I_0 denotes the logarithmic derivative of the Bessel function I_0. Here, $v_k v_k^T$ is the rank one matrix obtained from the direction v_k.

Once the gradients are computed, we can perform a gradient descent step

$$X_{ij}^{m+1} = \exp_{X_{ij}^m}(-\lambda_m \nabla D(X_{ij}^m)), \tag{22.42}$$

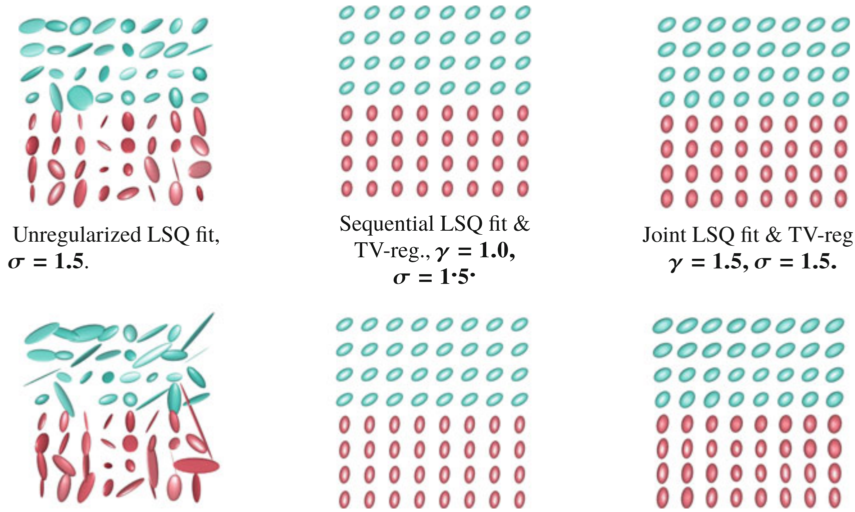
using (22.39) and (22.40) for ∇D, respectively, depending on weather the data term D is either given by (22.24) or by (22.27); here λ_m denotes the step size chosen at the m-th iteration.

22.3.3 Experiments

We evaluate the proposed method on synthetic data as well as on real data from the UCL Camino Diffusion MRI Toolkit [30]. The algorithm is implemented in C++ and the operations which are necessary for computing matrix roots, logarithms, and exponential functions have been implemented using Eigen v.3.2.4.[1] The parameter λ_m in Algorithm 2 is chosen as $\lambda_m = Cm^{-\frac{1}{2}}$, with $C = 100$. For the visualization, we used a modified version of the fanDTasia ToolBox [10].

At first, we performed a quantitative evaluation using synthetic test data similar to the one in Fillard et al. [36] and the original publication [16]. In short, we generated a ground truth volume that has two “phases” of tensors, used the Stejskal-Tanner equation to create ten ground truth DWIs and finally imposed Rician noise on

[1] Available at http://eigen.tuxfamily.org.



Unregularized LSQ fit, $\boldsymbol{\sigma = 1.5}$.

Sequential LSQ fit & TV-reg., $\boldsymbol{\gamma = 1.0}$, $\boldsymbol{\sigma = 1\cdot5\cdot}$

Joint LSQ fit & TV-reg., $\boldsymbol{\gamma = 1.5, \sigma = 1.5.}$

Unregularized Rice-MLE fit, $\boldsymbol{\sigma = 1.5.}$

Sequential Rice-MLE fit & TV-reg., $\boldsymbol{\gamma = 1.0,}$ $\boldsymbol{\sigma = 1.5.}$

Joint Rice-MLE fit & TV-reg., $\boldsymbol{\gamma = 3.5,}$ $\boldsymbol{\sigma = 1.5.}$

Fig. 22.5 Synthetic experiments (examples). Unregularized tensor fitting is severely affected by the noise which makes regularization necessary. All TV-based regularization methods yield edge-preserving regularization. We observe an undesired shrinking effect for the schemes based on the LSQ data term (22.24). The presented approach of combined fitting and regularization based on the model (22.32) using the Rice-MLE data term (22.27) yields the most convincing results

the produced DWIs. (The Rician noise distribution is given by the distribution of the radial part of a bivariate centered normally distributed random variable with covariance matrix being a multiple of the identity.) We compare the proposed combined method with the (uncombined) sequential baseline method which works as follows: we first fit the tensors using the data term (22.24) (or (22.32)) without any spatial regularization. Then, in a next step, we perform TV regularization on the fitted tensors as described in [83]. It can be observed in Fig. 22.5 that the combined fitting and TV denoising approach using the Rice-MLE data term (22.27) further improves the results obtained by the combined approach using the least squares (LSQ) data term (22.24); in particular, it does not show the aforementioned shrinking effect. Furthermore, we observe that the proposed approach yields better results than the corresponding baseline approach which first fits the tensors using the respective data term and subsequently performs TV regularization. For details on the quantitative evaluation we refer the reader to [16].

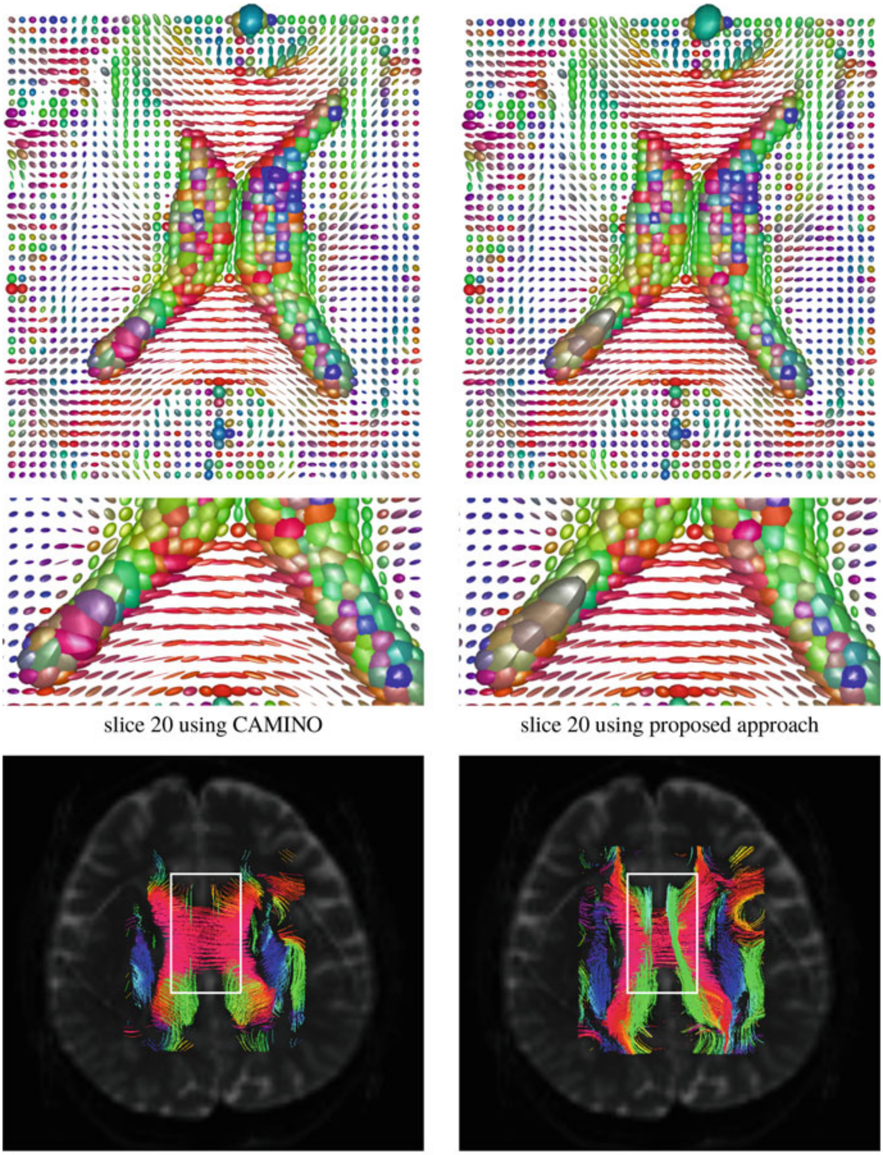
In order to demonstrate the potential of the proposed method on real data, we have applied it to diffusion weighted data of the human brain from the Camino project [30]. We reconstructed the tensors in three dimensions with the proposed approach using joint tensor fitting and TV-regularization using the Rice-MLE data term, where we performed 1000 steps ($S = 1000$). We estimated a Rician noise level of $\sigma_r = 16777.5$ using the method proposed in [46]. Since the maximum DWI magnitude is of order 10^5, magnitude of the unweighted measurements is of order 10^6 the noise level can be considered as low to moderate. For comparison, we computed the DTI volumes using the Camino software; see Fig. 22.6. We first observe the denoising capabilities of the proposed scheme. Comparing the upper left panel and the upper right panel of Fig. 22.6, we further observe that the tensors obtained by the Camino software are slightly smaller than the tensors obtained by the presented method. This effect might be explained by the fact that the CAMINO software uses a least squares fit on the real data (containing noise well modeled by Rician noise) which causes shrinkage and by the fact that the Rice-MLE data term of the presented method seems more suited to the noise and so avoids shrinkage.

In order to indicate the relevance of the proposed method for fiber tracking, we employed the fiber tracking module implemented in 3D Slicer (www.slicer.org). For all experiments we used the following settings: 1.0 seed spacing (in voxel), 0.3 min seed FA, 0.3 stopping min FA, 0.1 mm integration step, 0 mm minimum path length, 2000 mm maximum path length. The lower left panel in Fig. 22.6 shows the fiber tracking results obtained with the UKFT-module (Slicer plugin) which implements the method of [13]. The lower right panel finally shows the result obtained with the proposed reconstruction algorithm exhibiting an improved reconstruction of the (vertical) fibers corresponding to the cingulum (indicated by white squares), which has been confirmed by a clinical expert.

22.3.4 Discussion

We have presented an approach for combined tensor fitting and edge-preserving regularization which is a TV regularization approach in a combined (non-flat) manifold and inverse problem setup; in particular, we have considered an energy with a data fidelity term adapted to Rician noise on the DWIs. As minimization strategy, we have developed a generalized forward-backward scheme. We have applied the derived algorithms to real DTI data and shown its benefit for fiber tracking.

The affine invariant metric on Pos_3 we employ is well-established in DTI and used in various contexts; see Pennec et al. [61] or Cheng et al. [29]. In particular this metric corresponds to the Fisher-Rao metric on normal distributions which yields a strong statistical motivation to precisely consider this metric. Instead of endowing Pos_3 with the Fisher-Rao metric (and thus considering it as a manifold) there are approaches which impose different mathematical structures. One approach is to consider them as the positive cone in the space of matrices and to equip it with the Euclidean distance; see, for instance, [49, 82, 85]. Methods based on this concept typically have to ensure that the computations done in the ambient space do not leave the cone of positive matrices. This is usually achieved by using projections which are a somewhat problematic concept in this context, since the positive matrices form an open set which means that they have no boundary onto which one could project.



slice 20 using CAMINO

slice 20 using proposed approach

Fiber tracking using UKFT module

Fiber tracking using proposed tensor fitting

Fig. 22.6 Experiments using the Camino Dataset. We compare the presented approach with the reconstruction result obtained by the Camino software (http://cmic.cs.ucl.ac.uk/camino/) in the upper panels. We observe that the tensors reconstructed with this software are slightly smaller than the tensors reconstructed with the presented approach, which is especially visible in the respective close-ups. In the lower panels, we apply the fiber tracking implemented in Slicer (www.slicer.org) to the result of the respective reconstruction method. As indicated by the white rectangles, the presented method leads to an improved reconstruction of the (vertical) fibers corresponding to the cingulum

Also, using the Euclidean metric for DTI data, a swelling effect has been reported in [5, 74]. In this case, the dispersion of the corresponding covariance matrices, i.e., their determinant, tends to be larger than the original ones [5] when reconstructing noisy data with known ground truth. Another mathematical structure for DTI is the so-called log-Euclidean framework as employed by Arsigny et al. [5, 6] and Fillard et al. [36]. By using this metric, one essentially works in the tangent space of the identity matrix and then solves the present problem in this linear space. One of the main advantages of the log-Euclidean approach is that it is computationally very efficient. This advantage comes with not being affine invariant and the drawback of being restricted to a particular base point, i.e., the identity matrix. Going to the tangent space at the identity matrix yields good results for nearby points, but one loses quality the further the considered points are away.

Because the positive definiteness of the reconstructed tensors is a non-linear constraint and as such hard to enforce [36] there is a large body of literature on DTI reconstruction. One can distinguish four types of approaches: (1) fitting the tensors independently per voxel while imposing constraints to enforce the positive definiteness of the reconstructed tensors [47, 81], (2) denoising the DWIs first and then fitting the tensors individually [9, 11, 52], (3) regularizing the tensors after reconstruction [5, 22, 23, 28, 40, 61, 74], and (4) reconstructing and regularizing the tensors simultaneously [6, 27, 36, 56, 59, 76]. Methods related to the latter approach are more intricate, but they show the best performance with respect to reconstruction quality [36, 75]. The presented approach falls in the last category. Due to the amount, diversity, and variety of related work, we refer the interested reader to [16] for a detailed discussion.

The major limitation of classical DTI appears in the representation of intravoxel crossings of fibers [4, 78]. Here, the diffusion process is no longer well described by a single tensor and leads to falsely fitted plate or sphere shaped tensors. In order to deal with crossings several approaches have been proposed, e.g., [4, 32, 38, 57]. They can be categorized into two groups: (1) the single-tensor-model is replaced by a multi-tensor one; see for instance [3, 60, 78]; (2) model-free approaches where the major assumption is that the diffusion process can be described by a orientation distribution function (ODF) on the 3D unit sphere; see for example [32, 42, 44, 73, 77]. The latter imaging approach is also called Q-ball imaging. For a review on both types of approaches we refer to [2]. Similar to standard DTI the model space in the Q-ball setup can be interpreted as a Riemannian manifold [39]. Hence, the framework presented here can be applied to the corresponding manifold as well. We refer the reader to the paper [84] where an a posteriori regularization method in the Q-ball setup is proposed and where the needed differential-geometric operations are described to implement the framework of this section in the Q-ball setting. The concrete application in the Q-ball setting is a promising topic of future work.

Finally, we point out that the employed model and algorithm are fairly general. In particular, to make the approach work for another manifold, we basically need an implementation of the gradient of the measurement operator as well as methods to compute the Riemannian exponential and its inverse.

22.4 Geometry Processing and Medical Image Segmentation

The main reason of defining and studying shape spaces is the wish to be able to compare two or multiple shapes in a mathematically meaningful way, i.e., by introducing a (Riemannian) metric. There are various approaches leading to different classes of shape spaces. The most straightforward way of categorizing them is possibly to distinguish discrete formulations which comprises the original approach by Kendall as well as most types of active and statistical shape models, e.g., [31, 41, 45], from continuous formulations, e.g., [12, 53, 68, 72]. We also refer to the Chapters 13, 14, 24 and 16 of this book which also deal with shape spaces. In this chapter, we consider (discrete) parametric shape spaces, where the parameters consist of a fixed number of two-dimensional points. To this end, let us consider polygonal discretizations of simple planar shapes, i.e., two-dimensional and non-intersecting closed curves. We obtain a simple n-gon which can be represented by a complex vector

$$z = (z_1, \ldots, z_n) \in \mathbb{C}^n, \tag{22.43}$$

where each entry z_i encodes the coordinates of one boundary point with its real and imaginary part. In the following, we employ the shape representation of Kendall [45] (see also Chapter 10 in this book or the book [33]) which is based on such closed polygonal lines. In particular, all quantities needed from an algorithmic point of view (in particular, the exponential mapping and its inverse) can be implemented very efficiently. We will use Kendall shapes, as well as Kendall preshapes which we will also call oriented Kendall shapes.

The advantage of using such parametric shape spaces is that they allow a geometrically meaningful treatment of shapes, e.g., by introducing invariance regarding translation or scale, without the need for learning the shape of an object class from a collection of previously segmented shapes. Particularly in cases, where no or little training data is available, this property comes in handy. The fact that such generic shape spaces are very powerful can also be seen when using them in a joint segmentation and regularization framework as discussed at the end of this section.

In Sect. 22.4.1 we discuss the manifold structure of the considered Kendall shape spaces, the basic model for denoising and describe a corresponding algorithm. In Sect. 22.4.2 we apply the scheme for regularizing slicewise segmentations of a part of an abdominal aorta. In Sect. 22.4.3 we discuss extensions to joint segmentation and regularization as well as to the multivariate situation.

22.4.1 Problem-Specific Manifold, Basic Model and Algorithm

We will briefly recall the original representation of Kendall [45]. For a concise description of this representation we refer to [37] for instance. According to Kendall, a shape is that which “is left when the effects associated with translation, scaling and rotation are filtered away.” More mathematically, the above operations form a (Lie) group, the group of similarity transforms. Then a shape is an equivalence class of polygons, where two polygons define the same shape/belong to the same equivalence class if they are congruent w.r.t. a similarity transform. A more general definition is obtained by replacing the group of similarity transforms by any reasonable group of transformations.

We first consider the translation part. We normalize the polygon (22.43) such that its center of mass equals the coordinate center. Hence, we may assume the representation (22.43) to be normalized w.r.t. translation, i.e.,

$$\sum_{i=1}^{n} z_i = 0 \in \mathbb{C}. \tag{22.44}$$

The mean value (translation vector) is stored for all shapes to position the regularized shapes correctly. This operation is usually done before applying the proposed method and can be reversed after the regularization. By normalizing z w.r.t. to translation we may represent shapes in the $(n-1)$-dimensional subspace

$$V_{n-1} = \{z \in \mathbb{C}^n : \sum_{i=1}^{n} z_i = 0\} \subset \mathbb{C}^n, \tag{22.45}$$

which can itself be identified with $\mathbb{C}^{n-1}$. Loosely speaking, normalizing w.r.t. translation removes *one complex degree of freedom.*

Frequently, we consider the real-valued representation of z via identifying $\mathbb{C}^n$ with $\mathbb{R}^{2n}$, i.e.,

$$x = (x_1^1, x_1^2, x_2^1, x_2^2, \ldots, x_n^1, x_n^2) \in \mathbb{R}^{2n}, \tag{22.46}$$

where $x_i^1 = \Re(z_i)$ and $x_i^2 = \Im(z_i)$ denote the real and imaginary part of z_i. In this representation, the normalization (22.44) reads $\sum_{i=1}^{n} x_i^1 = 0$, and $\sum_{i=1}^{n} x_i^2 = 0$. So, by normalizing x w.r.t. translation, we are removing *two real degrees of freedom*. Thus, the shape representation is restricted to the real subspace

$$V_{2n-2} = \{x \in \mathbb{R}^{2n} : \sum_{i=1}^{n} x_{2i} = 0, \text{ and } \sum_{i=1}^{n} x_{2i-1} = 0\} \subset \mathbb{R}^{2n}. \tag{22.47}$$

Next we consider scaling. We notice that a polygon $x \in V_{2n-2}$ can be scaled by multiplying all real components x_i with a real number $s > 0$ and still defines the same shape w.r.t. translation and scaling. In other words, all $y \in V_{2n-2}$ which lie on the real half-line

$$L_x = \{s \cdot x : s \in \mathbb{R}, s > 0\} \tag{22.48}$$

define the same *Kendall preshape* or *oriented Kendall shape* meaning that L_x is the equivalence class of all polygons which are equivalent w.r.t. translations and scaling. We emphasise that this shape representation is *not* invariant w.r.t. rotations. The set of all these equivalence classes can now be identified with the real unit sphere $S_{\mathbb{R}}^{2n-3}$. We note that by requiring scale invariance we are removing *another real degree of freedom*.

In order to incorporate rotation, we consider the complex representation (22.45). We observe that a polygon $z \in V_{n-1}$ can be scaled by a factor $s > 0$ and rotated by an angle $\theta \in [0, 2\pi)$ by multiplying all complex components z_i with the complex number $w = s\exp(i\theta) = s\cos(\theta) + is\sin(\theta)$ to define the same shape. Consequently, all polygons z which are equivalent w.r.t. translation, rotation, and scaling lie on the complex line

$$L_z = \{w \cdot z : w \in \mathbb{C}\backslash\{0\}\}. \tag{22.49}$$

In other words, the shape L_z is the equivalence class of all polygons which are equivalent w.r.t. rigid transformations and scaling. *The classical Kendall shape space* which consists of the set of all these equivalence classes can now be identified with the complex projective space $\mathbb{C}P^{n-2}$ or, equivalently, the complex unit sphere $S_{\mathbb{C}}^{n-2}$ (with antipodal points identified). We note that by requiring rotation and scale invariance we are removing *another complex degree of freedom*.

As a consequence, for the Kendall shape space, the exponential mapping and the inverse exponential mapping are given by the respective mappings of $S_{\mathbb{C}}^{n-2}$, i.e.,

$$\exp_z(v) = \cos(\phi)\cdot z + \frac{\|z\|\sin(\phi)}{\phi}\cdot v, \phi = \|v\| \tag{22.50}$$

and

$$\log_z(y) = \phi \cdot \frac{y - \Pi_z(y)}{\|y - \Pi_z(y)\|}, \phi = \arccos(\frac{|\langle z, y\rangle|}{\|z\|\,\|y\|}), \tag{22.51}$$

where $\Pi_z(y) = z \cdot \langle z, y\rangle \,/\, \|z\|^2$ denotes the projection of y onto z. We notice that $\langle\cdot,\cdot\rangle$ denotes the complex scalar product, i.e., $\langle z, y\rangle = \sum_{i=1}^n z_i\overline{y_i}$, where $\overline{\cdot}$ denotes the complex conjugation, and $\|\cdot\|$ is the norm induced by the complex scalar product.

For the oriented Kendall shape space, the exponential mapping and the inverse exponential mapping are given by the respective mappings of $S_{\mathbb{R}}^{n-3}$, -> i.e., by formulas (22.50) and (22.51) but this time with the real-valued scalar product $\langle x, y\rangle$ and its induced norm. To put it in a nutshell: By exchanging the scalar product for the computation of the exponential and the inverse exponential mappings we can switch between the rotationally invariant and the non-rotationally invariant representation. All mappings can be implemented very efficiently as they only require basic linear algebra subroutines (BLAS).

Model for Shape Denoising. We specify the general model explained in Sect. 22.1.2 to the situation of shape denoising. We here first consider a basic model as done in [15] and note that there is an extension proposed in [69] which incorporates an indirect measurement term as briefly discussed in Sect. 22.4.3. Since we deal with a time series $\mathbf{y} = (\mathbf{y}_i)_{i=1}^n$ of shapes as input, the basic model reads

$$\min_{\mathbf{x}\in\mathcal{M}^n} \sum_{i=1}^{n} h \circ \mathrm{d}(\mathbf{x}_i, \mathbf{y}_i) + \lambda\, h' \circ \mathrm{d}(\mathbf{x}_i, \mathbf{x}_{i+1}), \tag{22.52}$$

where d is the distance on the Kendall shape space or the oriented Kendall shape space. The model is similar to the model (22.14) in Sect. 22.2.1 with the Kendall shape space or the oriented Kendall shape space replacing the poses. (We note that historically the case of shapes was considered first.) As in Sect. 22.2.1 h, h' are potentials given by either $h(s) = s$, $h(s) = s^2/2$ or by the Huber potential (22.5) and $\lambda > 0$ is the regularization parameter. We refer to the discussion in Sect. 22.2.1.

Algorithm. In order to solve the problem (22.52) we employ a cyclic proximal point algorithm as described in Sect. 22.2.2 for pose signals. In order to implement all needed proximal mappings (cf. Sect. 22.2.2) in the Kendall shape space or the oriented Kendall shape space we only need explicit expressions for the distance and the Riemannian exponential mapping as given by (22.50) and (22.51).

We emphasize that this algorithm can be instantiated for any shape space by providing implementations of the corresponding exponential mapping and its inverse.

22.4.2 Experiments

We here restrict ourselves to considering one example from geometry processing and leave some space for possible extensions discussed in the next subsection.

Let us consider the slice-wise segmentation of organs, such as the abdominal part of the aorta from computed tomography angiography (CTA), cf. Fig. 22.7. The contrasted lumen of the aorta, was segmented with the method of Baust et al. [14]. The segmentation boundaries were discretized with 360 equally spaced points and the presented algorithm regularized the whole signal consisting of 68 shapes in 1.35 s, where we chose $\alpha = 15.0$ as well as ℓ_1 penalties for data term and regularizer.

As depicted in Fig. 22.7a, b, the presented algorithm successfully regularizes the segmentation of the aortic lumen. Thereby, it is particularly useful in removing little cusps and concavities which is shown in Fig. 22.7c where we colorized the original segmentation with the (signed) surface distance between the original and the regularized signal. The cusps correspond to erroneously segmented calcifications in the aortic wall. Since the algorithm does not alter the segmentation in an unreasonable way, it is perfectly suited for processing geometric models which are later used in biomechanical simulations.

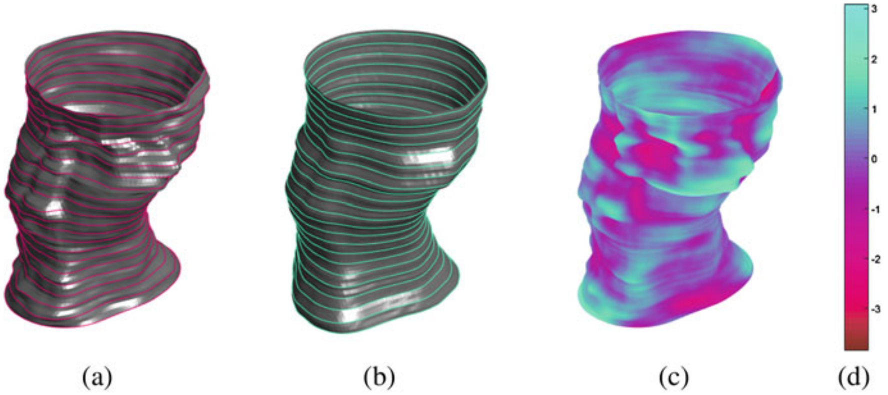


Fig. 22.7 Geometry processing example: Application of the presented algorithm to a segmentation of the lumen of the abdominal part of a human aorta. The model consists of 68 CTA slices segmented with the method of Baust et al. [14]. The original model is shown in (**a**) with every second shape highlighted in blue. The regularized signal is shown in (**b**) with every second contour highlighted in yellow. The method successfully removes little cusps and concavities of the original contours, where we colorized the original segmentation with the signed surface distance (in voxel) to the regularized model (**c**)

22.4.3 Extensions

Besides regularizing already existing shape signals, it is also possible to simultaneously generate and regularize them. Similar to the case of DTI, we have to define a suitable imaging operator $\mathcal{A}$ to be used in (22.2).

Let us assume that we are already given a shape signal $\mathbf{x} = (\mathbf{x}_i)_i$. As the oriented Kendall shapes are scale- and translation-invariant, the imaging operator has to endow each shape with these components in order to make them usable in a classical image segmentation scenario. Therefore, we actually have a series of measurement operators $\mathcal{A}_i$ which augment each oriented shape $\mathbf{x}_i$ with a translational component $t \in \mathbb{R}^2$ and a scaling factor $s > 0$ to obtain a particular implicit representation of the form

$$\mathcal{A}_{t,s}(\mathbf{x}_i) = \mu_{\text{in}} \mathbf{1}_{\Omega_{t,s}} + \mu_{\text{out}} \mathbf{1}_{\Omega_{t,s}^C}, \tag{22.53}$$

where $\Omega_{t,s}$ denotes the interior of the shape $\mathbf{x}_i$ anchored at t and normalized such that the sum over the squared distances to the anchor equals s^2. In order to keep the notation simple, we frequently drop the dependency of t and s on i. $\Omega_{t,s}^C$ denotes the complement of $\Omega_{t,s}$ which is the outer part determined by the curve $\mathbf{x}_i$. Furthermore, $\mu_{\text{in}} \in [0, 1]$ and $\mu_{\text{out}} \in [0, 1]$ denote the means of intensity values of I_i (ith frame) computed w.r.t. $\Omega_{x,s}$ and $\Omega_{x,s}^C$, respectively. This process is illustrated in Fig. 22.8.

Assuming that I_i is discretized on a $k \times l$-grid with intensity values normalized to [0, 1], we can now formulate the (frame-wise) measurement operation as a classical

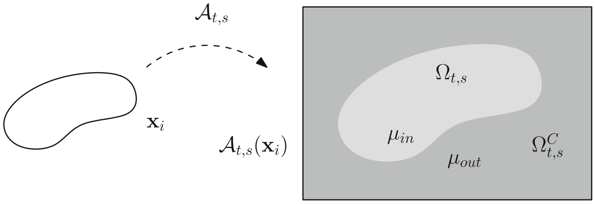


Fig. 22.8 Action of the measurement operator: $\mathcal{A}_{t,s}$ maps a Kendall shape $\mathbf{x}_i$ to a translated, i.e., positioned, and scaled version $\mathcal{A}_{t,s}(\mathbf{x}_i)$ in the image space. $\Omega_{t,s}$ and $\Omega_{t,s}^C$ denote the areas inside and outside $\mathcal{A}_{t,s}(\mathbf{x}_i)$, respectively

piece-wise constant segmentation problem, similar to the data term employed in the Chan-Vese model [25], i.e., the two-phase version of the Potts model [55]:

$$D(\mathcal{A}_i(\mathbf{x}_i), I_i) = \min_{t,s} \Big(\sum_{j\in\Omega_{t,s}} |(I_i)_j - \mu_{\text{in}}|^2 + \sum_{j\in\Omega_{t,s}^C} |(I_i)_j - \mu_{\text{out}}|^2 \Big), \tag{22.54}$$

where j denotes the (vectorized) pixel index. It is important to note that by keeping the shape fixed, the minimization problem in (22.54) reduces to a registration problem w.r.t. position and scale. The minimization over t and s in (22.54) thus reflects the necessary scale selection process as the shapes $\mathbf{x}_i$ do not carry the respective information any more. Since already one function evaluation requires solving a registration problem, a naive approach to the this problem is not reasonable from a computational perspective. Thus, we developed an approximate strategy in [69] based on the concept of active contour evolutions: Starting from a set of triplets $(\mathbf{x}_i, t_i, s_i)$, we compute a deformation field $\mathbf{v}_i$ which deforms the scaled and positioned version of $\mathbf{x}_i$. Next, we can easily estimate the scale and translational changes from $\mathbf{v}_i$ and update t_i and s_i in a gradient descent fashion. After normalizing $\mathbf{v}_i$ with respect to t_i and s_i, we are left with a tangent vector in $T_{\mathbf{x}_i}\mathcal{M}$, which can be used for an explicit gradient descent step. This way, we end up with solving the registration problem in an incremental way while solving the shape estimation problem in a forward-backward fashion similar to the DTI case.

It is further possible to extend the whole concept to shape fields. A typical application scenario is 3D+t cardiac MRI, where the human heart is imaged throughout the entire heart cycle in a slice-wise fashion; here, we obtain a shape signal for each slice and hence we term the collection of these slice-wise signal shape field as visualized in Fig. 22.9. Applying the presented framework to shape fields is relatively straightforward: Similar to the DTI case, cf. Sect. 22.3, we add regularization terms for both directions, i.e., in direction of the slices as well as in temporal direction. Then the algorithmic approach described in Algorithm 2 can be applied with the modification that the imaging/measurement operators depend on the individual frames.

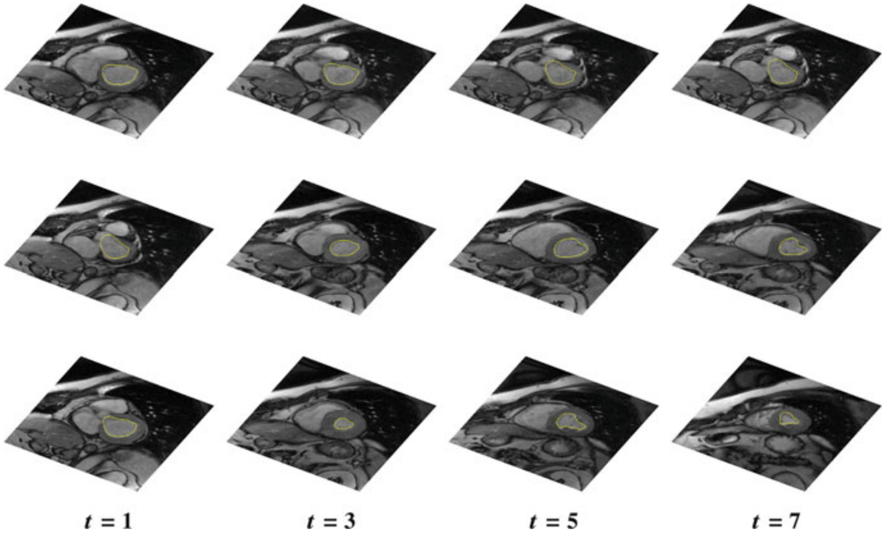


Fig. 22.9 Joint segmentation and regularization of a shape fields: We apply the presented framework to 3D+t short-axis MRI scans of the human heart. The two coordinate axes for regularization are the slice-direction (top to bottom) and the temporal direction (left to right)

22.4.4 Discussion

Both applications (cf. Figs. 22.7 and 22.9) demonstrate the broad applicability and potential of Riemannian approaches for both geometry processing and segmentation problems. We point out that the presented approach can be well combined with machine-learning-based techniques for segmentation, such as the recently proposed deep active contours framework by Rupprecht et al. [64]. Further extensions might also include an additional total-variation-based regularization of the translation and scale components which are currently only implicitly optimized via the active contour evolution.

22.5 Conclusion and Outlook

In this chapter, we have presented a collection of medical examples covering not only various applications, such as imaging, image compounding, segmentation and geometry processing, but also different imaging modalities themselves: Ultrasound, MRI, DTI, and CT. Depending on the chosen manifold, the presented framework can yield very efficient algorithms, such as in the case of pose regularization, where short run times are beneficial for many real-time applications. As discussed in the DTI case, it is also possible to extend the presented framework to more advanced setups such as Q-Ball imaging. With computers becoming more and more powerful, more and more image segmentation algorithms that tackle 3D data directly appear. Thus, we anticipate that the presented slice-wise segmentation and regularization approach might soon be replaced by approaches considering one-dimensional signals consisting of three-dimensional shapes.

We have mostly considered the application of TV regularization techniques in this chapter. They constitute a convincing compromise between quality and run time; they yield good regularization and the rather short runtimes are acceptable for many applications. In order to improve the quality (which comes with higher runtimes; cf. Sect. 22.2.3) it is interesting to further investigate higher order TV type methods such as total generalized variation [20] or different first order methods of Mumford-Shah type [84].

We hope that this overview inspires further applications. All the more, since the observation of medical image and sensor data being subject to physical and physiological constraints is a fairly general one and not restricted to any specific modality or application.

Acknowledgements M. Baust acknowledges the support of the Collaborative Research Centre 824 (Z2) *Imaging for Selection, Monitoring and Individualization of Cancer Therapies* funded by the German Science Foundation. A. Weinmann acknowledges support by the DFG Grants WE 5886/3-1 and WE 5886/4-1.

References

1. Afsham, N., Khallaghi, S., Najafi, M., Machan, L., Chang, S.D., Goldenberg, L., Black, P., Rohling, R., Abolmaesumi, P.: Filter-based speckle tracking for freehand prostate biopsy: theory, ex vivo and in vivo results. In: International Conference on Information Processing in Computer-Assisted Interventions, pp. 256–265. Springer, New York (2014)
2. Alexander, D.C.: Multiple-fiber reconstruction algorithms for diffusion MRI. Ann. N. Y. Acad. Sci. **1064**(1), 113–133 (2005)
3. Alexander, A.L., Hasan, K.M., Lazar, M., Tsuruda, J.S., Parker, D.L.: Analysis of partial volume effects in diffusion-tensor MRI. Magn. Reson. Med. **45**(5), 770–780 (2001)
4. Alexander, D.C., Barker, G.J., Arridge, S.R.: Detection and modeling of non-Gaussian apparent diffusion coefficient profiles in human brain data. Magn. Reson. Med. **48**(2), 331–340 (2002)
5. Arsigny, V., Fillard, P., Pennec, X., Ayache, N.: Fast and simple calculus on tensors in the log-Euclidean framework. In: Medical Image Computing and Computer-Assisted Intervention (MICCAI), pp. 115–122. Springer, Berlin (2005)
6. Arsigny, V., Fillard, P., Pennec, X., Ayache, N.: Log-Euclidean metrics for fast and simple calculus on diffusion tensors. Magn. Reson. Med. **56**(2), 411–421 (2006)
7. Bačák, M.: Computing medians and means in Hadamard spaces. SIAM J. Optim. **24**(3), 1542–1566 (2014)
8. Bacák, M., Bergmann, R., Steidl, G., Weinmann, A.: A second order nonsmooth variational model for restoring manifold-valued images. SIAM J. Sci. Comput. **38**(1), A567–A597 (2016)
9. Bao, L.J., Zhu, Y.M., Liu, W.Y., Croisille, P., Pu, Z.B., Robini, M., Magnin, I.E.: Denoising human cardiac diffusion tensor magnetic resonance images using sparse representation combined with segmentation. Phys. Med. Biol. **54**(6), 1435 (2009)

10. Barmpoutis, A., Vemuri, B., Shepherd, T., Forder, J.: Tensor splines for interpolation and approximation of DT-MRI with applications to segmentation of isolated rat hippocampi. IEEE Trans. Med. Imaging **26**(11), 1537–1546 (2007)
11. Basu, S., Fletcher, T., Whitaker, R.: Rician noise removal in diffusion tensor MRI. In: Medical Image Computing and Computer-Assisted Intervention (MICCAI), vol. 4190, pp. 117–125. Springer, Berlin (2006)
12. Bauer, M., Bruveris, M., Michor, P.W.: Overview of the geometries of shape spaces and diffeomorphism groups. J. Math. Imaging Vision **50**(1-2), 60–97 (2014)
13. Baumgartner, C., Michailovich, O., Levitt, J., Pasternak, O., Bouix, S., Westin, C.-F., Rathi, Y.: A unified tractography framework for comparing diffusion models on clinical scans. In: Computational Diffusion MRI Workshop of MICCAI, Nice, pp. 27–32 (2012)
14. Baust, M., Yezzi, A.J., Unal, G., Navab, N.: A Sobolev-type metric for polar active contours. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2011)
15. Baust, M., Demaret, L., Storath, M., Navab, N., Weinmann, A.: Total variation regularization of shape signals. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2075–2083 (2015)
16. Baust, M., Weinmann, A., Wieczorek, M., Lasser, T., Storath, M., Navab, N.: Combined tensor fitting and TV regularization in diffusion tensor imaging based on a Riemannian manifold approach. IEEE Trans. Med. Imaging **35**(8), 1972–1989 (2016)
17. Belta, C., Kumar, V.: Euclidean metrics for motion generation on SE (3). Proc. Inst. Mech. Eng. C J. Mech. Eng. Sci. **216**(1), 47–60 (2002)
18. Bergmann, R., Laus, F., Steidl, G., Weinmann, A.: Second order differences of cyclic data and applications in variational denoising. SIAM J. Imag. Sci. **7**(4), 2916–2953 (2014)
19. Bertsekas, D.: Incremental proximal methods for large scale convex optimization. Math. Program. **129**, 163–195 (2011)
20. Bredies, K., Holler, M., Storath, M., Weinmann, A.: Total generalized variation for manifold-valued data. SIAM J. Imag. Sci. **11**(3), 1785–1848 (2018)
21. Brubaker, M., Salzmann, M., Urtasun, R.: A family of mcmc methods on implicitly defined manifolds. In: Artificial Intelligence and Statistics, pp. 161–172 (2012)
22. Castaño Moraga, C.A., Westin, C.-F., Ruiz-Alzola, J.: Homomorphic filtering of DT-MRI fields. In: Ellis, R.E., Peters, T.M., Medical Image Computing and Computer-Assisted Intervention (MICCAI), Lecture Notes in Computer Science, vol. 2879, pp. 990–991. Springer, Berlin (2003)
23. Castaño Moraga, C.A., Lenglet, C., Deriche, R., Ruiz-Alzola, J.: A Riemannian approach to anisotropic filtering of tensor fields. Signal Process. **87**(2), 263–276 (2007)
24. Chambolle, A.: Finite-differences discretizations of the Mumford-Shah functional. ESAIM Math. Model. Numer. Anal. **33**(2), 261–288 (1999)
25. Chan, T., Vese, L.: Active contours without edges. IEEE Trans. Image Process. **10**(2), 266–277 (2001)
26. Cheeger, J., Ebin, D.: Comparison Theorems in Riemannian Geometry, vol. 9. North-Holland, Amsterdam (1975)
27. Chefd'hotel, C., Tschumperlé, D., Deriche, R., Faugeras, O.: Constrained flows of matrix-valued functions: application to diffusion tensor regularization. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 251–265 (2002)
28. Chefd'hotel, C., Tschumperlé, D., Deriche, R., Faugeras, O.: Regularizing flows for constrained matrix-valued images. J. Math. Imaging Vision **20**(1-2), 147–162 (2004)
29. Cheng, G., Salehian, H., Vemuri, B.C.: Efficient recursive algorithms for computing the mean diffusion tensor and applications to DTI segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 390–401. Springer, Berlin (2012)
30. Cook, P., Bai, Y., Nedjati-Gilani, S., Seunarine, K., Hall, M., Parker, G., Alexander, D.: Camino: open-source diffusion-MRI reconstruction and processing. In: 14th Scientific Meeting of the International Society for Magnetic Resonance in Medicine, pp. 2759 (2006)

31. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models-their training and application. Comput. Vis. Image Underst. **61**(1), 38–59 (1995)
32. Descoteaux, M., Angelino, E., Fitzgibbons, S., Deriche, R.: Regularized, fast, and robust analytical Q-ball imaging. Magn. Reson. Med. **58**(3), 497–510 (2007)
33. Dryden, I., Mardia, K.: Statistical Shape Analysis (1998)
34. Enayati, N., De Momi, E., Ferrigno, G.: A quaternion-based unscented kalman filter for robust optical/inertial motion tracking in computer-assisted surgery. IEEE Trans. Instrum. Meas. **64**(8), 2291–2301 (2015)
35. Esposito, M., Hennersperger, C., Göbl, R., Demaret, L., Storath, M., Navab, N., Baust, M., Weinmann, A.: Total variation regularization of pose signals with an application to 3D freehand ultrasound. IEEE Trans. Med. Imaging **38**(10), 2245–2258 (2019)
36. Fillard, P., Pennec, X., Arsigny, V., Ayache, N.: Clinical DT-MRI estimation, smoothing, and fiber tracking with log-Euclidean metrics. IEEE Trans. Med. Imaging **26**(11), 1472–1482 (2007)
37. Fletcher, T.P.: Geodesic regression and the theory of least squares on Riemannian manifolds. Int. J. Comput. Vis. **105**(2), 171–185 (2013)
38. Frank, L.R.: Characterization of anisotropy in high angular resolution diffusion-weighted MRI. Magn. Reson. Med. **47**(6), 1083–1099 (2002)
39. Goh, A., Lenglet, C., Thompson, P.M., Vidal, R.: A nonparametric Riemannian framework for processing high angular resolution diffusion images and its applications to ODF-based morphometry. NeuroImage **56**(3), 1181–1201 (2011)
40. Gur, Y., Sochen, N.: Fast invariant Riemannian DT-MRI regularization. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 1–7 (2007)
41. Heimann, T., Meinzer, H.-P.: Statistical shape models for 3d medical image segmentation: a review. Med. Image Anal. **13**(4), 543–563 (2009)
42. Hess, C.P., Mukherjee, P., Han, E.T., Xu, D., Vigneron, D.B.: Q-ball reconstruction of multimodal fiber orientations using the spherical harmonic basis. Magn. Reson. Med. **56**(1), 104–117 (2006)
43. Huber, P.J.: Robust estimation of a location parameter. Ann. Math. Stat. **35**(1), 73–101 (1964)
44. Jansons, K.M., Alexander, D.C.: Persistent angular structure: new insights from diffusion magnetic resonance imaging data. Inverse Prob. **19**(5), 1031–1046 (2003)
45. Kendall, D.G.: Shape manifolds, procrustean metrics, and complex projective spaces. Bull. Lond. Math. Soc. **16**(2), 81–121 (1984)
46. Koay, C., Basser, P.: Analytically exact correction scheme for signal extraction from noisy magnitude MR signals. J. Magn. Reson. **179**(2), 317–322 (2006)
47. Koay, C.G., Chang, L.C., Carew, J.D., Pierpaoli, C., Basser, P.J.: A unifying theoretical and algorithmic framework for least squares methods of estimation in diffusion tensor imaging. J. Magn. Reson. **182**(1), 115–125 (2006)
48. Kortier, H.G., Antonsson, J., Schepers, H.M., Gustafsson, F., Veltink, P.H.: Hand pose estimation by fusion of inertial and magnetic sensing aided by a permanent magnet. IEEE Trans. Neural Syst. Rehabil. Eng. **23**(5), 796–806 (2015)
49. Kwon, K., Kim, D., Kim, S., Park, I., Jeong, J., Kim, T., Hong, C., Han, B.: Regularization of DT-MR images using a successive Fermat median filtering method. Phys. Med. Biol. **53**(10), 2523 (2008)
50. Lang, A., Mousavi, P., Fichtinger, G., Abolmaesumi, P.: Fusion of electromagnetic tracking with speckle-tracked 3D freehand ultrasound using an unscented kalman filter. In: Medical Imaging 2009: Ultrasonic Imaging and Signal Processing, vol. 7265, pp. 72651A. International Society for Optics and Photonics (2009)
51. Lugez, E., Sadjadi, H., Joshi, C.P., Akl, S.G. and Fichtinger, G.: Improved electromagnetic tracking for catheter path reconstruction with application in high-dose-rate brachytherapy. Int. J. Comput. Assist. Radiol. Surg. **12**(4), 681–689 (2017)
52. Luo, J., Zhu, Y., Magnin, I.E.: Denoising by averaging reconstructed images: Application to magnetic resonance images. IEEE Trans. Biomed. Eng. **56**(3), 666–674 (2009)

53. Michor, P.W., Mumford, D.: Riemannian geometries on spaces of plane curves. J. Eur. Math. Soc. (JEMS) **8**, 1–48 (2003)
54. Moakher, M.: Means and averaging in the group of rotations. SIAM J. Matrix Anal. Appl. **24**(1), 1–16 (2002)
55. Mumford, D., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. Commun. Pure Appl. Math. **42**(5), 577–685 (1989)
56. Neji, R., Azzabou, N., Paragios, N., Fleury, G.: A convex semi-definite positive framework for DTI estimation and regularization. In: Lecture Notes in Computer Science, vol. 4841, pp. 220–229. Springer, Berlin (2007)
57. Özarslan, E., Mareci, T.H.: Generalized diffusion tensor imaging and analytical relationships between diffusion tensor imaging and high angular resolution diffusion imaging. Magn. Reson. Med. **50**(5), 955–965 (2003)
58. Park, F.C., Brockett, R.W.: Kinematic dexterity of robotic mechanisms. Int. J. Robot. Res. **13**(1), 1–15 (1994)
59. Pasternak, O., Sochen, N., Assaf, Y.: Variational regularization of multiple diffusion tensor fields. In Weickert, J., Hagen, H., Visualization and Processing of Tensor Fields, Mathematics and Visualization, pp. 165–176. Springer, Berlin (2006)
60. Pasternak, O., Assaf, Y., Intrator, N., Sochen, N.: Variational multiple-tensor fitting of fiber-ambiguous diffusion-weighted magnetic resonance imaging voxel. Magn. Reson. Imaging **26**(8), 1133–1144 (2008)
61. Pennec, X., Fillard, P., Ayache, N.: A Riemannian framework for tensor computing. Int. J. Comput. Vis. **66**, 41–66 (2006)
62. Pock, T., Chambolle, A.: Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In: IEEE International Conference on Computer Vision (ICCV), pp. 1762–1769. IEEE, Piscataway (2011)
63. Raguet, H., Fadili, J., Peyré, G.: A generalized forward-backward splitting. SIAM J. Imag. Sci. **6**(3), 1199–1226 (2013)
64. Rupprecht, C., Huaroc, E., Baust, M., Navab, N.: Deep Active Contours (2016). ArXiv preprint arXiv:1607.05074
65. Sadjadi, H., Hashtrudi-Zaad, K., Fichtinger, G.: Simultaneous electromagnetic tracking and calibration for dynamic field distortion compensation. IEEE Trans. Biomed. Eng. **63**(8), 1771–1781 (2016)
66. Sen, H.T. and Kazanzides, P.: Bayesian filtering to improve the dynamic accuracy of electromagnetic tracking. In: 2013 IEEE International Symposium on Robotic and Sensors Environments (ROSE), pp. 90–95. IEEE, Piscataway (2013)
67. Sijbers, J., den Dekker, A.J., Scheunders, P., Van Dyck, D.: Maximum-likelihood estimation of Rician distribution parameters. IEEE Trans. Med. Imaging **17**(3), 357–361 (1998)
68. Srivastava, A., Klassen, E., Joshi, S.H., Jermyn, I.H.: Shape analysis of elastic curves in euclidean spaces. IEEE Trans. Pattern Anal. Mach. Intell. **33**(7), 1415–1428 (2010)
69. Stefanoiu, A., Weinmann, A., Storath, M., Navab, N., Baust, M.: Joint segmentation and shape regularization with a generalized forward–backward algorithm. IEEE Trans. Image Process. **25**(7), 3384–3394 (2016)
70. Stejskal, E., Tanner, J.: Spin diffusion measurements: Spin echoes in the presence of a time-dependent field gradient. J. Chem. Phys. **42**(1), 288–292 (1965)
71. Storath, M., Weinmann, A., Frikel, J., Unser, M.: Joint image reconstruction and segmentation using the Potts model. Inverse Prob. **31**(2), 025003 (2015)
72. Sundaramoorthi, G., Yezzi, A., Mennucci, A.: Coarse-to-fine segmentation and tracking using sobolev active contours. IEEE Trans. Pattern Anal. Mach. Intell. **30**(5), 851–864 (2008)
73. Tournier, J.D., Calamante, F., Gadian, D.G., Connelly, A.: Direct estimation of the fiber orientation density function from diffusion-weighted MRI data using spherical deconvolution. NeuroImage **23**(3), 1176–1185 (2004)
74. Tschumperlé, D., Deriche, R.: Diffusion tensor regularization with constraints preservation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. I948–I953 (2001)

75. Tschumperlé, D., Deriche, R.: Variational frameworks for DT-MRI estimation, regularization and visualization. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 116–121 (2003)
76. Tschumperlé, D., Deriche, R.: Tensor field visualization with PDE's and application to DT-MRI fiber visualization. In: Workshop on Variational and Level Set Methods (2003)
77. Tuch, D.S.: Q-ball imaging. Magn. Reson. Med. **52**(6), 1358–1372 (2004)
78. Tuch, D.S., Reese, T.G., Wiegell, M.R., Makris, N., Belliveau, J.W., Wedeen, V.J.: High angular resolution diffusion imaging reveals intravoxel white matter fiber heterogeneity. Magn. Reson. Med. **48**(4), 577–582 (2002)
79. Vaccarella, A., De Momi, E., Enquobahrie, A., Ferrigno, G.: Unscented kalman filter based sensor fusion for robust optical and electromagnetic tracking in surgical navigation. IEEE Trans. Instrum. Meas. **62**(7), 2067–2081 (2013)
80. Wan, E.A., Van Der Merwe, R.: The unscented kalman filter for nonlinear estimation. In: Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000, pp. 153–158. IEEE, Piscataway (2000)
81. Wang, Z., Vemuri, B.C., Chen, Y., Mareci, T.H.: A constrained variational principle for direct estimation and smoothing of the diffusion tensor field from complex DWI. IEEE Trans. Med. Imaging **23**(8), 930–939 (2004)
82. Weickert, J., Hagen, H.: Visualization and Processing of Tensor Fields. Springer, Berlin (2006)
83. Weinmann, A., Demaret, L., Storath, M.: Total variation regularization for manifold-valued data. SIAM J. Imag. Sci. **7**(4), 2226–2257 (2014)
84. Weinmann, A., Demaret, L., Storath, M.: Mumford–Shah and Potts regularization for manifold-valued data. J. Math. Imaging Vision **55**(3), 428–445 (2016)
85. Welk, M., Weickert, J., Becker, F., Schnörr, C., Feddern, C., Burgeth, B.: Median and related local filters for tensor-valued images. Signal Process. **87**(2), 291–308 (2007)
86. Xiao, Y., Yan, C.X.B., Drouin, S., De Nigris, D., Kochanowska, A., Collins, D.L.: User-friendly freehand ultrasound calibration using Lego bricks and automatic registration. Int. J. Comput. Assist. Radiol. Surg. **11**(9), 1703–1711 (2016)
87. Zacur, E., Bossa, M., Olmos, S.: Left-invariant riemannian geodesics on spatial transformation groups. SIAM J. Imag. Sci. **7**(3), 1503–1557 (2014)
88. Zefran, M., Kumar, V., Croke, C.: Choice of Riemannian metrics for rigid body kinematics. In: ASME 24th Biennial Mechanisms Conference, vol. 2 (1996)

Chapter 23
The Riemannian and Affine Geometry of Facial Expression and Action Recognition

Mohamed Daoudi, Juan-Carlos Alvarez Paiva, and Anis Kacem

Contents

M. Daoudi (✉) · A. Kacem
IMT Lille-Douai, University of Lille, CNRS, UMR 9189, CRIStAL, Lille, France
e-mail: mohamed.daoudi@imt-lille-douai.fr

J.-C. Alvarez Paiva
University of Lille, CNRS-UMR-8524, Lille, France

Painlevé Laboratory, Villeneuve-d'Ascq, France

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_23

Abstract Recent advances in human 2D and 3D landmarks tracking have made it possible to model facial expression and action recognition as a temporal sequence of landmarks. We work directly with the Euclidean or affine invariants of landmarks. These invariants are represented as points in different shape spaces (Positive Semi-Definite (PSD) manifold, Grassmann manifold) and therefore their temporal evolution can be seen as a trajectory in these spaces. Using Riemannian geometry, these trajectories can be compared and classified, which has immediate applications in facial expression and action recognition.

23.1 Landmark Representation

In the last decades, automatic analysis of human behavior has been an active research topic, with applications that have been exploited in a number of different contexts, including video surveillance, semantic annotation of videos, entertainment, human computer interaction and home care rehabilitation, to say a few. For years, the approaches could be distinguished in two main classes: those operating on pixel values extracted from the RGB stream and those building upon the higher level representation of body skeletons and face landmarks.These latter approaches were supported by the diffusion of low cost RGB-D cameras (such as the Microsoft Kinect) that can operate in real-time, while reliably extracting the 3D coordinates of body joints. In this chapter, we will focus on designing effective landmark based solutions for facial expression and action recognition. One of our motivations for using landmarks representation is driven by the recent impressive advances in human landmark tracking. As mentioned above, recently landmark detection and tracking methods from human faces and bodies became reliable and accurate. They are robust to illumination changes that occur in RGB images. By considering the tracked landmarks instead of the original images, we take advantage of the robustness of tracking methods to these classical problems in Computer Vision and expect the same robustness for our landmark based solutions. Figure 23.1 shows examples of landmarks and skeleton.

Furthermore, considering only tracked landmarks reduces the complexity of the visual data. Instead of using a large number of pixels in each frame of the original

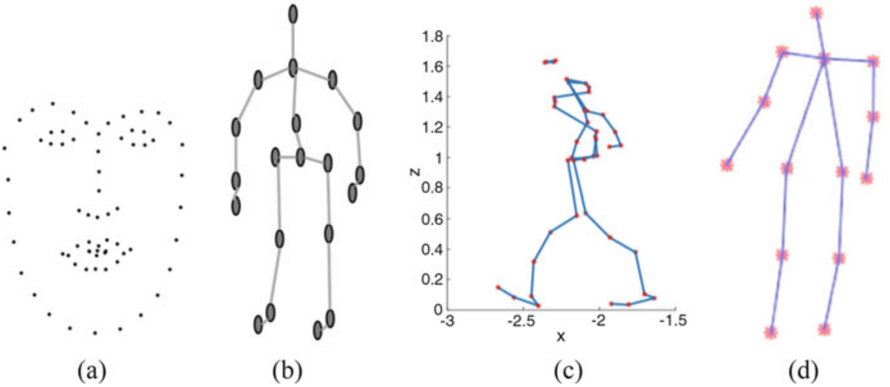


Fig. 23.1 (**a**) Facial landmarks, (**b**) Kinect skeleton, (**c**) Frame from a MoCap skeleton sequence, (**d**) OpenPose skeleton

video, which could make the analysis computationally intense, landmark trackers bring a brief summary of the frame by providing only a set of relevant 2D/3D points (the number of points typically varies from 15 to 90 points). Hence, landmark based solutions are expected to be more efficient and less computational expensive than other solutions, which makes them more suitable for real-time applications.

23.1.1 Challenges

While powerful and robust to many Computer Vision problems, human landmark tracking techniques generate temporal sequences of landmark configurations which exhibit several challenges:

- View variations: The 2D or 3D locations provided by the coordinates of the tracked landmarks are relative to the position of the camera. However, human behavioral signals belonging to the same category (e.g., drinking water), can occur in different positions w.r.t the camera.
- Rate variations: The human behavioral signals that we would like to analyze are subject to high temporal variations. For instance, two persons do not perform the same action (e.g., drinking water) at the same time and for the same duration. Consequently, we cannot simply compare the static landmark configurations of the two corresponding landmark sequences in order to know whether they are similar or not. Effective landmark based solutions should take into account these temporal (rate) variations in the analysis of human landmark sequences.
- Intra-class variations: Another challenge of stastical analyis of landmark sequences consists of the large variations that can be present within the same category of human behavioral signals. Indeed, behavioral signals of the same

category could be different from one person to another or even for the same person.[1]

- Inaccurate tracking and missing data: Despite the advances in tracking human landmarks, inaccurate tracking can occur especially in unconstrained environments and challenging conditions.

While there have been many efforts in the analysis of temporal sequences of landmarks, the problem is far from being solved and the current solutions are facing many technical and practical problems.

23.2 Static Representation

In this chapter we adopt the notations defined in [26, 28]. Let us consider an arbitrary sequence of landmark configurations $\{Z_0, \ldots, Z_\tau\}$. Each configuration Z_i $(0 \leq i \leq \tau)$ is an $n \times d$ matrix of rank d encoding the positions of n distinct landmark points in d dimensions. In our applications, we only consider the configurations of landmark points in two- or three-dimensional space (i.e., d=2 or d=3) given by, respectively, $p_1 = (x_1, y_1), \ldots, p_n = (x_n, y_n)$ or $p_1 = (x_1, y_1, z_1), \ldots, p_n = (x_n, y_n, z_n)$. We are interested in studying such sequences or curves of landmark configurations up to Euclidean motions. In the following, we will first propose a representation for static observations, then adopt a time-parametrized representation for temporal analysis.

As a first step, we seek a shape representation that is invariant up to Euclidean transformations (rotation and translation). Arguably, the most natural choice is the matrix of pairwise distances between the landmarks of the same shape augmented by the distances between all the landmarks and their center of mass p_0. Since we are dealing with Euclidean distances, it will turn out to be more convenient to consider the matrix of the squares of these distances. Also note that by subtracting the center of mass from the coordinates of the landmarks, these can be considered as *centered*: the center of mass is always at the origin. From now on, we will assume $p_0 = (0, 0)$ for $d = 2$ (or $p_0 = (0, 0, 0)$ for $d = 3$). With this provision, the augmented pairwise square-distance matrix $\mathcal{D}$ takes the form,

$$\mathcal{D} := \begin{pmatrix} 0 & \|p_1\|^2 & \cdots & \|p_n\|^2 \\ \|p_1\|^2 & 0 & \cdots & \|p_1 - p_n\|^2 \\ \vdots & \vdots & \vdots & \vdots \\ \|p_n\|^2 & \|p_n - p_1\|^2 & \cdots & 0 \end{pmatrix},$$

where $\| \cdot \|$ denotes the norm associated to the l^2-inner product $\langle \cdot, \cdot \rangle$. A key observation is that the matrix $\mathcal{D}$ can be easily obtained from the $n \times n$ Gram matrix

[1] Example taken from www.slideshare.net/NaverEngineering/human-action-recognition.

$G := ZZ^T$. Indeed, the entries of G are the pairwise inner products of the points $p_1, \ldots, p_n$,

$$G = ZZ^T, \tag{23.1}$$

$$G = \begin{pmatrix} \langle p_1, p_1 \rangle & \langle p_1, p_2 \rangle & \cdots & \langle p_1, p_n \rangle \\ \langle p_2, p_1 \rangle & \langle p_2, p_2 \rangle & \cdots & \langle p_2, p_n \rangle \\ \vdots & \vdots & \vdots & \vdots \\ \langle p_n, p_1 \rangle & \langle p_n, p_2 \rangle & \cdots & \langle p_n, p_n \rangle \end{pmatrix}$$

and the equality

$$\mathcal{D}_{ij} = \langle p_i, p_i \rangle - 2\langle p_i, p_j \rangle + \langle p_j, p_j \rangle, \quad 0 \leq i, j \leq n \ , \tag{23.2}$$

establishes a linear equivalence between the set of $n \times n$ Gram matrices and the augmented square-distance $(n + 1) \times (n + 1)$ matrices of distinct landmark points. On the other hand, Gram matrices of the form ZZ^T, where Z is an $n \times d$ matrix of rank d are characterized as $n \times n$ positive semidefinite matrices of rank d. For a detailed discussion of the relation between positive semidefinite matrices, Gram matrices, and square-distance matrices, we refer the reader to Section 6.2.1 of [14]. The space of these matrices, called the positive semidefinite cone $\mathcal{S}^+(d, n)$, is a not a vector space and is mostly studied when endowed with a Riemannian metric. In the next section, we will briefly review some basics of the Riemannian geometry of the manifolds of interest, then express the Riemannian geometry of the space of Gram matrices (i.e., positive semi-definite matrices of fixed rank).

23.3 Riemannian Geometry of the Space of Gram Matrices

23.3.1 Mathematical Preliminaries

A manifold is a topological space that is locally homeomorphic to the dim-dimensional Euclidean space $\mathbb{R}^{dim}$, where dim is the dimensionality of the manifold. A differentiable manifold is a topological manifold equipped with a differential structure that allows differential calculus on the manifold. The tangent space at a given point on a differentiable manifold is a vector space that consists of the tangent vectors of all possible curves passing through the point. A Riemannian manifold is a differentiable manifold equipped with a smoothly varying inner product on each tangent space. The family of inner products on all tangent spaces is known as the Riemannian metric of the manifold [24]. By defining a Riemannian metric on the manifold, one can exploit the vector space structure of the tangent space to define various geometric notions on the manifold. One can compute the *geodesic distance* between two points on the manifold which is the length of the shortest curve (i.e., *geodesic*) connecting these two points. Two other important operations in Riemannian manifolds are the logarithm (log) and exponential (exp) maps. To illustrate these two operations, let us consider two points X and Y lying on a Riemannian manifold $\mathcal{M}$. The logarithm map $\log_X(Y)$ of the point Y to the tangent space $T_X(\mathcal{M})$ attached to X results in a vector V in $T_X(\mathcal{M})$. This vector summarizes the path that should be taken in $\mathcal{M}$ to connect X and Y. In contrast, the exponential map $\exp_X(V)$ maps back the vector V to the manifold $\mathcal{M}$ resulting in a curve $\gamma(t)$ in $\mathcal{M}$ connecting X and Y. It is important to note that the computation of these operations depends on the nature of the manifold and the defined Riemannian metric.

Conveniently for us, the Riemannian geometry of the space of positive semidefinite matrices of fixed rank (i.e., Gram matrices) was studied in [11, 18, 37, 46]. To have a better understanding of the geometry of this space, we first define two manifolds that are extensively used in Computer Vision namely, the Grassmann manifold and the Riemannian manifold of positive definite matrices.

23.3.1.1 Grassmann Manifold

A Grassmann manifold $\mathcal{G}(d, n)$ is the set of the d-dimensional subspaces of $\mathbb{R}^n$, where $n > d$. A subspace $\mathcal{U}$ of $\mathcal{G}(d, n)$ is represented by an $n \times d$ matrix U, whose columns store an orthonormal basis of this subspace. Thus, U is said to span $\mathcal{U}$, and $\mathcal{U}$ is said to be the column space (or span) of U, and we write $\mathcal{U} = span(U)$. Indeed, the set of $n \times d$ matrices with orthonormal columns forms a manifold known as the Stiefel manifold $\mathcal{V}_{d,n}$. Points on $\mathcal{G}(d, n)$ are equivalence classes of $n \times d$ matrices with orthonormal columns (i.e., points on $\mathcal{V}_{d,n}$), where two matrices are equivalent if their columns span the same d-dimensional subspace. The geometry of the Grassmannian $\mathcal{G}(d, n)$ is then easily described by the map

$$span : \mathcal{V}_{d,n} \to \mathcal{G}(d, n) \ , \tag{23.3}$$

that sends an $n \times d$ matrix with orthonormal columns U to their span $span(U)$. Given two subspaces $\mathcal{U}_1 = span(U_1)$ and $\mathcal{U}_2 = span(U_2) \in \mathcal{G}(d, n)$, the geodesic curve connecting them is

$$span(U(t)) = span(U_1 \cos(\Theta t) + M \sin(\Theta t)) \ , \tag{23.4}$$

where Θ is a $d \times d$ diagonal matrix formed by the d *principal angles* between $\mathcal{U}_1$ and $\mathcal{U}_2$, while the matrix M is given by $M = (I_n - U_1 U_1^T) U_2 F$, with F being the pseudo-inverse of Θ. The Riemannian geodesic distance between $\mathcal{U}_1$ and $\mathcal{U}_2$ is given by

$$d_{\mathcal{G}}^2(\mathcal{U}_1, \mathcal{U}_2) = \|\Theta\|_F^2 \ . \tag{23.5}$$

23.3.1.2 Riemannian Manifold of Positive Definite Matrices

A symmetric $d \times d$ matrix R is said to be positive definite if and only if $v^{\mathrm{T}} R v > 0$ for every non-zero vector $v \in \mathbb{R}^d$. $\mathcal{P}_d$ is mostly studied when endowed with a Riemannian metric, thus forming a Riemannian manifold. A number of metrics have been proposed for $\mathcal{P}_d$, the most popular ones being the Affine-Invariant Riemannian Metric (AIRM) and the log-Euclidean Riemannian metric (LERM) [4]. In this study, we only consider the AIRM for its robustness [44].

With this metric, the geodesic curve connecting two SPD matrices R_1 and R_2 in $\mathcal{P}_d$ is

$$R(t) = R_1^{1/2} \exp(t \log(R_1^{-1/2} R_2 R_1^{-1/2})) R_1^{1/2} , \tag{23.6}$$

where $log(.)$ and $exp(.)$ are the matrix logarithm and exponential, respectively. The Riemannian distance between R_1 and R_2 is given by

$$d^2_{P_d}(R_1, R_2) = \| \log (R_1^{-1/2} R_2 R_1^{-1/2}) \|^2_{\mathrm{F}} , \tag{23.7}$$

where $\|.\|_F$ denotes the Frobenius matrix norm.

For more details about the geometry of the Grassmannian $\mathcal{G}(d, n)$ and the positive definite cone $\mathcal{P}_d$, readers are referred to [2, 8, 11, 38].

23.3.2 Riemannian Manifold of Positive Semi-Definite Matrices of Fixed Rank

Given an $n \times d$ matrix Z of rank d, its polar decomposition $Z = UR$ with $R = (Z^T Z)^{1/2}$ allows us to write the Gram matrix ZZ^T as UR^2U^T. Since the columns of the matrix U are orthonormal, this decomposition defines a map

$$\begin{aligned} \Pi : & \mathcal{V}_{d,n} \times \mathcal{P}_d \to \mathcal{S}^+(d, n) \\ & (U, R^2) \mapsto UR^2U^T \ , \end{aligned}$$

from the product of the Stiefel manifold $\mathcal{V}_{d,n}$ and the cone of $d \times d$ positive definite matrices $\mathcal{P}_d$ to the manifold $\mathcal{S}^+(d, n)$ of $n \times n$ positive semidefinite matrices of rank d. The map Π defines a principal fiber bundle over $\mathcal{S}^+(d, n)$ with fibers

$$\Pi^{-1}(UR^2U^T) = \{(UO, O^T R^2 O) : O \in \mathcal{O}(d)\} ,$$

where $\mathcal{O}(d)$ is the group of $d \times d$ orthogonal matrices. Bonnabel and Sepulchre [11] used this map and the geometry of the *structure space* $\mathcal{V}_{d,n} \times \mathcal{P}_d$ to introduce a Riemannian metric on $\mathcal{S}^+(d, n)$ and study its geometry.

23.3.2.1 Tangent Space and Riemannian Metric

The tangent space $T_{(U,R^2)}(\mathcal{V}_{d,n} \times \mathcal{P}_d)$ consists of pairs (M, N), where M is a $n \times d$ matrix satisfying $M^T U + U^T M = 0$ and N is any $d \times d$ symmetric matrix. Bonnabel and Sepulchre defined a *connection* (see [30, p. 63]) on the principal bundle $\Pi : \mathcal{V}_{d,n} \times \mathcal{P}_d \to \mathcal{S}^+(d, n)$ by setting the horizontal subspace $\mathcal{H}_{(U,R^2)}$ at the point (U, R^2) to be the space of tangent vectors (M, N) such that $M^T U = 0$ and N is an arbitrary $d \times d$ symmetric matrix. They also defined an inner product on $\mathcal{H}_{(U,R^2)}$: given two tangent vectors $A = (M_1, N_1)$ and $B = (M_2, N_2)$ on $\mathcal{H}_{(U,R^2)}$, set

$$\langle (A, B)\rangle_{\mathcal{H}_{U,R^2}} = \mathrm{tr}(M_1^T M_2) + k\,\mathrm{tr}(N_1 R^{-2} N_2 R^{-2}) \ , \tag{23.8}$$

where $k > 0$ is a real parameter.

It is easily checked that the action of the group of $d \times d$ orthogonal matrices on the fiber $\Pi^{-1}(UR^2U^T)$ sends horizontals to horizontals isometrically. It follows that the inner product on $T_{UR^2U^T}\mathcal{S}^+(d, n)$ induced from that of $\mathcal{H}_{(U,R^2)}$ via the linear isomorphism $D\Pi$ is independent of the choice of point (U, R^2) projecting onto UR^2U^T. This procedure defines a Riemannian metric on $\mathcal{S}^+(d, n)$ for which the natural projection

$$\begin{aligned} \rho : \mathcal{S}^+(d, n) &\to \mathcal{G}(d, n) \\ G &\mapsto range(G) \ , \end{aligned}$$

is a Riemannian submersion. This allows us to relate the geometry of $\mathcal{S}^+(d, n)$ with that of the Grassmannian $\mathcal{G}(d, n)$.

23.3.2.2 Pseudo-Geodesics and Closeness in $\mathcal{S}^+(d, n)$

Bonnabel and Sepulchre [11] defined the *pseudo-geodesic* connecting two matrices $G_1 = U_1 R_1^2 U_1^T$ and $G_2 = U_2 R_2^2 U_2^T$ in $\mathcal{S}^+(d, n)$ as the curve

$$\mathcal{C}_{G_1 \to G_2}(t) = U(t)R^2(t)U^T(t), \forall t \in [0, 1] \ , \tag{23.9}$$

where $R^2(t) = R_1 \exp(t \log R_1^{-1} R_2^2 R_1^{-1}) R_1$ is a geodesic in $\mathcal{P}_d$ connecting R_1^2 and R_2^2, and $U(t)$ is the geodesic in $\mathcal{G}(d, n)$ given by Eq. (23.4). They also defined the *closeness* between G_1 and G_2, $d_{\mathcal{S}^+}(G_1, G_2)$, as the square of the length of this curve:

$$d_{\mathcal{S}^+}(G_1, G_2) = d^2_{\mathcal{G}}(\mathcal{U}_1, \mathcal{U}_2) + k d^2_{\mathcal{P}_d}(R_1^2, R_2^2) = \|\Theta\|_F^2 + k \|\log R_1^{-1} R_2^2 R_1^{-1}\|_F^2 \ , \tag{23.10}$$

where $\mathcal{U}_i$ $(i = 1, 2)$ is the *span* of U_i and Θ is a $d \times d$ diagonal matrix formed by the principal angles between $\mathcal{U}_1$ and $\mathcal{U}_2$.

The closeness $d_{\mathcal{S}^+}$ consists of two independent contributions: the square of the distance $d_{\mathcal{G}}(span(U_1), span(U_2))$ between the two associated subspaces, and the square of the distance $d_{\mathcal{P}_d}(R_1^2, R_2^2)$ on the positive cone $\mathcal{P}_d$. Note that $C_{G_1 \to G_2}$ is not necessarily a geodesic and therefore, the closeness $d_{\mathcal{S}^+}$ is not a true Riemannian distance.

23.3.3 *Affine-Invariant and Spatial Covariance Information of Gram Matrices*

An alternative affine shape representation, considered in [8] and [43], associates to each configuration Z the d-dimensional subspace $span(Z)$ spanned by its columns. This representation, which exploits the geometry of the Grassmann manifold $\mathcal{G}(d, n)$ of d-dimensional subspaces in $\mathbb{R}^n$ is invariant under *all* invertible linear transformations. By fully encoding the set of all mutual distances between landmark points, the proposed Euclidean shape representation supplements the affine shape representation with the knowledge of the $d \times d$ positive definite matrix R^2 that lie on $\mathcal{P}_d$.

From the viewpoint of the landmark configurations Z_1 and Z_2, with $G_1 = Z_1 Z_1^T$ and $G_2 = Z_2 Z_2^T$, the closeness $d_{\mathcal{S}^+}$ encodes the distances measured between the affine shapes $span(Z_1)$ and $span(Z_2)$ in $\mathcal{G}(d, n)$ and between their spatial covariances in $\mathcal{P}_d$. Indeed, the spatial covariance of Z_i $(i = 1, 2)$ is the $d \times d$ symmetric positive definite matrix

$$C = \frac{Z_i^T Z_i}{n-1} = \frac{(U_i R_i)^T (U_i R_i)}{n-1} = \frac{R_i^2}{n-1} \ . \tag{23.11}$$

The weight parameter k controls the relative weight of these two contributions. Note that for $k = 0$ the distance on $\mathcal{S}^+(d, n)$ collapses to the distance on $\mathcal{G}(d, n)$. Nevertheless, the authors in [11] recommended choosing small values for this parameter.

23.4 Gram Matrix Trajectories for Temporal Modeling of Landmark Sequences

We are able to compare static landmark configurations based on their Gramian representation G, the induced space, and closeness introduced in the previous Section. We need a natural and effective extension to study their temporal evolution. Following [9, 43, 48], we defined curves $\beta_G \ : I \ \to \ \mathcal{S}^+(d, n)$ (I denotes the time domain, e.g., [0, 1]) to model the spatio-temporal evolution of elements on $\mathcal{S}^+(d, n)$. Given a sequence of landmark configurations $\{Z_0, \ldots, Z_\tau\}$ represented by their corresponding Gram matrices $\{G_0, \dots, G_\tau\}$ in $\mathcal{S}^+(d, n)$, the corresponding curve is the trajectory of the point $\beta_G(t)$ on $\mathcal{S}^+(d, n)$, when t ranges in [0, 1]. These curves are obtained by connecting all successive Gramian representations of shapes G_i and G_{i+1}, $0 \leq i \leq \tau - 1$, by pseudo-geodesics in $\mathcal{S}^+(d, n)$.[2]

23.4.1 Rate-Invariant Comparison of Gram Matrix Trajectories

A relevant issue to our classification problems is—how to compare trajectories while being invariant to rates of execution? One can formulate the problem of temporal misalignment as comparing trajectories when parameterized differently. The parameterization variability makes the distance between trajectories distorted. This issue was first highlighted by Veeraraghavan et al. [47] who showed that different rates of execution of the same activity can greatly decrease recognition performance if ignored. Veeraraghan et al. [47] and Abdelkader et al. [1] used the Dynamic Time Warping (DTW) for temporal alignment before comparing trajectories of shapes of planar curves that represent silhouettes in videos. Following the above-mentioned state-of-the-art solutions, we adopt here a DTW solution to temporally align our trajectories. More formally, given m trajectories $\{\beta_G^1, \beta_G^2, \dots, \beta_G^m\}$ on $\mathcal{S}^+(d, n)$, we are interested in finding functions γ_i such that the $\beta_G^i(\gamma_i(t))$ are matched optimally for all $t \in [0, 1]$. In other words, two curves $\beta_G^1(t)$ and $\beta_G^2(t)$ represent the same trajectory if their images are the same. This happens if, and only if, $\beta_G^2 = \beta_G^1 \circ \gamma$, where γ is a re-parameterization of the interval [0, 1]. The problem of temporal alignment is turned to find an optimal warping function $\gamma^\star$ according to,

$$\gamma^\star = \arg\min_{\gamma \in \Gamma} \int_0^1 d_{\mathcal{S}^+}(\beta_G^1(t), \beta_G^2(\gamma(t)))\,\mathrm{d}t\ , \tag{23.12}$$

where Γ denotes the set of all monotonically-increasing functions $\gamma : [0, 1] \to [0, 1]$. The most commonly used method to solve such optimization problem is DTW. Note that accommodation of the DTW algorithm to the manifold-value sequences can be achieved with respect to an appropriate metric defined on the underlying manifold $\mathcal{S}^+(d, n)$. Having the optimal re-parametrization function $\gamma^\star$, one can define a (dis-)similarity measure between two trajectories allowing a rate-invariant comparison:

$$d_{DTW}(\beta_G^1, \beta_G^2) = \int_0^1 d_{\mathcal{S}^+}(\beta_G^1(t), \beta_G^2(\gamma^\star(t)))\,\mathrm{d}t. \tag{23.13}$$

[2] To compute the polar decomposition, we used the SVD based implementation proposed in [21].

From now, we shall use $d_{DTW}(.,.)$ to compare trajectories in our manifold of interest $\mathcal{S}^+(d,n)$.

23.5 Classification of Gram Matrix Trajectories

Our trajectory representation reduces the problem of landmark sequence classification to that of trajectory classification in $\mathcal{S}^+(d,n)$. That is, let us consider $\mathcal{T} = \{\beta_G : [0,1] \to \mathcal{S}^+(d,n)\}$, the set of time-parameterized trajectories of the underlying manifold. Let $\mathcal{L} = \{(\beta_G^1, y^1), \dots, (\beta_G^m, y^m)\}$ be the training set with class labels, where $\beta_G^i \in \mathcal{T}$ and $y^i \in \mathcal{Y}$, e.g. $\mathcal{L} = \{Happiness, Sadness, Surprise, Fear, Disgust, Anger\}$ such that $y^i = f(\beta_G^i)$. The goal here is to find an approximation h to f such that $h : \mathcal{T} \to \mathcal{L}$. In Euclidean spaces, any standard classifier (e.g., standard SVM) may be a natural and appropriate choice to classify the trajectories. Unfortunately, this is no more suitable in our modeling, as the space $\mathcal{T}$ built from $\mathcal{S}^+(d,n)$ is non-linear. As mentioned and discussed in the previous chapter, a function that divides the manifold is rather a complicated notion compared with the Euclidean space. To overcome this issue, we adopt two classification schemes based on the (dis-)similarity measure d_{DTW} that uses the geometry-aware closeness $d_{\mathcal{S}^+}$ namely, k-Nearest Neighbor and Pairwise proximity function SVM classifiers.

23.5.1 Pairwise Proximity Function SVM

Inspired by a recent work of [6] for action recognition, we adopted the *pairwise proximity function SVM* (ppfSVM) [19, 20]. The ppfSVM requires the definition of a (dis-)similarity measure to compare samples. In our case, it is natural to consider the d_{DTW} defined in Eq. (23.13) for such a comparison. This strategy involves the construction of inputs such that each trajectory is represented by its (dis-)similarity to all the trajectories, with respect to d_{DTW}, in the dataset and then apply a conventional SVM to this transformed data [20]. The ppfSVM is related to the arbitrary kernel-SVM without restrictions on the kernel function [19].

Given m trajectories $\{\beta_G^1, \beta_G^2, \dots, \beta_G^m\}$ in $\mathcal{T}$, following [6], a proximity function $\mathcal{P}_{\mathcal{T}} : \mathcal{T} \times \mathcal{T} \to \mathbb{R}_+$ between two trajectories $\beta_G^1, \beta_G^2 \in \mathcal{T}$ is defined as,

$$\mathcal{P}_{\mathcal{T}}(\beta_G^1, \beta_G^2) = d_{DTW}(\beta_G^1, \beta_G^2) \ . \tag{23.14}$$

According to [19], there are no restrictions on the function $\mathcal{P}_{\mathcal{T}}$. For an input trajectory $\beta_G \in \mathcal{T}$, the mapping $\phi(\beta_G)$ is given by,

$$\phi(\beta_G) = [\mathcal{P}_{\mathcal{T}}(\beta_G, \beta_G^1), \dots, \mathcal{P}_{\mathcal{T}}(\beta_G, \beta_G^m)]^T \ . \tag{23.15}$$

The obtained vector $\phi(\beta_G) \in \mathbb{R}^m$ is used to represent a sample trajectory $\beta_G \in \mathcal{T}$. Hence, the set of trajectories can be represented by a $m \times m$ matrix P, where $P(i, j) = \mathcal{P}_{\mathcal{T}}(\beta_G^i, \beta_G^j)$, $i, j \in \{1, \ldots, m\}$. Finally, a linear SVM is applied to this data representation. Further details on ppfSVM can be found in [6, 19, 20].

23.6 Application to Facial Expression and Action Recognition

23.6.1 2D Facial Expression Recognition

We evaluated our approach also in the task of facial expression recognition from 2D landmarks. In this case, the landmarks are in a 2D coordinate space, resulting in a Gram matrix of size $n \times n$ of rank 2 for each configuration of n landmarks. The facial sequences are then seen as time-parameterized trajectories on $\mathcal{S}^+(2, n)$.

23.6.1.1 Datasets

We conducted experiments on two publicly available datasets—CK+, MMI.

Cohn-Kanade Extended (CK+) Dataset [36]—It contains 123 subjects and 593 frontal image sequences of posed expressions. Among them, 118 subjects are annotated with the seven labels—*anger* (An), *contempt* (Co), *disgust* (Di), *fear* (Fe), *happy* (Ha), *sad* (Sa) and *surprise* (Su). Note that only the two first temporal phases of the expression, i.e., neutral and onset (with apex frames), are present.

MMI Dataset [45]—It consists of 205 image sequences with frontal faces of 30 subjects labeled with the six basic emotion labels. In this dataset each sequence begins with a neutral facial expression, and has a posed facial expression in the middle; the sequence ends up with the neutral facial expression. The location of the peak frame is not provided as a prior information.

23.6.1.2 Experimental Settings and Parameters

All our experiments were performed once facial landmarks were extracted using the method proposed in [5] on the CK+, MMI, and Oulu-CASIA datasets. On the challenging AFEW dataset, we have considered the corrections provided in[3] after applying the same detector. The number of landmarks is $n = 49$ for each face.

[3] http://sites.google.com/site/chehrahome.

To evaluate our approach, we followed the experimental settings commonly used in recent works. Following [17, 25, 33], we have performed 10-fold cross validation experiments for the CK+, MMI, and Oulu-CASIA datasets. In contrast, the AFEW dataset was divided into three sets: training, validation and test, according to the protocols defined in EmotiW'2013 [16]. Here, we only report our results on the validation set for comparison with [16, 17, 33].

23.6.1.3 Results and Discussion

On CK+, the average accuracy is 96.87%. Note that the accuracy of the trajectory representation on $\mathcal{G}(2, n)$, following the same pipeline is 2% lower, which confirms the contribution of the covariance embedded in our representation.

An average classification accuracy of 79.19% is reported for the MMI dataset. Note that based on geometric features only, our approach grounding on both representations on $\mathcal{S}^+(2, n)$ and $\mathcal{G}(2, n)$ achieved competitive results with respect to the literature (see Table 23.1).

We highlight the superiority of the trajectory representation on $\mathcal{S}^+(2, n)$ over the Grassmannian. This is due to the contribution of the covariance part further to the conventional affine-shape analysis over the Grassmannian. Recall that k serves to balance the contribution of the distance between covariance matrices living in $\mathcal{P}_2$ with respect to the Grassmann contribution $\mathcal{G}(2, n)$. The optimal performance are achieved for the following values—$k^*_{CK+} = 0.081$, $k^*_{MMI} = 0.012$, $k^*_{Oulu-CASIA} = 0.014$ and $k^*_{AFEW} = 0.001$.

Comparative Study with the State-of-the-Art In Table 23.1, we compare our approach over the recent literature. Overall, our approach achieved competitive performance with respect to the most recent approaches. On CK+, we obtained the second highest accuracy. The ranked-first approach is DTAGN [25], in which two deep networks are trained on shape and appearance channels, then fused. Note that the geometry deep network (DTGN) achieved 92.35%, which is much lower than ours. Furthermore, our approach outperforms the ST-RBM [17] and the STM-ExpLet [33]. On the MMI dataset, our approach outperforms the DTAGN [25] and the STM-ExpLet [33]. However, it is behind ST-RBM [17].

On the Oulu-CASIA dataset, our approach shows a clear superiority to existing methods, in particular STM-ExpLet [33] and DTGN [25]. Elaiwat et al. [17] do not report any results on this dataset, however, their approach achieved the highest accuracy on AFEW. Our approach is ranked second showing a superiority to remaining approaches on AFEW.

Then, we have used different distances defined on $\mathcal{S}^+(2, n)$. Specifically, given two matrices G_1 and G_2 in $\mathcal{S}^+(2, n)$: (1) we used $d_{\mathcal{P}_n}$ to compare them by regularizing their ranks, i.e., making them n full-rank, and considering them in $\mathcal{P}_n$ (the space of n-by-n positive definite matrices), $d_{\mathcal{P}_n}(G_1, G_2) = d_{\mathcal{P}_n}(G_1 + \epsilon I_n, G_2 + \epsilon I_n)$; (2) we used the Euclidean flat distance $d_{\mathcal{F}^+}(G_1, G_2) = \|G_1 - G_2\|_F$, where $\|.\|_F$ denotes the Frobenius-norm. The closeness $d_{\mathcal{S}^+}$ between two elements

Table 23.1 Overall accuracy (%) on CK+ and MMI datasets

Method	CK+	MMI
$^{(A)}$ 3D HOG (from [25])	91.44	60.89
$^{(A)}$ 3D SIFT (from [25])	–	64.39
$^{(A)}$ Cov3D (from [25])	92.3	–
$^{(A)}$ STM-ExpLet [33] (10-fold)	94.19	75.12
$^{(A)}$ CSPL [56] (10-fold)	89.89	73.53
$^{(A)}$ F-Bases [39] (LOSO)	96.02	75.12
$^{(A)}$ ST-RBM [17] (10-fold)	95.66	81.63
$^{(A)}$ 3DCNN-DAP [32] * (15-fold)	87.9	62.2
$^{(A)}$ DTAN [25] * (10-fold)	91.44	62.45
$^{(A+G)}$ DTAGN [25] * (10-fold)	97.25	70.24
$^{(G)}$ DTGN [25] * (10-fold)	92.35	59.02
$^{(G)}$ TMS [23] (4-fold)	85.84	–
$^{(G)}$ HMM [50] (15-fold)	83.5	51.5
$^{(G)}$ ITBN [50] (15-fold)	86.3	59.7
$^{(G)}$ Velocity on $\mathcal{G}(n, 2)$[43]	82.8	–
$^{(G)}$ traj. on $\mathcal{G}(2, n)$ (10-fold)	94.25 ± 3.71	78.18 ± 4.87
$^{(G)}$ traj. on $\mathcal{S}^+(2, n)$ (10-fold)	96.87 ± 2.46	79.19 ± 4.62

Here, (A): appearance (or color); (G): geometry (or shape); $*$: Deep Learning based approach; last row: ours

Table 23.2 Distance performances and computational complexity on the CK+

Distance	CK+ (%)	Time (s)
Flat distance $d_{\mathcal{F}^+}$	93.78 ± 2.92	0.020
Distance $d_{\mathcal{P}_n}$ in $\mathcal{P}_n$	92.92 ± 2.45	0.816
Closeness $d_{\mathcal{S}^+}$	96.87 ± 2.46	0.055

of $\mathcal{S}^+(2, n)$ defined in Eq. (23.7) is more suitable, compared to the distance $d_{\mathcal{P}_n}$ and the flat distance $d_{\mathcal{F}^+}$ defined in literature. The results in Table 23.2 show the importance of being faithful to the geometry of the manifold of interest.

23.6.2 3D Action Recognition

Action recognition has been performed on 3D skeleton data as provided by a Kinect camera in different datasets. In this case, landmarks correspond to the estimated position of 3D joints of the skeleton (d=3). With this assumption, skeletons are represented by $n \times n$ Gram matrices of rank 3 lying on $\mathcal{S}^+(3, n)$, and skeletal sequences are seen as trajectories on this manifold.

23.6.2.1 Datasets

We performed experiments on two publicly available datasets showing different challenges. All these datasets have been collected with a Microsoft Kinect sensor.

UT-Kinect Dataset [52]—It contains 10 actions performed by 10 different subjects. Each subject performed each action twice resulting in 199 valid action sequences. The 3D locations of 20 joints are provided with the dataset.

Florence3D Dataset [41]—It contains 9 actions performed two or three times by 10 different subjects. Skeleton comprises 15 joints. This is a challenging dataset due to variations in the view-point and large intra-class variations.

23.6.2.2 Experimental Settings and Parameters

For all the datasets, we used only the provided skeletons. As discussed in Sect. 23.3.3, our body movement representation involves a parameter k that controls the contribution of two information: the affine shape of the skeleton at time t, and its spatial covariance. The affine shape information is given by the Grassmann manifold $\mathcal{G}(3, n)$, while the spatial covariance is given by the SPD manifold $\mathcal{P}_3$. We recall that for $k = 0$, the skeletons are considered as trajectories on the Grassmann manifold $\mathcal{G}(3, n)$. For each dataset, we performed a cross-validation grid search, $k \in [0, 3]$ with a step of 0.1, to find an optimal value k^*. To allow a fair comparison, we adopted the most common experimental settings in literature. For the UT-Kinect dataset, we used the *leave-one-out cross-validation* (LOOCV) protocol [52], where one sequence is used for testing and the remaining sequences are used for training. For the Florence3D dataset, a *leave-one-subject-out* (LOSO) schema is adopted following [13, 51, 53]. All our programs were implemented in Matlab and run on a 2.8 GHZ CPU. We used the multi-class SVM implementation of the LibSVM library [12].

23.6.2.3 Results and Discussion

In Table 23.3, we compare our approach with existing methods dealing with skeletons and/or RGB-D data. Overall, our approach achieved competitive results compared to recent state-of-the-art approaches. On the UT-Kinect dataset, we obtained an average accuracy of 96.48%. On the Florence3D dataset, we obtained an average accuracy of 88.07%.

From the reported results on the two different datasets, we can observe the large superiority of the Gramian representation over the Grassmann representation. For the Florence3D, we report an improvement of about 12%. For UT-Kinect, the performance increased by about 3%. Note that these improvements over the Grassmannian representation are due to the additional information of the spatial covariance given by the SPD manifold in the metric. The contribution of the spatial covariance is weighted with a parameter k. As discussed in Sect. 23.6.2.2, we performed a grid search cross-validation to find the optimal value k^* of this parameter. The optimal values are $k^* = 0.05$ and $k^* = 0.25$ for the the UT-Kinect and Florence3D respectively. These results are in concordance with the recommendation of Bonnabel and Sepulchre [11] to use relative small values of k.

Table 23.3 Overall accuracy (%) on the UT-Kinect and Florence3D datasets

	UT-Kinect		Florence3D	
Method	Prot.	Acc (%)	Prot.	Acc (%)
$^{(G+D)}$ 3D^{2}CNN [35]*	LOSO	95.5	–	–
$^{(G)}$ LARP [48]	5-fold	97.08	5-fold	90.88
$^{(G)}$ Gram Hankel [53]	LOOCV	100	–	–
$^{(G)}$ Motion trajectories [13]	LOOCV	91.5	LOSO	87.04
$^{(G)}$ Elastic func. coding [3]	5-fold	94.87	5-fold	89.67
$^{(G)}$ Mining key poses [51]	LOOCV	93.47	LOSO	92.25
$^{(G)}$ NBNN+parts+time [41]	–	–	LOSO	82
$^{(G)}$ LSTM-trust gate [34]*	LOOCV	97.0	–	–
$^{(G)}$ JL-distance LSTM[54]*	5-fold	95.96	–	–
Traj. on $\mathcal{G}(3, n)$	LOOCV	92.46	LOSO	75 ± 5.22
Traj. on $\mathcal{S}^+(3, n)$	LOOCV	96.48	LOSO	88.07 ± 4.8

Here, (D): depth; (C): color (or RGB); (G): geometry (or skeleton); *: Deep Learning based approach; last row: ours

23.7 Affine-Invariant Shape Representation Using Barycentric Coordinates

The analysis of moving landmarks may be distorted by view variations. The problem is more acute when it comes to dealing with 2D landmarks. Indeed, in the 2D case these distortions are due to undesirable projective transformations which should be filtered out to have a robust representation of 2D landmarks to view variations. These projective transformations are difficult to be filtered out, but they can be approximated by affine transformations, especially when the face is far from the camera [43]. In this section we briefly review the main definitions of the affine-invariance with *barycentric coordinates* and their use in 2D facial shape analysis [27].

In order to study the motion of an ordered list of n landmarks $Z_1(t), Z_2(t), \ldots,$ $Z_n(t)$, where t represents the time parametrization and $Z_i(t) = (x_i(t), y_i(t))$, $1 \leq i \leq n$, in the plane up to the action of an arbitrary affine transformation, a standard technique is to consider the span of the columns of the $n \times 3$ time-dependent matrix

$$M(t) := \begin{pmatrix} x_1(t) & y_1(t) & 1 \\ \vdots & \vdots & \vdots \\ x_n(t) & y_n(t) & 1 \end{pmatrix}.$$

If at any time t there exists a fixed triplet of landmarks forming a non-degenerate triangle, the rank of the matrix $M(t)$ is constantly equal to 3 and the span of its columns is a curve of three-dimensional subspaces in $\mathbb{R}^n$. In other words, a curve in the Grassmannian $\mathcal{G}(3, n)$, which is well known [8] to be an affine-invariant of

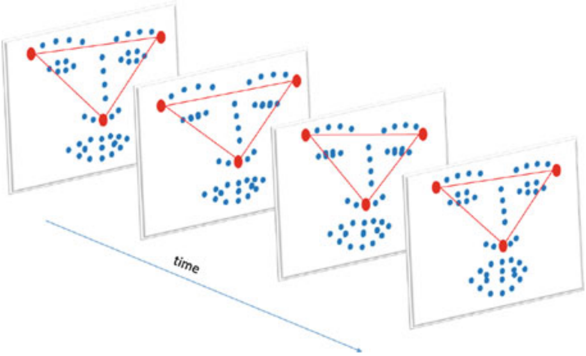


Fig. 23.2 Example of the automatically tracked 49 facial landmarks. The three red points denote the facial landmarks used to form the non-degenerate triangle required to compute the barycentric coordinates

the motion. This convenient way of filtering out the affine transformations opens the way to the use of metric and differential-geometric techniques in the study and classification of moving landmarks [3, 9, 13, 26, 27, 48].

It is worth noting that this representation in $\mathcal{G}(3, n)$ is equivalent to the Grassmann representation in $\mathcal{G}(2, n)$ which was studied and described in the previous work [26, 43]. The latter was obtained by centering the 2D landmarks and considering the span of the columns of the $n \times 2$ matrix as an affine-invariant representation in $\mathcal{G}(2, n)$ without adding a column of ones to the matrix formed by the 2D coordinates.

Another convenient and more classic way to filter out affine transformations is through the use of *barycentric coordinates*. This method can be applied given three of the landmarks which form a non-degenerate triangle throughout all their motion. Indeed, assume, without loss of generality, that $Z_1(t)$, $Z_2(t)$, and $Z_3(t)$ are the vertices of a non-degenerate triangle *for every value of* t. In the case of facial shapes, the right and left corners of the eyes and the tip of the nose are chosen to form a non-degenerate triangle (see the red triangle in Fig. 23.2). For $i = 4, .., n$ and at any time t, we can write

$$Z_i(t) = \lambda_{i1}(t)Z_1(t) + \lambda_{i2}(t)Z_2(t) + \lambda_{i3}(t)Z_3(t) \ ,$$

where the numbers $\lambda_{i1}(t)$, $\lambda_{i2}(t)$, and $\lambda_{i3}(t)$ satisfy

$$\lambda_{i1}(t) + \lambda_{i2}(t) + \lambda_{i3}(t) = 1.$$

The last condition renders the triplet of barycentric coordinates $(\lambda_{i1}(t), \lambda_{i2}(t), \lambda_{i3}(t))$ unique. In fact, it is equal to

$$(x_i(t), y_i(t), 1) \begin{pmatrix} x_1(t) & y_1(t) & 1 \\ x_2(t) & y_2(t) & 1 \\ x_3(t) & y_3(t) & 1 \end{pmatrix}^{-1} .$$

If T is an affine transformation of the plane, the barycentric representation of $TZ_i(t)$ in terms of the frame given by $TZ_1(t)$, $TZ_2(t)$, and $TZ_3(t)$ is still $(\lambda_{i1}(t), \lambda_{i2}(t), \lambda_{i3}(t))$. This allows us to derive the $(n-3) \times 3$ matrix

$$\Lambda(t) := \begin{pmatrix} \lambda_{41}(t) & \lambda_{42}(t) & \lambda_{43}(t) \\ \vdots & \vdots & \vdots \\ \lambda_{n1}(t) & \lambda_{n2}(t) & \lambda_{n3}(t) \end{pmatrix}.$$

as the affine shape representation of the moving landmarks.

23.7.1 Relationship with the Conventional Grassmannian Representation

A topological space $\mathcal{M}$ is a topological manifold of dimension dim if it is locally Euclidean. That means that every point $X \in \mathcal{M}$ has a neighborhood that is homeomorphic to an open subset of $\mathbb{R}^{dim}$. A coordinate chart (or just a chart on $\mathcal{M}$) is a pair (Σ, Φ), where Σ is an open subset of $\mathcal{M}$ and $\Phi : \Sigma \to \tilde{\Sigma}$ is homeomorphism from Σ to the open set $\tilde{\Sigma} \in \mathbb{R}^{dim}$. The definition of topological manifold implies that each point $X \in \mathcal{M}$ is contained in the domain of some coordinate chart [7]. In the case of the affine-invariant Grassmannian representation in $\mathcal{G}(3, n)$, the points on the Grassmannian corresponding to the facial landmarks are naturally contained in one of the standard charts. It turns out that passing to this chart is nothing more than taking the barycentric coordinates with respect to a specific triplet of landmark points.

In order to expose the basic relationship between the Grassmannian representation and the barycentric one, let us recall, in a particular case, the usual way to construct charts in the Grassmannian. If $\zeta \in \mathcal{G}(3, n)$ is a subspace that intersects the $(n-3)$-dimensional subspace

$$W = \{(0, 0, 0, x_4, \ldots, x_n) : x_i \in \mathbb{R}^n \text{ for } i \text{ between 4 and } n\}$$

only at the origin, and $\mathbf{x} = (x_1, \ldots, x_n)$, $\mathbf{y} = (y_1, \ldots, y_n)$, and $\mathbf{z} = (z_1, \ldots, z_n)$ is a basis for ζ, then the 3×3 matrix

$$\begin{pmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{pmatrix}$$

is invertible and the $(n-3) \times 3$ matrix

$$\begin{pmatrix} x_4 & y_4 & z_4 \\ \vdots & \vdots & \vdots \\ x_n & y_n & z_n \end{pmatrix} \begin{pmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{pmatrix}^{-1}$$

is independent of the chosen basis. In this way, the open and dense set of 3-dimensional subspaces transverse to W are put in a bijective correspondence with $\mathbb{R}^{(n-3)\times 3}$.

If we consider the curve in $\mathcal{G}(3, n)$ given by the span of the columns of the matrix

$$M(t) := \begin{pmatrix} x_1(t) & y_1(t) & 1 \\ \vdots & \vdots & \vdots \\ x_n(t) & y_n(t) & 1 \end{pmatrix}$$

and if the landmarks $Z_1(t) = (x_1(t), y_1(t))$, $Z_2(t) = (x_2(t), y_2(t))$, and $Z_3(t) = (x_3(t), y_3(t))$ form a non-degenerate triangle throughout all their motion, then composing this curve with a chart in the Grassmannian yields the curve of matrices

$$\begin{pmatrix} x_4(t) & y_4(t) & 1 \\ \vdots & \vdots & \vdots \\ x_n(t) & y_n(t) & 1 \end{pmatrix} \begin{pmatrix} x_1(t) & y_1(t) & 1 \\ x_2(t) & y_2(t) & 1 \\ x_3(t) & y_3(t) & 1 \end{pmatrix}^{-1} ,$$

which is just the curve $\Lambda(t)$ encoding the barycentric representation of the landmarks. For more details about the affine-invariance with barycentric coordinates, please refer to the page 81 of the book [10]. In what follows, we will consider the introduced affine-invariant vector Λ, with dimension $m = (n - 3) \times 3$, to represent a static facial shape and the curve $\Lambda(t)$ to denote a facial shape sequence.

23.8 Metric Learning on Barycentric Representation for Expression Recognition in Unconstrained Environments

Given the facial shape represented by the affine-invariant vector Λ, with dimension $m = (n - 3) \times 3$, we seek a suitable metric that is discriminative enough in terms of expression to compare them. The Euclidean distance, defined as the squared l_2-norm of the difference of the vectors, could be a reasonable choice since the defined shapes lie in Euclidean space. However, such distance disregards the specific nature of the considered facial shapes. To overcome this issue, we propose to learn a Mahalanobis distance instead of using the standard Euclidean distance [31]. Given two facial shapes represented by the affine-invariant vectors Λ_i and Λ_j in $\mathbb{R}^m$, the Mahalanobis distance is defined by

$$d^2_{l_{ij}}(\Lambda_i, \Lambda_j) = (\Lambda_i - \Lambda_j)^T A(\Lambda_i - \Lambda_j) , \qquad (23.16)$$

where A is a positive semi-definite (p.s.d) matrix of size $m \times m$. The problem of metric learning is then to find the best p.s.d matrix A that best discriminates the facial expressions, i.e., results in small distances when the facial shapes represent similar expressions and large distances when they represent different expressions.

Let $\mathcal{D} = \{(\Lambda_1, c_1), \ldots, (\Lambda_N, c_N)\}$ represent a set of affine-invariant shapes in $\mathbb{R}^m$ annotated with the corresponding expressions (e.g., $c =$'happy', 'angry', etc.). Let $\{\Lambda_i, \Lambda_j, \Lambda_k\}$ be a triplet of affine-invariant shapes from $\mathcal{D}$ such that (Λ_i, Λ_j) have the same label $(c_i = c_j)$, and (Λ_i, Λ_k) with different labels $(c_i \neq c_k)$. We aim to find an optimal p.s.d matrix A such that $d^2_{l_{ij}}(\Lambda_i, \Lambda_j) < d^2_{l_{ik}}(\Lambda_i, \Lambda_k)$. That is, we wish to find a p.s.d matrix A that minimizes $d^2_{l_{ij}} - d^2_{l_{ik}} = (\Lambda_i - \Lambda_j)^T A(\Lambda_i - \Lambda_j) - (\Lambda_i - \Lambda_k)^T A(\Lambda_i - \Lambda_k)$. In order to solve this optimization problem, we follow the convenient method described by Shen et al. [42], where a boosting is used. This method is based on the observation that any positive semidefinite matrix can be decomposed into a linear combination of trace-one rank-one matrices. It uses rank-one positive semidefinite matrices as weak learners within an efficient and scalable boosting-based learning process.

23.8.1 Experimental Results

In order to learn the metric, we use only peak frames from each facial sequence, where the expression reaches its peak. Since peak frames are difficult to detect in uncontrolled facial expressions, we performed the metric learning using extracted landmarks from CK+ dataset [36] which is captured in strict controlled conditions. In this dataset, 309 facial sequences of 118 subjects are annotated with the six labels (the six basic emotions). In all the sequences, the actors start by being neutral then perform the expression until reaching a peak. In our experiments, we only used the five last frames and the first frame from all the sequences. The labels of the five last frames are assigned according to the label of the sequence, while the label of the first frame is always considered as 'neutral'. A total number of 16,686 facial shapes are used for the training phase to learn the Mahalanobis distance.

To evaluate the proposed approach, we conducted experiments on the well-known AFEW dataset [15] collected from movies showing close-to-real-world conditions, which depict or simulate the spontaneous expressions in uncontrolled environment. The task is to classify each video clip into one of the seven expression categories (the six basic emotions plus the neutral). Note that our experiments are made once the facial landmarks are extracted using the method proposed in [5]. The three points used to form the non-degenerate triangle, essential to build the affine-invariant shapes from the landmarks, are the points positioned at the left and right corners of the eye and the nose tip.

All our programs were implemented in Matlab and run on a 2.8 GHZ CPU. We used the multi-class SVM implementation of the LibSVM library [12], and the codes given by [42] for the metric learning.

23.8.1.1 Results and Discussions

Following the experimental settings mentioned in the previous Section, we report an accuracy of 38.38%. Our results are outperformed by the Gram trajectory representation proposed in the previous chapter [26]. However, the execution time of comparing two arbitrary sequences on AFEW dataset is 0.064 s with the barycentric approach against 0.84 s with the Gram approach. In Table 23.4, we can observe that our results compared to the Gram approach are outperformed by only 1% while being 10 times faster.

To evaluate the different steps of the proposed pipeline, we performed baseline experiments. Firstly, we conducted the same experiments while using alternative representations and metrics. We compared our results with a conventional Grassmann affine-invariant representation coupled with a Riemannian metric given by the subspace angles [8, 43]. The achieved accuracy is around 2.5% lower than ours. We also replaced the learned Mahalanobis distance with a standard Euclidean distance. Here also, the performance decreases by about 3%. In Table 23.5, we show the achieved accuracies by the described alternative representations and metrics and the necessary execution time to compare two arbitrary facial shapes. One can observe that the proposed representation achieves better performance than the Grassmannian while being less time consuming. These results show the effectiveness of the proposed representation and the importance of the metric learning step in our pipeline. As mentioned in the previous Section, we used the five last (peak) frames from the sequences of CK+ dataset to learn the Mahalanobis distance. In Table 23.5, we provide the obtained accuracies when using one, two, five and seven last peak frames from each sequence. The highest accuracy is obtained with the last five frames.

Table 23.4 Overall accuracy AFEW dataset (FER with Barycentric representation)

Method	Accuracy (%)
$^{(A)}$ HOG 3D [29]	26.90
$^{(A)}$ HOE [49]	19.54
$^{(A)}$ 3D SIFT [40]	24.87
$^{(A)}$ LBP-TOP [55]	25.13
$^{(A)}$ EmotiW [16]	27.27
$^{(A)}$ STM [33]	29.19
$^{(A)}$ STM-ExpLet [33]	31.73
$^{(A)}$ SPDNet [22]	34.23
$^{(G)}$ Gram Trajectories [26]	39.94
$^{(G)}$ Ours	38.38

Table 23.5 Baseline experiments (FER with barycentric representation)

Distance	Accuracy (%)	Time (μs)
Subspace angles in $\mathcal{G}(3, n)$	36.81	2967
Euclidean distance	36.55	530
Mahalanobis distance d_l	38.38	568

Number of peak frames	Accuracy (%)
1 peak frame	37.07
2 peak frames	37.59
5 peak frames	38.38
7 peak frames	36.29

23.9 Conclusion

In this chapter, we proposed geometric tools for facial expression and action recognition based on the analysis of landmark sequences. Firstly, we proposed a novel geometric framework on Gram matrix trajectories. To overcome the non-linear nature of the space of Gram matrices, its Riemannian geometry was studied to derive suitable analyzing tools for the Gram matrix trajectories. Applications were shown to facial expression recognition from 2D landmarks tracked on the human face in RGB videos and 3D action recognition from 3D skeletons detected on the human body in depth streams. Secondly, we proposed an affine-invariant representation for the specific case of 2D facial landmarks based on their barycentric coordinates. While being related to the Gram matrix representation, the barycentric representation has the advantage of lying in Euclidean space where standard computational and machine learning tools are applicable. The barycentric representation was evaluated in facial expression recognition by applying a standard metric learning algorithm.

References

1. Abdelkader, M.F., Abd-Almageed, W., Srivastava, A., Chellappa, R.: Silhouette-based gesture and action recognition via modeling trajectories on riemannian shape manifolds. Comput. Vis. Image Underst. **115**(3), 439–455 (2011)
2. Absil, P.A., Mahony, R., Sepulchre, R.: Riemannian geometry of grassmann manifolds with a view on algorithmic computation. Acta Appl. Math. **80**(2), 199–220 (2004)
3. Anirudh, R., Turaga, P., Su, J., Srivastava, A.: Elastic functional coding of riemannian trajectories. IEEE Trans. Pattern Anal. Mach. Intell. **39**(5), 922–936 (2017)
4. Arsigny, V., Fillard, P., Pennec, X., Ayache, N.: Log-euclidean metrics for fast and simple calculus on diffusion tensors. Magn. Reson. Med. **56**(2), 411–421 (2006)
5. Asthana, A., Zafeiriou, S., Cheng, S., Pantic, M.: Incremental face alignment in the wild. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1859–1866 (2014)
6. Bagheri, M.A., Gao, Q., Escalera, S.: Support vector machines with time series distance kernels for action classification. In: IEEE Winter Conf. on Applications of Computer Vision (WACV), pp. 1–7 (2016)

7. Baralić, D.: How to understand grassmannians? Teach. Math. **XIV**(2), 147–157 (2011)
8. Begelfor, E., Werman, M.: Affine invariance revisited. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2087–2094 (2006)
9. Ben Amor, B., Su, J., Srivastava, A.: Action recognition using rate-invariant analysis of skeletal shape trajectories. IEEE Trans. Pattern Anal. Mach. Intell. **38**(1), 1–13 (2016)
10. Berger, M.: Geometry, vol. i-ii (1987)
11. Bonnabel, S., Sepulchre, R.: Riemannian metric and geometric mean for positive semidefinite matrices of fixed rank. SIAM J. Matrix Anal. Appl. **31**(3), 1055–1070 (2009)
12. Chang, C.C., Lin, C.J.: Libsvm: a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**(3), 27 (2011)
13. Devanne, M., Wannous, H., Berretti, S., Pala, P., Daoudi, M., Del Bimbo, A.: 3-D human action recognition by shape analysis of motion trajectories on Riemannian manifold. IEEE Trans. Cyber. **45**(7), 1340–1352 (2015)
14. Deza, M.M., Laurent, M.: Geometry of Cuts and Metrics, vol. 15. Springer, Berlin (2009)
15. Dhall, A., Goecke, R., Lucey, S., Gedeon, T.: Collecting large, richly annotated facial-expression databases from movies. IEEE MultiMedia **19**(3), 34–41 (2012)
16. Dhall, A., Goecke, R., Joshi, J., Wagner, M., Gedeon, T.: Emotion recognition in the wild challenge (EmotiW) challenge and workshop summary. In: International Conference on Multimodal Interaction (ICMI), pp. 371–372 (2013)
17. Elaiwat, S., Bennamoun, M., Boussaïd, F.: A spatio-temporal rbm-based model for facial expression recognition. Pattern Recogn. **49**, 152–161 (2016)
18. Faraki, M., Harandi, M.T., Porikli, F.: Image set classification by symmetric positive semi-definite matrices. In: IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1–8 (2016)
19. Graepel, T., Herbrich, R., Bollmann-Sdorra, P., Obermayer, K.: Classification on pairwise proximity data. In: Advances in Neural Information Processing Systems, pp. 438–444 (1999)
20. Gudmundsson, S., Runarsson, T.P., Sigurdsson, S.: Support vector machines and dynamic time warping for time series. In: IEEE World Congress on Computational Intelligence, pp. 2772–2776 (2008)
21. Higham, N.J.: Computing the polar decomposition with applications. SIAM J. Sci. Stat. Comput. **7**(4), 1160–1174 (1986)
22. Huang, Z., Van Gool, L.J.: A Riemannian network for spd matrix learning. In: AAAI, vol. 2, p. 6 (2017)
23. Jain, S., Hu, C., Aggarwal, J.K.: Facial expression recognition with temporal modeling of shapes. In: IEEE International Conference on Computer Vision Workshops (ICCV), pp. 1642–1649 (2011)
24. Jayasumana, S., Hartley, R.I., Salzmann, M., Li, H., Harandi, M.T.: Kernel methods on riemannian manifolds with gaussian RBF kernels. IEEE Trans. Pattern Anal. Mach. Intell. **37**(12), 2464–2477 (2015)
25. Jung, H., Lee, S., Yim, J., Park, S., Kim, J.: Joint fine-tuning in deep neural networks for facial expression recognition. In: IEEE International Conference on Computer Vision, ICCV, pp. 2983–2991 (2015)
26. Kacem, A., Daoudi, M., Ben Amor, B., Alvarez-Paiva, J.C.: A novel space-time representation on the positive semidefinite cone for facial expression recognition. In: IEEE International Conference on Computer Vision (ICCV) (2017)
27. Kacem, A., Daoudi, M., Alvarez-Paiva, J.C.: Barycentric Representation and Metric Learning for Facial Expression Recognition. In: IEEE International Conference on Automatic Face and Gesture Recognition . IEEE, Xi'an (2018)
28. Kacem, A., Daoudi, M., Ben Amor, B., Berretti, S., Alvarez-Paiva, J.C.: A novel geometric framework on gram matrix trajectories for human behavior understanding. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–1 (2019)
29. Kläser, A., Marszalek, M., Schmid, C.: A spatio-temporal descriptor based on 3d-gradients. In: British Machine Vision Conference (BMVC), pp. 1–10 (2008)

30. Kobayashi, S., Nomizu, K.: Foundations of Differential Geometry, vol. 1. Interscience Publishers, New York (1963)
31. Kulis, B.: Metric learning: a survey. Found. Trends Mach. Learn. **5**(4), 287–364 (2013)
32. Liu, M., Li, S., Shan, S., Wang, R., Chen, X.: Deeply learning deformable facial action parts model for dynamic expression analysis. In: Asian Conference on Computer Vision, pp. 143–157 (2014)
33. Liu, M., Shan, S., Wang, R., Chen, X.: Learning expressionlets on spatio-temporal manifold for dynamic facial expression recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1749–1756 (2014)
34. Liu, J., Shahroudy, A., Xu, D., Wang, G.: Spatio-Temporal LSTM with Trust Gates for 3D Human Action Recognition, pp. 816–833. Springer, Cham (2016)
35. Liu, Z., Zhang, C., Tian, Y.: 3d-based deep convolutional neural network for action recognition with depth sequences. Image Vis. Comput. **55**, 93–100 (2016)
36. Lucey, P., Cohn, J.F., Kanade, T., Saragih, J.M., Ambadar, Z., Matthews, I.A.: The extended Cohn-Kanade dataset (CK+): a complete dataset for action unit and emotion-specified expression. In: IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPRW), pp. 94–101 (2010)
37. Meyer, G., Bonnabel, S., Sepulchre, R.: Regression on fixed-rank positive semidefinite matrices: a riemannian approach. J. Mach. Learn. Res. **12**(Feb), 593–625 (2011)
38. Pennec, X., Fillard, P., Ayache, N.: A riemannian framework for tensor computing. Int. J. Comput. Vis. **66**(1), 41–66 (2006)
39. Sariyanidi, E., Gunes, H., Cavallaro, A.: Learning bases of activity for facial expression recognition. IEEE Trans. Image Process. **PP**(99), 1–1 (2017)
40. Scovanner, P., Ali, S., Shah, M.: A 3-dimensional sift descriptor and its application to action recognition. In: International Conference on Multimedia, pp. 357–360 (2007)
41. Seidenari, L., Varano, V., Berretti, S., Bimbo, A., Pala, P.: Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 479–485 (2013)
42. Shen, C., Kim, J., Wang, L., Hengel, A.: Positive semidefinite metric learning with boosting. In: Advances in neural information processing systems, pp. 1651–1659 (2009)
43. Taheri, S., Turaga, P., Chellappa, R.: Towards view-invariant expression analysis using analytic shape manifolds. In: IEEE International Conference on Automatic Face and Gesture Recognition and Workshops (FG), pp. 306–313 (2011)
44. Tuzel, O., Porikli, F., Meer, P.: Pedestrian detection via classification on Riemannian manifolds. IEEE Trans. Pattern Anal. Mach. Intell. **30**(10), 1713–1727 (2008)
45. Valstar, M.F., Pantic, M.: Induced disgust, happiness and surprise: an addition to the mmi facial expression database. In: International Conference on Language Resources and Evaluation, Workshop on EMOTION, pp. 65–70. IEEE, Malta (2010)
46. Vandereycken, B., Absil, P.A., Vandewalle, S.: Embedded geometry of the set of symmetric positive semidefinite matrices of fixed rank. In: IEEE/SP Workshop on Statistical Signal Processing (SSP), pp. 389–392 (2009)
47. Veeraraghavan, A., Chellappa, R., Roy-Chowdhury, A.: The function space of an activity. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. 959–968 (2006)
48. Vemulapalli, R., Arrate, F., Chellappa, R.: Human action recognition by representing 3D skeletons as points in a Lie group. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 588–595 (2014)
49. Wang, L., Qiao, Y., Tang, X.: Motionlets: Mid-level 3d parts for human motion recognition. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013, pp. 2674–2681 (2013)
50. Wang, Z., Wang, S., Ji, Q.: Capturing complex spatio-temporal relations among facial muscles for facial expression recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3422–3429 (2013)

51. Wang, C., Wang, Y., Yuille, A.L.: Mining 3d key-pose-motifs for action recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2639–2647 (2016)
52. Xia, L., Chen, C.C., Aggarwal, J.K.: View invariant human action recognition using histograms of 3d joints. In: 2012 IEEE computer society conference on Computer vision and pattern recognition workshops (CVPRW), pp. 20–27. IEEE, Piscataway (2012)
53. Zhang, X., Wang, Y., Gou, M., Sznaier, M., Camps, O.: Efficient temporal sequence comparison and classification using Gram matrix embeddings on a riemannian manifold. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
54. Zhang, S., Liu, X., Xiao, J.: On geometric features for skeleton-based action recognition using multilayer lstm networks. In: IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 148–157 (2017)
55. Zhao, G., Pietikäinen, M.: Dynamic texture recognition using local binary patterns with an application to facial expressions. IEEE Trans. Pattern Anal. Mach. Intell. **29**(6), 915–928 (2007)
56. Zhong, L., Liu, Q., Yang, P., Liu, B., Huang, J., Metaxas, D.N.: Learning active facial patches for expression analysis. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2562–2569 (2012)

Chapter 24
Biomedical Applications of Geometric Functional Data Analysis

James Matuk, Shariq Mohammed, Sebastian Kurtek, and Karthik Bharath

Contents

Abstract In this chapter, we describe several biomedical applications of geometric functional data analysis methods for modeling probability density functions, amplitude and phase components in functional data, and shapes of curves and surfaces. We begin by reviewing parameterization-invariant Riemannian metrics and corresponding simplifying square-root transforms for each case. These tools

J. Matuk · S. Kurtek (✉)
Department of Statistics, The Ohio State University, Columbus, OH, USA
e-mail: matuk.3@osu.edu; kurtek.1@stat.osu.edu

S. Mohammed
Department of Computational Medicine and Bioinformatics, University of Michigan, Ann Arbor, MI, USA
e-mail: shariqm@umich.edu

K. Bharath
School of Mathematical Sciences, University of Nottingham, Nottingham, UK
e-mail: Karthik.Bharath@nottingham.ac.uk

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_24

allow for computationally efficient implementations of statistical procedures on the appropriate representation spaces, including computation of the Karcher mean and exploration of variability via principal component analysis. We then showcase applications of these tools in multiple biomedical case studies based on various datasets including Glioblastoma Multiforme tumors, Diffusion Tensor Magnetic Resonance Image-based white matter tracts and fractional anisotropy functions, electrocardiogram signals, endometrial tissue surfaces and subcortical surfaces in the brain.

24.1 Introduction

Improvements in medical data acquisition technology, especially non-invasive imaging technology, have resulted in proliferation of large, complex datasets. There are many goals in analyzing such data depending on the application of interest, ranging from assessment of regular aging patterns to diagnosis and monitoring of various diseases. The types of imaging data of interest greatly vary in their properties, e.g., functional Magnetic Resonance Imaging (fMRI) measures dynamic brain activity through changes in blood flow, structural Magnetic Resonance Imaging (MRI) produces images of the anatomy using magnetic fields and Diffusion Tensor Magnetic Resonance Imaging (DT-MRI) maps diffusion of water molecules in biological tissues. In spite of this apparent heterogeneity, many medical imaging datasets share two common characteristics: (1) the representation space of the data is fundamentally non-Euclidean and (2) the data is functional (infinite-dimensional) in nature. These two properties of the data introduce a major challenge for statistical analysis as most traditional statistical methods apply to data residing in relatively low-dimensional Euclidean spaces. Our focus in this book chapter is on representation and statistical analysis of various aspects of biomedical imaging data including (1) patterns of voxel values via probability density functions (pdfs, smoothed histograms of voxel intensities) [44], (2) elastic functional data that contains amplitude and phase variabilities [48], (3) shapes of curves [30, 47] and (4) shapes of surfaces representing objects in medical images [19, 35]. As will be seen later, all of these data types benefit from a Riemannian geometric approach to data analysis. To unify these different data objects of interest, we refer to them commonly as geometric data objects throughout.

Statistical analysis of geometric data objects starts with the definition of a suitable mathematical representation and metric that can be used for their comparison. Once an appropriate representation space and a Riemannian metric on that space have been defined, statistical analysis proceeds via the metric structure. In particular, this approach allows one to (1) compute summary statistics such as the mean and covariance, (2) explore variability in a sample via adaptations of principal component analysis and (3) define basic statistical models [20, 49]. We consider each of pdfs, elastic functional data, and shapes of curves and surfaces separately to define the relevant Riemannian geometric representation spaces. To tie all of the frameworks together, we point out the commonalities between the Riemannian geometry used for statistical analysis in each case.

We begin with statistical analysis of texture via a pdf representation. Texture here refers to the pattern of voxel values inside an object of interest in a medical image; it is a fundamental appearance property of objects in images [49]. We form the pdf by (1) vectorizing the relevant voxel values, (2) generating their histogram and (3) smoothing the histogram [44]. The result is a functional data object with two constraints: the pdf must be positive everywhere on its domain and it must integrate to one. The representation space of pdfs is the infinite-dimensional simplex, a constrained linear space. To define a Riemannian structure on this space, we use the well-known Fisher-Rao metric [25, 42, 46]. An important property of this metric is that it is invariant to reparameterization [7], a property used later for defining a Riemannian structure on the space of elastic functions and shapes.

The second type of geometric data objects of interest are elastic functions. Elastic functions naturally contain two different sources of variability: amplitude variability and phase, warping or parameterization variability [38]. A main goal in elastic functional data analysis is to separate these two sources of variability and define statistical methods to analyze them. The Riemannian setting for this type of analysis necessitates invariance to function reparameterization. Conveniently, we apply an extension of the Fisher-Rao metric used for pdfs in this setting [48].

Finally, we use methods from elastic shape analysis to study outlines (boundaries of objects resulting in curves and surfaces) representing objects in medical images [20, 30, 47]. The shape of such boundaries is a fundamental physical property of the objects, and provides indispensable information about the health and development of anatomical structures in the medical setting. The notion of shape is invariant to translations, scales, rotations and reparameterizations of the curves and surfaces [26]. In this setting, we use elastic Riemannian metrics which have been shown to have such desired invariances. These elastic metrics are also extensions of the Fisher-Rao metric introduced for pdfs.

In all of the above-mentioned settings, the initial Riemannian geometric structure of the representation spaces is quite complicated and necessitates numerical methods for simple tasks such as computing geodesic distances. Luckily, there exist square-root transforms in each of the cases that greatly simplify the geometry, and result in Riemannian geometric tools with analytical expressions. This, in turn, allows for development of large-scale data analytic approaches that can be applied in various biomedical settings.

Our focus in this book chapter is not on describing recent methodological advances in this area, but rather on elucidating various biomedical applications of geometric methods for functional data analysis. While we outline the relevant mathematical details to keep our discussion self-contained, the main aim is to showcase the breadth of applicability of the methods in medical imaging. As a result, our methodological descriptions are terse and avoid many technical details; we refer the interested readers to the recent books [49] and [20] for specific details. Additionally, we highlight two closely related chapters in this volume that present complementary material. In Chapter 13, the authors focus on the problem of registering different types of functional data as well as related mathematical/statistical properties; they also present many intuitive examples to introduce this topic. In Chapter 14, the authors provide an extension of the methods described in our chapter to trajectories on general manifolds and present examples that consider multimodal data. The rest of this chapter is organized as follows. Section 24.2 describes the Riemannian geometry of representation spaces for the four geometric data objects of interest: (1) pdfs, (2) elastic functional data, (3) shapes of curves and (4) shapes of surfaces. In Sect. 24.3, we describe a general nonparametric framework, based on tools provided by the Riemannian geometric backdrops, for computing summary statistics and assessing variability in random samples. Section 24.4 discusses multiple case studies for each type of geometric data object. Here, we draw on previous studies to showcase the breadth of biomedical applications of the described methods. Finally, we close with a brief summary in Sect. 24.5.

24.2 Mathematical Representation: Riemannian Metrics and Simplifying Transforms

We begin with a brief review of the different Riemannian metrics and representations for pdfs, amplitude and phase components of elastic functional data, shapes of curves and shapes of surfaces. In each case, we highlight a particular square-root transformation, which greatly simplifies the computational implementation of the framework. For more details on these approaches, please refer to Chapters 4 (pdfs and elastic functional data), 5 and 6 (shapes of open and closed curves, respectively) in [49], and [20] (shapes of surfaces). Throughout, we use $\|\cdot\|$ and $\langle\langle\cdot,\cdot\rangle\rangle$ to denote functional norms and inner products (not necessarily $\mathbb{L}^2$), and $|\cdot|$ and $\langle\cdot,\cdot\rangle$ to denote the norm and inner product in a finite-dimensional Euclidean space $\mathbb{R}^k$.

24.2.1 *Probability Density Functions*

Without loss of generality, our description focuses on univariate densities on [0, 1]. However, the methods described here can be generalized to the multivariate setting in a straightforward manner (see Section 4 in [44] for an example). Let $\mathcal{P}$ denote the Banach manifold of such pdfs defined as $\mathcal{P} = \{p : [0,1] \to \mathbb{R}_+ | \int_0^1 p(t)dt = 1\}$. For any point $p \in \mathcal{P}$, the tangent space is defined as $T_p(\mathcal{P}) = \{v : [0,1] \to \mathbb{R} | \int_0^1 v(t)dt = 0\}$; this is a vector space of all possible perturbations of the pdf p. We proceed to define a Riemannian metric on $\mathcal{P}$, which will be used to compute geodesic distances between two pdfs and summary statistics of samples of pdfs. The nonparametric Fisher-Rao Riemannian metric (simply FR metric hereafter), for any two tangent vectors $v_1, v_2 \in T_p(\mathcal{P})$ is defined as [25, 42, 46]

$$\langle\langle v_1, v_2\rangle\rangle_p = \int_0^1 v_1(t)v_2(t)\frac{1}{p(t)}dt. \tag{24.1}$$

The FR metric is invariant to reparameterizations of densities [7], a nice mathematical property. One drawback of this metric is the difficulty associated with computing geodesic paths and distances due to the fact that the metric changes from point to point on the space of pdfs, requiring numerical procedures.

To simplify computation, we choose an equivalent representation of the space $\mathcal{P}$ via the square-root density (SRD) representation [4]. Under this representation, the complicated FR metric becomes the standard $\mathbb{L}^2$ metric and the space of pdfs $\mathcal{P}$ becomes the positive orthant of the unit hypersphere in $\mathbb{L}^2$. In other words, we define an isometric transformation that greatly simplifies computing. The SRD is defined as a function $\psi = +\sqrt{p}$ (we omit the $+$ sign hereafter for notational convenience). Then, the inverse mapping is unique and is simply given by $p = \psi^2$. Hence, the space of all SRDs is given by $\Psi = \{\psi : [0, 1] \to \mathbb{R}_+ | \int_0^1 \psi(t)^2 dt = 1\}$. The $\mathbb{L}^2$ Riemannian metric on Ψ is defined as $\langle\langle w_1, w_2\rangle\rangle = \int_0^1 w_1(t)w_2(t)dt$, where $w_1,\ w_2 \in T_\psi(\Psi)$ and $T_\psi(\Psi) = \{w : [0, 1] \to \mathbb{R} | \int_0^1 \psi(t)w(t)dt = 0\}$.

As the Riemannian geometry of Ψ equipped with the $\mathbb{L}^2$ metric is well-known, geodesic paths and their lengths can now be computed analytically. The geodesic distance between $\psi_1,\ \psi_2 \in \Psi$ is simply given by

$$d(\psi_1, \psi_2) = \theta = \cos^{-1}\Big(\int_0^1 \psi_1(t)\psi_2(t)dt\Big). \tag{24.2}$$

The corresponding geodesic path between $\psi_1,\ \psi_2 \in \Psi$ is

$$\eta(\tau) = 1/\sin(\theta)\{\psi_1 \sin(\theta(1-\tau)) + \psi_2 \sin(\tau\theta)\}, \quad \tau \in [0, 1]. \tag{24.3}$$

It is easy to see that the geodesic distance θ is bounded above by $\pi/2$. In addition to geodesic paths and distances, we often use the exponential and inverse exponential maps for computing statistical summaries of a sample of pdfs. The exponential map at a point $\psi_1 \in \Psi$, denoted by $\exp : T_{\psi_1}(\Psi) \mapsto \Psi$, is defined as

$$\exp_{\psi_1}(w) = \cos(\|w\|)\psi_1 + \sin(\|w\|)(w/\|w\|), \tag{24.4}$$

where $\|w\| = \Big(\int_0^1 w(t)^2 dt\Big)^{1/2}$. The inverse exponential map, denoted by $\exp_{\psi_1}^{-1} : \Psi \mapsto T_{\psi_1}(\Psi)$, is given by

$$\exp_{\psi_1}^{-1}(\psi_2) = (\theta/\sin(\theta))(\psi_2 - \psi_1\cos(\theta)). \tag{24.5}$$

These two mappings can be used to transfer points from the nonlinear representation space Ψ to linear tangent spaces of Ψ, and vice versa.

24.2.2 Amplitude and Phase in Elastic Functional Data

One can extend the above FR metric-based framework to more general functional data. One difficulty that arises in this setting is the need for registration when comparing or modeling such observations. This is due to the fact that functional data often contains two forms of variability: amplitude and phase [38, 41, 48, 49]. Amplitude describes the vertical variability along the y-axis while phase describes the horizontal variability along the x-axis (also called domain warping), i.e., the parameterization of the functional observations. Thus, extracting phase variability from functional data through a registration procedure requires a metric that is invariant to reparameterization. As we have already established that the FR metric is invariant to reparameterizations of pdfs, we will use its extension for functional data.

We introduce some additional notation to formalize the discussion. Without loss of generality, we restrict our attention to absolutely continuous functions on the domain [0, 1], and focus only on nonlinear warpings of this domain; thus, we define the function space of interest as $\mathcal{F} = \{f : [0,1] \to \mathbb{R} | f \text{ is absolutely continuous}\}$. We use the set $\Gamma = \{\gamma : [0,1] \to [0,1] | \gamma(0) = 0,\ \gamma(1) = 1,\ \gamma \text{ is a diffeomorphism}\}$ to represent all possible nonlinear domain warpings. Then, for a function $f \in \mathcal{F}$, the composition $f \circ \gamma$ denotes the domain warping of f using γ, i.e., a reparameterization of the function f. To extend the FR metric for pdfs to this more general class of functions, we start with absolutely continuous functions $f : [0,1] \to \mathbb{R}$ such that $\dot{f} > 0$; call the set of such functions $\mathcal{F}_0$ and let $T_f(\mathcal{F}_0)$ denote the tangent space to $\mathcal{F}_0$ at f. For any $f \in \mathcal{F}_0$ and $v_1, v_2 \in T_f(\mathcal{F}_0)$, the FR metric can be redefined as [48]

$$\langle\langle v_1, v_2 \rangle\rangle_f = \int_0^1 \dot{v}_1(t)\dot{v}_2(t)\frac{1}{\dot{f}(t)}dt. \tag{24.6}$$

As in the case of densities, this metric is invariant to domain warpings, $\langle\langle v_1 \circ \gamma, v_2 \circ \gamma \rangle\rangle_{f\circ\gamma} = \langle\langle v_1, v_2 \rangle\rangle_f$, for all $\gamma \in \Gamma$, $f \in \mathcal{F}_0$ and $v_1, v_2 \in T_f(\mathcal{F}_0)$, but also difficult to work with computationally.

To alleviate this issue, we define a square-root transform similar to the SRD. Define the square-root slope function (SRSF) of f as $q = \text{sign}(\dot{f})\sqrt{|\dot{f}(t)|}$. Since we have assumed $\dot{f} > 0$, the SRSF in this case simply becomes $q = \sqrt{\dot{f}}$, i.e., the square-root of an unnormalized pdf. Importantly, under the SRSF representation, the FR metric becomes the standard $\mathbb{L}^2$ metric. While we have so far restricted our attention to functions with positive derivative, the SRSF allows us to treat more general cases. Next, we return to the space $\mathcal{F}$ of all absolutely continuous functions, i.e., $\dot{f}$ is allowed to take arbitrary values including zero (when $\dot{f} = 0$, the SRSF also takes value 0). Then, using the $\mathbb{L}^2$ metric on the space of all SRSFs corresponding to functions in $\mathcal{F}$, the FR metric implicitly extends from $\mathcal{F}_0$ to $\mathcal{F}$. If the function f is absolutely continuous then the resulting SRSF is square-integrable or an element of $\mathbb{L}^2([0,1], \mathbb{R})$ (simply $\mathbb{L}^2$ for brevity) [43]. The inverse mapping from an SRSF to its corresponding function is unique up to a vertical translation. If one additionally keeps track of the starting point $f(0)$, then the mapping is unique and is given by $f(t) = f(0) + \int_0^t q(s)|q(s)|ds$. Furthermore, the SRSF of a warped function $f \circ \gamma$ is given by $(q, \gamma) = (q \circ \gamma)\sqrt{\dot{\gamma}}$.

This basic setup allows us to define amplitude and phase mathematically. The amplitude of a function remains unchanged under warping, i.e., f and $f \circ \gamma$ have the same amplitude for any $\gamma \in \Gamma$. The amplitude is thus defined as the equivalence class $[f] = \{f \circ \gamma | \gamma \in \Gamma\}$, which contains all possible domain warpings of f. The space of all amplitudes is the quotient space $\mathcal{F}/\Gamma$. In contrast to amplitude, the definition of phase is only relative. Given two functions f_1 and f_2, the relative phase of f_2 with respect to f_1 is defined as

$$\gamma_{21} = \arg\min_{\gamma \in \Gamma} \|q_1 - (q_2 \circ \gamma)\sqrt{\dot{\gamma}}\|, \tag{24.7}$$

where q_1 and q_2 are the SRSFs of f_1 and f_2, respectively. This minimization is usually solved using the dynamic programming algorithm [43]. The optimization problem in Eq. (24.7) is referred to as the pairwise registration of f_2 to f_1.

Next, we focus on defining a distance for amplitude and phase components. The distance between amplitudes of two functions f_1 and f_2 is defined as

$$d_a(f_1, f_2) = d([q_1], [q_2]) = \min_{\gamma \in \Gamma} \|q_1 - (q_2 \circ \gamma)\sqrt{\dot{\gamma}}\| = \|q_1 - (q_2 \circ \gamma_{21})\sqrt{\dot{\gamma}_{21}}\|. \tag{24.8}$$

A geodesic path between two amplitude functions can then be constructed using a straight line connecting q_1 and $(q_2 \circ \gamma_{21})\sqrt{\dot{\gamma}_{21}}$. Similarly, in order to compare the phase components of the two functions f_1 and f_2, we use the relative phase between them, γ_{21}. Then, the phase distance is defined as

$$d_p(f_1, f_2) = \cos^{-1}\left(\int_0^1 \sqrt{\dot{\gamma}_{21}(t)}dt\right). \tag{24.9}$$

This definition is based on an adaptation of the FR metric to Γ, and is measured using the SRSFs of warping functions [27]. In fact, the SRSF of any warping function is simply an SRD. Thus, the phase distance uses the SRD representation introduced earlier to compute distances between warping functions. To construct a geodesic path between two warping functions, after transforming them to their SRSFs, one can simply use Eq. (24.3).

24.2.3 Shapes of Open and Closed Curves

The extension of methods for functional data analysis to curves in higher-dimensional Euclidean spaces comes from so-called elastic shape analysis. While functional data requires invariance to reparameterization only, shape analysis additionally requires invariance to translation, scale and rotation, also referred to as similarity shape-preserving transformations. As in the two previous sections, we begin by introducing a Riemannian metric, which is naturally invariant to all such transformations.

Let $f : \mathcal{D} \to \mathbb{R}^k,\ k > 1$ denote an absolutely continuous, parameterized curve in the Euclidean space $\mathbb{R}^k$ with the domain of parameterization given by $\mathcal{D} = [0, 1]$ for open curves and $\mathcal{D} = \mathbb{S}^1$ for closed curves. With a slight abuse of notation, let $\mathcal{F}$ denote the set of all such curves. While the framework described here applies to k-dimensional curves, biomedical applications generally consider 2D and 3D curves as data objects, as seen in later sections. The most difficult of the aforementioned invariances is that to parameterization, and it requires the definition of a nonstandard Riemannian metric on $\mathcal{F}$ referred to as the elastic metric. We begin by identifying the curve f with the pair (r, θ) where $r = |\dot{f}|$ is the speed function and $\theta = \frac{\dot{f}}{|\dot{f}|}$ is the angle function. The only information lost when passing from f to the pair (r, θ) is translation, which is one of the nuisance, shape-preserving transformations. Also, let $(\delta r_1, \delta\theta_1)$ and $(\delta r_2, \delta\theta_2)$ be two tangent vectors at (r, θ). Then, the elastic Riemannian metric is defined as

$$\langle\langle(\delta r_1, \delta\theta_1), (\delta r_2, \delta\theta_2)\rangle\rangle_{(r,\theta)} = a \int_{\mathcal{D}} \delta r_1(t)\delta r_2(t)\frac{1}{r(t)}dt + b \int_{\mathcal{D}} \delta\theta_1(t)^T \delta\theta_2(t) r(t) dt. \tag{24.10}$$

We note three important properties of this metric. First, it is a weighted combination of two terms, one capturing changes in the speed function, i.e., stretching deformations, and one capturing changes in the angle function, i.e., bending deformations. Second, the stretching term in the metric should look familiar: it is the same as the FR metric introduced earlier for densities. Third, this metric is invariant to reparameterizations of curves, in addition to translation, scaling and rotation. Unfortunately, as in the two previous cases, this metric is difficult to use in practice.

Fortunately, one can extend the SRSF representation introduced for functional data to this more general case. This new representation of curves is called the square-root velocity function (SRVF) [21] and is defined as $q = \sqrt{r}\theta = \frac{\dot{f}}{\sqrt{|\dot{f}|}}$. In fact, the SRVF and SRSF are equivalent for univariate curves. The SRVFs of absolutely continuous curves reside in $\mathbb{L}^2(\mathcal{D}, \mathbb{R}^k)$ (simply $\mathbb{L}^2$ for brevity). An important property of this representation is that the complicated elastic metric, with $a = 1/4$ and $b = 1$, simplifies to the standard $\mathbb{L}^2$ metric under the SRVF transform. We note that the SRVF is not the only transform that simplifies a specific instance of the elastic metric to the $\mathbb{L}^2$ metric; for alternative approaches see [2, 28, 53, 54]. We will use the SRVF to mathematically formalize the notion of shape so that any two curves that are within a translation, rescaling, rotation and reparameterization of each other are considered to be the same data object. Since the SRVF is a function of the derivative of the original curve, it is automatically translation invariant (this is obvious since the elastic metric is translation invariant). Forcing a unit length constraint on the curves results in unit $\mathbb{L}^2$ norm SRVFs, i.e., $\|q\|^2 = 1$. Hence, the set of unit length open curves is $C = \{q : [0, 1] \to \mathbb{R}^k | \|q\|^2 = 1\}$, i.e., a unit sphere in $\mathbb{L}^2$; C is also referred to as the pre-shape space. Restricting attention to closed curves, the pre-shape space becomes $C^c = \{q : \mathbb{S}^1 \to \mathbb{R}^k | \|q\|^2 = 1, \int_{\mathbb{S}^1} q(t)|q(t)|dt = 0\}$, which is a subspace of C due to the closure constraint. In the remainder, to keep the discussion general, we do not make a distinction between these two pre-shape spaces and simply use C. The rotation and reparameterization variabilities can be filtered out through a suitable definition of equivalence classes. Let $[q] = \{O(q, \gamma)|\gamma \in \Gamma, O \in SO(k)\}$ denote all possible rotations and reparameterizations of q, where $SO(k) = \{O \in \mathbb{R}^{k\times k}|O^T O = OO^T = 1, \det(O) = 1\}$ is the special orthogonal group of rotations, $\Gamma = \{\gamma : \mathcal{D} \to \mathcal{D}|\gamma$ is a diffeomorphism$\}$ is the set of (order-preserving) reparameterizations and $(q, \gamma) = (q \circ \gamma)\sqrt{\dot{\gamma}}$. Each equivalence class represents a shape uniquely and the collection of all equivalence classes is the shape space $\mathcal{S} = C/(SO(k) \times \Gamma)$. The final ingredient is the ability to compare shapes using a distance on $\mathcal{S}$. Under the SRVF representation, this distance is given by

$$d([q_1], [q_2]) = \min_{O\in SO(k),\gamma\in\Gamma} \cos^{-1}\Big(\int_{\mathcal{D}} q_1(t)^T O q_2(\gamma(t))\sqrt{\dot{\gamma}(t)}\Big)dt. \tag{24.11}$$

The optimization problem in Eq. (24.11) is solved using a combination of Procrustes analysis [10] and dynamic programming [43]. For visualization, a geodesic path between two shapes can be constructed using Eq. (24.3) with inputs q_1 and $O^*(q_2, \gamma^*)$, where O^* and γ^* denote the minimizers of Eq. (24.11).

24.2.4 Shapes of Surfaces

Lastly, we consider shape analysis of surfaces. This case evolves similarly to the case of curves. Again, with a slight abuse in notation, let $\mathcal{F}$ denote the space of smooth embeddings $f : \mathcal{D} \to \mathbb{R}^3$, where the domain of parameterization $\mathcal{D}$ can be a unit sphere (closed surfaces), a unit square (quadrilateral surfaces), a unit cylinder (cylindrical surfaces), a unit disk (hemispherical surfaces), etc. Furthermore, let Γ be the set of all diffeomorphisms of $\mathcal{D}$. We use $n(t) \in \mathbb{R}^3$ to denote the normal vector to the surface at the point $t \in \mathcal{D}$, i.e., $n(t) = \frac{\partial f}{\partial u}(t) \times \frac{\partial f}{\partial v}(t)$, where (u, v) are the coordinates on the domain $\mathcal{D}$. The infinitesimal area measure at a point t is given by $r(t) = |n(t)|$ and the normalized normal vector is $\tilde{n}(t) = \frac{n(t)}{r(t)}$. We will represent the surface f using the pair $(r, \tilde{n})$; as this representation depends on partial derivatives only, it is automatically invariant to translations. Let $(\delta r_1, \delta\tilde{n}_1)$ and $(\delta r_2, \delta\tilde{n}_2)$ be two tangent vectors at $(r, \tilde{n})$. A reparameterization invariant Riemannian metric on the space of surfaces is given by [19]

$$\langle\langle(\delta r_1, \delta\tilde{n}_1), (\delta r_2, \delta\tilde{n}_2)\rangle\rangle_{(r,\tilde{n})} = \frac{1}{4}\int_{\mathcal{D}} \delta r_1(t)\delta r_2(t)\frac{1}{r(t)}dt + \int_{D} \delta\tilde{n}_1(t)^T \delta\tilde{n}_2(t) r(t)dt. \tag{24.12}$$

Again, the first term in this metric resembles the FR metric introduced earlier, and captures changes in the infinitesimal areas of surface patches, i.e., stretching deformations. The second term captures changes in the direction of the unit normal vector, i.e., bending deformations. The metric in Eq. (24.12) is a special case of a more general elastic metric for surfaces [19]. Due to the difficulty of working with this metric in practice, we define an alternative representation of surfaces, called the square-root normal field (SRNF), which simplifies this metric to the standard $\mathbb{L}^2$ metric. The SRNF of a surface f is given by $q = \sqrt{r}\tilde{n} = \frac{n}{\sqrt{|n|}}$. The SRNF of a reparameterized surface $f \circ \gamma$, for a $\gamma \in \Gamma$, is given by $(q, \gamma) = (q \circ \gamma)\sqrt{J_\gamma}$, where J_γ is the determinant of the Jacobian of γ.

As in the case of curves, we seek a framework that is invariant to all shape-preserving transformations (translation, scale, rotation and reparameterization). The SRNF representation is automatically invariant to translations. To produce invariance to scaling, we rescale all surfaces to unit area, resulting in SRNFs with unit $\mathbb{L}^2$ norm. As in the case of curves, this amounts to restricting attention to the unit sphere in $\mathbb{L}^2$. We then define a distance on the shape space of surfaces by minimizing over equivalence classes of the form $[q] = \{O(q, \gamma) | \gamma \in \Gamma,\ O \in SO(3)\}$

$$d([q_1], [q_2]) = \min_{O \in SO(3), \gamma \in \Gamma} \cos^{-1}\left(\int_{\mathcal{D}} q_1(t)^T O q_2(\gamma(t))\sqrt{J_\gamma(t)}\right) dt. \tag{24.13}$$

As in the case of curves, the optimal rotation is found using Procrustes analysis [10]. Computation of the optimal reparameterization requires a gradient descent algorithm [29]. A geodesic path between two shapes can be constructed using Eq. (24.3) with inputs q_1 and $O^*(q_2, \gamma^*)$, where O^* and γ^* are the minimizers of Eq. (24.13).

24.3 Nonparametric Metric-Based Statistics

We provide a general recipe for computing the sample mean, covariance and performing principal component analysis (PCA). Our tools rely on Karcher means for metric spaces and local linear approximations via the Riemannian structure. Since all four geometric data objects described in Sect. 24.2 rely on $\mathbb{L}^2$ Riemannian geometry, we provide a single description here for brevity.

24.3.1 Karcher Mean

The sample Karcher mean [24] of a collection of points (i.e., pdfs, amplitude functions, phase functions or shapes) $x_1, \ldots, x_n$ from a metric space $(\mathcal{X}, d)$ is defined as the minimizer of the Karcher variance

$$\hat{\mu} = \arg\min_{x \in \mathcal{X}} \frac{1}{n} \sum_{i=1}^{n} d(x, x_i)^2. \tag{24.14}$$

This definition, with slight modification when dealing with equivalence classes, is applicable to all four metric spaces discussed in Sect. 24.2. Computation of the Karcher mean is carried out using gradient-based algorithms [31, 36, 40], which generally iterate between three steps: (1) projection of data from the representation space to the linear tangent space at the current estimate of the mean via the inverse exponential map, (2) computation of the gradient of the cost function in Eq. (24.14), and (3) update of the current estimate of the mean using the exponential map. In the case of functional data, the Karcher mean is used as a template for mutliple function registration. That is, once the Karcher mean is estimated, the amplitude components of all functions are defined through pairwise registration to the Karcher mean; this also results in the phase component, computed with respect to the mean [48].

24.3.2 Covariance Estimation and Principal Component Analysis

Exploration of variability in a sample of geometric data objects can be carried out by choosing local coordinates in the vicinity of the Karcher mean $\hat{\mu}$. The Riemannian structure allows one to conveniently linearize the data representation space via the tangent space at the mean, $T_{\hat{\mu}}$, and to select Euclidean coordinates in this space.

As before, let $x_1, \ldots, x_n$ and $\hat{\mu}$ represent the data objects of interest and their Karcher mean, respectively. We begin by projecting each $x_i,\ i = 1, \ldots, n$ onto the tangent space at the mean using the inverse exponential map resulting in tangent vectors $v_1, \ldots, v_n$. Using this tangent space representation, we estimate the covariance matrix based on discretized versions of the tangent vectors denoted by $\boldsymbol{v}_i,\ i = 1, \ldots, n$. Assuming the dimension of each $\boldsymbol{v}_i$ is M, the sample covariance matrix is given by $K_M := 1/(n-1)\sum_{i=1}^{n} \boldsymbol{v}_i \boldsymbol{v}_i^T$. To study variability using PCA, we apply the spectral decomposition to the covariance matrix $K_M = U \Sigma U^T$, where the orthogonal matrix U contains the principal components (PCs) or principal directions of variability, and the diagonal matrix Σ contains the PC variances. In typical biomedical applications, the number of observations is smaller than the dimensionality of each tangent vector, i.e., $n < M$. Thus, there are at most $n-1$ positive values in the matrix Σ. The submatrix formed by the first r columns of U, U_r, spans the r-dimensional principal subspace of the observed data, and one can reexpress the data using coordinates in this subspace via principal coefficients computed as $c_i = U_r^T \boldsymbol{v}_i,\ i = 1, \ldots, n$. One can then use these principal coefficients for further statistical modeling, e.g., PC regression [3]. A common approach to modeling complex data objects is through tangent PCA-based models such as the truncated wrapped Gaussian distribution [30] or by directly modeling the principal coefficients.

24.4 Biomedical Case Studies

We focus on multiple biomedical case studies that consider (1) pdfs, (2) amplitude and phase in functional data, (3) shapes of curves, and (4) shapes of surfaces as data objects. While the theoretical underpinnings outlined in Sect. 24.2 consider infinite-dimensional data representations, computer implementation of these methods requires appropriate discretization. We represent pdfs and other univariate functions (amplitude/phase) as $1 \times N$ vectors, where N denotes the number of discretization points selected on the function domain. Shapes of curves are represented as $d \times N$ matrices, where $d = 2, 3$ depending on whether the curve is planar or 3D, and N is again the number of points selected on the curve domain. Finally, shapes of surfaces are represented as $N_1 \times N_2 \times 3$ arrays, where $N_1 \times N_2$ defines a discretization grid on the surface, and each point on the grid takes a value in $\mathbb{R}^3$.

24.4.1 Probability Density Functions

Assessment of Glioblastoma Multiforme Tumor Texture Variability Glioblastoma multiforme (GBM), also known as grade IV glioma, is the most common form of a malignant brain tumor in adults [15]. It is a morphologically heterogeneous disease with extremely poor prognosis; also, predicting the impact of standard cancer treatments such as chemotherapy and radiation therapy becomes considerably challenging. Thus, exploring tumor heterogeneity is critical in cancer research as inter- and intra-tumor differences have stymied the systematic development of targeted cancer therapies [9]. MRI is one of the modern medical imaging techniques that has been used to investigate tumor development in various contexts. MRI scans are primarily used to exhibit and evaluate the location, size, growth and progression of tumors, which serve as indicators for clinical decision making. Various physiological features are extracted by using voxel-level data to visualize the progression (or regression) of tumors. This is generally done by constructing voxel value histograms. However, in most cases, only simple summaries of the entire histograms are used for statistical analysis. This approach has two main drawbacks. First is the subjectivity in the choice of the number and location of the summary features (e.g., quantiles or percentiles, etc.). Second, and more importantly, these summary features fail to capture the entire information in a histogram of voxel intensities, and thus cannot detect small-scale and sensitive changes in the tumor due to treatment effects [23].

Alternatively, one can exploit the entire histogram, or its corresponding smoothed density profile, for the tumor region in an MRI. This was the approach taken in a recent paper that introduces DEMARCATE, a self-contained pipeline for geometric clustering and validation of GBM tumor texture profiles [44]. Semi-automated segmentation methods [1] can be employed to delineate the tumor region in the

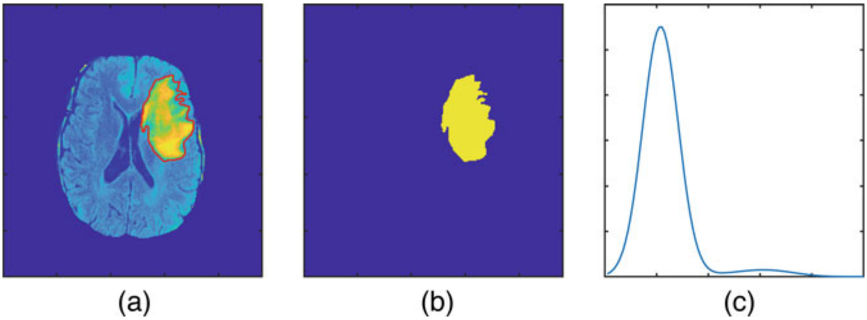


Fig. 24.1 (**a**) MRI slice for a subject with GBM; the delineated region corresponds to the tumor. (**b**) Mask identifying the tumor region. (**c**) Estimated voxel intensity pdf corresponding to the tumor

whole brain MRI scan. In subsequent analyses we use the voxel-level information from the axial slice with the largest tumor area only. This is done for simplicity of visualization and can be easily extended to the full 3D tumor. Figure 24.1a shows a single slice of an MRI scan for a subject with GBM, where the delineated region corresponds to the tumor. This region is displayed as a binary mask in panel (b). The voxel values inside the tumor are used to compute a Gaussian kernel density estimate (a pdf), which is displayed in panel (c); it contains detailed and refined information about the voxel-level tumor characteristics. Hence, under this setup, a sample of GBM scans is represented by a sample of voxel value pdfs corresponding to the tumor region in the MRI scan of each subject. For a more detailed description of the image processing pipeline, we refer the interested reader to [44]. The imaging data used in this study was retrieved from The Cancer Imaging Archive (www.cancerimagingarchive.net).

Next, we consider a comparison of two subjects based on their voxel value pdfs. Figure 24.2a, b shows the MRI slice for two subjects, and the corresponding pdfs of the tumor intensity values. The geodesic path between the two pdfs under the FR metric is shown in Fig. 24.2c. The displayed geodesic was discretized with five equally spaced points on the interior of the path. Finally, we consider a random sample of ten subjects with GBM. The densities for these ten subjects (dashed), along with their Karcher mean (solid red) are displayed in Fig. 24.3a. The Karcher mean in this case provides a simple summary of the sample of voxel intensity pdfs, and was computed using the FR Riemannian framework. We do not display the corresponding MRI slices in this case for brevity (note that there doesn't exist a unique MRI slice corresponding to the Karcher mean pdf). Given an estimate of the Karcher mean, we perform PCA and show the first principal direction of variability in the given sample. This result is provided in Fig. 24.3b and reflects the relative heights of the different modes in the sample of voxel value pdfs. While not shown here, principal coefficients can be subsequently used as covariates in regression models, e.g., to predict subject survival [3].

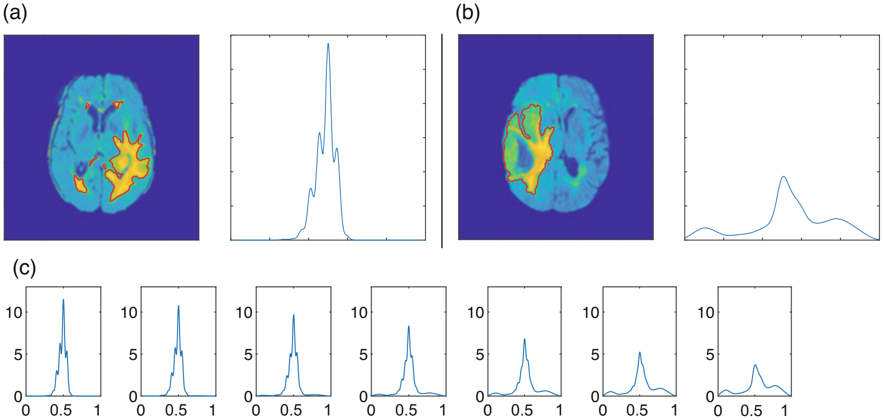


Fig. 24.2 (**a** and **b**) MRI slices from two different GBM subjects, with the pdf corresponding to the tumor intensity values. (**c**) Geodesic path between the pdfs for subject (**a**) and subject (**b**)

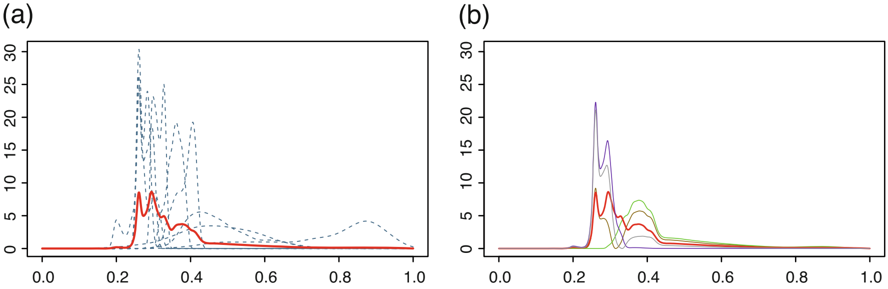


Fig. 24.3 (**a**) Karcher mean (solid red line) of a random sample of ten voxel value pdfs (dashed lines) extracted from tumor regions of GBM subjects. (**b**) Principal direction of variability in the sample, displayed at −2, −1, 0, +1, +2 standard deviations around the mean (red)

24.4.2 Amplitude and Phase in Elastic Functional Data

Automatic Segmentation and Clustering of Electrocardiogram Signals The electrocardiogram (ECG) is a cheap and widely-applied diagnostic tool for assessment of various heart diseases including myocardial infarction (MI). Automated algorithms, based on sound mathematical and statistical principles, that can accurately and efficiently analyze ECG signals are thus useful in monitoring and identifying the risk or onset of a particular disease. The ECG captures fluctuations in electrical potential of the heart muscle on the body surface and results in a vector that represents the magnitude and direction of the electric field generated through the heart [8]. The ECG represents an example of a highly periodic biomedical signal. The two main challenges in analyzing such data include (1) automatic segmentation

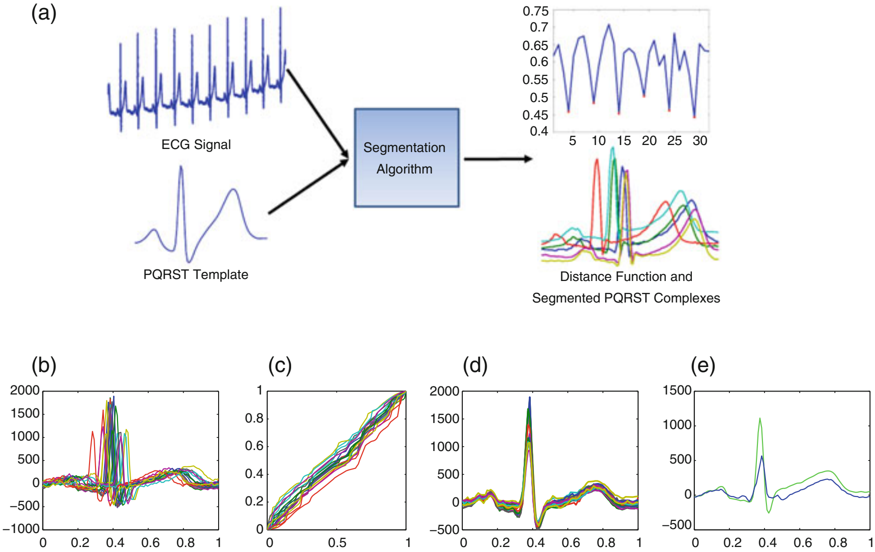


Fig. 24.4 (**a**) Pictorial description of automatic algorithm for segmentation of long ECG signals. Bottom: Registration of PQRST complexes to a common template. (**b**) Given PQRST complexes. (**c**) Phase component. (**d**) Amplitude component. (**e**) Comparison of average PQRST complex without (blue) and with (green) registration. Image courtesy of [32]

of cycles called PQRST complexes (PQRST refer to semantic features of each cycle: the P peak, QRS complex and the T peak) from a long temporal ECG signal, and (2) automatic registration of cycles to extract amplitude and phase variabilities of individual cycles. The ECG data used here for demonstrative purposes is a subset of the PTB Diagnostic ECG Database [5] obtained from Physionet [12].

In [32], the authors solve these two problems using techniques from elastic functional data analysis described in Sect. 24.2.2. First, they define an automatic signal segmentation algorithm based on a sliding window approach. In particular, they construct a PQRST complex template, based on the amplitude component of a few manually segmented PQRST complexes, and slide it along the long periodic signal. The cost function that is then used for segmentation is the phase distance, defined in Eq. (24.9), between the part of the long signal in the current window and the defined template. The PQRST cycles are identified as local minima of this cost function. Figure 24.4a provides a pictorial description of this process. Once the cycles have been extracted, their amplitude components are found by registering to a new common template. This result is displayed in the bottom panel of Fig. 24.4. In (b), we show the segmented PQRST complexes. The extracted phase components are displayed in (c) with corresponding amplitude components in (d). Finally, in (e), we compare the amplitude means computed without (blue) and with (green) registration. Note the enhanced features of the PQRST complex average computed after registration.

In addition to extracting the amplitude and phase components from PQRST complexes, the authors in [32] use the amplitude components for (1) classification of subjects as healthy controls or as having MI and (2) localization of the MI as anterior or inferior. The data they use for this experiment consists of 80 healthy control ECGs, 28 of which are repeated measures for the same subject, and 80 MI ECGs with no repeated measures. For each subject, they first segment the long ECG signal into corresponding PQRST complexes and then use the amplitude of the average PQRST complex for classification using the nearest neighbor procedure. They report an accuracy of MI classification of 90% by combining information from different ECG leads (the data contains a total of 15 different ECG signals called leads per subject). Also, they report a localization accuracy of 92.21%, again based on combining multiple single lead classifiers.

Assessment of Variability in DT-MRI Fractional Anisotropy Functions in Multiple Sclerosis DT-MRI is a neuroimaging modality that traces the diffusion of water molecules in the brain. A DT-MRI scan of a subject's brain provides a 3×3 tensor matrix, at each voxel in the image, that describes the constraints of local Brownian motion of water molecules. This information is essential to understanding white matter in the brain which constitutes areas made up of axons or tracts. Tracts connect neurons and allow for the transmittance of electric signals from one area of the brain to another, affecting overall brain function. Because the diffusion of water in tracts is anisotropic, tracts themselves can be extracted from the information contained in a DT-MRI, along with other quantities of interest that describe the quality of tract connection by summarizing its degree of anisotropy.

Here, we focus on Fractional Anisotropy (FA) measurements along tracts, which provide a voxel-wise summary of the eigenvalues of the diffusion tensors, denoted by $\lambda_1, \lambda_2, \lambda_3$. At each voxel, FA is given by the scalar quantity $FA = \sqrt{\frac{3}{2}}\sqrt{\frac{(\lambda_1-\bar{\lambda})^2+(\lambda_2-\bar{\lambda})^2+(\lambda_3-\bar{\lambda})^2}{\lambda_1^2+\lambda_2^2+\lambda_3^2}}$, where $\bar{\lambda} = \frac{\lambda_1+\lambda_2+\lambda_3}{3}$. A larger FA value indicates a large degree of anisotropy. For practitioners, this summary is interpreted as measuring the quality of connections between neurons connected by the tracts in a particular region of interest, and has been found to be a useful quantity to study subjects with various diseases, e.g., multiple sclerosis (MS) [13], Alzheimer's [39], etc. In the MS setting, the autoimmune disease causes lesions and damage to tracts that results in a decrease in FA. Thus, this quantity can be used to distinguish between healthy controls and subjects with MS, and to predict cognitive and motor disease outcomes. The data of interest takes a functional form, with the domain of the functions representing locations along tracts. Determining the voxels that the tracts pass through in the image is a practical challenge in itself and will be further discussed in Sect. 24.4.3.

The functional FA data we analyze here is available as part of the 'refund' package in R [14]. In particular, we study the mean and principal directions of variability in a sample of 66 subjects with MS whose FA values were measured at 55 locations along the right corticospinal tract that contributes to fine motor movements in ipsilateral limbs. The domain of parameterization for each FA function was

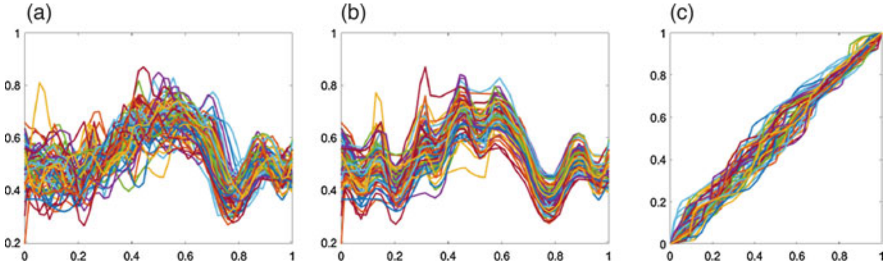


Fig. 24.5 (**a**) Observed FA functions. (**b**) Amplitude component. (**c**) Phase component

normalized to [0, 1] for convenience. It is important to note that due to differences in the geometry of different subjects' white matter, there generally exist both phase and amplitude variabilities in the FA functional data, as demonstrated next. The raw FA functions for the 66 subjects are shown in Fig. 24.5a. The amplitude components of the functions, after registration to a common template, are displayed in Fig. 24.5b. Finally, the warping functions which constitute the phase components are shown in Fig. 24.5c. Visual inspection of panel (b) reveals that the number of extreme values in the FA functions is roughly the same across subjects. The main source of variability in this case are the heights of the extreme values. The phase components in panel (c) suggest that the extreme values occur at different parameter values for different subjects, which is intuitive given natural geometric variability of the tracts across subjects. These insights are only made possible through the separation of amplitude and phase by registering all functions to a common template; such patterns are much more difficult to observe by looking at the observed functions in panel (a).

Figure 24.6 further highlights the importance of elastic functional data analysis methods by contrasting averages computed without (panel (a)) and with (panel (b)) registration. While the general patterns in the two means are similar, the amplitude mean in panel (b) reveals much more local structure through small peaks and valleys. Finally, to understand patterns of variability in the given sample of FA functions, we perform PCA on the amplitude components. Since the translation of the functions is also informative in this setting, we include it as an additional feature in the PCA model (it is appropriately weighted to make the scales of the two components, amplitude and translation, comparable). The first three principal directions of amplitude (and translation) variability are visualized in Fig. 24.6c–e. The first direction predominantly captures variability in translation as well as the initial portion of the functions, as some functions in the sample initially decrease and others increase. The second direction captures fine features of the different peaks and valleys of the FA functions, especially the fourth peak, as well as large amount of variability at the end of the functions. Finally, the third direction (and subsequent directions not displayed here) capture bigger differences in the relative heights of peaks and valleys.

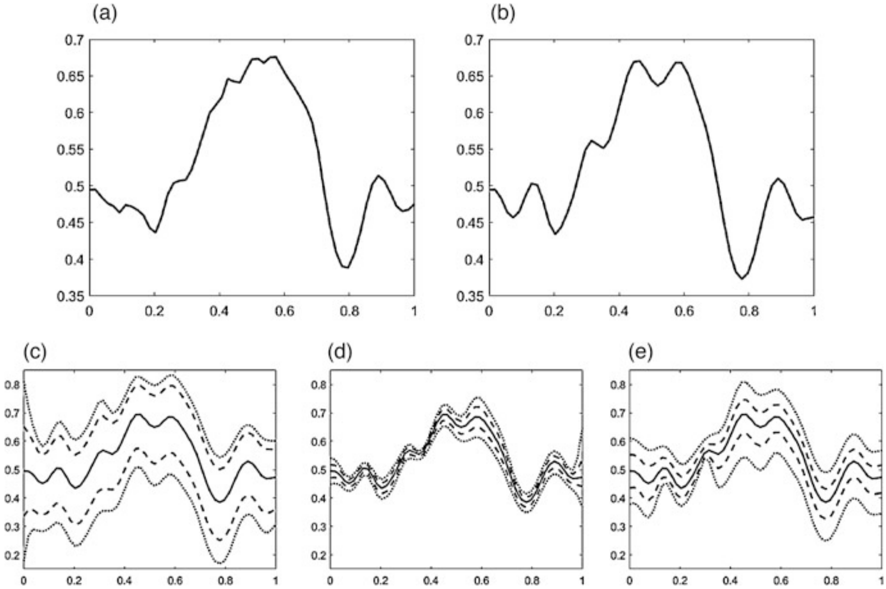


Fig. 24.6 (**a**) Pointwise mean of the FA functions in Fig. 24.5a. (**b**) Karcher mean of the FA functions after registration in Fig. 24.5b. (**c**)–(**e**) First three principal directions of amplitude (and translation) variability of the FA functions, respectively. We display a path of functions sampled at -2 and $+2$ (dotted lines), -1 and $+1$ (dashed lines), and 0 (solid line) standard deviations from the mean

24.4.3 Shapes of Open and Closed Curves

Comparison and Summarization of Planar Shapes of GBM Tumors We return to the study of the GBM tumor dataset, as described in Sect. 24.4.1. However, instead of modeling the internal texture of the tumors, we instead model the shapes of tumor outlines. This allows us to study growth patterns and shape heterogeneity of tumors, which are features that are complementary to voxel intensity values. Tumor shape is affected by the location of the tumor in the brain due to constraints posed by the brain anatomy such as white matter and blood vessels. In [3], the authors suggest that tumor shape could enhance our understanding of disease prognosis and help in prediction of therapeutic success. As in Sect. 24.4.1, the imaging data is a subset of The Cancer Imaging Archive, and the tumor shape is obtained through semi-automated segmentation; a segmented tumor is visualized in Fig. 24.1. The geometric data object of interest in this case is the red outline of the tumor rather than the entire MRI slice. In this case study, we consider 63 GBM tumor outlines, which are represented as planar closed curves. We focus on characterization of tumors through the visualization of geodesic paths, the Karcher mean shape and shape PCA. Similar results appear in [3]; the scope of their study is broader and additionally includes shape clustering, hypothesis testing and survival modeling.

We begin with a visualization of a geodesic path between two tumor shapes in Fig. 24.7. If the two endpoints of the geodesic path are a single subject's tumor at different timepoints, the points along the path can be viewed as an interpolation along different stages of tumor growth. This, in turn, can help a practitioner retrospectively understand how the subject's tumor has evolved over time without collecting MRI data at intermediate timepoints. On the other hand, when the two endpoints are shapes of tumors coming from two different subjects, as in Fig. 24.7, the path can help formulate a qualitative understanding of how tumor shapes differ in the population. In this case, the shapes of the tumors seem to differ by how bulbous or skinny their protrusions are. By viewing more subjects' tumors in Fig. 24.8, this seems to be a common discrepancy between the different subjects. The insight that this is a primary source of variability in GBM tumor shapes is formalized by viewing the principal directions of variability in the entire dataset; the first four directions are shown in Fig. 24.9. Notice that the first direction, which captures approximately 41% of the total variability, describes the types of differences in

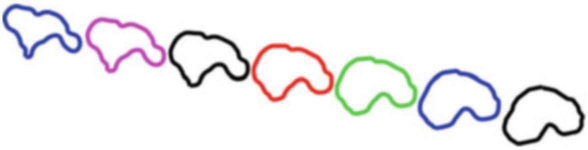


Fig. 24.7 Geodesic path between shapes of GBM tumors for two subjects (blue and black endpoints), sampled uniformly using five interior points along the path

Fig. 24.8 Five randomly selected GBM tumor outlines

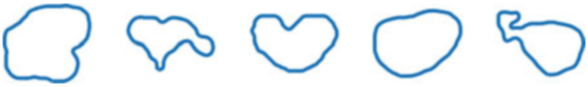
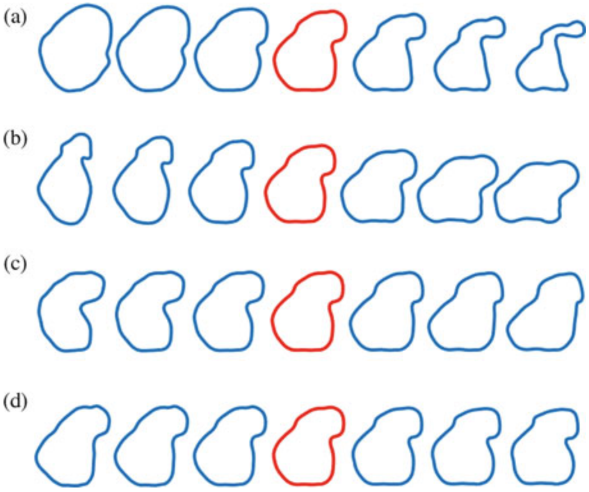


Fig. 24.9 (**a**)–(**d**) First four principal directions of shape variability in the GBM dataset, respectively, sampled at −3, −2, −1, 0, +1, +2 and +3 standard deviations around the mean shape (red)



protrusions described before. The remaining directions describe other shapes of the bulbous features of the tumors. The second, third and fourth principal directions of variation capture about 33%, 16%, and 10% of the total variability, respectively; essentially all of the variation is contained in these first four directions. This implies that a low dimensional model, based on these PCs, could be used for subsequent statistical analyses.

Clustering Shapes of 3D DT-MRI Tracts As previously mentioned in Sect. 24.4.2, DT-MRI tracts are of interest when studying structural connections between different regions of the brain. Tractography is the field of study concerned with discerning tracts from the tensor-based DT-MRIs [37]. Conventional tractography relies on the principle that water diffuses anisotropically in white matter tracts in a principal direction that is encoded in the diffusion tensor. This implies that the direction that the tract is pointing in a voxel will coincide with the eigenvector corresponding to the maximum eigenvalue of the diffusion tensor. Consequently, an entire tract can be traced using information from the observed diffusion tensors associated with voxel locations. The application described in [30] deals with tracts that connect Broca's and Wernicke's regions of the left hemisphere of the brain; these two regions are associated with the human language circuit. While two main routes of connection are widely recognized, there is an ongoing debate on whether the white matter tracts connecting the two regions can be further broken down into smaller subroutes.

The data in this study contains different numbers of fiber tracts for four subjects. We identify different routes of connectivity by clustering the shapes of these tracts using distance-based methods. This was also done in [30], but there the authors used shape in conjunction with other features of the tracts. To determine if the tracts can be put in different clusters representing major pathways connecting the regions of interest, a hierarchical clustering algorithm, with a complete linkage criterion, is used to cluster the observations for each individual based on the elastic shape distance defined in Eq. (24.11). The results for all four subjects are shown in Fig. 24.10. The tracts, represented as 3D open curves, are plotted in the top panel of the figure and are colored by cluster membership. In the middle panel, we show the pairwise shape distance matrix as an image, rearranged according to the computed clusters. Note the nice separation of clusters in this distance matrix. Finally, in the bottom panel, we show a plot of the tracts after applying multidimensional scaling (MDS) to the distance matrices. This 2D scatterplot provides a lower dimensional visualization of the clustered data. Some of the subjects exhibited tracts that could be separated into more than two clusters, e.g., Fig. 24.10b, c. The case for more than two clusters is hard to justify for the subject in Fig. 24.10a. Based on these results, it appears that the hypothesis that there are two or three main pathways connecting Broca's and Wernicke's regions in the left hemisphere is plausible for all of the subjects considered in this case study.

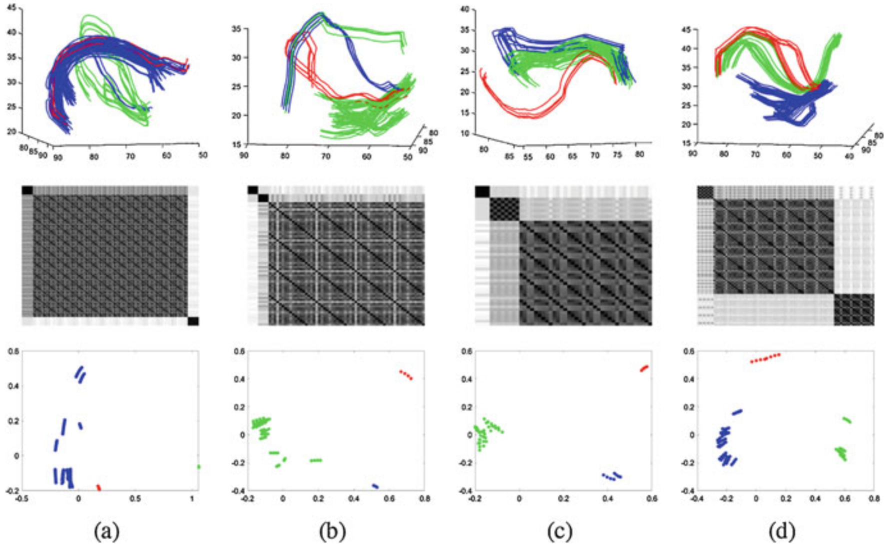


Fig. 24.10 (**a**)–(**d**) Results of hierarchichal shape clustering for four subjects. Top: Tracts colored by cluster membership. Middle: Image of distance matrix. Bottom: MDS plot of tracts

24.4.4 Shapes of Surfaces

Simulation of Endometrial Tissue Shapes We define a PCA-based statistical model for efficient simulation of random endometrial tissue shapes, which can be used for validation of various image processing algorithms such as multimodal registration of MRI and transvaginal ultrasound (TVUS). This is an important task in the context of diagnosis and surgery planning for endometriosis [45, 52], a complex gynecological disease in which endometrial cells appear outside their usual locations in the uterine cavity [6]. Endometriosis affects approximately 10% of women in the reproductive age group and may cause chronic pelvic pain, severe dysmenorrhea, infertility, rectal bleeding and digestive problems.

In this study [33, 34], we use real data from ten subjects with small endometrial implants in the pelvic area. The available data are cylindrically parameterized surfaces of endometrial tissues, reconstructed from 2D MRI slices. The entire dataset can be found in Figure 1(b) in [34]. There is a lot of variation in this data, and thus, parsimonious shape models are very important in this application. Of main interest is random generation of realistic endometrial tissue shapes as they'd appear in an MRI scan and a corresponding TVUS image. Unfortunately, endometrial tissue is soft and undergoes a significant deformation during TVUS imaging due to the transducer's pressure. Thus, in addition to generating a random endometrial tissue shape we must additionally apply a deformation on the surface of the shape that is consistent with the TVUS imaging protocol.

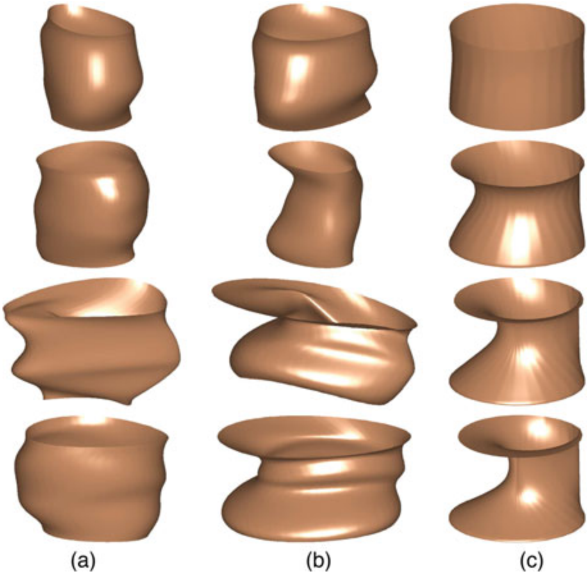


Fig. 24.11 (**a**) Randomly sampled shape from the Gaussian model resembling MRI data. (**b**) Random sample after additional deformation resembling TVUS data. (**c**) Deformation applied to the random sample displayed on a perfect cylinder. Image courtesy of [34]

To achieve the two goals outlined above, we first compute the Karcher mean of and perform PCA on the ten given endometrial tissue shapes. This allows us to express the data in terms of the principal coefficients. We model these coefficients using a simple zero-mean multivariate Gaussian distribution with the covariance structure informed by the PCA. A major advantage of this shape model is that it is very easy and computationally efficient to sample from. Figure 24.11a shows four randomly generated endometrial tissue shapes as they'd appear in an MRI. Then, to simulate the semi-synthetic deformation needed for the corresponding TVUS-based endometrial tissue shape, we define a simple diffusion model with different degrees of deformation on the previously computed Karcher mean; the deformation is centered at a randomly selected point on the mean. These deformations can then be transported from the Karcher mean to each of the random samples from our model using parallel transport [51]. Figure 24.11c displays the deformations applied to a perfect cylinder. The magnitude of deformation increases from the top row to the bottom row. Finally, the TVUS-based, deformed endometrial tissue shapes are displayed in (b). The random samples generated using this approach (as well as their deformed counterparts) naturally resemble the given data. In [34], the authors provide a thorough validation of their models and a formal assessment by a clinician.

Classification of Attention Deficit Hyperactivity Disorder (ADHD) via Shapes of Subcortical Structures Recently, many researchers have become interested in studying shape changes of brain structures and associating these changes with various diseases including Alzheimer's [22, 50], Parkinson's [11], autism [16] and ADHD [29], among others. Statistical analysis of the shapes of such structures plays a central role in the ability to diagnose and monitor such diseases, as well as to develop novel treatment strategies. The current standard of practice is to use clinical

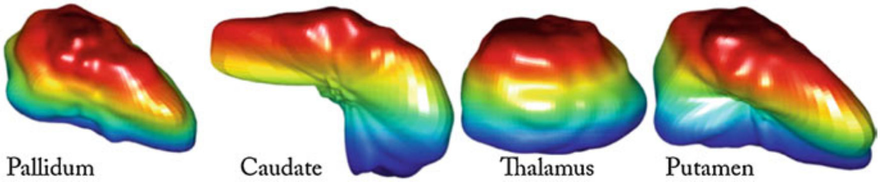


Fig. 24.12 Subcortical structures used for classification of ADHD. Image courtesy of [20]

symptoms, including various behavioral tests, to detect and quantify abnormalities due to disease status. Such an approach has clear limitations as the tests are often subjective and mainly qualitative, relying entirely on a doctor's assessment and judgment.

As an alternative, our final case study considers classification of ADHD based on the shapes of four distinct subcortical structures, represented as closed surfaces: pallidum, caudate, thalamus and putamen; a single example of each structure is displayed in Fig. 24.12. The surfaces of these subcortical structures were segmented from T1-weighted MRIs of young adults aged between 18 and 21 who were recruited from the Detroit Fetal Alcohol and Drug Exposure Cohort [17, 18]. Among the 34 subjects in this dataset, 19 were diagnosed with ADHD and the remaining 15 were controls. The classifier in this study was constructed in the following way. First, the training data was used to estimate the Karcher mean in each class. Then, shape PCA was used to define a Gaussian model on the principal coefficients. The resulting classifier simply uses the likelihood ratio under these two models to classify test shapes into control or ADHD classes. This classifier was applied in a leave-one-out manner to the above-described dataset, i.e., at each iteration a single case was left out for testing while the rest were used to learn the classification model. The best classification result obtained using this method was based on the shape of the left putamen, 94.1%. The shape of the right pallidum yielded a classification accuracy of 76.5%, and the shapes of the left caudate, left thalamus and right thalamus resulted in a slightly worse classification accuracy of 67.7%. Comprehensive results of this study are reported in [20], where the approach outlined here was compared to other classifiers and other shape representations [29].

24.5 Summary

We consider several biomedical applications of geometric functional data analysis. We begin by assessing variability in a sample of GBM voxel intensity pdfs to model tumor appearance. We then shift our focus to the use of elastic functional data methods for analyzing amplitude and phase components of electrocardiogram signals and FA functions extracted from DT-MRI. For the GBM tumor data, we additionally study shape variability of tumor outlines extracted from single MRI slices, which form planar closed curves. Shapes of white matter tracts in DT-MRI provide information about connectivity of different brain areas. We cluster particular sets of tracts to elucidate connection pathways between Broca's and Wernicke's areas, which are associated with the language circuit. Finally, we use shape models to simulate 3D endometrial tissue shapes, and to define classifiers for ADHD based on shapes of subcortical structures.

Acknowledgements We thank Arvind Rao for sharing the GBM tumor dataset, and acknowledge Joonsang Lee, Juan Martinez, Shivali Narang and Ganesh Rao for their roles in processing the MRIs used to produce the tumor outlines. We thank Zhaohua Ding for providing the DT-MRI tract dataset used in Sect. 24.4.3 and the dataset of subcortical structures for ADHD classification. Finally, we thank Chafik Samir for providing the endometrial tissue surfaces. Shariq Mohammed would like to acknowledge Institutional Research Support from The University of Michigan. Additionally, this work was supported in part by grants NSF DMS-1613054, NSF CCF-1740761, NSF CCF-1839252 and NIH R37-CA214955.

References

1. Bakas, S., Zeng, K., Sotiras, A., Rathore, S., Akbari, H., Gaonkar, B., Rozycki, M., Pati, S., Davatzikos, C.: GLISTRboost: combining multimodal MRI segmentation, registration, and biophysical tumor growth modeling with gradient boosting machines for glioma segmentation. In: BrainLes, pp. 144–155 (2015)
2. Bauer, M., Bruveris, M., Harms, P., Møller-Andersen, J.: A numerical framework for Sobolev metrics on the space of curves. SIAM J. Imag. Sci. **10**(1), 47–73 (2017)
3. Bharath, K., Kurtek, S., Rao, A., Baladandayuthapani, V.: Radiologic image-based statistical shape analysis of brain tumours. J. R. Stat. Soc. Ser. C **67**(5), 1357–1378 (2018)
4. Bhattacharyya, A.: On a measure of divergence between two statistical populations defined by their probability distributions. Bull. Calcutta Math. Soc. **35**, 99–109 (1943)
5. Bousseljot, R., Kreiseler, D., Schnabel, A.: Nutzung der EKG-Signaldatenbank CARDIODAT der PTB uber das internet. Biomedizinische Technik **40**(1), S317–S318 (1995)
6. Brosens, I., Puttemans, P., Campo, R., Gordts, S., Kinkel, K.: Diagnosis of endometriosis: Pelvic endoscopy and imaging techniques. Best Pract. Res. Clin. Obstet. Gynaecol. **18**(2), 285–303 (2004)
7. Čencov, N.N.: Statistical Decision Rules and Optimal Inference. American Mathematical Society, Providence (1982)
8. Clifford, G.D., Azuaje, F., McSharry, P.: Advanced Methods And Tools for ECG Data Analysis. Artech House, Inc., Boston (2006)
9. De Sousa, F., Melo, E., Vermeulen, L., Fessler, E., Medema, J.P.: Cancer heterogeneity—a multifaceted view. EMBO Rep. **14**(8), 686–695 (2013)
10. Dryden, I.L., Mardia, K.V.: Statistical Shape Analysis. Wiley, New York (1998)
11. Garg, A., Appel-Cresswell, S., Popuri, K., McKeown, M.J., Beg, M.F.: Morphological alterations in the caudate, putamen, pallidum, and thalamus in Parkinson's disease. Front. Neurosci. **9**, 101 (2015)
12. Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Mark, R., Mietus, J., Moody, G., Peng, C., Stanley, H.: PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. Circulation **101**(23), e215–e220 (2000)

13. Goldsmith, J., Crainiceanu, C.M., Caffo, B.S., Reich, D.S.: Penalized functional regression analysis of white-matter tract profiles in multiple sclerosis. NeuroImage **57**(2), 431–439 (2011)
14. Goldsmith, J., Scheipl, F., Huang, L., Wrobel, J., Gellar, J., Harezlak, J., Mclean, M., Swihart, B., Crainiceanu, L.X.C., Reiss, P., Chen, Y., Greven, S., Huo, L., Kunda, M.G., Park, S.Y., Miller, D., Staicu, A.M.: Refund: Regression with Functional Data (2018). https://cran.r-project.org/web/packages/refund/
15. Holland, E.C.: Glioblastoma multiforme: the terminator. Proc. Natl. Acad. Sci. **97**(12), 6242–6244 (2000)
16. Ismail, M., Keynton, R., Mostapha, M., Eltanboly, A., Casanova, M., Gimel'farb, G., El-Baz, A.: Studying autism spectrum disorder with structural and diffusion magnetic resonance imaging: A survey. Front. Hum. Neurosci. **10** (2016)
17. Jacobson, S., Jacobson, J., Sokol, R., Martier, S., Chiodo, L.: New evidence for neurobehavioral effects of in utero cocaine exposure. J. Pediatr. **129**(4) pp. 581–590 (1996)
18. Jacobson, S., Jacobson, J., Sokol, R., Chiodo, L., Corobana, R.: Maternal age, alcohol abuse history, and quality of parenting as moderators of the effects of prenatal alcohol exposure on 7.5-year intellectual function. Alcohol. Clin. Exp. Res. **28**, 1732–1745 (2004)
19. Jermyn, I.H., Kurtek, S., Klassen, E., Srivastava, A.: Elastic shape matching of parameterized surfaces using square root normal fields. In: European Conference on Computer Vision, pp. 804–817 (2012)
20. Jermyn, I.H., Kurtek, S., Laga, H., Srivastava, A.: Elastic Shape Analysis of Three-Dimensional Objects. Morgan & Claypool Publishers, San Rafael (2017)
21. Joshi, S., Klassen, E., Srivastava, A., Jermyn, I.: Removing shape-preserving transformations in square-root elastic (SRE) framework for shape analysis of curves. In: EMMCVPR, pp. 387–398 (2007)
22. Joshi, S., Xie, Q., Kurtek, S., Srivastava, A., Laga, H.: Surface shape morphometry for hippocampal modeling in Alzheimer's disease. In: International Conference on Digital Image Computing: Techniques and Applications (DICTA), pp. 1–8 (2016)
23. Just, N.: Improving tumour heterogeneity MRI assessment with histograms. Br. J. Cancer **111**(12), 2205 (2014)
24. Karcher, H.: Riemannian center of mass and mollifier smoothing. Commun. Pure Appl. Math. **30**(5), 509–541 (1977)
25. Kass, R.E., Vos, P.W.: Geometrical Foundations of Asymptotic Inference, vol. 908. Wiley, New York (2011)
26. Kendall, D.G.: Shape manifolds, procrustean metrics and complex projective spaces. Bull. Lond. Math. Soc. **16**, 81–121 (1984)
27. Kurtek, S.: A geometric approach to pairwise Bayesian alignment of functional data using importance sampling. Electron. J. Stat. **11**(1), 502–531 (2017)
28. Kurtek, S., Needham, T.: Simplifying transforms for general elastic metrics on the space of plane curves. ArXiv:1803.10894v1 (2018)
29. Kurtek, S., Klassen, E., Ding, Z., Jacobson, S., Jacobson, J., Avison, M., Srivastava, A.: Parameterization-invariant shape comparisons of anatomical surfaces. IEEE Trans. Med. Imaging **30**(3), 849–858 (2011)
30. Kurtek, S., Srivastava, A., Klassen, E., Ding, Z.: Statistical modeling of curves using shapes and related features. J. Am. Stat. Assoc. **107**(499), 1152–1165 (2012)
31. Kurtek, S., Su, J., Grimm, C., Vaughan, M., Sowell, R., Srivastava, A.: Statistical analysis of manual segmentations of structures in medical images. Comput. Vis. Image Underst. **117**, 1036–1050 (2013)
32. Kurtek, S., Wu, W., Christensen, G., Srivastava, A.: Segmentation, alignment and statistical analysis of biosignals with application to disease classification. J. Appl. Stat. **40**(6), 1270–1288 (2013)
33. Kurtek, S., Samir, C., Ouchchane, L.: Statistical shape model of elastic endometrial tissue surfaces. In: International Conference on Pattern Recognition Applications and Methods (2014)

34. Kurtek, S., Xie, Q., Samir, C., Canis, M.: Statistical model for simulation of deformable elastic endometrial tissue shapes. Neurocomputing **173**(P1), 36–41 (2016)
35. Laga, H., Xie, Q., Jermyn, I.H., Srivastava, A.: Numerical inversion of SRNF maps for elastic shape analysis of genus-zero surfaces. IEEE Trans. Pattern Anal. Mach. Intell. **39**(12), 2451–2464 (2017)
36. Le, H.: Locating Frechet means with application to shape spaces. Adv. Appl. Probab. **33**(2), 324–338 (2001)
37. Maier-Hein, K.H., Neher, P.F., Houde, J.C., Côté, M.A., Garyfallidis, E., Zhong, J., Chamberland, M., Yeh, F.C., Lin, Y.C., Ji, Q., Reddick, W.E., Glass, J.O., Chen, D.Q., Feng, Y., Gao, C., Wu, Y., Ma, J., Renjie, H., Li, Q., Westin, C.F., Deslauriers-Gauthier, S., González, J.O.O., Paquette, M., St-Jean, S., Girard, G., Rheault, F., Sidhu, J., Tax, C.M.W., Guo, F., Mesri, H.Y., Dávid, S., Froeling, M., Heemskerk, A.M., Leemans, A., Boré, A., Pinsard, B., Bedetti, C., Desrosiers, M., Brambati, S., Doyon, J., Sarica, A., Vasta, R., Cerasa, A., Quattrone, A., Yeatman, J., Khan, A.R., Hodges, W., Alexander, S., Romascano, D., Barakovic, M., Auría, A., Esteban, O., Lemkaddem, A., Thiran, J.P., Cetingul, H.E., Odry, B.L., Mailhe, B., Nadar, M.S., Pizzagalli, F., Prasad, G., Villalon-Reina, J.E., Galvis, J., Thompson, P.M., Requejo, F.D.S., Laguna, P.L., Lacerda, L.M., Barrett, R., Dell'Acqua, F., Catani, M., Petit, L., Caruyer, E., Daducci, A., Dyrby, T.B., Holland-Letz, T., Hilgetag, C.C., Stieltjes, B., Descoteaux, M.: The challenge of mapping the human connectome based on diffusion tractography. Nat. Commun. **8**(1), 1349 (2017)
38. Marron, J., Ramsay, J.O., Sangalli, L.M., Srivastava, A.: Functional data analysis of amplitude and phase variation. Stat. Sci. **30**(4), 468–484 (2015)
39. Oishi, K., Mielke, M., Albert, M., Lyketsos, C., Mori, S.: DTI analyses and clinical applications in Alzheimer's disease. In: Advances in Alzheimer's Disease, vol. 2, pp. 525–534 (2011)
40. Pennec, X.: Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements. J. Math. Imaging Vis. **25**(1), 127–154 (2006)
41. Ramsay, J., Silverman, B.: Functional Data Analysis. Springer, Berlin (2005)
42. Rao, C.R.: Information and the accuracy attainable in the estimation of statistical parameters. In: Breakthroughs in Statistics, pp. 235–247. Springer, Berlin (1992)
43. Robinson, D.T.: Functional Data Analysis and Partial Shape Matching in the Square Root Velocity Framework. Ph.D. thesis, Florida State University, Tallahassee (2012)
44. Saha, A., Banerjee, S., Kurtek, S., Narang, S., Lee, J., Rao, G., Martinez, J., Bharath, K., Rao, A., Baladandayuthapani, V.: DEMARCATE: Density-based magnetic resonance image clustering for assessing tumor heterogeneity in cancer. NeuroImage Clin. **12**, 132–143 (2016)
45. Samir, C., Kurtek, S., Srivastava, A., Canis, M.: Elastic shape analysis of cylindrical surfaces for 3D/2D registration in endometrial tissue characterization. IEEE Trans. Med. Imaging **33**(5), 1035–1043 (2014)
46. Srivastava, A., Jermyn, I.H., Joshi, S.H.: Riemannian analysis of probability density functions with applications in vision. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007)
47. Srivastava, A., Klassen, E., Joshi, S.H., Jermyn, I.H.: Shape analysis of elastic curves in Euclidean spaces. IEEE Trans. Pattern Anal. Mach. Intell. **33**, 1415–1428 (2011)
48. Srivastava, A., Wu, W., Kurtek, S., Klassen, E., Marron, J.S.: Registration of functional data using Fisher-Rao metric. ArXiv:1103.3817v2 (2011)
49. Srivastave, A., Klassen, E.P.: Functional and Shape Data Analysis. Springer, Berlin (2016)
50. Wang, L., Beg, M.F., Ratnanather, J.T., Ceritoglu, C., Younes, L., Morris, J.C., Csernansky, J.G., Miller, M.I.: Large deformation diffeomorphism and momentum based hippocampal shape discrimination in dementia of the Alzheimer type. IEEE Trans. Med. Imaging **26**(4), 462–470 (2007)
51. Xie, Q., Kurtek, S., Le, H., Srivastava, A.: Parallel transport of deformations in shape space of elastic surfaces. In: International Conference on Computer Vision (2013)
52. Yavariabdi, A., Samir, C., Bartoli, A., Ines, D.D., Bourdel, N.: Contour-based TVUS-MR image registration for mapping small endometrial implants. In: Abdominal Imaging, vol. 8198, pp. 145–154 (2013)

53. Younes, L.: Computable elastic distance between shapes. SIAM J. Appl. Math. **58**(2), 565–586 (1998)
54. Younes, L.: Elastic distance between curves under the metamorphosis viewpoint. ArXiv:1804.10155 (2018)